Matchable-Observable Linear Models and Direct Filter Tuning: An Approach to Multivariable Identification

Rodrigo Alvite Romano, *Member, IEEE*, and Felipe Pait, *Senior Member, IEEE*

***Abstract*—Identification of linear time-invariant multivariable systems can best be understood as comprising three separate problems: selection of system model structure, filter design, and parameter estimation itself. Approaching the first using matchable-observable models originally developed in the adaptive control literature and the second via direct or derivative-free optimization, effective least-squares algorithms can be used for parameter estimation. The accuracy, robustness and moderate computational demands of the methods proposed are demonstrated via simulations with randomly generated models and applied to identification using real process data. The results obtained are comparable or superior to the best results obtained using standard implementations of the algorithms described in the literature.**

***Index Terms*—Direct optimization, multivariable systems, parameter estimation, system identification.**

I. Multivariable Linear Identification

EFFICIENT and reliable techniques for identifying multivariable plants, which often present significant interactions between the multiple input-output channels, have attracted considerable research effort. Despite progress in the last decades [1], there is still room for improvement in the algorithms for identification of multi-input multi-output (MIMO) systems from input-output measurements [2]. Challenges include the development of robust and accurate algorithms capable of dealing with large scale systems, especially when the amount of available data is limited. State-space representations are among the most convenient descriptions of multivariable systems for identification purposes [3], [4]. Given a model parameterization, one approach to parameter estimation is to maximize the likelihood with respect to the observed data, usually using methods based on gradient search.

Parameterizations where the parameter space is not in one-to-one correspondence to the set of admissible process models can offer numerical advantages in the maximum likelihood (ML) framework [5]. One among these representations is the fully parameterized state-space model, in which all elements of all matrices are unrestricted and the dimension of the model's state is the only structural parameter which needs to be specified. Models of this type are overparameterized, making it necessary to deal with the extra degrees of freedom. In the data driven local coordinates (DDLC) method [6]–[8], at each iteration the optimization is regularized, so that the search is performed in a subspace which is of minimal dimension.

The expectation-maximization (EM) algorithm [9] computes ML estimates of fully parameterized state-space models without using gradient-based optimization. This method uses initial values of the system matrices to compute Kalman-smoothed state sequence estimates. These are then used to re-estimate the system matrices via linear regression. This smoothing/linear regression process is repeated until the relative decrease of a likelihood function falls below a predefined threshold. Thanks to its modest computational requirements, compared to gradient based search methods, EM methods can be useful in handling systems with high state dimension and multiple inputs and outputs [2], [10].

The ML approach requires solving non-convex, nonlinear optimization problems [8]. Depending on the initial parameter estimates, iterative search methods may lead to local minima far from the global solution [4]. This is a reason for interest in subspace system identification [5]. These non-iterative methods are effective in finding state-space MIMO models for several applications [11], without requiring specific structural decisions. Originally, subspace techniques were conceived considering fully parameterized state space models. A subspace-based identification algorithm for a minimal parameterization was proposed recently [12]. Subspace methods are simple and numerically efficient, but result in estimates which are not as accurate as the maximum likelihood ones [13]. The subspace approach can be used to provide an initial estimate for nonlinear optimization algorithms that compute ML estimates [4], [10].

The quasi-canonical MIMO state-space representations proposed in [14] in the context of adaptive control have the properties of observability, match-point controllability, and matchability, which will be further elaborated in Section III. As a consequence, no undesired pole-zero cancellations can appear, the number of model parameters is not excessive, linear least-squares estimation methods are applicable, and parameter estimation can be accomplished without the need for iterative or

Manuscript received September 3, 2015; revised May 2, 2016; accepted August 11, 2016. Date of publication August 25, 2016; date of current version April 24, 2017. This work was supported by the Instituto Mauá de Tecnologia (IMT), and by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), grant 2012/03719-3. Recommended by Associate Editor M. Verhaegen.

R. A. Romano is with Escola de Engenharia Mauá, Instituto Mauá de Tecnologia São Caetano do Sul SP, Brazil (e-mail: rromano@maua.br).

F. Pait is with Escola Politécnica da Universidade de São Paulo São Paulo SP, Brazil (e-mail: pait@lac.usp.br).

Digital Object Identifier 10.1109/TAC.2016.2602891

nonlinear optimization. This paper develops a novel multivariable system identification algorithm based on a discrete-time version of these matchable-observable design models, which, in spite of their desirable features, have to the best of our knowledge not yet been employed in system identification.

A recent kernel-based approach to linear system identification uses techniques inspired by the machine learning literature, in which hyper-parameters control the regularization of estimates of impulse response models [15]. The approach developed here also makes use of extra tunable parameters, which, as will be seen, have an explicit physical interpretation related to the location of the poles of the filters used to construct observers in predictor form. In contrast with kernel-based methods which use iterative adjustment of the hyper-parameters (see for instance [16]), in our proposal the extra parameters are tuned using derivative-free direct optimization.

In Section II we give an overview of the rationale behind the identification scheme we advocate. In Section III the structure of the matchable-observable linear identification models is presented, leading to especially convenient parameterizations of the models (III-A) and filters (III-B); in Section III-C the connection with prediction error methods is explained. Section IV describes the component algorithms of our MOLI-ZOFT scheme, namely, linear least-squares parameter estimation (IV-A), filter tuning via direct optimization (IV-B), and the selection of model structure (IV-C). Section V describes simulation results with random models (V-A), tests of the structure selection algorithm (V-B), and a case study with real data (V-C). Section VI suggests directions for further work.

II. The MOLI-ZOFT Approach

In this paper we contend that identification of linear time-invariant multivariable input-output systems can best be understood as comprising three separate problems. First is selection of a structure containing system models capable of approximating the input-output behavior of the process to be identified. Second is the design of filters that separate process signals from disturbances and measurement noise, in a manner conducive to parameter estimation. Third is parameter estimation itself.

Linearly parameterized state-space realizations may contain process models of varying McMillan degree, leading to pole-zero cancellations and other problematic phenomena. Some of these issues appear when using equation error methods based on autoregressive models such as ARX and its generalizations. To avoid dealing with parameter estimation in a disjoint and non-convex set, we are led to consider a more fundamental invariant: the list of observability indices of the models under consideration. The reason is that the coefficients of the transfer function matrices of a given McMillan degree form a set whose complicated geometry does not lend itself easily to the linear parameterizations which are preferable from an algorithmic point of view. Comparison of Examples 1 and 2 in Section III-A will serve to illustrate this point.

In single-input single-output (SISO) identification, model selection consists simply in picking the order of the transfer function to be estimated. Often one starts with a low-order model and augments its dimension until the increase of modeling precision no longer justifies the added complexity. With the MIMO quasi-canonical forms described in Section III, which were chosen to avoid the need for nonlinear optimization, simply increasing the McMillan degree of the parameterized model set is not enough. Rather than trying to estimate the "true" indices of the plant, we take a hint from the SISO case and, starting with low-order models, augment the elements of the list of observability indices one by one, as needed to improve identification.

As for the varied signals collectively referred to as "noise," one approach is to try to model their statistics. This is often pursued by nonlinear optimization, simultaneously estimating the input-to-output transfer function and the noise-to-output transfer function. An alternative is to perform iterative estimations of the noise by computing residues of successive parameter estimates. This was the route taken in a previous contribution [17], where multivariable matchable-observable models were employed with success.

We shall here take a more pragmatic, or Galilean, approach, which turns out to have a number of practical advantages over the use of regressors based on delayed sequences of inputs and outputs. The model structures we consider necessitate and possess a built-in set of observer poles, which act as a filter to the process noise. We propose to tune this observer via direct or derivative-free optimization methods, with the goal of minimizing a criterion that quantifies the accuracy of the model obtained with each set of observer poles. Considerable freedom is retained in the choice of the criterion; in simulations we have used one based on the variance-accounted-for. The direct optimization method we have chosen is based on computing the barycenter of a sequence of sets of identifier poles. The weight given to each test filter is the exponential of a measure of the quality of the best corresponding fit. The method is conceptually transparent, straightforward to code, and the free design parameters have an intuitive interpretation. The barycenter method is discussed in [18]. Direct optimization was used for filter tuning in recent work [19] which contains some of the simulations of Section V-A. Finally, parameter estimation itself reduces to linear regression and can be approached by standard least-squares methods.

The overall procedure we advocate combines the 3 phases hierarchically as follows. We start with a low-order model, with a given set of observability indices. For each value of the filter parameters, we compute the values of the model parameters that optimize a least-squares criterion. The estimation procedure results in an index of merit, the user-defined criterion J, which depends on the filter. We search for filter parameters via direct optimization of J, and the model parameters are determined as a consequence (the procedure does *not* attempt to find the jointly optimal set of process and filter parameter). Afterwards we increase the McMillan degree of the model by raising each observability index in the list. For each list of indices, the procedure of direct filter optimization, which in turn calls for the solution of a sequence of least-squares problems, is repeated. The list of indices which results in the largest improvement of J is selected. The process of augmenting the list of observability indices one by one is repeated until either the desired performance is achieved or the model dimension reaches an user defined upper bound. We call the method MOLI-ZOFT,

for Matchable-Observable Linear Identification with Zero-order Oracle Filter Tuning.[1]

Our algorithm may seem needlessly complicated, but that is not the case. First, the procedure for picking the model structure by augmenting the list of observability indices and thus increasing the model's McMillan degree is straightforward and does not lead to a combinatorial explosion of the number of candidate model structures. Second, direct optimization of the filters is computationally light and robust. It does not require manipulation of unwieldy and often unavailable explicit formulas for the dependence between the poles and the index of merit J. Other methods available in the direct or derivative-free optimization literature, besides the barycenter method we elected, might be tried, perhaps with similar success, because finding the precise optimal filter poles does *not* appear to be necessary for the satisfactory performance of the complete identification algorithm. Third, parameter estimation via linear least-squares can be done in closed-form for batch identification, or in a recursive manner. Computationally the crucial part, parameter estimation is called by the direct filter optimization procedure, which in turn is called by the model structure selection procedure. Proven least-squares estimation methods underpin the effectiveness of the algorithms we propose, and finite-time convergence is guaranteed by construction.

III. Matchable-Observable Linear Identification Model Structure

Any strictly proper transfer function of dimension $m \times p$ and McMillan degree n can be realized by the following state-space model structure with inputs $u_k \in \mathbb{R}^m$ and outputs $y_k \in \mathbb{R}^p$

$$x_{k+1} = \left(A + D(\theta)\left(I_p - G(\theta)\right)^{-1} C\right) x_k + B(\theta) u_k \tag{1}$$

$$y_k = \left(I_p - G(\theta)\right)^{-1} C x_k, \tag{2}$$

where θ denotes the model parameters, $I_p \in \mathbb{R}^p$ is an identity matrix, and (C, A) is a stable, observable and parameter-independent pair. A direct feedthrough term could be added to (2) but we have only considered strictly proper transfer function matrices in the present paper. The parameter matrix $G(\theta) \in \mathbb{R}^{p\times p}$ is strictly lower triangular. The other parameter matrices $D(\theta)$ and $B(\theta)$ take values in $\mathbb{R}^{n\times p}$ and $\mathbb{R}^{n\times m}$, respectively. This structure, based on previous work including [21]–[23], is fully spelled out in [14]. The dimension of the parameter space is

$$\dim \theta = n\,(m+p) + \frac{1}{2} p\,(p-1)\,.$$

For comparison, a fully parameterized model has $n(m+p) + n^2$ parameters.

We shall pick C and A in the form

$$C = \text{block diagonal } \{C_1, C_2, \ldots, C_p\} \tag{3}$$

$$A = \text{block diagonal } \{A_1, A_2, \ldots, A_p\}, \tag{4}$$

where the matrices C_i and A_i are constructed according to the following guidelines:

1) Given a list of observability indices

$$l = \{n_1, n_2, \ldots, n_p\}\,, \text{ s.t. } \sum_{i=1}^{p} n_i = n,$$

choose an arbitrary stable monic polynomial $\alpha(q)$ of degree $\underline{n} = \max\{l\}$, such that, for each $i \in \{1, \ldots, p\}$, $\alpha(q)$ has a real monic factor $\lambda_i(q)$ of degree n_i.

2) For each $i \in \{1, \ldots, p\}$, pick a single-output, n_i–dimensional observable pair $(C_i,\ A_i)$, so that $\lambda_i(q)$ is the characteristic polynomial of A_i.

Let $(\underline{C},\ \underline{A})$ denote a single-output, $\underline{n}$–dimensional observable pair such that the characteristic polynomial of $\underline{A}$ is $\alpha(q)$. Next, define a matrix-valued function $E_{\mathbb{I}}(\theta) : \mathbb{R}^{\underline{n}(m+p)} \to \mathbb{R}^n$

$$E_{\mathbb{I}}(\theta) = \begin{bmatrix} R_1 \mathcal{H}\left\{M_1 \begin{bmatrix} D_1(\theta) & B_1(\theta) \end{bmatrix}\right\} \\ \vdots \\ R_p \mathcal{H}\left\{M_p \begin{bmatrix} D_p(\theta) & B_p(\theta) \end{bmatrix}\right\} \end{bmatrix} \in \mathbb{R}^{n\times \underline{n}(m+p)}, \tag{5}$$

where matrices $[D_i(\theta)\ B_i(\theta)] \in \mathbb{R}^{n_i\times(m+p)}$ are obtained by partitioning the rows of $[D(\theta)\ B(\theta)] \in \mathbb{R}^{n\times(m+p)}$ according to the observability index list l. The left-invertible matrices $M_i \in \mathbb{R}^{\underline{n}\times n_i}$ are given by

$$C_i\left(qI_{n_i} - A_i\right)^{-1} = \underline{C}\left(qI_{\underline{n}} - \underline{A}\right)^{-1} M_i \tag{6}$$

and $R_i \in \mathbb{R}^{n_i\times \underline{n}}$ is a left inverse of M_i, that is $R_i M_i = I_{n_i}$. Furthermore, the linear transformation $\mathcal{H} : \mathbb{R}^{\underline{n}\times(m+p)} \to \mathbb{R}^{\underline{n}\times\underline{n}(m+p)}$ in (5) is defined by

$$S \mapsto \begin{bmatrix} \Upsilon\Gamma^T(s_1) & \cdots & \Upsilon\Gamma^T(s_{m+p}) \end{bmatrix}$$

where Υ is the inverse of the observability matrix of $(\underline{C}, \underline{A})$, $\Gamma(s_j)$ is the controllability matrix of $(\underline{A}, s_j)$ and s_j is the jth column of the matrix S.

Then, as shown in [14], it is possible to construct an $\underline{n}(m+p)$ -dimensional output predictor of the form

$$\xi_{k+1} = A_{\mathbb{I}}\xi_k + D_{\mathbb{I}} y_k + B_{\mathbb{I}} u_k \tag{7}$$

$$\hat{y}_k = C_{\mathbb{I}}(\theta)\xi_k + G_{\mathbb{I}}(\theta) y_k, \tag{8}$$

where[2]

$$A_{\mathbb{I}} = \text{ block diagonal } \{\underbrace{\underline{A}^T, \ldots, \underline{A}^T}_{m+p \text{ times}}\} \in \mathbb{R}^{\underline{n}(m+p)\times\underline{n}(m+p)} \tag{9}$$

$$B_{\mathbb{I}} = \left[\begin{array}{cccc} 0_{p\underline{n}\times m} & & & \\ \hline \underline{C}^T & 0 & \cdots & 0 \\ 0 & \underline{C}^T & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \underline{C}^T \end{array}\right] \in \mathbb{R}^{\underline{n}(m+p)\times m}$$

[1] The term zero-order oracle refers to the fact that the direct optimization algorithm employed for filter tuning is derivative-free: it makes use of the value of the function only [20].

[2] The superscript ^{T} indicates matrix transposition, the 0's in the equations of $B_{\mathbb{I}}$ and $D_{\mathbb{I}}$ denote matrices of suitable dimension filled with 0, and $0_{m\times p}$ is an $m \times p$ matrix of zeros.

$$D_{\mathbb{I}} = \begin{bmatrix} \underline{C}^T & 0 & \cdots & 0 \\ 0 & \underline{C}^T & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \underline{C}^T \\ \hline & 0_{m\underline{n}\times p} & & \end{bmatrix} \in \mathbb{R}^{\underline{n}(m+p)\times p}, \tag{10}$$

and

$$G_{\mathbb{I}}(\theta) = G(\theta) \tag{11}$$

$$C_{\mathbb{I}}(\theta) = CE_{\mathbb{I}}(\theta) \in \mathbb{R}^{p\times \underline{n}(m+p)}, \tag{12}$$

so that ξ_k is mapped to the state vector x_k through $E_{\mathbb{I}}(\theta)$, that is,

$$\hat{x}_k = E_{\mathbb{I}}(\theta)\xi_k, \tag{13}$$

where $\hat{x}_k$ is an asymptotic estimate of process model state x_k. With $E_{\mathbb{I}}(\theta)$ as constructed above the linking equations below hold:

$$E_{\mathbb{I}}(\theta)A_{\mathbb{I}} = AE_{\mathbb{I}}(\theta) \tag{14}$$

$$E_{\mathbb{I}}(\theta)B_{\mathbb{I}} = B(\theta) \tag{15}$$

$$E_{\mathbb{I}}(\theta)D_{\mathbb{I}} = D(\theta). \tag{16}$$

The main motivation for the use of the model structure just described is that the dynamic behavior of ξ_k does not depend on θ, so the output predictor (7)–(8) is linear in the model's parameter matrices $C_{\mathbb{I}}$ and $G_{\mathbb{I}}$. This is possible thanks to the structure of model (1)–(2), whose poles can be assigned via the output-injection matrix $D(\theta)$. We are not aware of any structure with this property which does not incorporate the output change of variables here effected by the matrix G, or that does not require specification of a list of observability indices [24]. Although the dimension of the observer is not minimal, the extra modes are stable by construction.

The parameterization described in this section has some additional properties which makes it particularly suitable for multivariable system identification. First, model (1)–(2) is observable for all parameter values. This follows from observability of each pair $(C_i,\ A_i)$, the block-diagonal structure of $(C,\ A)$, and the fact that observability is not affected either by output injection or by nonsingular changes of variables. Second, the model is match-point controllable, that is to say, it does not exhibit pole-zero cancellations when its transfer function matches that of a process in the admissible set of LTI processes with m inputs, p outputs, McMillan degree n, and list of observability indices l. This is evident from the fact that the model only contains n integrators. Third, it is capable of matching all such transfer functions, as shown in [14].

The remainder of this section describes a parameterization devised using model (1)–(2) together with identifier (7)–(8), and a parameterization of the filter constructed by the choice of the pair $(\underline{C}, \underline{A})$.

A. Parameterization of MOLI Models

From (12) and (5), it is clear that the matrices M_i directly influence the parameters of the model represented through the observer realized as in (7)–(8). Therefore, before addressing the parameter estimation task, it is worth discussing how to compute such matrices. For this purpose the following lemma is introduced:

Lemma 1: For each $i \in \{1, \ldots, p\}$, consider the polynomial

$$\kappa_i(q) \triangleq \frac{\alpha(q)}{\lambda_i(q)} = q^{\underline{n}-n_i} + \kappa_{i,\underline{n}-n_i} q^{\underline{n}-n_i-1} + \ldots + \kappa_{i,2}q + \kappa_{i,1}$$

If the observable pairs (C_i, A_i) are matrices in right companion form A_i and $C_i = [0\ \ \cdots\ \ 0\ \ 1]$, then each matrix M_i which satisfies (6) is Toeplitz and its first column is given by

$$m_{i,1} = \begin{bmatrix} \kappa_{i,1} & \cdots & \kappa_{i,\underline{n}-n_i} & 1 & 0 & \cdots & 0 \end{bmatrix}^T.$$

Proof: Let $\lambda_i(q) = q^{n_i} + \alpha_{i,n_i} q^{n_i-1} + \ldots + \alpha_{i,2}q + \alpha_{i,1}$ be the characteristic polynomial of A_i, then the adjoint matrix of $qI_{n_i} - A_i$ can be decomposed as [24]

$$\operatorname{adj}\ (qI_{n_i} - A_i) = \begin{bmatrix} 1 & \alpha_{i,n_i} & \cdots & \alpha_{i,2} \\ 0 & 1 & & \alpha_{i,3} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} q^{n_i-1} \\ q^{n_i-2} \\ \vdots \\ 1 \end{bmatrix}$$

$$\times \begin{bmatrix} 1 & q & \cdots & q^{n_i-1} \end{bmatrix} - \lambda_i(q) \begin{bmatrix} 0 & 1 & q & \cdots & q^{n_i-2} \\ & 0 & 1 & \ddots & \vdots \\ & & 0 & \ddots & q \\ & & & \ddots & 1 \\ & \bigcirc & & & 0 \end{bmatrix}.$$

As A_i are right companion matrices and each C_i is a row vector filled with zeros except for a "1" in its last column, (6) can be rewritten as

$$\begin{bmatrix} 1 & q & \cdots & q^{n_i-1} \end{bmatrix} \frac{\alpha(q)}{\lambda_i(q)} = \begin{bmatrix} 1 & q & \cdots & q^{\underline{n}} \end{bmatrix} M_i. \tag{17}$$

Thus, according to (17), the lth column of M_i is given by

$$m_{i,l} = \Big[\underbrace{0\ \cdots\ 0}_{l-1 \text{ times}} \quad \kappa_{i,1}\ \cdots\ \kappa_{i,\underline{n}-n_i} \quad 1 \quad \underbrace{0\ \cdots\ 0}_{\underline{n}-n_i-l \text{ times}} \Big]^T.$$

■

Remark 1: The lemma gives a straightforward expression for M_i in (6), which depends only on the coefficients of the polynomial $\alpha(q)/\lambda_i(q)$. The formula in Lemma 1 is more explicit and computationally simpler than the procedure proposed in [14] to obtain the matrices M_i.

Provided that the pairs (C_i, A_i) are stable and observable, (3)–(4) may be parameterized arbitrarily. To take advantage of the simplified calculation of the matrices M_i presented in Lemma 1, from this point on A_i is taken to be right companion and $C_i = [0\ \ \cdots\ \ 0\ \ 1]$, for each $i \in \{1, \ldots, p\}$. Next, a useful

result concerning the mapping of the elements in $B(\theta)$ and $D(\theta)$ into the matrix $C_{\mathbb{I}}(\theta)$ is presented.

Lemma 2: For each $i \in \{1, \ldots, p\}$, define

$$\mathcal{M}_i = \text{block diagonal} \left\{ \underbrace{M_i, \ldots, M_i}_{m+p \text{ times}} \right\} \in \mathbb{R}^{\underline{n}(m+p) \times n_i (m+p)}.$$

The ith row of $C_{\mathbb{I}}(\theta)$ is given by

$$C_{\mathbb{I},i}(\theta) = \left[\mathcal{M}_i \text{ vec } \left\{ \left[D_i(\theta) \quad B_i(\theta) \right] \right\} \right]^T, \tag{18}$$

where $\text{vec}\{X\}$ stacks the columns of the matrix X on top of each other.

Proof: The structure of M_i enables us to pick matrices R_i, subject to $R_i M_i = I_{n_i}$, whose last rows are equal to $\underline{C}$. This implies that (12) can be rewritten as

$$C_{\mathbb{I}}(\theta) = \begin{bmatrix} \underline{C} & & \bigcirc \\ & \ddots & \\ \bigcirc & & \underline{C} \end{bmatrix} \begin{bmatrix} \mathcal{H} \left\{ M_1 \left[D_1(\theta) \; B_1(\theta) \right] \right\} \\ \vdots \\ \mathcal{H} \left\{ M_p \left[D_p(\theta) \; B_p(\theta) \right] \right\} \end{bmatrix}.$$

Let $z_{i,l}$ denote the lth column of $Z_i \triangleq M_i \left[D_i(\theta) \; B_i(\theta) \right] \in \mathbb{R}^{\underline{n} \times (m+p)}$. It was shown in [14, see Lemma 3] that: $\underline{C} \Upsilon \Gamma^T(z_{i,l}) = z_{i,l}^T$. Using such result, it follows that

$$\begin{aligned} C_{\mathbb{I}}(\theta) &= \begin{bmatrix} z_{1,1}^T & z_{1,2}^T & \cdots & z_{1,m+p}^T \\ \vdots & \vdots & & \vdots \\ z_{p,1}^T & z_{p,2}^T & \cdots & z_{p,m+p}^T \end{bmatrix} \\ &= \begin{bmatrix} \{ M_1 \left[D_1(\theta) & B_1(\theta) \right] \}^T \\ & \vdots \\ \{ M_p \left[D_p(\theta) & B_p(\theta) \right] \}^T \end{bmatrix}. \end{aligned} \tag{19}$$

Finally, (19) together with the identity

$$\text{vec} \left\{ M_i \left[D_i(\theta) \quad B_i(\theta) \right] \right\} = \mathcal{M}_i \text{vec} \left\{ \left[D_i(\theta) \quad B_i(\theta) \right] \right\}$$

imply that (18) holds. ■

Remark 2: Lemma 2 shows how to obtain $C_{\mathbb{I}}(\theta)$ without computing $E_{\mathbb{I}}(\theta)$. Knowledge of the matrices M_i is sufficient to determine how the parameters of the state-space canonical representation (1)–(2) are mapped into the predictor (7)–(8). Implementation of the parameter estimation algorithm detailed in Section IV-A is thus simplified.

Example 1: Suppose $m = 2$, $p = 2$, $n = 4$ and $l = \{3, 1\}$. Let $\alpha(q) = q^3 + \alpha_3 q^2 + \alpha_2 q + \alpha_1$ be an arbitrary stable polynomial of degree $\underline{n} = 3$, whose roots are q_1, q_2 and q_3. Choose $\lambda_1(q) = \alpha(q)$ and $\lambda_2(q) = q - q_1$, which has degree n_2 and divides $\alpha(q)$ as required. In this case, (1) becomes

$$x_{k+1} = \mathfrak{A} x_k + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix} u_k, \tag{20}$$

where

$$\mathfrak{A} = \begin{bmatrix} 0 & 0 & -\alpha_1 & 0 \\ 1 & 0 & -\alpha_2 & 0 \\ 0 & 1 & -\alpha_3 & 0 \\ 0 & 0 & 0 & -q_1 \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ d_{31} & d_{32} \\ d_{41} & d_{42} \end{bmatrix} \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & g_{21} & 1 \end{bmatrix}}_{= (I_p - G(\theta))^{-1} C}.$$

Applying the result presented in Lemma 1 we get $M_1 = I_3$ and $M_2 = \begin{bmatrix} q_2 q_3 & -(q_2 + q_3) & 1 \end{bmatrix}^T$. Next, pick left inverses $R_1 = I_3$ and $R_2 = [0 \;\; 0 \;\; 1]$. The state ξ_k in (7) has dimension 12. Using (18), the rows of the 2×12 parameter matrix $C_{\mathbb{I}}(\theta)$ are

$$\begin{aligned} C_{\mathbb{I},1}(\theta) &= \begin{bmatrix} d_{11} & d_{21} & d_{31} & d_{12} & d_{22} & d_{32} \\ b_{11} & b_{21} & b_{31} & b_{12} & b_{22} & b_{32} \end{bmatrix} \\ C_{\mathbb{I},2}(\theta) &= \begin{bmatrix} d_{41} & d_{42} & b_{41} & b_{42} \end{bmatrix} \mathcal{M}_2^T \\ &= \begin{bmatrix} q_2 q_3 d_{41} & -(q_2 + q_3) d_{41} & d_{41} \\ q_2 q_3 d_{42} & -(q_2 + q_3) d_{42} & d_{42} \\ q_2 q_3 b_{41} & -(q_2 + q_3) b_{41} & b_{41} \\ q_2 q_3 b_{42} & -(q_2 + q_3) b_{42} & b_{42} \end{bmatrix}. \end{aligned}$$

If we adopt $\alpha(q) = q^{\underline{n}}$, the second row of $C_{\mathbb{I}}$ becomes

$$C_{\mathbb{I},2}(\theta) = [0 \;\; 0 \;\; d_{41} \;\; 0 \;\; 0 \;\; d_{42} \;\; 0 \;\; 0 \;\; b_{41} \;\; 0 \;\; 0 \;\; b_{42}].$$

The case in which all the eigenvalues of the matrix A are set at the origin of the complex plane is dealt with in more depth in Section III-C.

Example 2: Consider the class of systems with the same number of inputs and outputs and the same McMillan degree as in Example 1, but now with observability indices $l = \{2, 2\}$. In this case, $\alpha(q)$ must be a second order monic stable polynomial, which is represented by $\alpha(q) = q^2 + \alpha_2 q + \alpha_1$. As $n_1 = n_2$, an obvious choice is $\lambda_1(q) = \lambda_2(q) = \alpha(q)$, leading to a state equation as the same form of (20), but with

$$\mathfrak{A} = \begin{bmatrix} 0 & -\alpha_1 & 0 & 0 \\ 1 & -\alpha_2 & 0 & 0 \\ 0 & 0 & 0 & -\alpha_1 \\ 0 & 0 & 1 & -\alpha_2 \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ d_{31} & d_{32} \\ d_{41} & d_{42} \end{bmatrix} \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & g_{21} & 0 & 1 \end{bmatrix}}_{= (I_p - G(\theta))^{-1} C}.$$

In this case ξ_k has dimension 8 and the matrix $C_{\mathbb{I}}(\theta)$ is 2×8. Using Lemma 1 we obtain $M_1 = M_2 = R_1 = R_2 = I_2$, yielding

$$C_{\mathbb{I}}(\theta) = \begin{bmatrix} d_{11} & d_{21} & d_{12} & d_{22} & b_{11} & b_{21} & b_{12} & b_{22} \\ d_{31} & d_{41} & d_{32} & d_{42} & b_{31} & b_{41} & b_{32} & b_{42} \end{bmatrix}.$$

The parameterizations of Examples 1 and 2 are distinct but overlapping: it is not difficult to come up with numeric examples of transfer functions that could be realized by systems with different lists of indices. For such systems, there is no unique way to choose one structure over the other, so it is appropriate to refer to the parameterizations as quasi-canonical. Although

the parameter space is of the same dimension for both types of systems, there is no meaningful way to endow the combined set with the structure of a connected differentiable manifold. The dimension of the regressor matrix changes with the list of indices, but the number of parameters does not. In both examples it is 17, that is, one extra parameter in addition to the minimal dimension [7], but far less than 32 for a fully parameterized 4-state, 2-input, 2-output model.

The dimensionality of the parameter space is not minimal—models with only 16 parameters capable of matching the same class of systems could be constructed—but the models illustrated in the examples give error equations which are linear in the parameters, and numerical issues can be easily dealt with via regularization.

B. Filter Parameterization

The roots of $\alpha(q)$ determine the poles of the observer (7)–(8), which is in regressor form. The polynomial $\alpha(q)$ can be seen as the denominator of a prefilter transfer function to the input–output data, without taking into account the process dynamics. This point of view is substantiated by the following result.

Proposition 1: Consider the transfer function $L(q) = q^{\underline{n}}/\alpha(q)$. Let $y^f_{i,k} = L(q)y_{i,k}$ and $u^f_{i,k} = L(q)u_{i,k}$ be filtered versions of the ith output and input sequences, respectively. If the pair $(\underline{C}, \underline{A})$ is constructed as in Lemma 1, then

$$\xi_k = \begin{bmatrix} y^f_{1,k-\underline{n}} & \cdots & y^f_{1,k-1} & \cdots & y^f_{p,k-\underline{n}} & \cdots & y^f_{p,k-1} \\ u^f_{1,k-\underline{n}} & \cdots & u^f_{1,k-1} & \cdots & u^f_{m,k-\underline{n}} & \cdots & u^f_{m,k-1} \end{bmatrix}^T. \tag{21}$$

Proof: Notice that (7) can also be written as

$$\xi_k = \left(I_{m+p} \otimes \operatorname{adj}\left(qI_{\underline{n}} - \underline{A}^T\right)\right)\left[D_{\mathbb{I}}\ B_{\mathbb{I}}\right]\frac{I_{m+p}}{\alpha(q)}\begin{bmatrix} y_k \\ u_k \end{bmatrix},$$

where $\otimes$ denotes the Kronecker product. Using (9)–(10), provided $\underline{C} = [0\ \cdots\ 0\ \ 1]$ and $\underline{A}$ is in companion form, it is immediate to obtain (21). ■

Due to the equivalence between prefilters and noise models (see Section 14.4 in [3]), the MOLI parametrization incorporates a noise model with a factor $q^{-\underline{n}}\alpha(q)I_{m+p}$. In other words, $\alpha(q)$ can be designed to filter out the noise in the measured data. The filter parameters play a role similar to that of the hyper–parameters which, in some machine–learning inspired system identification approaches [15], are responsible for the smoothness of estimates of impulse response coefficients. Instead of treating all roots of $\alpha(q)$ as free parameters, one may simplify the design and search for the prefilter's dominant poles only.

For $\underline{n} \geq 2$, discretized versions of second–order polynomials with cutoff frequency ω_c and damping ζ are used to parameterize $\alpha(q)$, that is, the roots corresponding to the dominant modes are given by

$$q_{1,2} = e^{-\omega_c\left(\zeta \pm j\sqrt{1-\zeta^2}\right)T_s},$$

where T_s denotes the sampling period. Polynomials of higher degrees are constructed by successively introducing second–order factors with the same damping, each faster than the previous one by a constant factor. In the present work we chose this factor to be equal to 2; for extra design flexibility one could treat the ratio between successively faster filter poles as an additional design parameter. If the degree of $\alpha(q)$ is odd, a real factor $(q - q_{\underline{n}})$ is added lastly, so that $q_{\underline{n}} = |q_{\underline{n}-1}|^2$, where $q_{\underline{n}-1}$ denotes one of the roots of the second-order factor included in the preceding iteration.

When constructed in this manner, $\alpha(q)$ is parameterized in terms of a vector $\eta = [\omega_c\ \zeta]^T$, and the search for its coefficients may be treated as a 2–dimensional optimization problem in a convex set (in the less interesting case which happens if $n = p$ and $\underline{n} = 1$, $\alpha(q) = q - e^{-\omega_c T_s}$ and $\eta = \omega_c$ is 1–dimensional).

C. Regressor Model and Connections With Prediction Error Methods

Now a relationship between the MOLI parameterization and state–space innovation models is presented.

Proposition 2: Consider the regression model based on (8)

$$\epsilon_k = (I_p - G_{\mathbb{I}}(\theta))y_k - C_{\mathbb{I}}(\theta)\xi_k$$

Let $v_k = (I_p - G(\theta))\, y_k$ be an output transformation. The residuals ϵ_k are equivalent to the innovations of the state–space representation

$$\begin{aligned} \hat{x}_{k+1} = &\left(A + D(\theta)\left(I_p - G(\theta)\right)^{-1}C\right)\hat{x}_k + B(\theta)u_k \\ &+ D(\theta)\left(I_p - G(\theta)\right)^{-1}\epsilon_k \\ v_k = &\, C\hat{x}_k + \epsilon_k, \end{aligned} \tag{22}$$

where $\hat{x}_k$ denotes the estimate of x_k, given θ.

Proof: From (11)–(12) and (13) we get

$$y_k = \left(I_p - G(\theta)\right)^{-1}C\hat{x}_k + \left(I_p - G(\theta)\right)^{-1}\epsilon_k.$$

Analogously, substituting (14)–(16) into (7) results

$$\hat{x}_{k+1} = A\hat{x}_{k+1} + D(\theta)y_k + B(\theta)u_k. \tag{23}$$

By combining the preceding expressions and applying the output transformation from y_k to v_k, we obtain the innovation form (22). ■

Because the model employed in parameter estimation can be expressed in the innovation form (22), it follows that the proposed approach belongs to the broad class of prediction error methods [3]. In the present formulation the Kalman gain matrix entries are determined by $D(\theta)$ and $G(\theta)$, which in turn depend on the choice of $\alpha(q)$. The following result relates these design parameters.

Proposition 3: Define $\sigma_i = n_i + \sigma_{i-1}(\sigma_0 = 0)$. If the eigenvalues of $\underline{A}$ are chosen to be zero, which implies in $\alpha(q) = q^{\underline{n}}$, the model (1)–(2) leads to a MIMO ARX structure

$$\mathcal{A}(q)y_k = \mathcal{B}(q)u_k + \epsilon_k \tag{24}$$

where

$$\mathcal{A}(q)=\begin{bmatrix}\mathcal{A}_{11}(q) & \cdots & \mathcal{A}_{1p}(q)\\ \vdots & \ddots & \vdots\\ \mathcal{A}_{p1}(q) & \cdots & \mathcal{A}_{pp}(q)\end{bmatrix}$$

$$\mathcal{A}_{ij}(q)=\begin{cases}1-d_{\sigma_i j}q^{-1}-\ldots-d_{(\sigma_i-n_i+1)j}q^{-n_i}, & \text{for } i=j\\ -g_{ij}-d_{\sigma_i j}q^{-1}-\ldots-d_{(\sigma_i-n_i+1)j}q^{-n_i}, & \text{for } i>j\\ -d_{\sigma_i j}q^{-1}-\ldots-d_{(\sigma_i-n_i+1)j}q^{-n_i}, & \text{for } i<j\end{cases}$$

and

$$\mathcal{B}(q)=\begin{bmatrix}\mathcal{B}_{11}(q) & \cdots & \mathcal{B}_{1m}(q)\\ \vdots & \ddots & \vdots\\ \mathcal{B}_{p1}(q) & \cdots & \mathcal{B}_{pm}(q)\end{bmatrix},\mathcal{B}_{ij}(q)=\sum_{r=1}^{n_i}b_{(\sigma_i-r+1)j}q^{-r}.$$

Proof: The equation error model (24) is obtained straightforwardly from the input–output description of (22):

$$v_k=C\left(qI_n-A\right)^{-1}\left(D(\theta)y_k+B(\theta)u_k\right)+\epsilon_k.$$

Firstly, as in Lemma 1, assume that $C_i=[0\ \cdots\ 0\ \ 1]$ and A_i are right companion matrices. For $\alpha(q)=q^{\underline{n}}$, it follows that

$$C\left(qI_n-A\right)^{-1}=\begin{bmatrix}q^{-n_1}\ \cdots\ q^{-1} & & \bigcirc\\ & \ddots & \\ \bigcirc & & q^{-n_p}\ \cdots\ q^{-1}\end{bmatrix}.$$

Defining the polynomial matrices

$$\mathcal{A}(q)=I_p-G(\theta)-C\left(qI_n-A\right)^{-1}D(\theta)$$

$$\mathcal{B}(q)=C\left(qI_n-A\right)^{-1}B(\theta),$$

we obtain the matrix fraction description (24). Although a particular form of (C,A) has been assumed, there is no loss of generality as it can been verified that the main result also holds for matrices $TAT^{-1},TB(\theta),TD(\theta),CT^{-1}$ and any nonsingular transformation T. ■

If we choose $\alpha(q)=q^{\underline{n}}$, the information vector will be composed exclusively of input–output observations, and the parameters of the state–space representation (1)–(2) will appear explicitly in a linear regression model. The connection with MIMO ARX structures can also be handled as a special case of Proposition 1. As the simulation results later in the paper will make clear, freedom to choose observer poles not at the origin of the complex plane can have a very positive impact on the performance of the estimation algorithm.

To summarize this section, model (1)–(2) is capable of matching the class of transfer functions of interest in this paper. The observer (7)–(8), which is in predictor form, can be constructed using equations (11) to (16) as shown in [14], and the parameters to be estimated (namely θ) appear only in readout equation (8), in a linear manner. The next section develops estimation algorithms for the models presented.

IV. Estimation Algorithms

We have argued that system identification can be profitably split into three separate tasks. We shall present the algorithms following the hierarchy from the bottom up: parameter estimation, filter tuning, and model structure selection. It will be seen that the procedure is guaranteed to converge because each step described in IV-A, IV-B, and IV-C ends in finite time.

A. Parameter Estimation

In light of Lemma 1, which shows how to compute M_i given $\alpha(q)$, and using (8), (18) and (21), introduce the regressor or information vector relative to the ith output at instant k as

$$\varphi_{i,k}=\begin{cases}\mathcal{M}_1^T\xi_k, & \text{if } i=1\\ \xi_k^T\mathcal{M}_i\quad y_{1,k}\quad\cdots\quad y_{i-1,k}^T, & \text{if } i\neq 1\end{cases}$$

where $y_{i,k}$ indicates the ith output at instant k as well as the respective parameter vector

$$\theta_i=\begin{cases}\{[D_i(\theta)\ \ B_i(\theta)]\}, & \text{if } i=1\\ \begin{bmatrix}\{[D_i(\theta)\quad B_i(\theta)]\}^T & g_{i1} & \cdots & g_{i(i-1)}\end{bmatrix}^T, & \text{if } i\neq 1\end{cases}$$

where the symbols g_{ij} denote the elements of the matrix $G(\theta)$. The parameters of the state–space description (1)–(2) can be estimated efficiently by solving

$$\hat{\theta}_i=\begin{cases}\arg\min_{\theta_i}\ \sum_k\left(y_{i,k}-\varphi_{i,k}^T\theta_i\right)^2, & \text{if } i=1\\ \arg\min_{\theta_i}\ \sum_k\left(y_{i,k}-\varphi_{i,k}^T\theta_i\right)^2+\gamma\theta_i^T\theta_i, & \text{if } i\neq 1.\end{cases}\tag{25}$$

The term $\gamma\theta_i^T\theta_i$ is introduced for $i>1$ to avoid singularities in the parameter estimation (Tikhonov regularization [25]). This was made necessary because of the parameters in the matrix $G(\theta)$, which were introduced in order to obtain linear error equations. It is known that (25) approaches the minimum norm solution as the regularization term γ diminishes. In simulations we have used $\gamma=1$, but smaller values do not lead to significant modifications. Thus for all practical purposes the solution used is a least–norm one.

The dynamics of ξ_k, given in (7), are determined by constant matrices $\underline{A}$ and $\underline{C}$, therefore the sequence $\varphi_{i,k}$ does not depend on θ, and (25) is a regularized linear least–squares problem which is guaranteed to have a solution. According to the results in Section III, ξ_k is computed directly using (21), and the parameters of (1)–(2) can be obtained without the explicit value of the matrix $E_{\mathbb{I}}(\theta)$, which would be needed only in the calculation of the state estimate $\hat{x}_k$ from ξ_k as in (13). The characteristic polynomial of $\underline{A}$, namely $\alpha(q)$, is a free design parameter of the estimation algorithm. We shall now explain how we intend to tune it.

B. Filter Tuning

We shall select the free parameters of $\alpha(q)$, denoted η in Section III-B, by means of derivative–free direct optimization. Consider a set $\Omega=\{\eta_1,\ldots,\eta_{n_\eta}\}$ of candidate values of η, and let $J(\eta,\mathcal{Z})$ be a functional which quantifies the performance of

the values $\eta \in \Omega$ given a data set $\mathcal{Z} = \{y_1, u_1, \ldots, y_N, u_N\}$ of length N. Pick the values of the filter parameters according to the formula

$$\eta^* = \frac{\sum_{r=1}^{n_\eta} \eta_r e^{-\mu J(\eta_r, \mathcal{Z})}}{\sum_{r=1}^{n_\eta} e^{-\mu J(\eta_r, \mathcal{Z})}}, \tag{26}$$

where η_r denotes the rth element of Ω and $\mu \in \mathbb{R}$ is an appropriately chosen positive constant. In formula (26), η^* represents the barycenter of the points η_r with weights $e^{-\mu J(\eta_r, \mathcal{Z})}$. By construction, the selected point η^* will be a convex combination of the points η_r, so the choice of Ω defines a convex set in which the search is to be performed. The elements of the Ω may be fixed, contain elements chosen according to some previous calculation, or obtained using some sort of randomization. The flexibility this method offers over simply choosing among the elements of Ω will be exploited in Section V-C to improve and speed up calculations.

The intuition behind taking the barycenter as a substitute for the hard–to–compute value of η which minimizes $J(\eta, \mathcal{Z})$ is that the points η_r which lead to low values of the cost criterion are given more weight than those that lead to high values. This is a direct optimization method: only the numerical values of J for a selected set of arguments need to be computed, and neither the derivatives of J nor a closed–form mathematical expression for the objective function J itself are required. Alternatives to the barycenter method, which was selected for its simplicity and robustness, are described in the literature [26].

The filter selection algorithm is as follows: for each candidate η_r the corresponding polynomial $\alpha(q)$ is constructed as described in Section III-B. Then the respective models are obtained by solving the linear least–squares problem (25), and the functional $J(\eta_r, \mathcal{Z})$ is computed for each $r \in \{1, \ldots, n_\eta\}$. The polynomial $\alpha(q)$ is found using the weighted average η^*. Choosing Ω with finite cardinality is enough to guarantee convergence of the search procedure.

If there is prior knowledge about the frequency range of the dynamics that should be modeled, such information can guide the choice of Ω. In the case of a "black–box" approach, one might simply adopt a set of candidate vectors η which correspond to the characteristic equation of discrete-time filters with a pair of dominant poles uniformly distributed inside the unit circle. Because of convexity, there is no risk of obtaining an η^* that leads to a polynomial $\alpha(q)$ with unstable roots. This strategy allows flexibility in choosing the functional $J(\cdot)$ to reflect the application. One particular choice based on the concept of variance–accounted–for is employed in Section V.

C. Model Structure Selection

Up to this point, the algorithm was described for a fixed list of the observability indices l. Here the determination of a suitable l is addressed. Before describing the methods we have in mind, it is important to recognize the existence of a developed literature concerning the estimation of the structural invariants of multivariable systems.

Structural identification for MIMO dynamic systems parameterized through quasi–canonical forms is more complex and difficult to implement than for SISO systems. Theoretical advances dating from the 1970s focus on methods for estimating the Kronecker indices [1], [27]. In the pioneering approach of [28], determination of structural indices of a controllable canonical form was treated using singularity tests on input–output data matrices, without requiring the construction of an intermediate parametric model. However, in addition to numerical difficulties caused by singularity tests, the method may fail if knowledge about the noise statistics is unavailable. The structural parameters of a canonical state–space representation can also be determined by testing the determinant of a certain matrix whose entries are the correlation functions of the output sequence [29].

The issues with the estimation of structural indices are threefold. First, estimation may be difficult to perform or prone to numerical and other errors caused by the fragility of algorithms that involve determining matrix rank. Second, the "true" process with well-defined observability indices is a Platonic object whose very existence is not altogether compatible with the realities of unmodeled high–order dynamics, nonlinearities, and time–variation. Finally, there are no uniquely defined, or properly canonical, forms for multivariable systems, which are best described by a set of overlapping quasi–canonical parameterizations. Thus it is questionable whether determining the "correct" invariants is crucial for the identification of real processes. In practice, one of the overlapping structures may give better results than the others. Although parameterizations capable of matching all processes of a given McMillan degree have been employed in the literature, testing more than one structure for each McMillan degree and choosing according to some performance criterion appears to offer quite satisfactory results, a fact that is borne out by the simulations, particularly those in Section V-B.

In light of the preceding arguments, we employ the following strategy to select the list of observability indices. Rather than trying to determine the model internal structure for a certain dynamic order n, the structure selection procedure starts from the list $l = \{1, \ldots, 1\}$, which corresponds to the simplest p-output model from a structural point of view (cases with outputs with observability index zero are of limited interest). Next, p new models are estimated using a set of linear regressions driven by the direct optimization method of Section IV-B, each one structured from a new list obtained by raising the index relating to one of the outputs. Then, one picks the list l which results in the best performance, according to a criterion $J(\cdot)$ established beforehand. The strategy is depicted in Fig. 1.

In order to keep the number of steps finite, an upper bound $n_{\max}$ to the McMillan degree is defined, and the model order is chosen over the interval between $p \leq n \leq n_{\max}$ as a compromise between complexity and performance. As the variance–accounted–for (VAF) is used in the present work as a performance measure, the lowest model order whose average VAF of the outputs is within 1% of the best attained one is selected.

This simple heuristic circumvents the problem posed by the number of the candidate structures, which increases combinatorially with model degree. The price is that not all possible lists of observability indices are considered and one might end up

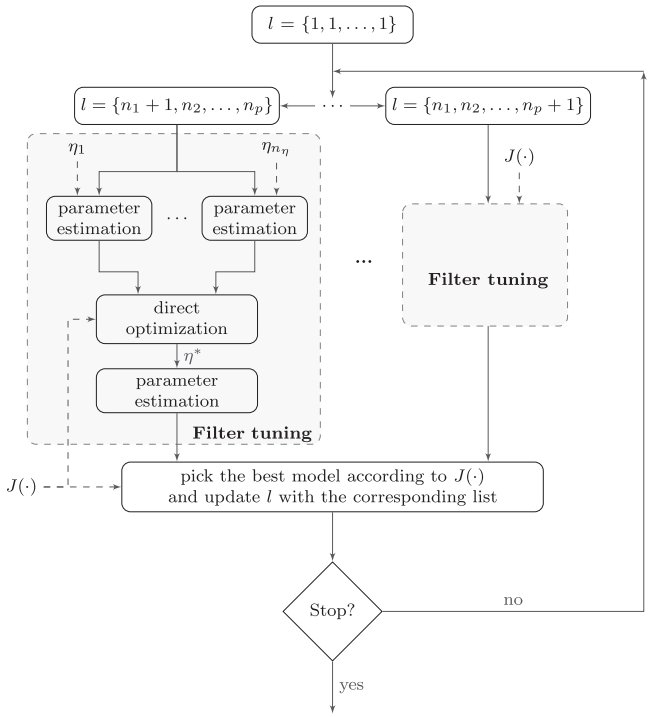

Fig. 1. Model structure selection, filter tuning, and parameter estimation.

with a more complex model than necessary. The effectiveness of the method is confirmed through a real data case study in Section V-C.

V. Simulation and Experimental Results

The performance of the MOLI-ZOFT identification algorithm is first tested using data generated from simulated models. The tests in V-A do not address structure selection—the identification models have a fixed McMillan degree and a fixed list of observability indices. In V-B, we used a few cases where the scheme with a fixed structure fails to give an acceptable result to test the structure selection algorithm. Data from a multivariable industrial process was used in V-C to test the complete algorithm and compare it with well-established estimation methods.

A. Simulation With Random Models

To cover a wide range of situations, a Monte Carlo experiment comprising 500 simulations was performed. In each simulation, the data set is generated using a randomly chosen, 3-input, 3-output stable model of McMillan degree n. The models are generated using the `drss` function of the Matlab Control Systems Toolbox$^{\mathrm{TM}}$ [30]; those that are not BIBO-stable were discarded. The models are excited with independent zero-mean Gaussian white noise sequences of unit variance; a new realization is generated for each simulation. The simulated outputs are also corrupted by Gaussian noise sequences colored by random autoregressive moving average filters of order n. The noise covariances are adjusted so that the signal-to-noise (SNR) ratio in each output is 5 in variance, or 7dB. In each run, the models are identified using $N = 500$ input and output observations.

The accuracy, computational efficiency, and robustness of the MOLI-ZOFT algorithms is evaluated in comparison with four methods.

- The PO-MOESP (Past Output-Multivariable Output-Error State-sPace) subspace method [5] (implemented via the software described in [31]).
- The expectation maximization (EM) algorithm developed in [9] and implemented in the University of Newcastle Identification Toolbox (UNIT) [10].
- Gradient-based search of models in the controllable canonical (CAN) form [3].
- Gradient-based search using the Data-Driven Local Coordinates (DDLC) parameterization [6].

The two gradient-based approaches are available in the widely used MATLAB System Identification Toolbox$^{\mathrm{TM}}$ [32] through the `pem` function, which was used with the property `Focus` set to three different options: "`Predicton`", "`Stability`" and "`Simulation`" (henceforth `pred`, `stab` and `sim`, respectively). The first incorporates np extra parameters to model the noise; the second forces the model to be stable;[3] and the third employs an output-error method. The property `nk` was set to `[1,1,1]`, in order to not consider the feedthrough term. The EM and gradient-based search methods terminate when the relative decrease of the cost at an iteration falls below 10^{-4} (the cost function is taken as the determinant of $\mathcal{E}^T\mathcal{E}$, where $\mathcal{E} \in \mathbb{R}^{N\times p}$ is the residual matrix), or after 40 iterations. These iterative methods were initialized using a CVA-weighted subspace estimate achieved using the `n4sid` function as discussed in [32]. The PO-MOESP block-size parameter was set to $2n$.

To measure accuracy, consider the variance-accounted-for criterion (VAF_i) [31]

$$\mathrm{VAF}_i = \max\left\{1 - \frac{\mathrm{var}\,(\mathbf{y}_i - \hat{\mathbf{y}}_i)}{\mathrm{var}\,(\mathbf{y}_i)}, 0\right\} \times 100\%,$$

where $\mathbf{y}_i$ is the observed ith output sequence and $\hat{\mathbf{y}}_i$ is its estimate, which is computed using only the deterministic part of each identified model. The operator $\mathrm{var}\,(\cdot)$ denotes the sample based variance estimate of the argument. The closer VAF_i is to 100%, the lower the energy of the modeling error in the ith output. To prevent disturbances from masking the validation results, the criterion VAF_i is computed using a noiseless data set, based on a different input realization.

For the MOLI-ZOFT algorithm, the list of observability indices is fixed at $l = \left\{\frac{n}{3}, \frac{n}{3}, \frac{n}{3}\right\}$. The design variable $\alpha(q)$ is obtained by the direct optimization procedure presented in IV-B. The set Ω is generated using three different damping ratios $\zeta = \{0.3, 0.575, 0.85\}$ and six cutoff frequencies $\omega_c = \{0.1\pi, 0.26\pi, 0.42\pi, 0.58\pi, 0.74\pi, 0.9\pi\}$, so that a wide frequency range (limited to the Nyquist frequency) is covered.

[3] By default, the option `Stability` restricts the magnitude of the poles to be less than or equal to 1, so the `pem` function can generate BIBO-unstable models.

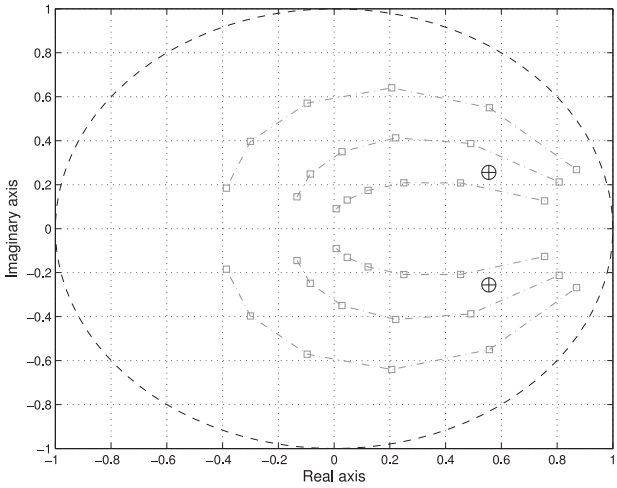

Fig. 2. Dominant poles of candidate filters.

TABLE I
ACCURACY: AVERAGE VAF

	$n=6$	$n=9$	$n=12$	$n=15$	$n=18$
MOLI–ZOFT	95.8	96.3	96.8	97.0	97.3
PO-MOESP	90.3	84.8	84.2	84.0	81.8
EM	96.0	97.2	96.8	96.0	94.2
$\text{CAN}_{\texttt{pred}}$	98.7	98.4	97.7	95.0	93.2
$\text{CAN}_{\texttt{stab}}$	98.7	98.7	98.6	97.6	97.2
$\text{CAN}_{\texttt{sim}}$	97.3	95.6	95.9	95.4	95.2
$\text{DDLC}_{\texttt{pred}}$	98.7	98.9	98.6	98.3	97.4
$\text{DDLC}_{\texttt{stab}}$	98.7	98.9	98.6	98.2	97.8
$\text{DDLC}_{\texttt{sim}}$	96.9	94.2	92.9	90.1	89.0

Hence, $n_\eta = 18$ candidate prefilters are considered in the computation of $\alpha(q)$. The dominant poles associated to each candidate are depicted in Fig. 2. The dominant poles resulting from the direct optimization in one among the 500 runs of the Monte Carlo experiment, for $n = 6$, are shown with "$\oplus$". The functional $J(\eta, \mathcal{Z})$ is based on an average of the VAF_i for the outputs

$$J(\eta, \mathcal{Z}) = \frac{1}{p}\sum_{i=1}^{p} \min\left\{\frac{\text{var}\left(\mathbf{y}_i - \hat{\mathbf{y}}_i(\eta)\right)}{\text{var}\left(\mathbf{y}_i\right)}, 1\right\}.$$

To assess the accuracy of the algorithms, we consider the average value of $\frac{1}{p}\sum_{i=1}^{p}\text{VAF}_i$ over 500 trials, namely, $\frac{1}{500}\sum_{j=1}^{500}\frac{1}{p}\sum_{i=1}^{p}\text{VAF}_i$, which is reported in Table I. The MOLI-ZOFT method provides performance comparable to the best results, which were achieved by the gradient-based search method with options `Prediction` and `Stability`. For higher dimensional models, the trend appears favorable to the new method. This is not unexpected: because of numerical complexity, performance of algorithms using nonlinear optimization problems tends to degrade in high-dimensional problems.

For the purpose of robustness analysis, a failure is deemed to occur when $\text{VAF}_i \leq 50\%$ for some output i. Table II shows the number of failures of each method.

TABLE II
ROBUSTNESS: NUMBER OF IDENTIFICATION FAILURES

	$n=6$	$n=9$	$n=12$	$n=15$	$n=18$
MOLI-ZOFT	0	1	0	0	0
PO-MOESP	84	149	131	109	124
EM	17	15	22	31	57
$\text{CAN}_{\texttt{pred}}$	0	7	12	42	63
$\text{CAN}_{\texttt{stab}}$	0	2	0	3	2
$\text{CAN}_{\texttt{sim}}$	4	14	12	7	10
$\text{DDLC}_{\texttt{pred}}$	0	0	0	0	6
$\text{DDLC}_{\texttt{stab}}$	0	0	1	0	0
$\text{DDLC}_{\texttt{sim}}$	6	20	24	43	57

TABLE III
ROBUSTNESS: NUMBER OF UNSTABLE ESTIMATED MODELS

	$n=6$	$n=9$	$n=12$	$n=15$	$n=18$
MOLI-ZOFT	0	0	0	0	0
PO-MOESP	30	70	70	69	90
EM	3	5	8	7	17
$\text{CAN}_{\texttt{pred}}$	0	3	11	19	43
$\text{CAN}_{\texttt{stab}}$	0	3	11	11	33
$\text{CAN}_{\texttt{sim}}$	52	71	78	64	57
$\text{DDLC}_{\texttt{pred}}$	0	1	4	2	6
$\text{DDLC}_{\texttt{stab}}$	0	0	7	3	9
$\text{DDLC}_{\texttt{sim}}$	71	118	188	269	322

The MOLI-ZOFT algorithm and gradient-based search applied to the DDLC parameterization provide the best results, and the former is more robust in higher dimensional cases.

A different type of failure occurs when the estimated model is unstable, although the data comes from a BIBO stable process. Even if the modeling error is small, an unstable plant is an object qualitatively different from a stable one. A controller designed on the basis of an unstable model for a plant that is in fact stable is likely to be impractical. Table III shows that MOLI-ZOFT does not result in models with spurious unstable poles. This was the purpose of working with match-point controllable models.

The other methods are less robust in this respect, especially for larger dimensions. Even when the stability constraint is enforced by means of the setting `Focus`, gradient-based search ($\text{CAN}_{\texttt{stab}}$ and $\text{DDLC}_{\texttt{stab}}$) still occasionally return models with poles on the unit circle.

Remark 3: In these experiments the output data were corrupted by colored noise generated by stable random filters. Some methods are not designed to work well in this case; for example, output-error methods such as $\text{CAN}_{\texttt{sim}}$ and $\text{DDLC}_{\texttt{sim}}$ are unable to achieve minimum variance estimates under colored noise conditions. The scenario seems particularly challenging for PO-MOESP. This may be partly explained by design parameters inherent to subspace methods. The weighting matrix W_1 in [33], which influences data prefiltering, is taken as the identity in PO-MOESP subspace algorithm.

As a rule, the price of better accuracy is increased computational complexity, but MOLI-ZOFT turns out to be the fastest in virtually all investigated scenarios, in some cases by a whole order of magnitude, while still achieving a competitive accuracy.

TABLE IV
AVERAGE RUNTIME IN SECONDS

	$n=6$	$n=9$	$n=12$	$n=15$	$n=18$
MOLI-ZOFT	0.08	0.12	0.13	0.14	0.15
PO-MOESP	0.21	0.30	0.42	0.56	0.66
EM	0.43	0.54	0.62	1.36	1.31
CAN_{pred}	0.61	1.34	2.55	3.58	4.27
CAN_{stab}	0.61	1.16	1.98	3.72	4.36
CAN_{sim}	0.61	1.14	2.11	3.05	4.13
$\text{DDLC}_{\text{pred}}$	0.54	1.15	2.33	3.65	4.71
$\text{DDLC}_{\text{stab}}$	0.53	0.97	1.75	3.64	4.74
DDLC_{sim}	0.53	0.96	1.91	3.03	4.51

TABLE V
ROBUSTNESS: IDENTIFICATION FAILURES WITH SMALLER DATASET

	$n=6$	$n=9$	$n=12$	$n=15$	$n=18$
MOLI-ZOFT	2	2	0	0	1
PO-MOESP	110	183	205	280	–
EM	10	27	46	123	212
CAN_{pred}	3	19	44	96	108
CAN_{stab}	0	1	5	13	11
CAN_{sim}	20	28	29	17	45
$\text{DDLC}_{\text{pred}}$	0	0	11	11	6
$\text{DDLC}_{\text{stab}}$	0	0	8	6	27
DDLC_{sim}	21	66	91	99	111

TABLE VI
ROBUSTNESS: UNSTABLE ESTIMATED MODELS WITH SMALLER DATASET

	$n=6$	$n=9$	$n=12$	$n=15$	$n=18$
MOLI-ZOFT	0	0	0	1	0
PO-MOESP	47	91	109	148	–
EM	2	7	15	44	73
CAN_{pred}	3	16	25	52	67
CAN_{stab}	3	12	17	27	41
CAN_{sim}	77	103	104	99	114
$\text{DDLC}_{\text{pred}}$	1	4	8	15	25
$\text{DDLC}_{\text{stab}}$	2	11	10	16	21
DDLC_{sim}	90	206	285	354	420

Table IV presents the average runtime of each identification algorithm as a function of the McMillan degree adopted in the generation of the simulation models.

When the number of available observations is smaller, the difference in numerical robustness between MOLI-ZOFT and the other algorithms becomes more pronounced, especially so in tests with higher order plants. The number of failures observed in a test with $N = 250$ samples is reported in Table V. The implementation of PO-MOESP we tested [31] was unable to identify models of order greater than 16 with this smaller dataset.

While all methods tend to exhibit more failures when identification is performed with fewer samples, performance of MOLI-ZOFT deteriorated primarily in cases of lower order models.

Unstable estimated models remain very rare when employing MOLI-ZOFT with $N = 250$, as shown in Table VI. We also performed a few tests with $N = 1000$ and $N = 2000$ samples. We do not present the results here because performance improves only marginally in most aspects and tests, and the conclusions concerning relative accuracy, robustness, and runtime remain qualitatively the same.

TABLE VII
STRUCTURE SELECTION ALGORITHM FOR FAILURE CASE WITH $n = 9$, DATASET OF TABLE V

List l	VAF_1	VAF_2	VAF_3	$\frac{1}{3}\sum_{i=1}^{3}\text{VAF}_i$
$\{5,2,2\}$	52.2%	63.6%	54.3%	56.7%
$\{3,3,3\}$	40.0%	56.8%	47.0%	49.7%
$\{4,2,3\}$	48.5%	59.0%	52.9%	53.5%
$\{4,2,4\}$	67.3%	74.0%	71.0%	70.8%
$\{5,2,4\}$	73.0%	77.5%	75.4%	75.3%
$\{5,2,5\}$	75.2%	79.4%	75.4%	77.5%
$\{5,3,5\}$	73.8%	79.8%	77.1%	76.9%

The analysis carried out from data generated by the Monte Carlo experiment confirms that the design goals for the identification method were met: numerical robustness, accuracy and computational efficiency. The simulations showed that MOLI-ZOFT is robust, as efficient as the subspace methods, and as accurate as nonlinear iterative techniques which are available from commercial and academic packages.

B. Structure Selection Tests

In some of the tests above the results obtained with fixed structure were unsatisfactory. We used these cases as tests to the structure selection algorithm.

Remark 4: The list $\{\frac{n}{3}, \frac{n}{3}, \frac{n}{3}\}$ works well in the majority of the random trials because generically the dimensions of the observability subspaces are about equal (see Corollary 5.4 in [34]). Nevertheless, there are cases when other lists result in better models.

In the one case shown in Table II when MOLI-ZOFT gave unacceptable results, the VAF_i values were 34.4%, 93.8% and 86.2%. In this case, we applied the structure selection approach of Section IV-C, and instead of $\{3,3,3\}$, we obtained for $n = 9$ the list of indices $\{4,2,3\}$, resulting in $\text{VAF}_1 = 56.2\%, \text{VAF}_2 = 91.7\%$ and $\text{VAF}_3 = 92.1\%$, which are better than the failure threshold.

Likewise, in Table V the failures for $n = 6$ and $n = 18$ could have been avoided applying the structure selection algorithm, which resulted in models with list of observability indices $\{3,1,2\}$ and $\{8,3,7\}$. Concerning the failure for $n = 9$, the list $\{4,2,3\}$ obtained with the heuristic from Section IV-C would result in a single failure, although there is a list with the same McMillan degree, namely $\{5,2,2\}$, for which such failure would not occur. In this particular case, the strategy was unable to find the best choice for the list of observability indices.

In practice, when the performance is not satisfactory, an increase in model order is advisable. Table VII compares the VAF results for lists $\{5,2,2\}, \{3,3,3\}$, and for the list sequence determined by the structure selection algorithm for $9 \leq n \leq 13$. Our algorithm overlooks list $\{5,2,2\}$ and only finds a satisfactory model for McMillan degree 12. This result exemplifies the disadvantage of not testing all possible lists.

With the data corresponding to the unstable estimated model in Table VI for $n = 15$, the structure selection algorithm picks

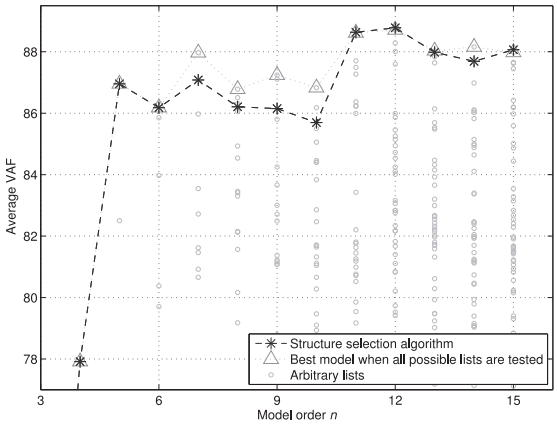

Fig. 3. Model accuracy for each candidate list of indices, for McMillan degree from 4 to 15.

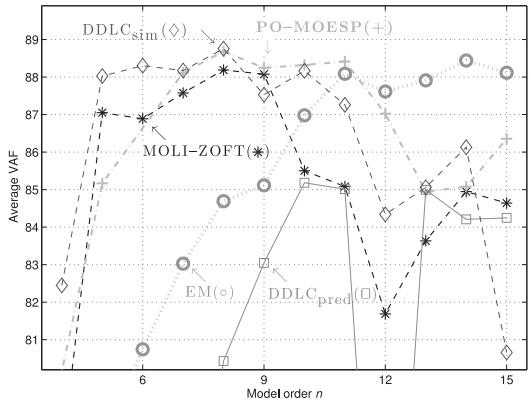

Fig. 4. Average VAF of the estimated outputs by models having orders $4 \leq n \leq 15$.

the list $l = \{3, 10, 2\}$, resulting in a stable model with acceptable VAF despite the smaller number of measurements available.

C. A Real-Data Case Study

The MOLI-ZOFT algorithm was applied to the identification of a real four-stage industrial evaporator for reducing the water content of a product. The data set is available from the Daisy database [35]. The system inputs are feed flow (u_1), vapor flow to the first evaporator stage (u_2) and cooling water flow (u_3). The three outputs are the dry matter content (y_1), the flow (y_2) and the temperature of the outcoming product (y_3). The evaporator data set comprises more than 6000 samples collected on two different days, but only 500 samples were used for estimation and another 500 for validation, a situation more commonly encountered in industrial practice.

Direct optimization is carried out considering a set Ω of 18 candidates to η which are generated by $\zeta = \{0.6, 0.7, 1.1\}$ and $\omega_c = \{0.02\pi, 0.0355\pi, 0.0631\pi, 0.1122\pi, 0.1995\pi, 0.3548\pi\}$. Due to the low-pass nature of the process, in contrast to Section V-A, the poles of the candidate filters are concentrated in low frequencies, to enhance model fit in this range. Model structure selection is performed using the procedure described in Section IV-C. Unlike the simulation test in which the model order was supposed to be known beforehand, for the test with real data a modified randomized version of the direct optimization is used. Instead of picking all elements of the set Ω, at each step the barycenter is computed using seven values of η: the barycenter computed in the previous step, three elements chosen at random from the original Ω, and three other random candidates centered on the barycenter, with normal distribution and 20% of variance of the sets ζ and ω_c.

Considering candidate structures up to $n = 15$ and starting from $l = \{1, 1, 1\}$, the following list sequence was obtained: $\{1, 2, 1\}$, $\{1, 3, 1\}$, $\{1, 4, 1\}$, $\{1, 5, 1\}$, $\{2, 5, 1\}$, $\{3, 5, 1\}$, $\{3, 6, 1\}$, $\{4, 6, 1\}$, $\{4, 7, 1\}$, $\{4, 8, 1\}$, $\{5, 8, 1\}$ and $\{6, 8, 1\}$. The average VAF computed using the validation dataset for each of these models is presented in Fig. 3. To evaluate the effectiveness of the search, the performance achieved by all possible lists of observability indices (i.e., all combinations of p indices n_i, such that $\sum_{i=1}^{p} n_i = n$) are shown using the symbols "○". For each McMillan degree, the result achieved with the best list l of all possibilities is highlighted by $\triangle$.

The structure selection algorithm was able to find the best list for most McMillan degrees in this test. Even in cases in which the algorithm does not select the best list, the performance achieved with the selected list and the best choice are very similar. The discrepancy between degrees 7 to 10 occurs because among all the possible lists for $n = 7$, the higher average VAF (88%) is attained by $l = \{3, 3, 1\}$, while our selection algorithm picked $l = \{1, 5, 1\}$. As the lists are different, so is the resulting α. From there on, direct optimization is calculated with different candidates of η selection algorithm structure. According to the accuracy of the models estimated using the structure selection procedure (indicated by * in Fig. 3), increases in the model complexity are effective up to order 11, for which $l = \{4, 6, 1\}$.

We compared MOLI-ZOFT to other multivariable state-space identification methods using the evaporator data. The average VAF provided by each method of order 4 to 15 is reported in Fig. 4. The method using controllable canonical forms was not superior to the other algorithms and was left out for ease of visualization. The results with method DDLC$_{\texttt{stab}}$ are also not shown because they are almost identical to those with DDLC$_{\texttt{pred}}$. Performance of PO-MOESP for $n = 6$ is not depicted because an unstable model was obtained.

The average VAF achieved with the method proposed is higher than 86% for virtually all models with $n \geq 5$. This result is comparable to the accuracy achieved by PO-MOESP and DDLC$_{\texttt{sim}}$, the algorithms which achieved the best performance for this dataset. MOLI-ZOFT does not seem prone to overmodeling, as the performance does not degrade for $n \geq 11$. As a practical matter one may select the model order considering the trade-off between complexity and accuracy. The lowest order models whose VAF is within 1% of the best VAF are presented in Table VIII. We see that MOLI-ZOFT furnishes an accurate model of reasonable complexity.

Remark 5: The results obtained with this real data set differ somewhat from the ones seen in Section V-A. Here the output-error method (DDLC$_{\texttt{sim}}$) was superior than DDLC$_{\texttt{pred}}$, PO-MOESP was much more competitive, and the EM method

TABLE VIII
SELECTED MODEL ORDER AND VAF ATTAINED BY EACH METHOD

	Order n	Average VAF
MOLI-ZOFT	11	88.6
PO-MOESP	8	88.7
EM	11	88.1
$\text{DDLC}_{\text{pred}}$	10	85.2
DDLC_{sim}	8	88.8

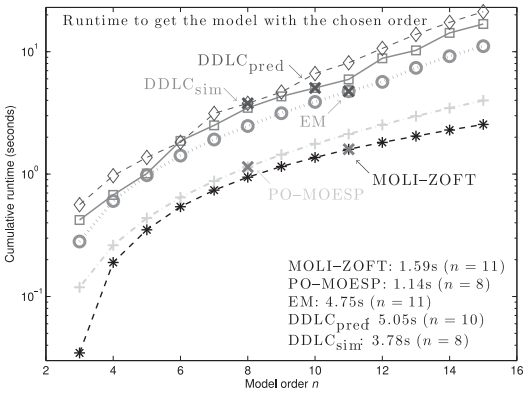

Fig. 5. Cumulative runtime (in seconds) spent to estimate models up to a specific order.

performed better as well. These differences are due to process and noise spectrum profiles in the evaporator data. A deeper comparison of the various competing methods is outside the scope of this paper.

The identification algorithms were also applied to other data ranges of 500 samples (recall that only a fraction of the available data was used in the previous test). The structure selection strategy works equally well in finding a suitable list of observability indices, which is not necessarily the same for each piece of data. Despite variations from the results reported in Table VIII, MOLI-ZOFT is comparable or superior to the other approaches.

We analyzed the computational effort needed to identify a real process in which the model order is not a priori known. Fig. 5 presents the cumulative runtime necessary to estimate models from $n = p$ to a certain order. The time spent to estimate the models with suitable orders is indicated by crosses.

Even with the increase in computational effort due to the structure selection procedure (remember that p different lists are considered for each n), MOLI-ZOFT remains the fastest algorithm. In addition, the computation time spent by the proposed method to obtain a model with $n = 11$ is comparable to the time needed by the subspace approach to obtain a model with $n = 8$, when only a single block-size value (equal to $2n$) is assessed for each n. In comparison to the computational complexity of algorithms based on nonlinear optimization, the performance of MOLI-ZOFT is even more compelling.

VI. CONCLUSION

A novel multivariable identification method based on overlapping quasi-canonical parameterizations is proposed. The parameters of the state-space model matrices enter linearly in the regression equation, and estimation can be performed via linear least-squares. No assumptions concerning the noise statistics are necessary. Instead, the poles of a prefilter, which is intrinsic to the parameterization adopted, are determined by means of derivative-free optimization, using an objective function which can be freely chosen according to the needs of the application in mind, even if its dependence on the filter parameters happens to follow a formula that is unknown or unwieldy. An effective heuristic procedure is introduced to find a suitable set of invariants, without the need to test all possible list of observability indices.

The MOLI-ZOFT methodology splits the overall identification problem into three stages, namely, parameter estimation, filter tuning and model structure selection. It proved to be as effective as, and more robust than, other multivariable identification algorithms described in the literature. In experiments with randomly generated models and with real data, MOLI-ZOFT showed the best balance between model complexity, accuracy and computational requirements. When smaller measurement data sets are available, we have found the robustness of the method particularly remarkable.

The most challenging identification problems appear when the model is to be determined in real time. Nonlinear optimization and iterative methods can become problematic. This is where our methodology's adaptive control heritage shows its mettle. A real-time extension of the MOLI-ZOFT framework would go along the following lines. First, the designer would select time scales for computation of identification models. At the faster rate, model parameters would be estimated using a recursive version of the least-squares technique described in Section IV-A. Periodically, a figure of merit of the estimated model, such as the average VAF over a time window, would be evaluated, and used to tune the filter parameters via direct optimization in a manner analogous to Section IV-B. The fact that a recursive formula for the barycenter is simple to obtain is one of the main reasons for our choice of this derivative-free optimization method over many others available in the literature. At a lower rate, model structure would be reevaluated and changed using a modified version of the algorithm in Section IV-C. Through the use of recursive optimization for both the filter design and the parameter estimation stages, computation time can be cut dramatically, both for real-time and batch applications. This is indeed the goal of the randomized strategy employed in Section V-C. Another application that benefits from the techniques developed here, and particularly from the use of predictors whose dynamic behavior is parameter-independent, is the estimation of continuous-time models using sampled data. Further results will be reported in a forthcoming paper. We have also collaborated in employing the techniques developed here in a much more challenging problem, the identification of nonlinear and time-varying systems using linear parameter-varying models. Our state-space representations and optimization techniques nicely complement machine learning approaches based on the Least-Squares Support Vector Machine (LS-SVM) framework, with results reported in [36].

The methodology we advocate does require the use of techniques that are perhaps not widespread in the identification literature, namely direct optimization and the quasi-canonical structures of Section III. Rather than mourn the loss of simplicity, and blame the steeper learning curve that one needs to climb in order to describe model structures based on the set of observability indices, we should rejoice in the understanding that the improved performance and data economy is brought to us by the richly developed literature on realization theory and quasi-canonical forms for linear systems.

Acknowledgment

The authors wish to thank the reviewers for their detailed reading of the submission and their probing questions.

References

[1] M. Gevers, "A personal view of the development of system identification," *IEEE Control Syst. Mag.*, vol. 26, no. 6, pp. 93–105, Dec. 2006.
[2] B. Ninness, "Some system identification challenges and approaches," in *Proc. 15th IFAC Symp. Syst. Ident.*, Saint-Malo, France, 2009, pp. 1–20.
[3] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1999.
[4] T. McKelvey, "Identification of state–space models from time and frequency data," Ph.D. dissertation, Linköping University, Linköping, Sweden, 1995.
[5] M. Verhaegen and V. Verdult, *Filtering and System Identification: A Least Squares Approach*. London, U.K., Cambridge University Press, 2007.
[6] T. McKelvey, A. Helmersson, and T. Ribarits, "Data driven local coordinates for multivariable linear systems and their application to system identification," *Automatica*, vol. 40, no. 9, pp. 1629–1635, 2004.
[7] T. Ribarits, M. Deistler, and T. McKelvey, "An analysis of the parametrization by data driven local coordinates for multivariable linear systems," *Automatica*, vol. 40, no. 5, pp. 789–803, 2004.
[8] A. Wills and B. Ninness, "On gradient–based search for multivariable system estimates," *IEEE Trans. Autom. Control*, vol. 53, no. 1, pp. 298–306, 2008.
[9] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, 2005.
[10] B. Ninness, A. Wills, and A. Mills, "UNIT: A freely available system identification toolbox," *Control Engineering Practice*, vol. 21, no. 5, pp. 631–644, 2013.
[11] N. F. Al-Muthairi, S. Bingulac, and M. Zribi, "Identification of discrete-time MIMO systems using a class of observable canonical-form," *IEE Proc. Control Theory Appl.*, vol. 149, no. 2, pp. 125–130, 2002.
[12] G. Mercère and L. Bako, "Parameterization and identification of multivariable state-space systems: A canonical approach," *Automatica*, vol. 47, no. 8, pp. 1547–1555, 2011.
[13] P. Stoica and M. Jansson, "MIMO system identification: State-space and subspace approximations versus transfer function and instrumental variables," *IEEE Trans. Signal Processing*, vol. 48, no. 11, pp. 3087–3099, 2000.
[14] A. S. Morse and F. M. Pait, "MIMO design models and internal regulators for cyclicly switched parameter–adaptive control systems," *IEEE Trans. Autom. Control*, vol. 39, no. 9, pp. 1809–1818, 1994.
[15] G. Pillonetto, F. Dinuzzo, T. Chen, G. D. Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, pp. 657–682, 2014.
[16] T. Chen and L. Ljung, "Implementation of algorithms for tuning parameters in regularized least squares problems in system identification," *Automatica*, vol. 49, no. 7, pp. 2213–2220, Jul. 2013.
[17] R. A. Romano and F. Pait, "Linear multivariable identification using observable state space parametrizations," in *Proc. 52nd IEEE Conf. Decision Control*, Firenze, Italy, Dec. 2013, pp. 1429–1434.
[18] F. Pait, "Reading Wiener in Rio," in *Proc. IEEE Conf. Norbert Wiener 21st Century*, 2014, pp. 1–4.
[19] R. A. Romano and F. Pait, "Direct filter tuning and optimization in multivariable identification," in *Proc. 53rd IEEE Conf. Decision Control*, Los Angeles, CA USA, Dec. 2014, pp. 1798–1803.
[20] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*. Dordrecht, The Netherlands, Kluwer Academic Publishers, 2004.
[21] D. G. Luenberger, "Canonical forms for linear multivariable systems," *IEEE Trans. Autom. Control*, vol. AC-12, pp. 290–293, 1967.
[22] P. Brunovský, "A classification of linear controllable systems," *Kybernetika*, vol. 6, pp. 173–188, 1970.
[23] A. S. Morse, "Representations and parameter identification of multi-output linear systems," in *Proc. 13th IEEE Conf. Decision Control*, Dec. 1974, pp. 301–306.
[24] T. Kailath, *Linear Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
[25] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. Philadelphia, PA USA: SIAM, 1998.
[26] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*, ser. MPS-SIAM series on optimization. Philadelphia, PA USA: SIAM, 2009.
[27] A. Królikowski, "Model structure selection in linear system identification —survey of methods with emphasis on the information theory approach," , Eindhoven University of Technology, Eindhoven, The Netherlands, Tech. Rep. 82-E-126, 1982.
[28] R. Guidorzi, "Canonical structures in the identification of multivariable systems," *Automatica*, vol. 11, pp. 361–374, 1975.
[29] H. El-Sherief, "Multivariable system structure and parameter identification using the correlation method," *Automatica*, vol. 17, no. 3, pp. 541–544, 1981.
[30] *Control Systems Toolbox™ User's Guide, Version 9.1*, The MathWorks, Inc., Natick, MA, 2011.
[31] M. Verhaegen, V. Verdult, and N. Bergboer, *Filtering and System Identification: An Introduction to Using Matlab Software*, Delft, The Netherlands, Delft University of Technology, 2007.
[32] L. Ljung, *System Identification Toolbox™ User's Guide, Version 7.4*, The MathWorks, Inc., Natick, MA, 2011.
[33] P. V. Overschee and B. D. Moor, "A unifying theorem for three subspace system identification algorithms," *Automatica*, vol. 31, no. 12, pp. 1853–1864, 1995.
[34] W. M. Wonham, *Linear Multivariable Control: A Geometric Approach*, 3rd ed., ser. Applications of mathematics. Berlin, Germany, Springer-Verlag, 1985.
[35] B. L. R. De Moor, "DaISy: Database for the identification of systems," Dept. Elect. Eng., ESAT/STADIUS, KU Leuven. Leuven, Belgium. http://homes.esat.kuleuven.be/~smc/daisy/, Feb. 2008.
[36] R. A. Romano, P. L. dos Santos, F. Pait, T.-P. A. Perdicoúlis, and J. A. Ramos, "Machine learning barycenter approach to identify LPV state-space models," in *Proc. Amer. Control Conf.*, Boston, MA USA, Jul. 2016, pp. 6351–6356.

Rodrigo Alvite Romano (M'11) received the B.S. degree in electrical engineering from Faculdade de Engenharia Industrial, São Bernardo do Campo, Brazil, and the the M.Sc. and D.Sc. degrees in electrical engineering from Escola Politécnica da Universidade de São Paulo, in 2006 and 2010, respectively.

He is an Associate Professor at the Instituto Mauá de Tecnologia, São Caetano do Sul, Brazil. His current research interests include system identification and control of dynamic systems.

Felipe Pait (SM'02) received the M.S. degree in electrical engineering from the University of São Paulo, São Paulo, Brazil and received the Ph.D. degree from Yale University, New Haven, CT, in 1993.

He is an Associate Professor at the University of S Paulo, Brazil, having dedicated substantial efforts to curriculum reform and multidisciplinary engineering education initiatives. He has worked on adaptive control and applications. Currently he is interested applying randomized optimization algorithms to classical open questions of adaptive control design.