

On Gradient-Based Search for Multivariable System Estimates

Adrian Wills and Brett Ninness, *Member, IEEE*

Abstract—This paper addresses the design of gradient-based search algorithms for multivariable system estimation. In particular, the paper here considers so-called “full parametrization” approaches, and establishes that the recently developed “data-driven local coordinate” methods can be seen as a special case within a broader class of techniques that are designed to deal with rank-deficient Jacobians. This informs the design of a new algorithm that, via a strategy of dynamic Jacobian rank determination, is illustrated to offer enhanced performance.

Index Terms—Gradient-based search (GBS), parameter estimation, system identification.

I. INTRODUCTION

IN THE FIELD of dynamic system identification, the so-called maximum likelihood (ML) principle and its relations, such as prediction error (PE) techniques, play a key role. There are several reasons for this, including the existence of a large body of supporting theory, the very general applicability of ML/PE methods, and their statistical optimality [2], [9], [12], [21].

Despite these advantages of ML/PE methods, their practical deployment is not always straightforward. This is largely due to the nonconvex optimization problems that are often implied. Typically, these are solved via a gradient-based search (GBS) strategy such as a Newton or similar type method [5], [12], [21].

The success of this sort of approach depends on the curvature of the associated cost function being optimized, and this is further dependent on the chosen system parametrization. Selecting the latter can be difficult, particularly in the multivariable case where the cost contours resulting from natural canonical state–space parametrizations imply poor numerical conditioning during the GBS [4], [17].

Indeed, the possibility of avoiding these parametrization-based difficulties is one of the key reasons for the recent intense interest in state–space-subspace-based system identification (4SID) methods [10], [22], [24]. With these techniques, every element of every matrix in the state–space model is estimated, which this paper terms a “fully parametrized” model structure.

More recently, the use of these fully parametrized structures has been investigated in the context of the GBS for ML/PE estimates [1], [11], [15], [18], [23]. An essential point of this paper

is to recognize that a full parametrization is an overparametrization, but that a minimal parametrization that is (locally) linearly related to the full parametrization can be simply derived.

These local representations have been dubbed “data-driven local coordinates” (DDLDC). Their employment reduces computational requirements, and also (usually) avoids rank deficiency of the PE Jacobian. This simplifies the computation of search directions. These features render DDLDC-based gradient search as a very effective means for finding ML/PE estimates of multivariable systems, and, indeed, it has become the default method for multivariable system estimation implemented in the widely used Matlab System Identification Toolbox [13].

This paper is also directed at the development of GBS methods for the ML/PE estimation of multivariable systems. Full parametrizations are also employed, and attention is focused on strategies for computing search directions that are “robust” to poor numerical conditioning and rank deficiency of the Jacobian.

This effectively involves discarding elements of the Jacobian matrix that lie in its own kernel. A first main result of this paper is that if this discarding is implemented whereby a certain fixed dimensional subspace of the Jacobian is eliminated in the computation of a search direction, then the ensuing direction is identical to that obtained via DDLDC methods, although by quite different means.

This result may be of independent interest in terms of providing insight into the search mechanism inherent to DDLDC techniques. However, its main significance here is in suggesting an extended approach wherein an DDLDC parametrization is used in a first stage in order to minimize computation load. A second stage is then introduced, wherein search directions are computed via *dynamic* Jacobian rank determination.

The utility of this approach is demonstrated by simulation-based empirical analysis over a range of scenarios.

II. PROBLEM SETTING

This paper considers the problem of system estimation using the following innovations-form state–space model structure:

$$\begin{bmatrix} x_{t+1} \\ y_t \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} + \begin{bmatrix} K \\ I \end{bmatrix} v_t. \quad (1)$$

Here, $u_t \in \mathbf{R}^m$ is the observed input, $y_t \in \mathbf{R}^p$ is the observed output, $v_t \in \mathbf{R}^p$ is a zero mean independent identically distributed (i.i.d.) stochastic process that models measurement corruptions, and the state $x_t \in \mathbf{R}^n$.

In order to compute estimates, this paper will employ a *full parametrization* of the system matrices in (1). Specifically, this paper will address the estimation of a parameter vector $\theta \in \mathbf{R}^{n_\theta}$

Manuscript received May 8, 2006. Recommended by Associate Editor H. Hjalmarsson.

The authors are with the School of Electrical Engineering and Computer Science, The University of Newcastle, Newcastle, N.S.W. 2308, Australia (e-mail: adrian.wills@newcastle.edu.au; brett.ninness@newcastle.edu.au).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2007.914953

given as

$$\theta \triangleq [\text{vec}\{A\}^T, \text{vec}\{B\}^T, \text{vec}\{C\}^T, \text{vec}\{D\}^T, \text{vec}\{K\}^T]^T. \quad (2)$$

Here, the $\text{vec}\{\cdot\}$ operator is one which forms a vector from a matrix by stacking its columns on top of one another. Via this and the innovations form of the structure (1), the steady-state mean square optimal one-step-ahead predictor $\hat{y}_{t|t-1}(\theta)$ associated with a model parametrized by θ can be simply expressed as [12]:

$$\begin{aligned} \hat{x}_{t+1|t} &= (A - KC)\hat{x}_{t|t-1} + (B - KD)u_t + Ky_t \\ \hat{y}_{t|t-1}(\theta) &= C\hat{x}_{t|t-1} + Du_t. \end{aligned} \quad (3)$$

Therefore, with the assumption that $\{v_t\}$ is Gaussian distributed as¹

$$v_t \sim \mathcal{N}(0, \sigma^2 I_p), \quad \sigma^2 \in \mathbf{R}^+ \quad (4)$$

and neglecting constant terms that are immaterial to the estimation process, the associated log likelihood function for the data is given as

$$L(\theta, \sigma^2) = -\frac{Np}{2} \log \sigma^2 - \frac{1}{\sigma^2} \|e(\theta)\|_2^2. \quad (5)$$

Here, the PE vector $e(\theta)$ is defined as

$$e(\theta) \triangleq [y_1^T - \hat{y}_{1|0}^T(\theta), \dots, y_N^T - \hat{y}_{N|N-1}^T(\theta)]^T. \quad (6)$$

Notice that, according to (5), there is an essential decoupling between the estimation of σ^2 and the elements of the parameter vector θ defined in (2). Namely, under the model structure (1), the ML estimate $\hat{\theta}$ is given as an element satisfying

$$\hat{\theta} \in \{\theta \in \mathbf{R}^{n_\theta} : \|e(\theta)\|_2 \leq \|e(\bar{\theta})\|_2, \forall \bar{\theta} \in \mathbf{R}^{n_\theta}\}. \quad (7)$$

If the Gaussian assumption (4) is violated, then the criterion (7) will no longer yield an ML solution. However, it will still specify a minimum PE norm estimate that will, asymptotically in observed data length, possess statistical properties that are closely related to those of an ML solution [12, Ch. 9], and that has contributed to its widespread adoption.

Balancing these attractive features, $\hat{\theta}$ defined by (7) cannot be specified in closed form due to the nonlinear dependence of $e(\theta)$ on θ . In recognition of this, the previous work [1], [11], [15], [18], [23] has focused on this problem of finding minima of $\|e(\theta)\|$, and has explored a gradient search approach. This paper is also directed at studying these methods, and seeks to propose, analyze, and empirically substantiate effective variants of them.

III. GRADIENT-BASED SEARCH METHODS

Gradient search strategies have long been employed in a wide variety of system identification applications [12]. These methods are commonly motivated ([5, Ch. 10]) by the use of a linear approximation of $e(\theta)$ about a current guess θ_k of a minimizer

according to

$$e(\theta) \approx e(\theta_k) + J(\theta_k)(\theta - \theta_k) \quad (8)$$

where $J(\theta_k)$ denotes the Jacobian matrix

$$J(\theta_k) \triangleq \left. \frac{\partial e(\theta)}{\partial \theta} \right|_{\theta=\theta_k}. \quad (9)$$

Here, according to (6) and with θ_i denoting the i th element in θ

$$\frac{\partial e(\theta)}{\partial \theta_i} = - \left[\frac{\partial \hat{y}_{1|0}^T}{\partial \theta_i}, \dots, \frac{\partial \hat{y}_{N|N-1}^T}{\partial \theta_i} \right]^T \quad (10)$$

where according to the model (1) and associated predictor (3)

$$\frac{\partial \hat{y}_{t|t-1}}{\partial \theta_i} = \frac{\partial C}{\partial \theta_i} \hat{x}_{t|t-1} + C \frac{\partial \hat{x}_{t|t-1}}{\partial \theta_i} + \frac{\partial D}{\partial \theta_i} u_t \quad (11)$$

where

$$\begin{aligned} \frac{\partial \hat{x}_{t+1|t}}{\partial \theta_i} &= \frac{\partial}{\partial \theta_i} (A - KC) \hat{x}_{t|t-1} + (A - KC) \frac{\partial \hat{x}_{t|t-1}}{\partial \theta_i} \\ &\quad + \frac{\partial}{\partial \theta_i} (B - KD) u_t + \frac{\partial K}{\partial \theta_i} y_t. \end{aligned} \quad (12)$$

In this case, the ensuing approximation

$$\min_{\theta} \|e(\theta)\|_2 \approx \min_{\theta} \|e(\theta_k) + J(\theta_k)(\theta - \theta_k)\|_2 \quad (13)$$

by virtue of being affine in θ does have a solution that can be found in closed form. However, this solution is only unique if the Jacobian $J(\theta_k)$ is of full column rank.

Both the Levenberg–Marquardt and “damped” Gauss–Newton approaches ([5, Ch. 10], [19, Ch. 10], and [6, Ch. 6]) use the linear approximation (8) and are, undoubtedly, the most famous gradient-based iterative search methods used in solving problem (7). Indeed, both methods involve the calculation of a search direction p and also (e.g., in the “damped” Gauss–Newton case) a step length α , such that the $(k+1)$ th iterate θ_{k+1} in the search for $\hat{\theta}$ is found from the previous iterate θ_k by

$$\theta_{k+1} = \theta_k + \alpha p \quad (14)$$

where the search direction p for both methods is obtained as

$$\begin{aligned} p &\in \{d \in \Delta \subseteq \mathbf{R}^{n_\theta} : \|e(\theta_k) + J(\theta_k)d\|_2 \\ &\leq \|e(\theta_k) + J(\theta_k)\bar{d}\|_2, \forall \bar{d} \in \Delta\}. \end{aligned} \quad (15)$$

In the Levenberg–Marquardt method, Δ is chosen in an adaptive manner according to how well the local approximation (8) predicts the actual algorithm performance.

Alternatively, when a “damped” Gauss–Newton technique is used instead, the search region is taken as $\Delta = \mathbf{R}^{n_\theta}$, and a second stage is introduced to compute a step length $\alpha > 0$ such that

$$\|e(\theta_k + \alpha p)\|_2 < \|e(\theta_k)\|_2. \quad (16)$$

The update (14) is then applied for the value of α found in this second stage, which typically starts with $\alpha = 1$, and then, reduces it (for example, by halving the previous value of α) until (16) is satisfied.

¹The more general case of spatial correlation can easily be handled [18], but at the expense of extra notation that will detract from the essential arguments to follow.

In both cases, it can be shown that the search direction defined by (15) is any vector p that satisfies

$$[J(\theta_k)^T J(\theta_k) + \lambda I]p = -J(\theta_k)^T e(\theta_k), \quad \lambda \geq 0 \quad (17)$$

where $\lambda \in \mathbf{R}$ is taken as zero in the Gauss–Newton method, and is some nonnegative value that ensures that $p \in \Delta$ in the Levenberg–Marquardt case.

IV. DDLC PARAMETRIZED SEARCH

The full parametrization (2) is an overparametrization, in that, θ is of dimension $n_\theta = n^2 + n(m + 2p) + mp$ that is n^2 more than any minimal (injective) parametrization of dimension [15]

$$n_\beta = n_\theta - n^2 = n(m + 2p) + mp. \quad (18)$$

This implies that, since the Jacobian $J(\theta_k)$ has n_θ columns, it is rank deficient with a kernel of dimension (at least) n^2 , and possibly higher.² Therefore, the search direction (17) in the damped Gauss–Newton case of $\lambda = 0$ is not uniquely defined.

In reaction to this, several authors have proposed the use of a certain $n_\theta - n^2$ dimensional minimal parametrization that is related to the full parametrization (2) via an affine transformation, and that has been dubbed DDLC [1], [11], [15], [18], [23].

In this previous work, the key idea has been to identify the set of systems parametrized by $\theta \in \mathbf{R}^{n_\theta}$ that are input–output equivalent. This can be efficiently described by a mapping $S(\theta, T): \mathbf{R}^{n_\theta} \times \mathbf{R}^{n \times n} \rightarrow \mathbf{R}^{n_\theta}$ that depends on an arbitrary invertible matrix $T \in \mathbf{R}^{n \times n}$ according to

$$S(\theta, T) = \begin{bmatrix} \text{vec}\{T^{-1}AT\} \\ \text{vec}\{T^{-1}B\} \\ \text{vec}\{CT\} \\ \text{vec}\{D\} \\ \text{vec}\{T^{-1}K\} \end{bmatrix}. \quad (19)$$

This mapping is clearly nonlinear with respect to T . However, since for a given θ , $S(\theta, T)$ viewed as a function of T is differentiable with respect to T on $M_n \triangleq \{T \in \mathbf{R}^{n \times n} : \det(T) > 0\}$ [11], a linear approximation applying locally for a perturbation ΔT around $T = I_n$ may be derived as

$$S(\theta, I_n + \Delta T) \approx \theta + Q \text{vec}\{\Delta T\} \quad (20)$$

where (\otimes) is the Kronecker tensor product

$$Q \triangleq \left. \frac{\partial S(\theta, T)}{\text{vec}\{T\}^T} \right|_{T=I_n} = - \begin{bmatrix} A^T \otimes I_n - I_n \otimes A \\ B^T \otimes I_n \\ -I_n \otimes C \\ \mathcal{O}_{m \times n^2} \\ K^T \otimes I_n \end{bmatrix}. \quad (21)$$

This implies that a parameter space update in the search direction $p \triangleq Q \text{vec}\{\Delta T\}$ for any ΔT will locally yield a system with equivalent input–output properties, and hence, an un-

changed value of $\|e(\theta)\|_2$. Therefore, it seems reasonable to restrict search directions to be orthogonal to the columns of Q .

In recognition of this, the works [1], [11], [14], [18], [23] have suggested the use of a local coordinate structure, termed DDLC, that is minimal in the sense that distinct points in parameter space correspond to non–input–output equivalent systems. More specifically, a vector $\beta \in \mathbf{R}^{n_\theta - n^2}$ is used to parametrize this local coordinate system according to

$$\theta(\beta) = \theta + P\beta \quad (22)$$

where the columns of P are chosen via, for example, a singular value decomposition (SVD) or a QR factorization of Q in (21), and satisfy the requirements

$$P^T P = I, \quad P^T Q = 0, \quad \mathcal{R}(P) \oplus \mathcal{R}(Q) = \mathbf{R}^{n_\theta}. \quad (23)$$

Here, $\mathcal{R}(Q) \triangleq \{x : x = Qy \text{ for some } y\}$ is the column space of Q and similarly for P .

Thus, according to the local parametrization (22), θ can only move in directions that are a linear combination of the columns of P , i.e., in directions $P\beta$. In this case, the (local to θ_k) optimization problem (13) is replaced by

$$\min_{\beta} \|e(\theta_k) + J(\theta_k)P\beta\|_2. \quad (24)$$

Denoting a solution as q then, via (22), this implies a search direction

$$\theta_k(q) - \theta_k(0) = \theta_k + Pq - \theta_k = Pq. \quad (25)$$

As explored by [1], [14], [18], a benefit of solving (24) is that β typically has dimension n^2 less than that of θ and it seems reasonable to expect that the computational load is, therefore, diminished. Indeed, the Jacobian matrix of $e(\theta_k + P\beta)$ with respect to β , defined as

$$J_{\text{DDLC}}(\theta_k) \triangleq \left. \frac{\partial e(\theta_k + P\beta)}{\partial \beta} \right|_{\beta=0} = J(\theta_k)P \quad (26)$$

has n^2 less columns than that of $J(\theta_k)$. Substituting this relationship into (24) then implies that (24) may be expressed as

$$\min_{\beta} \|e(\theta_k) + J_{\text{DDLC}}(\theta_k)\beta\|_2 \quad (27)$$

with $J_{\text{DDLC}}(\theta_k)$ computed, in practice, via (9)–(12) with the required derivatives of system matrices in (12) obtained via (22) as

$$\frac{\partial \theta(\beta)}{\partial \beta_i} = \frac{\partial}{\partial \beta_i} (\theta + P\beta) = P_i \quad (28)$$

with P_i denoting the i th column of P .

Therefore, according to the previous discussion, a Levenberg–Marquardt or Gauss–Newton method may be used to solve (7) by computing a search direction q as

$$\begin{aligned} q &\in \{d \in \Delta \subseteq \mathbf{R}^{n_\beta} : \|e(\theta_k) - J_{\text{DDLC}}(\theta_k)d\|_2 \\ &\leq \|e(\theta_k) - J_{\text{DDLC}}(\theta_k)\bar{d}\|_2, \forall \bar{d} \in \Delta\} \end{aligned} \quad (29)$$

where $n_\beta \triangleq \text{rank}(P)$. This, in turn, is satisfied by any q that solves

$$[J_{\text{DDLC}}(\theta_k)^T J_{\text{DDLC}}(\theta_k) + \lambda I]q = -J_{\text{DDLC}}(\theta_k)^T e(\theta_k). \quad (30)$$

²The details are not essential to the paper here, but depend on certain observability and controllability-type conditions on the system $[A, B, C, D, K]$ implied by $\theta \in \mathbf{R}^{n_\theta}$. See [15, Lemma 4] for details.

V. RAPPROCHEMENT BETWEEN FULL AND DDLC PARAMETRIZED SEARCH

Computing a search direction p via the solution of (17) or an update direction q via solution of (30) is (relatively) straightforward in the Levenberg–Marquardt situation, since $\lambda > 0$ ensures positive definiteness of the left-hand side coefficient matrix in (17) and (30), and hence, the uniqueness of either p or q .

However, when a Gauss–Newton search strategy is employed in which $\lambda = 0$, then this same coefficient matrix may well be rank deficient. It will certainly be rank deficient in the situation when p pertains to a fully parametrized search direction, or when nonminimal system estimates occur.

The need to deal with this sort of rank deficiency is well recognized in the general theory of gradient-based optimization. It is routinely handled [19, Ch. 10] by the employment of a pseudoinverse [8, Sec. 5.5.3] of the possibly rank-deficient coefficient matrix.

The most common implementation of this approach employs an SVD of the Jacobian matrix since this allows a straightforward and computationally robust means to compute the pseudoinverse. To provide further detail on this point, define the SVD of $J(\theta)$ as

$$J(\theta) = \underbrace{[U_1, U_2]}_U \underbrace{\begin{bmatrix} S_1 & \emptyset \\ \emptyset & \emptyset \end{bmatrix}}_S \underbrace{\begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}}_{V^T} = U_1 S_1 V_1^T. \quad (31)$$

Concentrating for a moment on the full parametrization approach, the unique minimum two-norm solution [8] to (17) for any value of $\lambda \geq 0$ (i.e., both Levenberg–Marquardt and Gauss–Newton methods) is given by (31) and

$$p = -V_1 (S_1^2 + \lambda I)^{-1} S_1 U_1^T e(\theta). \quad (32)$$

In what follows, and consistent with the literature mentioned earlier, p as computed via this method that accommodates rank deficiency in the Jacobian $J(\theta_k)$ will be termed “robust” GBS.

Moving to the DDLC parametrized case, the same robust approach requires the SVD of the Jacobian matrix $J_{\text{DDLC}}(\theta)$ defined as

$$J_{\text{DDLC}}(\theta) = \bar{U}_1 \bar{S}_1 \bar{V}_1^T \quad (33)$$

that provides a minimum two-norm solution to (30) given by

$$q = -\bar{V}_1 (\bar{S}_1^2 + \lambda I)^{-1} \bar{S}_1 \bar{U}_1^T e(\theta). \quad (34)$$

Although the search directions p and q are obtained using different SVD’s (31) and (33) [associated with different Jacobian matrices $J(\theta)$ and $J_{\text{DDLC}}(\theta)$], the following Theorem establishes a main point of this paper.

The θ -space search direction implied by the robust GBS (32) and the DDLC parametrized robust GBS (34) are identical.

Theorem 5.1: Let Q be given by (21) and a corresponding matrix P satisfy the equations in (23). Further, let p and q be given by (32) and (34), respectively. Then:

$$p = Pq. \quad (35)$$

Proof: Since $S(\theta, T)$ parametrizes a system that is input–output equivalent to that parametrized by θ , then:

$$e(\theta) = e(S(\theta, T)). \quad (36)$$

Therefore, differentiating both sides with respect to $\text{vec}\{T\}$ at the point $T = I_n$ provides [recall the definition (21)] via the SVD (31):

$$0 = \left. \frac{\partial e(S(\theta, T))}{\partial \text{vec}\{T\}^T} \right|_{T=I_n} = J(\theta)Q = U_1 S_1 V_1^T Q. \quad (37)$$

Therefore, since $U_1 S_1$ is full rank:

$$V_1^T Q = 0. \quad (38)$$

Hence, for any $z \in \mathcal{R}(V_1)$, it follows that $z \in \mathcal{N}(Q^T) \triangleq \{x : Q^T x = 0\}$, which from (23) implies that $z \in \mathcal{R}(P)$. Since z was arbitrary, then $\mathcal{R}(V_1) \subseteq \mathcal{R}(P)$.

Therefore, $V_1 = PR$ for some matrix R . Furthermore, since $P^T P = I$ and $V_1^T V_1 = I$, then $I = V_1^T V_1 = R^T P^T P R = R^T R$. Therefore, from (26) and (31), it follows that

$$\begin{aligned} J_{\text{DDLC}}(\theta) &= U_1 S_1 V_1^T P \\ &= U_1 S_1 R^T P^T P \\ &= U_1 S_1 R^T \end{aligned}$$

that is, by definition, the (thin) SVD of $J_{\text{DDLC}}(\theta)$. Therefore, q obtained via

$$q = -R(S_1^2 + \lambda I)^{-1} S_1 U_1^T e(\theta) \quad (39)$$

is the unique minimum two-norm solution to (30) for any $\lambda \geq 0$ [8, Th. 5.5.1]. Moreover, since $P^T V_1 = P^T P R = R$, then $V_1 = P R = P P^T V_1$ and

$$\begin{aligned} Pq &= -\underbrace{P P^T}_{V_1} V_1 (S_1^2 + \lambda I)^{-1} S_1 U_1^T e(\theta) \\ &= -V_1 (S_1^2 + \lambda I)^{-1} S_1 U_1^T e(\theta) \\ &= p. \end{aligned}$$

■

That this result implies θ -space equivalent search directions follows by noting that the fully parametrized robust gradient search update is, according to (14)

$$\theta_{k+1} = \theta_k + \alpha p \quad (40)$$

while the DDLC parametrized update is, according to (25):

$$\theta_{k+1} = \theta_k + \alpha Pq. \quad (41)$$

In relation to this, note that (26) allows (30) to be written as

$$P^T [J(\theta_k)^T J(\theta_k) + \lambda I] Pq = -P^T J(\theta_k)^T e(\theta_k). \quad (42)$$

Therefore, if p solves (17) and is in the column range space of P so that for some q it holds that $Pq = p$, then via (42), it is straightforward that q satisfies (30).

The point of the preceding theorem is that for the *particular choice* of p given by (32) (there are infinitely many when $\lambda = 0$), p is, in fact, in the column range space of P , and for the *particular choice* of q (again there may be infinitely many) given by (39),

the relationship $Pq = p$ holds. That is, it establishes that when using SVD's to find minimal norm solutions to the perhaps underdetermined equations (17) and (30), equivalent θ -space search directions result.

It should be emphasized that these aspects of the relationship between GBS schemes employing the DDLc versus full parametrization are implicit in the motivation and discussion of previous work such as [14]–[16]. However, to the authors knowledge, Theorem 5.1 is the first explicit analysis of the search direction equivalence.

VI. EXTENDED GAUSS–NEWTON APPROACH

The arguments of the previous sections establish that, on the one hand, the DDLc corresponds to a particular choice of singular value truncation when applying fully parametrized GBS, and on the other hand, employing DDLc parametrizations in a GBS context requires the computation of n^2 fewer columns of the Jacobian matrix, which can offer significant computational savings.

In light of this, the remainder of this paper proposes and profiles an extended approach whereby, in addition to using the DDLc, further singular value truncations are chosen online and adaptively in the interests of overall algorithm performance.

In particular with the notation that the diagonal entries of S_1 are denoted as an ordered sequence $\{s_1, \dots, s_k\}$ with $s_m \geq s_{m+1}$, this paper proposes to adaptively truncate them by restricting the singular value ratio s_r/s_1 (for some $r \leq k$) to some small value γ . More precisely, given γ , then r is chosen as the smallest singular value satisfying $s_r \geq s_1\gamma$.

Moreover, this paper proposes that γ be changed online according to the size of the previous step length α according to the following reasoning. If α is small, say $\alpha < 0.5^5$ (corresponding to five bisections of the step length), then this is likely to imply that the local linear approximation of $e(\theta)$ is poor. Therefore, the singular value ratio should be increased (i.e., increase γ) so as to focus only on dominating directions. *Vice versa*, if $\alpha = 1$ on the previous iteration, then the local linear approximation is likely to be accurate. This implies that it is worthwhile including search directions additional to dominating ones. Hence, γ should be decreased so that more singular values and corresponding directions are considered.

The precise details of how this paper proposes these ideas be implemented are encapsulated in the following algorithm definition, which broadly speaking, is structured in a similar manner to standard gradient-based line search algorithms.

- 1) In a first stage, the search direction is computed, which combines the singular value ratio selection with the computationally efficient DDLc parametrization as outlined in Section IV. In addition to this, a regularization term may be employed to ensure that a descent direction is generated.
- 2) In a second stage, the bisection line search strategy is used to make sufficient progress along this search direction, and depending on the progress made, the dynamic singular value ratio is modified accordingly.

Algorithm 1: Enhanced Gauss–Newton based search for DDLc parametrized system estimates

Given an initial guess $\theta_0 \in \Theta$, initialize parameters

$$\begin{aligned} \beta &\in (0, 1/2), & \eta_{\min} &> 0, & \nu &\in (0, 1), & \eta &= 0, \\ \gamma &\in (0, 1], & \alpha_{\min} &> 0, & \epsilon &> 0 & k &= 0. \end{aligned}$$

Compute descent direction:

- 1) Compute a matrix P_k that satisfies the requirements in (23) for θ_k .
- 2) Compute the PE vector $e(\theta_k)$, its Jacobian $J_{\text{DDLc}}(\theta_k)$ and the gradient

$$g_k \triangleq J_{\text{DDLc}}^T(\theta_k)e(\theta_k).$$

- 3) Compute the SVD

$$J_{\text{DDLc}}(\theta_k) = USV^T. \quad (43)$$

- 4) Let $\{s_1, \dots, s_m\}$ be the ordered singular values from S such that $s_i \geq s_{i+1}$.
- 5) Perform the following.
 - a) Find the index r of the smallest singular value that satisfies $(s_r + \eta) \geq \gamma(s_1 + \eta)$; let U_r, V_r be the first r columns of U and V , respectively, and set S_r as a diagonal matrix:

$$S_r \triangleq \text{diag} \left\{ s_1 + \frac{\eta}{s_1}, \dots, s_r + \frac{\eta}{s_r} \right\}.$$

- b) Compute a search direction q_k as

$$q_k = -V_r S_r^{-1} U_r^T e(\theta_k). \quad (44)$$

- c) If

$$-q_k^T g_k \geq \nu \|q_k\| \|g_k\| \quad (45)$$

then go to step 6.

- d) If $\gamma > \epsilon$, then let $\gamma \mapsto \max\{\epsilon, 0.25\gamma\}$, otherwise let $\eta \mapsto \max\{\eta_{\min}, 2\eta\}$.
 - e) Go to step 5a.

Compute step length:

- 6) Initialize the step length $\alpha = 1$ and perform the following: while $V_N(\theta_k + \alpha P_k q_k) \geq V_N(\theta_k) + \alpha \beta q_k^T g_k$, then update

$$\alpha \mapsto 0.5\alpha.$$

- 7) If $\alpha = 1$, then update

$$\gamma \mapsto \max\{\epsilon, 0.25\gamma\}, \quad \eta \mapsto \eta/2.$$

- 8) If $\alpha \leq \alpha_{\min}$, then update $\gamma \mapsto \min\{1, 2\gamma\}$.

- 9) Set $\theta_{k+1} = \theta_k + \alpha P_k q_k$.

- 10) Check termination conditions:

if

$$g_k^T [J_{\text{DDLc}}^T(\theta_k) J_{\text{DDLc}}(\theta_k) + \epsilon I]^{-1} g_k \leq \epsilon$$

then stop.

Otherwise update $k \mapsto k + 1$ and go to step 1.

Steps 5a–5e are designed to ensure that the search direction q_k is not moving tangential to the cost contour, i.e., the search direction q_k and gradient g_k are not orthogonal with each other. This, in turn, ensures that the algorithm can make progress in direction q_k . More specifically, the mechanism used to ensure this can be explained as follows.

- 1) In the first (default) case, γ is reduced by a constant factor (1/4) at each iteration in the interests of increasing the allowed condition number of the Jacobian matrix, thus including more “singular-value directions” in q_k . This is repeated a finite number of times until $\gamma \leq \epsilon$ at which point an alternative strategy is invoked. Note that without making further assumptions on the condition number of J_{DDLc} , there is no guarantee of satisfying (45) using this strategy alone.
- 2) In the alternative (fallback) case, a regularization term η (initially set to $\eta_{\text{min}} > 0$) is included so that q_k now satisfies [cf. (44)]

$$[J_{\text{DDLc}}^T(\theta_k)J_{\text{DDLc}}(\theta_k) + \eta I] q_k = -g_k. \quad (46)$$

In this case, the algorithm increases η (by a factor of 2) at each iteration in the hope of satisfying (45).

In terms of making progress, the aforementioned algorithm will continue to decrease the cost function V_N until the gradient vector g_k is close to zero (see step 10), which indicates that θ_k is close to a stationary point. The fact that Algorithm 1 finds such a point is established in the following theorem.

Theorem 6.1: Assuming that $V_N(\theta)$ is continuously differentiable and Lipschitz continuous on $\Theta \in \mathbf{R}^{n_\theta}$, then Algorithm 1 generates iterates θ_k such that corresponding gradient vectors g_k tend to zero, i.e.,

$$\lim_{k \rightarrow \infty} g_k = 0.$$

Proof: Lemma A.1 shows that for each iteration of Algorithm 1, the search direction q_k will satisfy (45). Also, [5, Th. 6.3.2] shows that there exists a step length $\alpha \in (0, 1]$ that satisfies the Wolfe condition $V_N(\theta_k + \alpha P_k q_k) \geq V_N(\theta_k) + \alpha \beta q_k^T g_k$. Therefore, immediate application of [5, Th. 6.3.3] establishes that $\lim_{k \rightarrow \infty} g_k = 0$. ■

While it is reassuring that Algorithm 1 converges, this does not illustrate the utility of employing a dynamic rank determination strategy as outlined earlier. In fact, the aforementioned theorem does not depend on this feature at all. However, it is essential to demonstrate the overall performance of Algorithm 1, and with this in mind, an empirical study of its performance relative to existing approaches forms the remainder of this paper.

Note that a related approach wherein Jacobian rank is determined by a fixed threshold on the smallest singular value (as opposed the adaptive numerical conditioning threshold proposed above) is briefly studied in [16], which appears to be the only previous work related to Algorithm 1.

VII. EMPIRICAL STUDY

This section is devoted to profiling the performance of the proposed recursive Gauss–Newton (rGN) multivariable estimation Algorithm 1 across a range of estimation problems.

For this purpose, it is compared to the GBS routine available in the pem.m function provided in the Matlab (release 2006b) System Identification Toolbox Version 6.0 (SIT6). It is also examined relative to an expectation-maximization (EM) algorithm developed in [7]. In addition, for the two gradient-based algorithms rGN and GBS, we also consider two different parametrizations, namely the DDLc one described in Section IV and the separable-least-squares DDLc (slsDDLc) parametrization detailed in [20].

This implies five different algorithm/parametrization combinations as listed as follows.

- 1) Robust Gauss–Newton Algorithm 1, using the DDLc parametrization. This is denoted as rGN-DDLc.
- 2) Robust Gauss–Newton Algorithm 1, but modified to incorporate the slsDDLc parametrization. This is denoted as rGN-slsDDLc.
- 3) GBS as implemented in the SIT6 pem.m function and using the DDLc parametrization of Section IV. This is denoted as GBS-DDLc.
- 4) GBS implemented via SIT6 pem.m and using slsDDLc parametrization implemented via the software [3]. This is denoted as GBS-slsDDLc.
- 5) Expectation maximization denoted as EM.

Although this paper is primarily concerned with providing multivariable estimates, the profiling begins by considering single-input single-output (SISO) systems. The rationale is that by illustrating the dividends of Algorithm 1 even in the relatively simple SISO case where it might otherwise be thought unnecessary, the main theme of the importance of adaptive Jacobian subspace dimension is underlined.

In order to cover a broad class of problems, Monte–Carlo analysis is employed that involves 500 runs over randomly chosen third-order SISO systems in the form of (1) using different input and noise realizations for each run. Furthermore, each run involves $N = 500$ samples of the input signal and measurement noise, which were generated according to $u_t \sim \mathcal{N}(0, 1)$, $v_t \sim \mathcal{N}(0, 0.01)$.

Within this set of simulations signified by S1, the initialization of θ_0 was performed both by initial deployment of an N4SID subspace estimation algorithm (denoted as case S1a) and also by simply using a random value (denoted as case S1b). This latter situation is included in order to study robustness to initial value and robustness against capture in local minima. All methods were run for 200 iterations.

Table I profiles the performance of the five algorithms listed earlier by showing the number of failures for each of them, where a failure is defined to be a situation in which

$$\|e(\theta)\|^2 > 1.3 \sum_{t=1}^N \|v_t\|^2. \quad (47)$$

Considering the relative failure rates by this criterion shown in columns S1a and S1b of Table I, this initial study illustrates

TABLE I
NUMBER OF FAILURES FOR DIFFERENT ALGORITHMS (ROWS) UNDER
DIFFERENT CONDITIONS (COLUMNS)

	S1a	S1b	S2a	S2b	S3a	S3b
EM	0	0	0	5	1	23
rGN-DDLC	0	7	0	7	0	13
GBS-DDLC	0	18	0	20	5	97
rGN-slsDDLDC	0	22	2	238	6	488
GBS-slsDDLDC	0	30	3	269	24	500

In each case, 500 trials were conducted

that with good initialization, all five algorithms offer equivalent performance, while for poor initialization, the EM appears to be the most robust and the rGN outperforms GBS using both DDLC and slsDDLDC parametrizations.

Further, Monte–Carlo-based simulation study with input, output, and state dimensions increased to $m = 2$, $p = 2$, and $n = 8$, and all other parameters as in the previous SISO case was also performed. The results, denoted as S2 with epithets a and b according to N4SID and random initialization are also presented in Table I.

Again considering relative failure rates, this provides clear evidence that the SISO-evidenced conclusions, in terms of enhanced robustness of Algorithm 1, apply (it seems with greater emphasis) in the multivariable situation that has been the main impetus for this paper.

Finally, the input, output, and state dimensions were increased to $m = 3$, $p = 3$, and $n = 18$. The Monte–Carlo results for this case are denoted as S3 (a and b for N4SID and random initialization) in Table I. The very high failure rates in this case for some algorithms in comparison to the quite low rate for the rGN-DDLC underlines the effectiveness, in a multiple-input multiple-output (MIMO) setting, of Algorithm 1.

In particular, note that column S3a in Table I indicates that, for a higher dimensional system (18 state, three inputs/outputs), and even for a “good” N4SID initialization, use of Algorithm 1 may substantially reduce the failure rate compared (for example) with the GBS-DDLC. It is perhaps worth noting that Table I illustrates that the EM algorithm [7] also offers a robust approach.

While the systems under study in Table I were all chosen randomly, the work in [20] on the slsDDLDC parametrization for the GBS has proposed a series of eight specific two-output autoregressive moving average (ARMA) test examples.

The systems are of the form (1) (but with B and D removed) where the state dimensions are 2, 4, 6, 8, 10, 12, 14, and 16. These will be denoted by M_1, M_2, \dots, M_8 here. Furthermore, [20] proposes parameter initializations for each of the eight systems denoted as “close” and “far” from the true parameter values according to random perturbations with a particular variance.

The methods 1–5 profiled in Table I were applied to these ARMA systems proposed in [20]. Tables II and III show the number of failures for each of the eight models for “close” and “far” initial estimates, respectively. One hundred Monte–Carlo runs were conducted. The results further evidence the robustness of the rGN-DDLC approach proposed here.

TABLE II
NUMBER OF FAILURES FOR DIFFERENT ALGORITHMS (ROWS) AND FOR
DIFFERENT MODELS (COLUMNS) WHEN USING “CLOSE” INITIAL ESTIMATES

	M_1	M_2	M_3	M_4	M_8
EM	1	1	0	0	0
rGN-DDLC	0	0	0	0	0
GBS-DDLC	0	2	0	2	1
rGN-slsDDLDC	0	1	0	1	1
GBS-slsDDLDC	0	2	1	3	1

Note that columns M_5 , M_6 , and M_7 are not shown because all corresponding entries are zero. In all, 100 trials were conducted.

TABLE III
NUMBER OF FAILURES FOR DIFFERENT ALGORITHMS (ROWS) AND FOR
DIFFERENT MODELS (COLUMNS) WHEN USING “FAR” INITIAL ESTIMATES

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
EM	0	14	5	5	0	0	0	0
rGN-DDLC	0	36	11	13	1	0	0	4
GBS-DDLC	2	36	15	27	2	4	3	7
rGN-slsDDLDC	0	38	13	26	0	0	0	6
GBS-slsDDLDC	0	38	17	28	0	0	0	6

In all, 100 trials were conducted

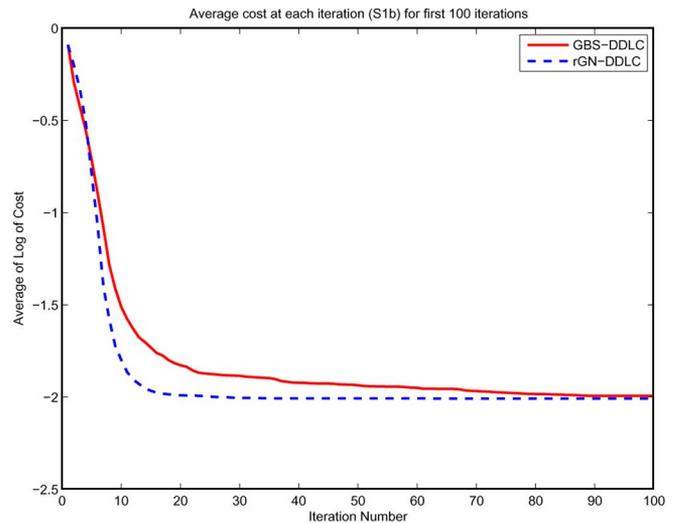


Fig. 1. Average cost at each iteration for S1b experiment.

Finally, while the empirical study has so far focused on failure or otherwise according to the final cost criterion (47), the rate of convergence is also of practical significance. In consideration of this, a comparison was made between rGN-DDLC and GBS-DDLC algorithms by averaging the evolution of $\|e(\theta_k)\|^2$ over all 500 Monte–Carlo runs profiled in Table I after failure runs are censored.

The results are illustrated in Figs. 1–3. Note that only the case of random initialization is presented, but results for N4SID derived initializations are similar. Furthermore, note that only two of the five algorithms considered in Table I are profiled, namely Algorithm 1 rGN-DDLC and the next fastest converging method SIT6 pem.m denoted as GBS-DDLC.

Clearly, this evidence indicates that the enhanced robustness illustrated in Tables I–III of the rGN-DDLC method proposed in this paper is not at the expense of convergence rate. For example, for the eight-state two-input, two-output systems profiled in Fig. 2, the rGN-DDLC method of this paper is illustrated to

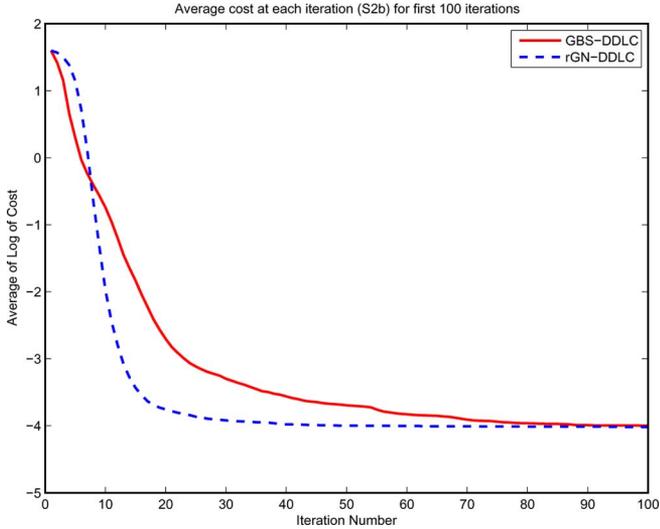


Fig. 2. Average cost at each iteration for S2b experiment.

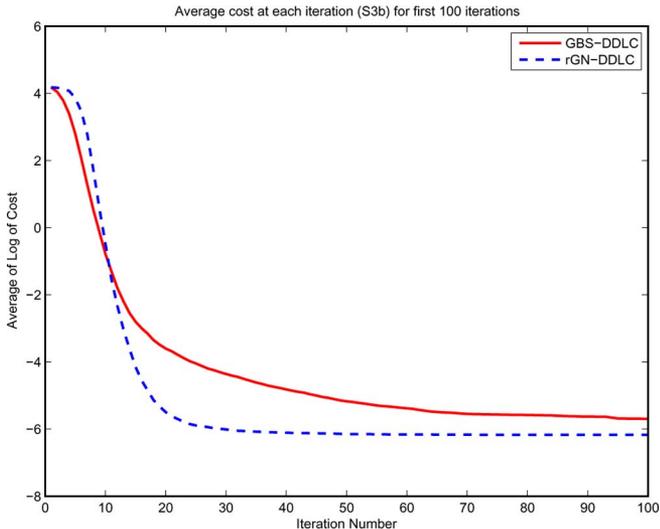


Fig. 3. Average cost at each iteration for S3b experiment.

converge to the minimum cost in (on average) less than half the number of iterations of the nearest competitor GBS-DDLC.

VIII. CONCLUSION

When employing the GBS for a wide class of optimization problems, it is standard to assess Jacobian rank and employ a pseudoinverse when necessary in the computation of search direction. This procedure is commonly termed “robust” GBS.

This paper has established that, when considering the optimization problems implied by the PE estimation of multivariable systems, the recently proposed DDLC parametrization delivers GBS directions, and hence, final estimates that are identical to those provided by the aforementioned and more standard robust GBS technique.

That is, DDLC methods can be viewed as a special case of robust approaches wherein Jacobian rank is fixed according to the nature of the underlying model structure. In this paper,

this observation has led to the development of a new algorithm wherein (effective) Jacobian rank is adaptively modulated not according to model structure, but according to inferences on the local performance of ensuing search directions.

At the same time, since the chosen rank is necessarily less than that considered by DDLC methods, and since this parametrization implies computational cost savings, the final algorithm proposed here is a hybrid that employs a DDLC parametrization in a first step.

Readers wishing to assess the methods proposed in this paper are invited to visit <http://sigpromu.org> to download a MATLAB implementation provided as part of a general system identification toolbox.

APPENDIX

Lemma A.1: There exists a finite positive integer ℓ such that the search direction q_k obtained via (44), i.e.,

$$q_k = - [J_{\text{DDLC}}^T(\theta_k) J_{\text{DDLC}}(\theta_k) + 2^\ell \eta_{\min} I]^{-1} g_k \quad (48)$$

satisfies (45).

Proof: For pedagogical reasons, we drop the subscript k . Define the function $f(\eta)$ as [cf. (45)]

$$f(\eta) \triangleq \frac{-q^T g}{\|q\| \|g\|} = \frac{g^T M_\eta g}{(g^T M_\eta^2 g)^{1/2} \|g\|} \quad (49)$$

$$M_\eta \triangleq [H + \eta I]^{-1}, \quad H \triangleq J_{\text{DDLC}}^T(\theta) J_{\text{DDLC}}(\theta). \quad (50)$$

If there exists η^* such that $f(\eta^*) = \nu$ and $f(\eta) > \nu$ for all $\eta > \eta^*$, then $\ell \triangleq \lceil \log_2(\eta^*/\eta_{\min}) \rceil$ will satisfy (45) via (48). To show that such an η^* exists, we will show that $\lim_{\eta \rightarrow \infty} f(\eta) = 1$ and that $f(\eta)$ is a monotonically increasing function, hence via the intermediate value theorem since $\nu \in (0, 1)$, then $f(\eta^*) = \nu$ for some $\eta^* \in (f(\eta_{\min}), \infty)$. In terms of the limit, it follows that

$$\begin{aligned} \lim_{\eta \rightarrow \infty} f(\eta) &= \lim_{\eta \rightarrow \infty} \frac{g^T [\frac{1}{\eta} H + I]^{-1} g}{(g^T [\frac{1}{\eta} H + I]^{-2})^{1/2} \|g\|} \\ &= \frac{g^T g}{\|g\| \|g\|} = 1. \end{aligned}$$

In terms of showing that $f(\eta)$ is monotonically increasing, it suffices to show that its derivative with respect to η is positive. The derivative is given by

$$f'(\eta) = \frac{(g^T M_\eta g)(g^T M_\eta^3 g) - (g^T M_\eta^2 g)^2}{\|q\|^3 \|g\|} \quad (51)$$

and since $\|q\|^3 \|g\| > 0$, it suffices to show that the numerator of (51) is positive. Since M_η is positive definite and symmetric, we may define $x \triangleq M_\eta^{1/2} g$ and $y \triangleq M_\eta^{3/2} g$, in which case $(x^T y)^2 \leq \|x\|^2 \|y\|^2$ via the Cauchy–Schwarz inequality. ■

REFERENCES

- [1] N. H. Bergboer, V. Verdult, and M. H.G. Verhaegen, “An efficient implementation of maximum likelihood identification of LTI state–space models by local gradient search,” in *Proc. 41st IEEE CDC*, Las Vegas, NV, Dec. 10–13, 2002, vol. 1, pp. 616–621.

- [2] P. E. Caines, *Linear Stochastic Systems*. New York: Wiley, 1988.
- [3] M. Deistler, W. Scherrer, and T. Ribarits, "SlsDDLc addition to Matlab's System Identification Toolbox Version 6.0," personal communication, Aug. 2005.
- [4] M. Deistler, "System identification—General aspects and structure," in *Model Identification and Adaptive Control*. New York: Springer-Verlag, 2000.
- [5] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
- [6] R. Fletcher, *Practical Methods of Optimization*. Chichester, U.K.: Wiley, 1987.
- [7] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, Oct. 2005.
- [8] G. Golub and C. V. Loan, *Matrix Computations*. Baltimore, MD: The Johns Hopkins Univ. Press, 1989.
- [9] E. J. Hannan and M. Deistler, *The Statistical Theory of Linear Systems*. New York: Wiley, 1988.
- [10] W. Larimore, "Canonical variate analysis in identification, filtering and adaptive control," in *Proc. 29th IEEE Conf. Decis. Control*, 1990, pp. 596–604.
- [11] L. H. Lee and K. Poolla, "Identification of linear parameter-varying systems using nonlinear programming," *J. Dyn. Syst., Manage., Control*, vol. 121, pp. 71–78, Mar. 1999.
- [12] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.
- [13] L. Ljung, *MATLAB System Identification Toolbox Users Guide, Version 6*. Natick, MA: The Mathworks, 2004.
- [14] T. McKelvey and A. Helmersson, "System identification using an over-parametrized model class—Improving the optimization algorithm," in *Proc. 36th IEEE Conf. Decis. Control*, San Diego, CA, Dec. 1997, pp. 2984–2989.
- [15] T. McKelvey, A. Helmersson, and T. Ribarits, "Data driven local coordinates for multivariable linear systems and their application to system identification," *Automatica*, vol. 40, pp. 1629–1635, 2004.
- [16] T. McKelvey, "Identification of State-Space Models from Time and Frequency Data," Ph.D. dissertation, Linköping University, Linköping, Sweden, 1995.
- [17] T. McKelvey, "Discussion: 'On the use of minimal parametrizations in multivariable ARMAX identification' by R.P. Guidorzi," *Eur. J. Control*, vol. 4, pp. 93–98, Feb. 1998.
- [18] T. McKelvey and A. Helmersson, "A dynamical minimal parametrization of multivariable linear systems and its application to optimization and system identification," in *Proc. 14th World Congress IFAC*, Beijing, China, 1999, vol. H, pp. 7–12.
- [19] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York: Springer-Verlag, 1999.
- [20] T. Ribarits, M. Deistler, and B. Hanzon, "On new parametrization for the estimation of linear state-space models," *Int. J. Adaptive Control Signal Process.*, vol. 18, pp. 717–743, 2005.
- [21] T. Söderström and P. Stoica, *System Identification*. New York: Prentice-Hall, 1989.
- [22] P. van Overschee and B. De Moor, *Subspace Identification for Linear Systems*. Norwell, MA: Kluwer, 1996.
- [23] V. Verdult, N. Bergboer, and M. Verhaegen, "Maximum likelihood identification of multivariable bilinear state-space systems by projected gradient search," in *Proc. 41st IEEE CDC*, Las Vegas, NV, Dec. 10–13, 2002, pp. 1808–1813.
- [24] M. Verhaegen, "Identification of the deterministic part of MIMO state space models in innovations form from input-output data," *Automatica*, vol. 30, no. 1, pp. 61–74, Jan. 1994.



Adrian Wills was born in Orange, N.S.W., Australia. He received the B.E.(elec.) and Ph.D. degrees in electrical engineering from The University of Newcastle, Callaghan Campus, Newcastle, N.S.W., in 1999 and 2003, respectively.

Since 2003, he has been a Postdoctoral Fellow at The University of Newcastle, where he is engaged in research on the area of system identification.



Brett Ninness (S'88–M'89) was born in Singleton, Australia, in 1963. He received the B.E., M.E., and Ph.D. degrees in electrical engineering from The University of Newcastle, Newcastle, N.S.W., Australia, in 1986, 1991, and 1994, respectively.

Since 1993, he has been with the School of Electrical Engineering and Computer Science, The University of Newcastle, where he is currently a Professor. He is the author or coauthor of about 100 papers in journals and conference proceedings. He has been a member of the editorial board of the *Automatica*, and

is currently the Editor-in-Chief of the *Institute of Electrical Engineering (IEE) Control Theory and Applications*. His current research interests include the areas of system identification and stochastic signal processing.

Prof. Ninness has been a member of the editorial board of the *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*. He organized the 14th International Federation of Automatic Control (IFAC) Symposium on System Identification in Newcastle, Australia, in 2006.