Robust Maximum-Likelihood Estimation of Multivariable Dynamic Systems

Stuart Gibson* Brett Ninness†

Abstract

This paper examines the problem of estimating linear time-invariant state-space system models. In particular it addresses the parametrization and numerical robustness concerns that arise in the multivariable case. These difficulties are well recognised in the literature, resulting (for example) in extensive study of subspace based techniques, as well as recent interest in "data driven" local co-ordinate approaches to gradient search solutions. The paper here proposes a different strategy that employs the Expectation Maximisation (EM) technique. The consequence is an algorithm that is iterative, and locally convergent to stationary points of the (Gaussian) Likelihood function. Furthermore, theoretical and empirical evidence presented here establishes additional attractive properties such as numerical robustness, avoidance of difficult parametrization choices, the ability to estimate unstable systems, the ability to naturally and easily estimate non-zero initial conditions, and moderate computational cost. Moreover, since the methods here are Maximum-Likelihood based, they have associated known and asymptotically optimal statistical properties.

1 Introduction

A fundamental and widely-applicable approach to the problem of obtaining parametric models from observed data involves adopting a statistical framework and then selecting as estimated model, that which maximises the likelihood of the observed data. Schemes guided by this principle are known as Maximum Likelihood (ML) methods and, due to the fact that they have been studied for almost a century, they benefit from a very large and sophisticated body of supporting theory [6, 19, 25, 42] This theoretical underpinning allows, for example, important practical issues such as error analysis and performance trade-offs to be addressed. Moreover, it provides a rationale for using such methods since it is generally (but not universally) true that ML estimators are asymptotically optimal, in that they asymptotically (in observed data length) achieve the Cramér–Rao Lower Bound [18].

However, despite their theoretical advantages, the practical deployment of ML methods is not always straightforward. This is largely due to the non-convex optimisation problems that are often implied. Since these cannot be solved in closed-form, they are typically attacked via a gradient-based search strategy based on Newton's method or one of its derivatives [12] The ultimate success of such approaches depends on its curvature with respect to the model parameters. The curvature, in turn, is dependent on the chosen model parametrization, and the selection of these can be difficult,

*This work was supported by the Australian Research Council. This author is with the School of Electrical Engineering and Computer Science, The University of Newcastle, Australia and can be contacted at email: stuart.gibson@newcastle.edu.au or FAX: +61 2 49 21 69 93

†This author is also with the School of Electrical Engineering and Computer Science, The University of Newcastle, Australia and can be contacted at email: brett@ee.newcastle.edu.au or FAX: +61 2 49 21 69 93

particularly in the multivariable case where the cost contours resulting from natural canonical state-space parametrizations imply poor numerical conditioning during a gradient-based search [9, 28, 29].

Fully-parametrized state-space models (i.e. those in which all elements of all matrices are unconstrained) provide an obvious alternative to canonical structures, at least for modelling input-output behaviour, since they provide a very general, compact and simple framework within which to represent finite-dimensional multivariable systems. These are employed by the class of "subspace-based" estimation methods that have attracted great interest over the last several years as a practical solution for finding multiple-input, multiple output (MIMO) system models [44, 43, 10, 24]. Via this approach, and as opposed to ML strategies, estimates are found in closed form without need for iterative search. While the utility of these algorithms is widely recognised, it is equally acknowledged that, depending on the problem conditions, their accuracy may be less than that offered by ML or prediction error estimates [21, 27].

In reaction to this, it has recently been established that fully parametrized models can also be coupled with ML and prediction error criterions via algorithms that identify, at each iteration of a gradient based search, a search subspace of minimal dimension [2, 26, 30].

Inspired by these issues, this paper explores a different approach to the problem of finding ML estimates of fully-parametrized state-space models from multivariable observations. More specifically, the work here employs the Expectation Maximisation (EM) algorithm as a means of computing ML estimates. The EM algorithm enjoys wide popularity and acceptance in a broad variety of fields of applied statistics. For example, areas as disparate as signal processing and dairy science routinely use the method [3, 8].However, despite this acceptance and success in other fields, it could be argued that in systems and control settings, the EM algorithm is not as well understood, accepted and utilised as it may deserve to be. With this in mind this paper seeks to make the contribution of establishing, via both theoretical and empirical evidence, that EM algorithm based techniques are a highly competitive alternative for solving multivariable control-relevant ML estimation problems.

It is important to note that there have been previous works using the EM algorithm in control-related problems. In [20] the problem of single-input, single-output (SISO) ARX model estimation on the basis of censored data sets was considered, and was further addressed in [17]. Furthermore, the work [39], appearing in the statistics literature, addressed the control relevant issue of time series estimation via an EM algorithm based solution and with respect to a particular class of time series models. Finally, the works [7, 13] address theoretical aspects of EM approaches for parameter estimation of continuous time diffusions.

By way of contrast, this paper deals with a different set of estimation problems by including the possibility of exogenous inputs, by progressing beyond polynomial SISO ARX structures to the employment of MIMO state-space models and by allowing full ARMA noise model estimation.

In relation to these contributions, an essential development here is the derivation of a numerically robust implementation. This is of key importance since the poor numerical properties of any naïve implementation limit the feasible state and input-output dimensions of any estimated system to such a degree as to severely curtail its practical utility. On the other hand, with the robust implementation derived here, the method proves to be exceptionally dependable, and capable of handling quite high dimensions of state, input number and output number.

Additionally this paper makes further contributions by providing a self-contained introduction to the EM approach and its underlying principles, as well as then profiling, both theoretically and empirically, the performance of the specific EM algorithm based technique developed here for the purposes of LTI MIMO estimation.

2 The Expectation-Maximisation (EM) algorithm

As just mentioned, the EM algorithm has a long history within the mathematical statistics community. The idea underlying the algorithm was first proposed in [1] and then presented in its current, highly developed form several years later with the publication of [11]. Since then has been widely applied not only in the area of mathematical statistics [31, 33], but has also in areas relevant to control such as signal processing, pattern, and speech recognition [36, 41].

However, the fundamental ideas and principles underlying the method do not seem to have permeated widely within the control community. This motivates the following introductory section, intended as a concise overview of the fundamentals of the EM algorithm.

The first essential point is that the EM algorithm is designed to compute the Maximum-Likelihood (ML) estimate $\widehat{\theta}_{ML}$ of a parameter vector θ on the basis of an observed data set Y, for which the likelihood of this data is written as $p_\theta(Y \mid U)$ of the data. That is, the EM method is an algorithm to find

$$\widehat{\theta}_{ML} \in \{\theta \in \Theta : p_\theta(Y \mid U) \geq p_{\theta'}(Y \mid U) \; \forall \; \theta' \in \Theta\}. \tag{1}$$

Here U denotes some further (and at this stage, unspecified) information, $p_\theta(\cdot \mid \cdot)$ is a conditional probability density function that is parametrized by a vector $\theta \in \mathbf{R}^d$, while $\Theta \subset \mathbf{R}^d$ a compact subset of candidate parameter vectors from which $\widehat{\theta}_{ML}$ is to be chosen and is chosen as a closed hypercube in $\mathbf{R}^d$. As implied by equation (1), the ML estimate $\widehat{\theta}_{ML}$ need not be unique. This point will be addressed further in the sequel.

This formalism is well known with regard to control relevant system identification problems. The traditional approach is to recognise (1) as a particular case of a general class of optimisation problems where the cost is smooth enough for a gradient based search algorithm to be used [25, 34, 42].

However, instead of exploiting any smoothness of $p_\theta(Y \mid U)$, the EM algorithm takes a different approach by utilising a more fundamental characteristic of the cost that arises by virtue of it being a probability density function; viz.

$$\int_{\mathbf{R}^N} p_\theta(Y \mid U) \, \mathrm{d}Y = 1, \qquad \forall \theta. \tag{2}$$

Engaging this mechanism involves the postulate of a so-called 'complete data set' that contains not only what was actually observed, Y, but also another set of data, X, which one might wish were available, but in fact is not. The data set X itself is usually termed the 'missing data' and its composition is left largely up to the user. Its choice is usually the key design step in the use of the EM algorithm. The approach taken here is to select the missing data so that an associated likelihood maximisation problem can be simply and robustly solved in closed form.

With this in mind, the principles underlying the EM Algorithm depend first on the application of Bayes' Rule, which delivers

$$p(Y)p(X \mid Y) = p(X, Y)$$

and hence,

$$\log p_\theta(Y \mid U) = \log p_\theta(X, Y \mid U) - \log p_\theta(X \mid Y, \, U). \tag{3}$$

This provides an explicit link between a 'wished for' log-likelihood function $\log p_\theta(X, Y \mid U)$ that depends on *unavailable* observations X, and the log-likelihood $\log p_\theta(Y \mid U)$ which is actually available, and for which a maximiser is sought.

With this link in mind, the essential idea underlying the EM algorithm is to approximate $\log p_\theta(X, Y \mid U)$ by a function $\mathcal{Q}(\theta, \theta')$, which is a projection onto a space defined by available observations, and a

current estimate θ' of the likelihood maximiser. That is,

$$\log p_\theta(X, Y \mid U) \approx \mathcal{Q}(\theta, \theta') \triangleq \mathbf{E}_{\theta'}\{\log p_\theta(X, Y \mid U) \mid Y,\ U\}. \tag{4}$$

Here, $\mathbf{E}_{\theta'}\{\cdot\}$ is the expectation operator with respect to an underlying probability density function defined by the true system parameters being θ'. This invites the obvious question as to how maximisation (or approximate maximisation) of $p_\theta(X, Y \mid U)$, that depends on unavailable data X, relates to the *prima facie* more practical issue of maximising $L(\theta) = p_\theta(Y \mid U)$. To study this, note that $L(\theta)$ may be written in terms of $\mathcal{Q}(\theta, \theta')$ according to (3) as

$$\begin{aligned} L(\theta) \triangleq \log p_\theta(Y \mid U) &= \mathbf{E}_{\theta'}\{\log p_\theta(Y \mid U) \mid Y,\ U\} &(5)\\ &= \mathbf{E}_{\theta'}\{\log p_\theta(X, Y \mid U) \mid Y,\ U\} - \mathbf{E}_{\theta'}\{\log p_\theta(X \mid Y,\ U) \mid Y,\ U\} \\ &= \mathcal{Q}(\theta, \theta') - \mathcal{V}(\theta, \theta'), &(6)\end{aligned}$$

where $\mathcal{Q}(\theta, \theta')$ was defined above in (4) and

$$\mathcal{V}(\theta, \theta') \triangleq \mathbf{E}_{\theta'}\{\log p_\theta(X \mid Y,\ U) \mid Y,\ U\}. \tag{7}$$

Note that the equality (5) follows since the arguments in the integration implied by the expectation are fixed by the conditioning on (Y, U) (see, for example, Theorem 2.9 of [22]). Furthermore, it is straightforward (see the following proof of Theorem 5.2) that $\mathcal{V}(\theta', \theta') - \mathcal{V}(\theta, \theta')$ is the Kullback-Leibler distance between the two density functions $p_\theta(X \mid Y,\ U)$ and $p_{\theta'}(X \mid Y,\ U)$. Hence, via property (2), $\mathcal{V}(\theta', \theta') - \mathcal{V}(\theta, \theta') \geq 0$ for all θ, θ'.

Therefore, the decomposition (6) establishes that

$$\mathcal{Q}(\theta, \theta') > \mathcal{Q}(\theta', \theta') \ \Rightarrow\ L(\theta) > L(\theta'). \tag{8}$$

That is, any new θ which increases $\mathcal{Q}(\theta, \theta')$ above it's old value $\mathcal{Q}(\theta', \theta')$ *must also* increase the likelihood function $L(\theta)$.

This principle then leads to the following definition for one iteration of the EM algorithm starting from an estimate $\widehat{\theta}_k$ of $\widehat{\theta}_{ML}$ and updating to a (better) one $\widehat{\theta}_{k+1}$ according to

1. **E Step**

$$\text{Calculate:} \qquad \mathcal{Q}(\theta, \widehat{\theta}_k); \tag{9}$$

2. **M Step**

$$\text{Compute:} \qquad \widehat{\theta}_{k+1} = \underset{\theta \in \Theta}{\arg\max}\ \mathcal{Q}(\theta, \widehat{\theta}_k). \tag{10}$$

One iteration of equations (9) and (10) is rarely enough to obtain a satisfactory approximation to $\widehat{\theta}_{ML}$ and thus an EM algorithm is usually composed of more than one iteration. The net effect of applying this algorithm to the ML problem is to replace the single $\arg\max$ operation on $L(\theta)$ (see equation (1)) with a succession of $\arg\max$ operations on the function $\mathcal{Q}(\cdot, \cdot)$. Clearly, this strategy is only sensible when the task of computing and maximising $\mathcal{Q}(\theta, \widehat{\theta}_k)$ is much easier than that of maximising $L(\theta)$ directly. In practice, this turns out to depend largely upon the composition of the missing data set X and, as such, using an EM algorithm is not always an appropriate strategy. However, it will now be demonstrated that it is very suitable for solving a wide range of dynamic system estimation problems of engineering relevance.

3 Application of the EM algorithm to LTI Parameter Estimation

The estimation problem considered in this paper is to obtain a ML parameter estimate of an $n^{th}-$order system described by the state-space equations

$$\begin{bmatrix} x_{t+1} \\ y_t \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} + \begin{bmatrix} w_t \\ v_t \end{bmatrix} \tag{11}$$

given that the distribution of the initial state is Normal with unknown mean μ and positive definite covariance matrix P_1, i.e.

$$x_1 \sim \mathcal{N}(\mu, P_1), \tag{12}$$

and a sequence of input-output data samples (U_N, Y_N), where

$$U_N \triangleq \{u_1, u_2, \cdots, u_N\} \qquad \text{and} \qquad Y_N \triangleq \{y_1, y_2, \cdots, y_N\}. \tag{13}$$

In equation (11), the vector sequence $\{x_t \in \mathbf{R}^n\}$ represents the evolution of the system's state, while $\{w_t\}$ and $\{v_t\}$ model random disturbances. It is assumed that the latter two sequences can be modelled as temporally independent random variables with a positive-definite joint covariance matrix and the following Normal distribution:

$$\begin{bmatrix} w_t \\ v_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \right). \tag{14}$$

Estimating the parameters of a system described by equations (11), (12) and (14) amounts to estimating the elements of the constant matrices A, B, C, D, Q, R, S, P_1 and the vector μ. For convenience these quantities shall be collected into a vector as follows:

$$\theta^T \triangleq \left[\text{vec}\{\Gamma\}^T, \text{vec}\{\Pi\}^T, \text{vec}\{P_1\}^T, \mu^T \right], \tag{15}$$

where

$$\Gamma \triangleq \begin{bmatrix} A & B \\ C & D \end{bmatrix} \qquad \text{and} \qquad \Pi \triangleq \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}, \tag{16}$$

and the vec$\{\cdot\}$ operator creates a vector from a matrix by stacking its columns on top of one another.

Now, with this definition of the parameter vector θ in mind, the log-likelihood function $L(\theta)$ for the system described above is then [25]

$$\begin{aligned} L(\theta) &= -\frac{Np}{2}\log(2\pi) - \frac{1}{2}\sum_{t=1}^{N} \log\det(CP_{t|t-1}C^T + R) \\ &- \frac{1}{2}\sum_{t=1}^{N} (y_t - \widehat{y}_{t|t-1})^T [CP_{t|t-1}C^T + R]^{-1} (y_t - \widehat{y}_{t|t-1}), \end{aligned} \tag{17}$$

where ($Y_0 \triangleq \varnothing$)

$$\widehat{y}_{t|t-1} \triangleq \mathbf{E}_\theta\{y_t | Y_{t-1}\} \tag{18}$$

is the mean square optimal one-step ahead prediction of the system output and

$$P_{t|t-1} \triangleq \mathbf{E}_\theta\{(x_t - \widehat{x}_{t|t-1})(x_t - \widehat{x}_{t|t-1})^T\} \tag{19}$$

is the covariance matrix associated with the state-estimate $\widehat{x}_{t|t-1} \triangleq \mathbf{E}_\theta\{x_t \mid Y_{t-1}\}$. Both of these quantities may be calculated by the well-known Kalman predictor, a version of which will be presented in Lemma 3.2.

Now, an essential observation is that if, in addition to the measurements Y_N and U_N, the state sequence

$$X_{N+1} \triangleq \{x_1, x_2, \ldots, x_{N+1}\} \tag{20}$$

were available then it would be possible to extract an estimate of θ from equation (11) using simple linear regression techniques. Since knowledge of X_{N+1} would so radically simplify the estimation problem it is taken here as the EM algorithm's missing data.

According to the definition of the EM algorithm (see equations (9) and (10)) the first step is to determine the function $\mathcal{Q}(\theta, \theta')$ defined by (4). This is achieved via the following result.

Lemma 3.1. *Consider the model structure (11),* (12)*, (14). If the missing data is $X \triangleq X_{N+1}$ (defined by equation* (20)*) and the input-output data is $U \triangleq U_N$ and $Y \triangleq Y_N$ (see* (13)*), then the function $\mathcal{Q}(\theta, \theta')$ defined in* (4) *is given by*

$$\begin{aligned} -2\mathcal{Q}(\theta, \theta') &= \log\det P_1 + \operatorname{Tr}\left\{P_1^{-1}\mathbf{E}_{\theta'}\{(x_1-\mu)(x_1-\mu)^T \mid Y_N, U_N\}\right\} \\ &\quad + N\log\det\Pi + N\ \operatorname{Tr}\left\{\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\right\}, \end{aligned} \tag{21}$$

where

$$z_t^T \triangleq [x_t^T, u_t^T], \qquad \xi_t^T \triangleq [x_{t+1}^T, y_t^T], \qquad \Phi \triangleq \frac{1}{N}\sum_{t=1}^{N}\mathbf{E}_{\theta'}\{\xi_t\xi_t^T \mid Y_N, U_N\}, \tag{22}$$

$$\Psi \triangleq \frac{1}{N}\sum_{t=1}^{N}\mathbf{E}_{\theta'}\{\xi_t z_t^T \mid Y_N, U_N\}, \qquad \Sigma \triangleq \frac{1}{N}\sum_{t=1}^{N}\mathbf{E}_{\theta'}\{z_t z_t^T \mid Y_N, U_N\}. \tag{23}$$

Proof. The derivation here draws on that presented in [39] wherein a simpler time series modelling situation was considered. Those ideas are extended here to allow for exogenous inputs together with full ARMA modelling of the measurement noise component while not requiring (as in [39]) non-minimal state dimension. To begin, repeated application of Bayes' Rule, and use of Markov properties implied by (11) yields

$$\begin{aligned} & p_\theta(Y_N, X_{N+1} \mid U_N) \\ = \ & p_\theta(x_{N+1}, y_N | Y_{N-1}, X_N, U_N) p_\theta(Y_{N-1}, X_N | U_N) \\ = \ & p_\theta(x_{N+1}, y_N | x_N, u_N) p_\theta(x_N, y_{N-1} | Y_{N-2}, X_{N-1}, U_N) p_\theta(Y_{N-2}, X_{N-1} | U_N) \\ = \ & p_\theta(x_{N+1}, y_N | x_N, u_N) p_\theta(x_N, y_{N-1} | x_{N-1}, u_{N-1}) p_\theta(Y_{N-2}, X_{N-1} | U_N) \\ \vdots \ & \qquad\qquad \vdots \qquad\qquad \vdots \qquad\qquad (24) \\ = \ & p_\theta(x_1) \prod_{t=1}^{N} p_\theta(x_{t+1}, y_t \mid x_t, u_t). \qquad\qquad (25) \end{aligned}$$

Furthermore, straightforwardly from equations (11), (12), (14) and (16)

$$p_\theta(x_1) \sim \mathcal{N}(\mu, P_1) \qquad \text{and} \qquad p_\theta\left(\left[\begin{array}{c} x_{t+1} \\ y_t \end{array}\right] \middle| \, x_t, u_t\right) \sim \mathcal{N}(\Gamma z_t, \Pi). \tag{26}$$

Therefore, using the relationships (26) and excluding terms that are independent of the quantities to be estimated, equation (25) may be expressed as

$$\begin{aligned} -2\log p_\theta(Y_N, X_{N+1}|U_N) &= \log\det P_1 + (x_1-\mu)^T P_1^{-1}(x_1-\mu) + N\log\det\Pi \\ &\quad + \sum_{t=1}^{N}(\xi_t - \Gamma z_t)^T\Pi^{-1}(\xi_t - \Gamma z_t). \end{aligned} \tag{27}$$

Applying the conditional expectation operator $\mathbf{E}_{\theta'}\{\cdot \mid Y_N, U_N\}$ to both sides of equation (27) yields (21). □

Note that, according to the definitions (22) and (23), all quantities making up $\mathcal{Q}(\theta,\theta')$ may be computed from the elements

$$\widehat{x}_{t|N} \triangleq \mathbf{E}_{\theta'}\{x_t \mid Y_N, U_N\}, \qquad \mathbf{E}_{\theta'}\{x_t x_t^T \mid Y_N, U_N\}, \qquad \text{and} \qquad \mathbf{E}_{\theta'}\{x_t x_{t-1}^T \mid Y_N, U_N\} \tag{28}$$

which, according to the model structure of interest (11) and noise assumptions (14), may be computed using a standard Kalman smoother, except for the last element in (28), which requires a non-standard augmentation. This is made explicit in the following lemma.

Lemma 3.2. *Let the parameter vector θ' be composed of the elements of the A, B, C, D, Q, R, S, P_1 and μ, themselves defining the system* (11)*,* (12)*,* (14)*. Then*

$$\mathbf{E}_{\theta'}\{y_t x_t^T \mid Y_N, U_N\} = y_t\widehat{x}_{t|N}^T, \tag{29}$$

$$\mathbf{E}_{\theta'}\{x_t x_t^T \mid Y_N, U_N\} = \widehat{x}_{t|N}\widehat{x}_{t|N}^T + P_{t|N}, \tag{30}$$

$$\mathbf{E}_{\theta'}\{x_t x_{t-1}^T \mid Y_N, U_N\} = \widehat{x}_{t|N}\widehat{x}_{t-1|N}^T + M_{t|N}, \tag{31}$$

where $\widehat{x}_{t|N}$, $P_{t|N}$, and $M_{t|N}$ are calculated via the reverse-time recursions

$$J_t \triangleq P_{t|t}\overline{A}^T P_{t+1|t}^{-1} \tag{32}$$

$$\widehat{x}_{t|N} = \widehat{x}_{t|t} + J_t\left[\widehat{x}_{t+1|N} - \overline{A}\widehat{x}_{t|t} - \overline{B}u_t - SR^{-1}y_t\right] \tag{33}$$

$$P_{t|N} = P_{t|t} + J_t\left[P_{t+1|N} - P_{t+1|t}\right]J_t^T, \tag{34}$$

for $t = N, \ldots, 1$, and

$$M_{t|N} = P_{t|t}J_{t-1}^T + J_t(M_{t+1|N} - \overline{A}P_{t|t})J_{t-1}^T \tag{35}$$

for $t = N, \ldots, 2$ and the matrices $\overline{A}$, $\overline{B}$, $\overline{Q}$ are defined as

$$\overline{A} \triangleq A - SR^{-1}C, \qquad \overline{B} \triangleq B - SR^{-1}D, \qquad \overline{Q} \triangleq Q - SR^{-1}S^T. \tag{36}$$

The quantities $\widehat{x}_{t|t}, P_{t|t}, P_{t|t-1}$ required by equations (33) *through* (35) *are themselves pre-computed using the Kalman Filter equations*

$$P_{t|t-1} = \overline{A}P_{t-1|t-1}\overline{A}^T + \overline{Q} \tag{37}$$

$$K_t = P_{t|t-1}C^T\left(CP_{t|t-1}C^T + R\right)^{-1} \tag{38}$$

$$P_{t|t} = P_{t|t-1} - K_t C P_{t|t-1} \tag{39}$$

$$\widehat{x}_{t|t-1} = \overline{A}\widehat{x}_{t-1|t-1} + \overline{B}u_{t-1} + SR^{-1}y_{t-1} \tag{40}$$

$$\widehat{x}_{t|t} = \widehat{x}_{t|t-1} + K_t\left(y_t - C\widehat{x}_{t|t-1} - Du_t\right) \tag{41}$$

with $t = 1, \ldots, N$. The recursion (35) *is initialised with*

$$M_{N|N} = (I - K_N C)\overline{A}P_{N-1|N-1}. \tag{42}$$

Proof. Equations (33)-(34) are the well-known Rauch–Tung–Striebel recursions for fixed interval Kalman Smoothing of the system (11), (12), (14) [22] once transformed as follows:

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_t \\ &= Ax_t + Bu_t + w_t + SR^{-1}\underbrace{(y_t - Cx_t - Du_t - v_t)}_{=0} \\ &= (A - SR^{-1}C)x_t + (B - SR^{-1}D)u_t + SR^{-1}y_t + \overline{w}_t, \\ y_t &= Cx_t + Du_t + v_t, \end{aligned}$$

where now

$$\begin{bmatrix} \overline{w}_t \\ v_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} Q - SR^{-1}S^T & 0 \\ 0 & R \end{bmatrix}\right).$$

These depend on the Kalman Filtered quantities $\widehat{x}_{t|t}, P_{t|t}, P_{t|t-1}$ which again are very well known as being computable via equations (37) through (41) [22]. The expressions (35) and (42) are established in Property P4.3 of [40]. □

With the computation of $\mathcal{Q}(\theta, \theta')$ established as being straightforward, attention now turns, according to equation (10), to its maximisation. This is also straightforward (by design) as established below.

Lemma 3.3. *Let Σ satisfy $\Sigma > 0$ and let θ be partitioned as $\theta \in \begin{bmatrix} \beta, & \eta \end{bmatrix}$, where β parametrizes Γ, μ, and η parametrizes Π, P_1. Then*

$$\widehat{\beta} = \arg\max_{\theta \in \Theta} Q(\theta, \theta')$$

is given by

$$\Gamma = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \Psi\Sigma^{-1}, \qquad \mu = \widehat{x}_{1|N}, \tag{43}$$

where the matrices Φ, Ψ, Γ are defined in equations (22) *and* (23) *and it is assumed that the closed hypercube Θ is sufficiently large to contain these values.*

Furthermore, Π and P_1 given by

$$\Pi = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = \Phi - \Psi\Sigma^{-1}\Psi^T, \qquad P_1 = P_{1|N} \tag{44}$$

form a stationary point of $\mathcal{Q}(\cdot, \theta')$ with respect to η, and both Π and P_1 defined by equation (44) *are guaranteed positive semi-definite.*

Finally, if Π and P_1 given by (44) *are positive definite, the input sequence, $\{u_t\}$, is persistently exciting (in the sense of* (52)*) and θ' implies a controllable and observable system, then the point Π, P_1 is more than a stationary point, it is a global maximiser of $\mathcal{Q}(\cdot, \theta')$ with respect to η.*

Proof. To prove equation (43), note that the terms

$$\mathrm{Tr}\left\{P_1^{-1}\mathbf{E}_{\theta'}\{(x_1 - \mu)(x_1 - \mu)^T \mid Y_N, U_N\}\right\} = \mathrm{Tr}\left\{P_1^{-1}[(\widehat{x}_{1|N} - \mu)(\widehat{x}_{1|N} - \mu)^T + P_1]\right\}$$

and

$$\mathrm{Tr}\left\{\Pi^{-1}[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T]\right\} \;=\; \mathrm{Tr}\left\{\Pi^{-1}[(\Gamma - \Psi\Sigma^{-1})\Sigma(\Gamma - \Psi\Sigma^{-1})^T + \Phi - \Psi\Sigma^{-1}\Psi^T]\right\}$$

in (21) are clearly (globally) minimised with respect to the elements of β the choices in equation (43).

The expression for Π in equation (44) follows by application of Lemma C.1 and the chain rule to compute

$$\frac{\mathrm{d}}{\mathrm{d}\Pi}\log\det\Pi+\frac{\mathrm{d}}{\mathrm{d}\Pi}\mathrm{Tr}\{\Pi^{-1}(\Phi-\Psi\Sigma^{-1}\Psi^T)\}=\Pi^{-1}-\Pi^{-1}(\Phi-\Psi\Sigma^{-1}\Psi^T)\Pi^{-1}$$

which is clearly zero for the choice of Π in (44). The expression for P_1 follows in a similar manner. Since Π and P_1 are both covariance matrices they are, by construction, positive semi-definite.

Finally, it follows immediately from Lemma C.2 that equations (43) and (44) describe a global maximiser of $\mathcal{Q}(\cdot,\theta')$.

□

Note that an essential feature of the maximisation steps (43) and (44) is that they do not impose or require any particular parametrization of the system matrices to be estimated. As in the case of subspace based methods, and data-driven local co-ordinate gradient search methods, this imbues the algorithm with an enhanced degree of robustness by avoiding well known pitfalls associated with MIMO parametrization choices. Furthermore, it allows the EM-based approach proposed here to smoothly mesh with the aforementioned alternatives, particularly as a means for providing initialisation of the algorithm via a subspace based approach.

Additionally, note that an estimate $\widehat{\mu}_k$ of any non-zero initial state conditions arises very easily and naturally, implying that the algorithm can provide dynamic system estimates on the basis of non steady-state operating data.

At the same time, this lack of imposed parametrization is achieved by a full (and hence over) parametrization, which raises the issue of whether there is a variance penalty for employing a model structure with a non-minimal parametrization. This issue has been addressed in [35] where it was established that, in fact, for quantities that are unaltered by parametrization choice (such as frequency response function) the variance of an estimate is the same for minimally and non-minimally parametrized model structures.

4 Robust Algorithm Implementation

When collected, the results in Lemmas 3.1, 3.2 and 3.3 deliver the "naïve" EM-based algorithm for estimating the parameters of the system description (11), (12), (14).

EM Algorithm 4.1. (Naïve Implementation) *EM Algorithm for ML Estimation*

1. *Let $k=0$ and initialise estimates at $\widehat{\theta}_0=[A,B,C,D,Q,R,S,P_1,\mu]$;*
2. *Using the system specification $\widehat{\theta}_k=[A,B,C,D,Q,R,S,P_1,\mu]$, perform the Kalman-Filter recursions (37)-(41) followed by the Kalman Smoother (type) recursions* (32) *through* (35) *in order to compute Φ,Ψ and Σ (defined in equations* (22) *and* (23)*);*
3. *Maximise $\mathcal{Q}(\theta,\widehat{\theta}_k)$ with respect to θ by choosing A,B,C,D,Q,R,S,P_1 and μ according to equations* (43) *and* (44) *to obtain a new estimate $\widehat{\theta}_{k+1}$;*
4. *If the associated likelihood sequence $\{L(\widehat{\theta}_k)\}$ has converged then terminate, otherwise increment k and return to step 2.*

While the above provides a formal algorithm specification, the algorithm is termed "naïve" since the question of robust and efficient implementation requires further consideration. In particular, the experience of the authors is that any approach involving the above formal algorithm specification leads to unsatisfactory results for all but rather trivial state and input/output dimension. In particular, it is essential that the estimated covariance matrix composed of Q, R and S (see equation (14)) is symmetric and positive semidefinite at all iterations, despite the limitation of finite precision arithmetic.

On the other hand, if a considered approach is taken to dealing with the limitations of finite precision computation, then Algorithm 4.1 can be implemented in a very robust fashion, as will now be detailed.

4.1 The Robust E-Step

A major part of the E-step of EM Algorithm 4.1 consists of computing the covariance matrices $\{P_{t|N}\}$, using a set of Kalman smoothing recursions. The latter are used to construct the matrices Σ and Φ (see equations (22) and (23)). It is possible to ensure that these matrices are positive semi-definite and symmetric, as required, by employing a "square-root" filtering strategy. Such a scheme uses recursions that propagate the matrix square-roots of $P_{t|t-1}$, $P_{t|t}$ and $P_{t|N}$ rather than the full matrices themselves. The matrices can then computed as, for example, $P_{t|N} = P_{t|N}^{1/2} P_{t|N}^{T/2}$, where $P_{t|N}^{T/2}$ is shorthand for $(P_{t|N}^{1/2})^T$.

There are several approaches to square-root filtering, but the method chosen here is based on the methods proposed in [23] and employs the numerically robust and efficient $\mathcal{QR}$ factorisation [16] which, for an arbitrary matrix M decomposes it as $M = \mathcal{QR}$ where $\mathcal{Q}$ is a unitary matrix and $\mathcal{R}$ an upper-triangular matrix.

Consider first the recursion (34) which propagates $P_{t|N}$ backwards in time t. We seek a recursion which instead propagates a square root $P_{t|N}^{1/2}$ for which $P_{t|N} = P_{t|N}^{T/2} P_{t|N}^{1/2}$ satisfies (34). For this purpose, the following $\mathcal{QR}$ factorisation is useful.

$$\begin{bmatrix} P_{t|t}^{T/2}\overline{A}^T & P_{t|t}^{T/2} \\ \overline{Q}^{T/2} & 0 \\ 0 & P_{t+1|N}^{T/2} J_t^T \end{bmatrix} = \mathcal{QR} = \mathcal{Q} \begin{bmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ 0 & \mathcal{R}_{22} \\ 0 & 0 \end{bmatrix}. \tag{45}$$

Here, $\mathcal{R}$ is partitioned conformally to the left-hand side of (45). Now, exploiting the unitary nature of $\mathcal{Q}$, and multiplying this equation on the left by its transpose we obtain

$$\begin{bmatrix} \mathcal{R}_{11}^T\mathcal{R}_{11} & \mathcal{R}_{11}^T\mathcal{R}_{12} \\ \mathcal{R}_{12}^T\mathcal{R}_{11} & \mathcal{R}_{12}^T\mathcal{R}_{12} + \mathcal{R}_{22}^T\mathcal{R}_{22} \end{bmatrix} = \begin{bmatrix} P_{t+1|t} & \overline{A}P_{t|t} \\ P_{t|t}\overline{A}^T & P_{t|t} + J_t P_{t+1|N} J_t^T \end{bmatrix}.$$

Equating the lower right submatrices then indicates that

$$\begin{aligned} \mathcal{R}_{22}^T\mathcal{R}_{22} &= P_{t|t} + J_t P_{t+1|N} J_t^T - \mathcal{R}_{12}^T\mathcal{R}_{12} \\ &= P_{t|t} + J_t P_{t+1|N} J_t^T - \mathcal{R}_{12}^T\mathcal{R}_{11}(\mathcal{R}_{11}^T\mathcal{R}_{11})^{-1}\mathcal{R}_{11}^T\mathcal{R}_{12} \\ &= P_{t|t} + J_t P_{t+1|N} J_t^T - P_{t|t}\overline{A}^T(P_{t+1|t})^{-1}\overline{A}P_{t|t} \\ &= P_{t|t} + J_t\left[P_{t+1|N} - P_{t+1|t}\right]J_t^T \end{aligned} \tag{46}$$

and therefore, equations (34) and (46) imply that

$$\mathcal{R}_{22}^T \mathcal{R}_{22} = P_{t|N} \tag{47}$$

satisfies (34) so that $P_{t|N}^{1/2} = \mathcal{R}_{22}$.

Turning now to the forward time recursions (37)-(39) for the Kalman filtered state covariance $P_{t|t}$, a recursion is sought which propagates a square root $P_{t|t}^{1/2}$. Again, a $\mathcal{QR}$ factorisation is useful, this time of the form

$$\begin{bmatrix} R^{1/2} & CP_{t|t-1}^{1/2} \\ 0 & P_{t|t-1}^{1/2} \end{bmatrix}^T = \mathcal{QR} = \mathcal{Q} \begin{bmatrix} \overline{\mathcal{R}}_{11} & \overline{\mathcal{R}}_{12} \\ 0 & \overline{\mathcal{R}}_{22} \end{bmatrix}, \tag{48}$$

where a similar procedure to the one used above reveals that $\overline{\mathcal{R}}_{22}^T = P_{t|t}^{1/2}$ and therefore that $P_{t|t} = \overline{\mathcal{R}}_{22}^T \overline{\mathcal{R}}_{22}$ satisfies (39). Finally, the factorisation (48) requires the matrix square root $P_{t|t-1}^{1/2}$, and for this purpose the factorisation

$$\begin{bmatrix} P_{t-1|t-1}^{T/2} \overline{A}^T \\ \overline{Q}^{T/2} \end{bmatrix} = \mathcal{Q} \begin{bmatrix} \widetilde{\mathcal{R}}_1 \\ 0 \end{bmatrix} \tag{49}$$

can be performed so that by (37)

$$\widetilde{\mathcal{R}}_1^T \widetilde{\mathcal{R}}_1 = \overline{A} P_{t-1|t-1} \overline{A}^T + \overline{Q} = P_{t|t-1}.$$

Thus $P_{t|t-1}^{1/2} = \widetilde{\mathcal{R}}_1^T$.

4.2 The Robust M-Step

Note that a key aspect of the preceding robust implementation of the E-step is that the square root terms can be used to form matrices Σ and Φ (see equations (22) through (23)) that are guaranteed to be non-negative and symmetric, even in the presence of finite precision computation.

Since Σ is square, then standard pivoting and Gaussian elimination (as, for example, implemented by the Matlab / operator) provides a numerically robust means for computing A, B, C and D according to (43).

The computation of the Q, R and S matrix updates, again according to equation (44) requires more careful thought. For example, due to the subtractions involved in (44), positive semi-definiteness could easily be lost via any naïve implementation. Furthermore, it is again important to guarantee symmetry. An ad-hoc method for addressing this problem would be to implement (44) directly, and then ensure symmetry of Π by averaging it with its transpose. This, however, only ensures symmetry, and not positive semi-definiteness.

With this in mind, the paper here develops a numerically robust means for computing (44) via performing the following Cholesky factorisation

$$\begin{aligned} \begin{bmatrix} \Sigma & \Psi^T \\ \Psi & \Phi \end{bmatrix} &= \begin{bmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ 0 & \mathcal{R}_{22} \end{bmatrix}^T \begin{bmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ 0 & \mathcal{R}_{22} \end{bmatrix} \\ &= \begin{bmatrix} \mathcal{R}_{11}^T \mathcal{R}_{11} & \mathcal{R}_{11}^T \mathcal{R}_{12} \\ \mathcal{R}_{12}^T \mathcal{R}_{11} & \mathcal{R}_{12}^T \mathcal{R}_{12} + \mathcal{R}_{22}^T \mathcal{R}_{22} \end{bmatrix}, \end{aligned} \tag{50}$$

where all matrices are partitioned conformally to the left-hand side. Equating the lower right sub-matrices yields

$$\Phi = \mathcal{R}_{12}^T\mathcal{R}_{12} + \mathcal{R}_{22}^T\mathcal{R}_{22} = R_{12}^T\mathcal{R}_{11}\left(\mathcal{R}_{11}^T\mathcal{R}_{11}\right)^{-1}\mathcal{R}_{11}^T\mathcal{R}_{12} + \mathcal{R}_{22}^T\mathcal{R}_{22}.$$

This implies that Π may be expressed in terms of Cholesky factors as

$$\Pi = \Phi - \Psi\Sigma^{-1}\Psi^T = \Phi - R_{12}^T\mathcal{R}_{11}\left(\mathcal{R}_{11}^T\mathcal{R}_{11}\right)^{-1}\mathcal{R}_{11}^T\mathcal{R}_{12} = \mathcal{R}_{22}^T\mathcal{R}_{22}.$$

That is, Q, R and S may be calculated at each iteration using the formula $\Pi = \mathcal{R}_{22}^T\mathcal{R}_{22}$. This approach simultaneously guarantees both the symmetry and non-negative definiteness of the result.

In relation to this, in the interests of maximum robustness, it is important to employ a Cholesky factorisation method that (unlike that native to Matlab) can cope with rank deficient matrices. For this purpose, the experience of the authors is that either of the methods presented in [16] (as Algorithm 4.2.4) or in [45] are appropriate.

Finally, the computation of P_1 in a manner guaranteeing its symmetry and positive semi-definiteness is trivial since $P_1^{1/2} = P_{t|N}^{1/2}$ is already calculated in the E-step (see equation (47)).

4.3 The Robust Algorithm

Summarising the ideas in Section 4.1 and 4.2 we present the following numerically robust algorithm, which is a central contribution of this paper.

EM Algorithm 4.2. *Numerically Robust EM-based Algorithm*

1. *Let $k = 0$ and initialise estimates at $\widehat{\theta}_0 = [A, B, C, D, Q, R, S, P_1, \mu]$;*
2. *Using the system specification $\widehat{\theta}_k = [A, B, C, D, Q, R, S, P_1, \mu]$, compute, for $t = 1, \ldots, N$, the sequences $\{\widehat{x}_{t|t}\}$ $\{P_{t|t-1}^{1/2}\}$ and $\{P_{t|t}^{1/2}\}$ via $\mathcal{QR}-$decompositions*

$$\begin{bmatrix} P_{t-1|t-1}^{T/2}\overline{A}^T \\ \overline{Q}^{T/2} \end{bmatrix} = \mathcal{Q}\begin{bmatrix} \mathcal{R}_1 \\ 0 \end{bmatrix}, \qquad P_{t|t-1}^{1/2} = \mathcal{R}_1^T$$

$$\begin{bmatrix} R^{1/2} & CP_{t|t-1}^{1/2} \\ 0 & P_{t|t-1}^{1/2} \end{bmatrix}^T = \mathcal{Q}\begin{bmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ 0 & \mathcal{R}_{22} \end{bmatrix}, \qquad P_{t|t}^{1/2} = \mathcal{R}_{22}^T$$

and the Kalman filter recursions with initialisation $\widehat{x}_{1|0} = \mu$, $P_{1|0} = P_1$

$$\begin{aligned} K_t &= P_{t|t-1}^{1/2}P_{t|t-1}^{T/2}C^T\left(CP_{t|t-1}^{1/2}P_{t|t-1}^{T/2}C^T + R\right)^{-1} \\ \widehat{x}_{t|t} &= (I - K_tC)\left(\overline{A}\widehat{x}_{t-1|t-1} + \begin{bmatrix} \overline{B} & SR^{-1} \end{bmatrix}\begin{bmatrix} u_{t-1} \\ y_{t-1} \end{bmatrix}\right) + K_t\left(y_t - Du_t\right), \end{aligned}$$

where

$$\overline{A} \triangleq A - SR^{-1}C, \qquad \overline{B} \triangleq B - SR^{-1}D, \qquad \overline{Q} \triangleq Q - SR^{-1}S^T. \tag{51}$$

3. *Compute, for $t = N, \ldots, 1$, the smoothed sequences $\{\widehat{x}_{t|N}\}$ and $\{P^{1/2}_{t|N}\}$ via $\mathcal{QR}-$decomposition*

$$\begin{bmatrix} P^{T/2}_{t|t}\overline{A}^T & P^{T/2}_{t|t} \\ \overline{Q}^{T/2} & 0 \\ 0 & P^{T/2}_{t+1|N}J^T_t \end{bmatrix} = \mathcal{Q}\begin{bmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ 0 & \mathcal{R}_{22} \\ 0 & 0 \end{bmatrix}, \qquad P^{1/2}_{t|N} = \mathcal{R}^T_{22}$$

and the recursion

$$\widehat{x}_{t|N} = \widehat{x}_{t|t} + J_t\left(\widehat{x}_{t+1|N} - \overline{A}\widehat{x}_{t|t} - \begin{bmatrix} \overline{B} & SR^{-1} \end{bmatrix}\begin{bmatrix} u_t \\ y_t \end{bmatrix}\right),$$

where $J_t \triangleq P_{t|t}\overline{A}^T P^{-1}_{t+1|t}$. Also calculate the sequence $\{M_{t|N}\}$ via equation (35) *with initialisation* (42) *for $t = N, \ldots, 2$.*

4. *Calculate the matrices*

$$\begin{aligned} \mathbf{E}_{\theta'}\{y_t x^T_t \mid Y_N, U_N\} &= y_t\widehat{x}^T_{t|N}, \\ \mathbf{E}_{\theta'}\{x_t x^T_t \mid Y_N, U_N\} &= \widehat{x}_{t|N}\widehat{x}^T_{t|N} + P^{1/2}_{t|N}P^{T/2}_{t|N}, \\ \mathbf{E}_{\theta'}\{x_t x^T_{t-1} \mid Y_N, U_N\} &= \widehat{x}_{t|N}\widehat{x}^T_{t-1|N} + M_{t|N}, \end{aligned}$$

and construct Φ, Ψ and Σ as described in equations (22) *and* (23)*.*

5. *Maximise $\mathcal{Q}(\theta, \widehat{\theta}_k)$ with respect to θ to obtain a new estimate $\widehat{\theta}_{k+1}$ as follows.*

   (a) *Compute new estimates of A, B, C and D via equation* (43) *using pivoting and Gaussian elimination.*

   (b) *Use the the modified Cholesky factorisation [45] to calculate new estimates of Q, R and S as*

$$\begin{bmatrix} \Sigma & \Psi^T \\ \Psi & \Phi \end{bmatrix}^{T/2} = \begin{bmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ 0 & \mathcal{R}_{22} \end{bmatrix},$$

$$\Pi = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = \mathcal{R}^T_{22}\mathcal{R}_{22}.$$

   (c) *Set $\mu = \widehat{x}_{1|N}$ and $P_1 = P^{1/2}_{1|N}P^{T/2}_{1|N}$, where $\widehat{x}_{1|N}$ and $P_1 = P^{1/2}_{1|N}$ are computed via Kalman smoothing in the E-step.*

6. *If the associated likelihood sequence $\{L(\widehat{\theta}_k)\}$ has converged via some measure (e.g. relative decrease below a threshold) terminate, otherwise increment k and return to step 2.*

5 Algorithm Properties

With this definition of a numerically robust version of the EM-based algorithm in hand, the paper now proceeds to investigate some of its convergence properties via a series of theoretical and empirical studies. For the purposes of theoretical analysis, the following standing assumptions will be imposed.

Standing Assumptions 5.1.

1. *The set of candidate parameter vectors, Θ, is a closed and bounded hypercube in $\mathbf{R}^d$;*
2. *Both $L(\theta)$ and $\mathcal{Q}(\theta, \theta')$ are bounded and continuous for all $\theta, \theta' \in \Theta$.*

5.1 Uniqueness of Iterations

Minimally-parametrized model structures, (those for which each input-output system behaviour corresponds to only one point in parameter space) have traditionally been favoured in system identification.

By way of contrast, the model structure used by the EM algorithm is clearly not minimally-parametrized since any similarity transformation of the state-space matrices in equation (11) leads to another potentially valid model with the same input-output behaviour and thus the same likelihood value. It is therefore natural to question whether, $\{\widehat{\theta}_k\}$, the sequence of estimates generated by EM Algorithm 4.2, are well defined and if so what their convergence properties are. We begin with the issue of well-posedness.

Theorem 5.1. *Suppose that $\widehat{\theta}_k$ parametrizes a controllable and observable system with $\Pi, P_1 > 0$, and that for the given data length N, the input sequence $\{u_t\}$ satisfies*

$$\frac{1}{N}\sum_{t=1}^{N} u_t u_t^T > 0. \tag{52}$$

Then Σ, defined by equation (23)*, is positive definite and $\widehat{\theta}_{k+1}$ is uniquely defined.*

Proof. See Appendix A. □

Clearly, the condition (52) can be considered a "persistence of excitation" requirement which forces the input sequence to be informative.

5.2 Limit Points of the Algorithm

With the well-posedness of the definition of the algorithm iterates $\{\widehat{\theta}_k\}$ established, the paper now turns to the question of convergence. In relation to this, there exist some general results on the convergence of EM algorithm iterates, which are scattered among a number of papers in the statistical literature (see, for example, [4, 11, 32, 33, 46]). In what follows, we draw on these results in a manner tailored to the particular case of Algorithms 4.1 and 4.2. To begin, we establish the important property that Algorithms 4.1 and 4.2 generate a sequence of estimate iterates $\{\widehat{\theta}_k\}$ for which the associated sequence of likelihoods $\{L(\widehat{\theta}_k)\}$ is non-decreasing.

Theorem 5.2. *Let $\widehat{\theta}_{k+1}$ be generated from $\widehat{\theta}_k$ by an iteration of EM Algorithm 4.2. Then*

$$L(\widehat{\theta}_{k+1}) \geq L(\widehat{\theta}_k) \qquad \forall k = 0, 1, 2, \ldots \tag{53}$$

with equality if and only if both

$$\mathcal{Q}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) = \mathcal{Q}(\widehat{\theta}_k, \widehat{\theta}_k) \tag{54}$$

and

$$p_{\widehat{\theta}_{k+1}}(X \mid Y, U) = p_{\widehat{\theta}_k}(X \mid Y, U) \tag{55}$$

for almost all X.

Proof. See also [11]. Equation (6) yields

$$L(\widehat{\theta}_{k+1}) - L(\widehat{\theta}_k) = \left[\mathcal{Q}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) - \mathcal{Q}(\widehat{\theta}_k, \widehat{\theta}_k)\right] + \left[\mathcal{V}(\widehat{\theta}_k, \widehat{\theta}_k) - \mathcal{V}(\widehat{\theta}_{k+1}, \widehat{\theta}_k)\right] \quad (56)$$

$$\geq \mathcal{V}(\widehat{\theta}_k, \widehat{\theta}_k) - \mathcal{V}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) \quad (57)$$

where (57) follows from (56) upon using the assumption that $\{\widehat{\theta}_k\}$ is generated by an EM algorithm. Via the definition of $\mathcal{V}(\cdot, \widehat{\theta}_k)$

$$\mathcal{V}(\widehat{\theta}_k, \widehat{\theta}_k) - \mathcal{V}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) = \int \log\left(\frac{p_{\widehat{\theta}_k}(X|X_N, Y_N)}{p_{\widehat{\theta}_{k+1}}(X|Y_N)}\right) p_{\widehat{\theta}_k}(X|Y_N)\,\mathrm{d}X. \quad (58)$$

Furthermore, $\log x \leq x - 1$ with equality if, and only if, $x = 1$. Consequently, since the area under $p_\theta(X|Y)$ is one for any value of θ,

$$\begin{aligned}\mathcal{V}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) - \mathcal{V}(\widehat{\theta}_k, \widehat{\theta}_k) &\leq \int \left(\frac{p_{\widehat{\theta}_{k+1}}(X|Y)}{p_{\widehat{\theta}_k}(X|Y)} - 1\right) p_{\widehat{\theta}_k}(X|Y)\,\mathrm{d}X \\ &= \int p_{\widehat{\theta}_{k+1}}(X|Y)\,\mathrm{d}X - \int p_{\widehat{\theta}_k}(X|Y)\,\mathrm{d}X = 0\end{aligned}$$

which establishes the inequality (53) and hence the important likelihood monotonicity property of the algorithm. Establishing the necessary and sufficient conditions (54) (55) for equality in (53) involves recognising (54) is clear. Equation (55) however requires a more elaborate argument which involves application of Lemma C.3 to (58). □

Note that while this shows that Algorithms 4.1 and 4.2 will never lead to a new estimate $\widehat{\theta}_{k+1}$ with a likelihood lower than a preceding one $\widehat{\theta}_k$, it also establishes that in the more usual case when the maximisation step yields $\mathcal{Q}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) > \mathcal{Q}(\widehat{\theta}_k, \widehat{\theta}_k)$ then in fact the underlying likelihood is strictly increased; i.e. $L(\widehat{\theta}_{k+1}) > L(\widehat{\theta}_k)$. Furthermore, note that this fundamental property of the algorithm is a direct consequence of the essential likelihood function property.

$$\int p_\theta(x)\mathrm{d}x = 1, \qquad \forall\theta. \quad (59)$$

It may be significant that this, and hence the likelihood monotonicity of the EM algorithm, holds even when the likelihood function is discontinuous, and hence even in situations for which gradient based search methods are inapplicable.

To continue the analysis, notice that Standing Assumptions 5.1 together with a requirement that $\Pi > 0$ ensure that the likelihood function $L(\cdot)$ (as defined by equation (17)) is bounded on Θ. Since $\{L(\widehat{\theta}_k)\}$ is monotonically increasing it is clear that this sequence must converge. The values to which is does converge are established by the following lemma.

Lemma 5.1. *Let $\{\widehat{\theta}_k\} \subseteq \Theta$ be a sequence of estimates generated by EM Algorithms 4.1 or 4.2 for which each element parametrizes a controllable and observable system with $\Pi, P_1 > 0$. Then a limit point of $\{\widehat{\theta}_k\}$, $\theta^\star$, is a stationary point of $L(\theta)$ and the sequence $\{L(\widehat{\theta}_k)\}$ converges monotonically to $L(\theta^\star)$.*

Proof. Denote the following sets:

$$\begin{aligned}\mathcal{Y}_\beta &\triangleq \{(\theta, L(\theta)) : \theta \in \Theta, L(\theta) \leq L(\beta)\}, \\ \mathcal{Z}_\beta &\triangleq \{(\theta, L(\theta)) : \theta \in \Theta\}, \\ \mathcal{X}_\beta &\triangleq \{(\theta, L(\theta)) : \theta \in \Theta, L(\theta) \geq L(\beta)\}.\end{aligned}$$

Since it is a fundamental property of continuous functions (see Lemma 10.12 of [15]) that the sets $\mathcal{Y}_\beta$ and $\mathcal{Z}_\beta$ are closed, then it follows that any finite intersections and unions of them are also closed (Theorem 4.18 of [14]). In particular,

$$\mathcal{X}_\beta = \mathcal{Y}_\beta \backslash (\mathcal{Y}_\beta \cap \mathcal{Z}_\beta)$$

is a closed subset in $\mathbf{R}^d \times \mathbf{R}$ and hence compact.

Via this and the assumption that $\mathcal{Q}(\theta, \theta')$ is continuous in both arguments, all the conditions of Theorem 2 of [46] are satisfied. Application of this results then completes the proof. □

Note that although this result establishes that the sequence of likelihoods $\{L(\widehat{\theta}_k)\}$ is convergent to a value $L(\theta^\star)$ associated with (local) maximiser $\theta^\star$ of $L(\theta)$, this does not immediately imply that the parameter estimates $\{\widehat{\theta}_k\}$ themselves converge to that local maximiser $\theta^\star$.

In particular, since a non-minimal parametrization is involved, it is natural to be concerned that the algorithm could lead to a "wandering" of parameter estimates that are parameter-space different, but system-wise equivalent, and hence all implying the same likelihood $L(\theta^\star)$. In fact, this is not the case, since as soon the sequence $\{\widehat{\theta}_k\}$ arrives at a stationary point $\theta^\star$ of $L(\theta)$, then all further iterations of the algorithm remain at this same *parameter-space* stationary point.

Corollary 5.1. *Let $\widehat{\theta}_k$ parametrize a controllable and observable system with $\Pi, P_1 > 0$. Suppose that the input sequence U_N satisfies the condition* (52)*. Then*

$$L(\widehat{\theta}_{k+1}) \geq L(\widehat{\theta}_k) \tag{60}$$

with equality if and only if $\widehat{\theta}_{k+1} = \widehat{\theta}_k$.

Proof. According to Theorem 5.2, we need only show that $\widehat{\theta}_{k+1} = \widehat{\theta}_k$ if and only if both

$$\mathcal{Q}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) = \mathcal{Q}(\widehat{\theta}_k, \widehat{\theta}_k) \qquad \text{and} \qquad p_{\widehat{\theta}_{k+1}}(X \mid Y, U) = p_{\widehat{\theta}_k}(X \mid Y, U)$$

for almost all X. Clearly, the "only if" part is trivial. To address the "if" component, note that if $\mathcal{Q}(\widehat{\theta}_{k+1}, \widehat{\theta}_k) = \mathcal{Q}(\widehat{\theta}_k, \widehat{\theta}_k)$ then according to the definition of the EM Algorithm, both $\widehat{\theta}_{k+1}$ and $\widehat{\theta}_k$ must be maximisers of the function $\mathcal{Q}(\cdot, \widehat{\theta}_k)$. On the other hand, Theorem 5.1 demonstrates that $\mathcal{Q}(\cdot, \widehat{\theta}_k)$ has only one maximiser. Thus $\widehat{\theta}_{k+1} = \widehat{\theta}_k$. □

5.3 Rate of Convergence of Parameter Estimates

The previous results establish the convergence properties of EM Algorithm 4.1 (and its numerically robust version Algorithm 4.2) about a stationary point $\theta^\star$ of the likelihood. This section now turns to understanding the convergence rate of Algorithm 4.2 and the factors affecting it.

This is a more difficult question to address, but to try to gain insight an analysis locally about a stationary point $\theta^\star$ of $L(\theta)$ provides the following approximate expression characterising the evolution of the "estimation error" $\widehat{\theta}_k - \theta^\star$.

Theorem 5.3. *Suppose that $\widehat{\theta}_k$ parametrizes a controllable and observable system with $\Pi, P_1 > 0$ and that $\theta^\star$ is a stationary point of the likelihood function $L(\cdot)$. Then*

$$\widehat{\theta}_{k+1} - \theta^\star = \left[I - \mathcal{I}_{XY}^{-1}(\widehat{\theta}_k) \mathcal{I}_Y(\widehat{\theta}_k) \right] (\widehat{\theta}_k - \theta^\star) + o(\|\widehat{\theta}_{k+1} - \widehat{\theta}_k\|^2) + o(\|\widehat{\theta}_k - \theta^\star\|^2) \tag{61}$$

where

$$\mathcal{I}_Y(\widehat{\theta}_k) \triangleq -\left.\frac{\partial^2}{\partial\theta\partial\theta^T}\log p_\theta(Y|U)\right|_{\theta=\widehat{\theta}_k} \tag{62}$$

and

$$\mathcal{I}_{XY}(\widehat{\theta}_k) \triangleq -\mathbf{E}_{\widehat{\theta}_k}\left\{\left.\frac{\partial^2}{\partial\theta\partial\theta^T}\log p_\theta(X,Y\mid U)\right|Y,U\right\}\Big|_{\theta=\widehat{\theta}_{k+1}} = -\left.\frac{\partial^2}{\partial\theta\partial\theta^T}\mathcal{Q}(\theta,\widehat{\theta}_k)\right|_{\theta=\widehat{\theta}_{k+1}}. \tag{63}$$

The latter is guaranteed to be invertible provided input condition (52) *is satisfied.*

Proof. See Appendix B. □

This establishes that locally around a stationary point $\theta^\star$, the EM algorithm proposed in this paper exhibits an estimation error $\widehat{\theta}_k - \theta^\star$ that approximately evolves according to the autonomous system given by (61) with order $o(\cdot)$ terms neglected. In particular, this suggests that convergence locally around $\theta^\star$ will be rapid if the matrices $\mathcal{I}_{XY}(\widehat{\theta}_k)$ and $\mathcal{I}_Y(\widehat{\theta}_k)$ are similar.

In relation to this, note that according to (6) and the fact that $\mathcal{V}(\theta',\theta') - \mathcal{V}(\theta,\theta')$ is a Kullback–Liebler distance, then $L(\theta) = \mathcal{Q}(\theta,\theta)$ and hence $\mathcal{Q}(\theta,\theta)$ mimics the *value* of the likelihood $L(\theta)$. Theorem 5.3 now further establishes that if $\mathcal{Q}(\theta,\theta)$ is also a good approximation to the second order properties of the likelihood function, that is if $\mathcal{I}_{XY}(\widehat{\theta}_k) \approx \mathcal{I}_Y(\widehat{\theta}_k)$, then local convergence will be rapid. Moreover, at least locally about $\theta^\star$, the rate of parameter convergence will be governed by the smallest eigenvalue of $\mathcal{I}_{XY}^{-1}(\theta^\star)\mathcal{I}_Y(\theta^\star)$ – the larger the dominating eigenvalue the faster the parameters will converge.

To study the relationship between $\mathcal{I}_{XY}$ and $\mathcal{I}_Y$, note that as a consequence of (6),

$$\mathcal{I}_Y(\theta^\star) = \mathcal{I}_{XY}(\theta^\star) - \mathcal{I}_X(\theta^\star), \tag{64}$$

where

$$\mathcal{I}_X(\theta^\star) \triangleq -\mathbf{E}_{\theta^\star}\left\{\left.\frac{\partial^2}{\partial\theta\partial\theta^T}\log p_\theta(X\mid Y,U)\right|Y,U\right\}\Big|_{\theta=\theta^\star}$$

is the "missing data" X information matrix. All three information matrices are positive semi-definite by construction and hence regardless of the nature of the missing data X

$$\mathcal{I}_Y(\theta^\star) \leq \mathcal{I}_{XY}(\theta^\star). \tag{65}$$

Intuitively then, the more assistance the missing data provides in the solution of an EM iteration the less effective that iteration will be.

Furthermore, since $\mathcal{I}_{XY}(\theta^\star)$ and $\mathcal{I}_Y(\theta^\star)$ are both positive definite by construction, an immediate consequence of (65) is that the eigenvalues of $I - \mathcal{I}_{XY}^{-1}(\theta^\star)\mathcal{I}_Y(\theta^\star)$ must be real and lie in the interval $[0,1]$ since

$$\lambda_i\{I - \mathcal{I}_{XY}^{-1}(\theta^\star)\mathcal{I}_Y(\theta^\star)\} = 1 - \underbrace{\lambda_i\{\mathcal{I}_Y^{T/2}(\theta^\star)\mathcal{I}_{XY}^{-1}(\theta^\star)\mathcal{I}_Y^{1/2}(\theta^\star)\}}_{\in[0,1]} \in [0,1], \tag{66}$$

where $\lambda_i\{M\}$ denotes the i^{th} eigenvalue of a matrix M and $\mathcal{I}_Y^{1/2}(\theta^\star)$ is a positive definite matrix such that $\mathcal{I}_Y(\theta^\star) = \mathcal{I}_Y^{1/2}(\theta^\star)\mathcal{I}_Y^{T/2}(\theta^\star)$. This implies that locally around $\theta^\star$ the EM algorithm can be expected to display an exponential convergence rate.

6 Computational Load

The final performance issue to be addressed is that of the computational load of the proposed EM Algorithms 4.1 and 4.2. This will be addressed by comparing the load relative to the main alternative method for ML estimation of MIMO systems, which is that provided by recent methods of gradient search via data driven local co-ordinates (DDLC) [2, 9, 26, 30, 29, 38].

In addressing computational overhead, there are two key issues. Firstly, there is the requirements per iteration, which we will measure here in terms of necessary floating point operations (FLOPs). Secondly, it is important to consider the number of iterations. Unfortunately, it is impossible to be precise with regard to the latter. The experience of the authors is that (as will be demonstrated in the next section) the number of iterations required by the EM methods proposed here, are typically (roughly) equal to the number required by the alternative DDLC method.

In consideration of this, the paper will profile computational requirement via FLOP load per iteration, and for this purpose a detailed audit of the FLOP count for the key stages of the robust EM Algorithm 4.2 are provided in Table 1 (Recall m is the number of inputs, p is the number of outputs, n is the model state dimension, and N is the data length) where the FLOP counts provided are for the computation of all N quantities such as $P_{t|t}^{1/2}$ where necessary. Assuming the typical case of state

Computation	Equations	Number of FLOPs required
$\{P_{t\|t-1}^{1/2}\}$	(49)	$\frac{10}{3}n^3N$
$\{P_{t\|t}^{1/2}\}$	(48)	$\frac{32}{3}n^3N$
$\{K_t\}$	(38)	$(7pn+\frac{1}{3}p^2+\frac{5}{2}p+2n^2)pN$
$\{\widehat{x}_{t\|t}\}$	(40),(41)	$(3n+p+8np+2n^2+2nm+2mp)N$
$\{P_{t\|N}^{1/2}\}$	(45)	$\frac{56}{3}n^3N$
$\{J_t\}$	(32)	$(4n+2)n^2N$
$\{\widehat{x}_{t\|N}\}$	(33)	$(3+2m+4n+2p)nN$
$\{M_{t\|N}\}$	(35)	$(8n+2)n^2N$
Φ,Ψ,Σ	(22), (23)	$(9n^2+m^2+6nm+6np+2pm+2p^2)N$
Γ	(43)	$(m+n)^2(2+\frac{7}{3}n+\frac{1}{3}m+2p)$
Π	(44)	$\frac{1}{3}(2n+m+p)^3+(n+p)^3$

Table 1: *FLOP count for each of the key steps of the robust Algorithm 4.2.*

dimension n being of the order (or larger) than the input/output dimensions m, p then indicates that the total FLOP count per iteration required by Algorithm 4.2 is $O(n^3N)$.

Turning now to a DDLC approach, we refer the reader to [2, 26, 30, 29] for the details of the method. The essential point is that at each iteration, since the methods involved are gradient-search based, then the gradient itself needs to be computed for each of the components parametrizing the $n(m+2p)+mp$ dimensional manifold of minimal representations. This involves running an n state filter, with FLOP count $O(n^2N)$ (there is no sparsity in any of the matrices involved) for each of these $n(m+2p)+mp$ parameters, which leads to a FLOP count of $O(\max(p,m)n^3N)$.

For low numbers of inputs and outputs, these FLOP counts of $O(n^3N)$ and $O(\max(p,m)n^3N)$ for (respectively) Algorithm 4.2 and a DDLC gradient search approach are roughly equivalent, and establish Algorithm 4.2 as competitive from a computational load point of view. However, for larger numbers of inputs and outputs, there can be an appreciable difference in the FLOP count due to the DDLC count being the Algorithm 4.2 count of n^3N scaled by $\max(p,m)$. As such, and also in con-

sideration of the numerical robustness we have observed empirically, we suggest that Algorithm 4.2 should especially be considered for the case of high dimension MIMO estimation.

7 Empirical Study

Having derived the robust Algorithm 4.2 and provided a theoretical analysis of its properties, this section provides a brief empirical study of the performance of the EM methods proposed here. To begin with, we compare the results obtained by Algorithm 4.2 to those provided by the main alternatives, which are subspace-based techniques, and data driven local co-ordinate (DDLC) gradient search approaches.

The comparison method employed here is that employed in [27] for a similar profiling purpose, and wherein a Monte–Carlo approach is used. To be more specific, our first empirical study involves the simulation and estimation of 250 different, stable, two-input, two-output, $5^{th}-$order systems in innovations form that were generated randomly using the Matlab `drss()` command. For each of these 250 estimation experiments, a new length $N = 500$ i.i.d. input realisation $u_t \sim \mathcal{N}(0, I)$, and a new i.i.d. innovations realisation $e_t \sim \mathcal{N}(0, 0.04I)$ was generated.

Three estimates were formed for each of these 250 data sets. Firstly, a (CVA-weighted) subspace estimate [24] was found using the Matlab command `n4sid()` with the options `Focus` and `N4Weight` set to 'Prediction' and 'CVA', respectively. Secondly, a DDLC based gradient search, initialised at the subspace estimate, was used to find a Maximum-Likelihood estimate as proposed in [29] and implemented via the Matlab `pem()` function (with the algorithm's `Tolerance` set to 0, and `Focus` to Prediction). Finally, a further Maximum-Likelihood estimate was found via the robust EM Algorithm 4.2. In the latter case the algorithm was initialised with the subspace-based estimate and with the arbitrarily selected noise covariance estimates of $Q = I$ and $R = 0.2I$. Both iterative algorithms were forced to run for 100 iterations.

For each of these three estimation methods, and for each of 250 estimation experiments, the mean prediction error cost associated with the estimated model of

$$E_N(\widehat{\theta}_k) \triangleq \frac{1}{pN}\sum_{t=1}^{N}(y_t - \widehat{y}_{t|t-1})^T(y_t - \widehat{y}_{t|t-1}), \tag{67}$$

was computed, where the term $\widehat{y}_{t|t-1}$ (implicitly a function of θ) above is defined by equation (18).

The performance of Algorithm 4.2, as measured by $E_N(\widehat{\theta}_k)$, and relative to subspace and DDLC gradient search is then illustrated in figure 1. There, each star, represents one data set, and two estimation experiments, with the co-ordinates of each star being set by the value of $E_N(\widehat{\theta}_k)$ for each of two estimation methods. A star on the indicated diagonal line then represents equal cost $E_N(\widehat{\theta}_k)$ for both methods, while a star below the line represents lower cost (and hence superiority in a prediction error sense) for the methods indicated on the vertical axis.

With this in mind, Figure 1(a) illustrates that following an initial subspace estimation step by a further EM algorithm refinement can often lead to a final estimate that is superior in a mean-square prediction error sense. Of course, under Gaussian conditions, it will also lead to an estimate that is superior in a maximum likelihood sense as the preceding theoretical analysis has established.

Figure 1(b) illustrates a further point. Namely, in terms of choosing between either a DDLC co-ordinate gradient search, or the EM-based Algorithm 4.2 as a means of refining an initial subspace-based estimate, then Algorithm 4.2 is quite competitive; there were many realisations were the EM method led to substantially lower cost than DDLC, and only a few where EM led to higher cost, in which cases the relative difference was smaller (than in the reverse case).

Of course, no general conclusion can be drawn on the basis of specific, even Monte–Carlo based, simulation examples. However, the authors have noted that the EM-based search appears to be particularly robust in avoiding capture in local minima, but at the expense of best-case convergence speed.

To provide an illustration of this, consider the case of a particular 8^{th}-order two input two output fixed multivariable system $[A, B, C, D]$ which is specified as

$$[A, B, C, D] \triangleq \mathcal{ZOH}\left\{\begin{bmatrix} \frac{1}{s^2+1.1s+0.1} & \frac{3}{s^2+2.5s+1} \\ \frac{1}{s^2+s+0.21} & \frac{1}{s^2+1.2s+0.32} \end{bmatrix}\right\} \tag{68}$$

$$= \begin{bmatrix} \frac{0.3550q^{-1}+0.2465q^{-2}}{1-1.2727q^{-1}+0.3329q^{-2}} & \frac{0.7092q^{-1}+0.3114q^{-2}}{1-0.7419q^{-1}+0.0821q^{-2}} \\ \frac{0.3619q^{-1}+0.2594q^{-2}}{1-1.2374q^{-1}+0.3679q^{-2}} & \frac{0.3397q^{-1}+0.2277q^{-2}}{1-1.1196q^{-1}+0.3012q^{-2}} \end{bmatrix} \tag{69}$$

and for which $N = 1000$ data points are simulated with i.i.d. input $u_t \sim \mathcal{N}(0, I)$, and i.i.d. measurement noise $e_t \sim \mathcal{N}(0, 0.125I)$. This simulation with different input and noise realisations was repeated until 100 "successful" DDLC co-ordinate gradient based search estimates were found from an initialisation of

$$\widehat{G}(q, \widehat{\theta}_0) \triangleq \begin{bmatrix} \frac{0.1}{q^2-q+0.25} & \frac{0.1}{q^2-1.40q+0.49} \\ \frac{0.1}{q^2-1.20q+0.36} & \frac{0.1}{q^2-0.80q+0.16} \end{bmatrix}.$$

Here, "successful" was taken to mean that the final prediction error variance was within 30% of the measurement noise variance, and for each data set an ML estimate was also found by EM-Algorithm 4.2. The sample-average trajectory of diminishing prediction error cost $E_N(\widehat{\theta}_k)$ over these 100 successful runs is shown in Figure 2.

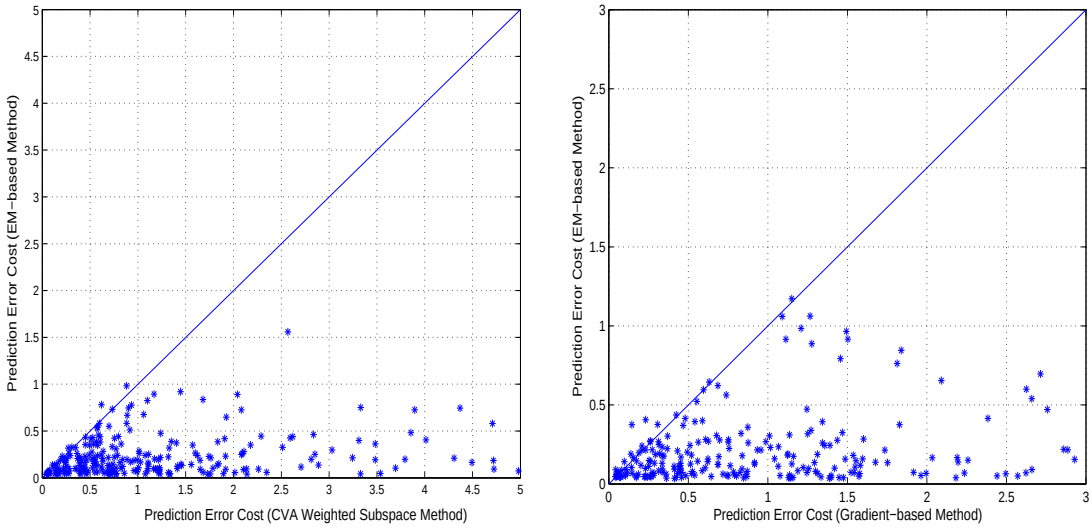
The left hand plot shows the case of Gaussian measurement noise, while the right hand plot illustrates uniformly distributed measurement noise. Note that in both cases, the best case DDLC gradient search performance over all the simulation runs was superior to the average behaviour of the EM algorithm.

However, there is a factor that balances this best case performance. Due to high variability in the DDLC gradient search performance over the simulation runs, the average performance, after "non successful" runs were censored from the averaging, is slightly inferior to that of the average EM algorithm convergence rate, for which there were no non-successful runs that required culling. Indeed, due to the observed low variability in EM algorithm behaviour, its average performance is representative of its worst case as well.

While again this simulation provides only one specific robustness example, the experience of the authors is that it appears to be a more general principle. Namely, in comparison to DDLC based gradient search, the EM algorithm presented in this paper is particularly reliable in its ability to avoid becoming locked in local minima, but perhaps at the cost of slower convergence rate.

A further robustness aspect is illustrated in the right hand plot of figure 2, wherein the case of uniform distributed measurement noise is presented. This is a situation where the assumptions underlying the ML criterion employed in this paper are violated, but nevertheless the robust EM algorithm presented here still converges reliably to an estimate that is the global minimiser of prediction error.

Note that in this case, no non-convergent trajectories were culled in computing the average EM behaviour but the 9% of the DDLC gradient search runs which did not converge were censored before the average trajectory illustrated in Figure 2(b) was computed. As before, the best-case performance of the gradient based scheme was significantly faster than the average for the EM algorithm.



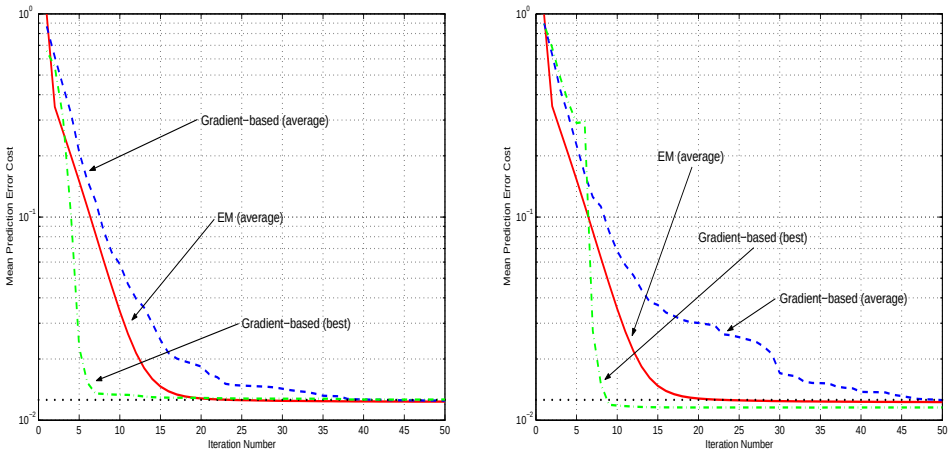
(a) (b)

Figure 1: *Simulation 1 - Monte-Carlo Study Results. The left-hand figure shows a comparison of the CCA weighted subspace and EM algorithm ML estimates. The right-hand figure shows a comparison of DDLC parametrized gradient-search versus EM algorithm ML estimates.*

8 Conclusions

The contribution of this paper is to derive a numerically robust and algorithmically reliable method for the estimation of possibly high dimension LTI MIMO systems. The key principle underlying the methods proposed here is the Expectation-Maximisation technique. Attractive features of the algorithm derived and studied here include numerical robustness, ability to deal with non-smooth likelihood surfaces, ease of initial condition estimation, moderate computational cost which scales well with the number of model states and input-output dimension, and robustness to local-minima attraction.

Balancing this, the convergence rate of the EM methods derived here have been observed as being generally slower than the best case performance of the main alternative methods of DDLC parametrized gradient search, although the average behaviour of the two methods have been observed by the authors as basically equivalent.



(a) Normally-distributed noise

(b) Uniformly-distributed noise

Figure 2: *Simulation 2 - Sample average prediction-error cost $E_N(\widehat{\theta}_k)$ evolution for DDLC parametrized gradient search via Matlab 6.5's* `pem()` *function (dashed line) versus EM-Algorithm 4.2 (solid line)). Also shown is the best case performance of the* `pem()` *function (dot-dashed lines) and the global minimum value of the expected likelihood $\boldsymbol{E}\{L(\theta)\}$ (dotted line). Sub-figure (a) shows the case of Gaussian measurement noise and sub-figure (b) shows the case of uniformly distributed measurement noise.*

A Proof of Theorem 5.1

Proof. Define the following quantities:

$$\mathcal{A} \triangleq \frac{1}{N}\sum_{t=1}^{N} \widehat{x}_{t|N}\widehat{x}_{t|N}^T, \qquad \mathcal{B} \triangleq \frac{1}{N}\sum_{t=1}^{N} \widehat{x}_{t|N}u_t^T,$$

$$\mathcal{C} \triangleq \frac{1}{N}\sum_{t=1}^{N} u_t u_t^T, \qquad \mathcal{D} \triangleq \frac{1}{N}\sum_{t=1}^{N} P_{t|N}.$$

These allow Σ to be expressed as

$$\Sigma = \frac{1}{N}\sum_{t=1}^{N} \mathbf{E}_{\widehat{\theta}_k}\left\{ \begin{bmatrix} x_t \\ u_t \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix}^T \middle| Y_N, U_N \right\} = \begin{bmatrix} \mathcal{A}+\mathcal{D} & \mathcal{B} \\ \mathcal{B}^T & \mathcal{C} \end{bmatrix}.$$

By construction,

$$\begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{B}^T & \mathcal{C} \end{bmatrix} \geq 0$$

and therefore, by virtue of the fact that $\mathcal{C} > 0$,

$$\begin{bmatrix} \mathcal{A}-\mathcal{B}\mathcal{C}^{-1}\mathcal{B}^T & 0 \\ 0 & \mathcal{C} \end{bmatrix} = \begin{bmatrix} I & -\mathcal{B}\mathcal{C}^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{B}^T & \mathcal{C} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\mathcal{C}^{-1}\mathcal{B}^T & I \end{bmatrix} \geq 0.$$

Since the model is controllable and observable and Π is positive definite, Lemma C.4 proves that $\mathcal{D} > 0$. Therefore

$$\left[\begin{array}{cc} \mathcal{A} + \mathcal{D} - \mathcal{B}\mathcal{C}^{-1}\mathcal{B}^T & 0 \\ 0 & \mathcal{C} \end{array} \right] > 0$$

and thus

$$\left[\begin{array}{cc} \mathcal{A} + \mathcal{D} & \mathcal{B} \\ \mathcal{B}^T & \mathcal{C} \end{array} \right] = \left[\begin{array}{cc} I & \mathcal{B}\mathcal{C}^{-1} \\ 0 & I \end{array} \right] \left[\begin{array}{cc} \mathcal{A} + \mathcal{D} - \mathcal{B}\mathcal{C}^{-1}\mathcal{C}^T & 0 \\ 0 & \mathcal{C} \end{array} \right] \left[\begin{array}{cc} I & 0 \\ \mathcal{C}^{-1}\mathcal{B}^T & I \end{array} \right] > 0.$$

As a consequence, according to (43) and (44), θ_{k+1} is uniquely defined. □

B Proof of Theorem 5.3

Proof. A linear Taylor's expansion of $\frac{\partial L(\theta)}{\partial \theta}$ about $\widehat{\theta}_k$ provides

$$\frac{\partial L(\theta)}{\partial \theta} = \frac{\partial L(\widehat{\theta}_k)}{\partial \theta} + \frac{\partial^2 L(\widehat{\theta}_k)}{\partial \theta \partial \theta^T}(\theta - \widehat{\theta}_k) + o(\|\theta - \widehat{\theta}_k\|^2). \tag{70}$$

Letting $\theta = \theta^\star$ and noting that $\frac{\partial L(\theta^\star)}{\partial \theta} = 0$, then establishes that

$$\frac{\partial^2 L(\widehat{\theta}_k)}{\partial \theta \partial \theta^T}(\widehat{\theta}_k - \theta^\star) = \frac{\partial L(\widehat{\theta}_k)}{\partial \theta} + o(\|\theta^\star - \widehat{\theta}_k\|^2). \tag{71}$$

Also by Taylor expansion,

$$\begin{aligned} \left.\frac{\partial \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta}\right|_{\theta=\widehat{\theta}_k} &= \left.\frac{\partial \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta}\right|_{\theta=\widehat{\theta}_{k+1}} + \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_k - \widehat{\theta}_{k+1}) + o(\|\widehat{\theta}_{k+1} - \widehat{\theta}_k\|^2) \\ &= \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_k - \widehat{\theta}_{k+1}) + o(\|\widehat{\theta}_{k+1} - \widehat{\theta}_k\|^2), \end{aligned} \tag{72}$$

where the second line follows from the first on noticing that $\{\widehat{\theta}_k\}$ is generated by an EM algorithm. Now, according to Lemma C.5,

$$\frac{\partial L(\widehat{\theta}_k)}{\partial \theta} = \left.\frac{\partial \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta}\right|_{\theta=\widehat{\theta}_k}$$

so that by combining equations (71) and (72) we obtain

$$\frac{\partial^2 L(\widehat{\theta}_k)}{\partial \theta \partial \theta^T}(\widehat{\theta}_k - \theta^\star) = \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_k - \widehat{\theta}_{k+1}) + o(\|\widehat{\theta}_{k+1} - \widehat{\theta}_k\|^2) + o(\|\theta^\star - \widehat{\theta}_k\|^2).$$

Therefore,

$$\begin{aligned} \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_{k+1} - \theta^\star) &= \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_k - \theta^\star) + \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_{k+1} - \widehat{\theta}_k) \\ &\quad + o(\|\widehat{\theta}_{k+1} - \widehat{\theta}_k\|^2) + o(\|\theta^\star - \widehat{\theta}_k\|^2) \\ &= \left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \theta \partial \theta^T}\right|_{\theta=\widehat{\theta}_{k+1}} (\widehat{\theta}_k - \theta^\star) - \frac{\partial^2 L(\widehat{\theta}_k)}{\partial \theta \partial \theta^T}(\widehat{\theta}_k - \theta^\star) \\ &\quad + o(\|\widehat{\theta}_{k+1} - \widehat{\theta}_k\|^2) + o(\|\theta^\star - \widehat{\theta}_k\|^2). \end{aligned} \tag{73}$$

Since Lemma C.2 proves that $\frac{\partial^2 \mathcal{Q}(\theta,\widehat{\theta}_k)}{\partial\theta\partial\theta^T}\Big|_{\theta=\widehat{\theta}_{k+1}} > 0$, the result follows directly from equation (73). □

C Technical Lemmata

Lemma C.1. *Suppose $M, N \in \mathbf{R}^{n\times n}$ and M is invertible. Then*

$$\frac{\partial}{\partial M}\log\det M = M^{-T}, \qquad \frac{\partial}{\partial M}\mathrm{Tr}\{M^{-1}N\} = -M^{-T}N^TM^{-T}, \qquad \frac{\partial}{\partial M}\mathrm{Tr}\{MN\} = N^T.$$

Proof. See [18]. □

Lemma C.2. *Let the parameter vector $\widehat{\theta}_k$ parametrize a controllable and observable system with $\Pi, P_1 > 0$. Assume that the input condition* (52) *are satisfied. Then*

$$\begin{aligned} &\frac{\partial^2}{\partial\theta\partial\theta^T}\mathcal{Q}(\theta,\widehat{\theta}_k)\Big|_{\theta=\widehat{\theta}_{k+1}} \qquad (74)\\ &= \begin{bmatrix} -N\Sigma\otimes\Pi^{-1} & 0 & 0 & 0\\ 0 & -\frac{N}{2}\Pi^{-1}\otimes\Pi^{-1} & 0 & 0\\ 0 & 0 & -\frac{1}{2}P_{1|N}^{-1}\otimes P_{1|N}^{-1} & 0\\ 0 & 0 & 0 & -P_{1|N}^{-1}\end{bmatrix} > 0, \end{aligned}$$

where θ and $\mathcal{Q}(\theta,\widehat{\theta}_k)$ are defined by equations (15) *and* (21)*, the matrix Π is defined by equation* (43) *and Φ, Ψ and Σ are defined in equations* (22) *and* (23) *with $\theta' \triangleq \widehat{\theta}_k$.*

Proof. According to Lemma 3.1,

$$\begin{aligned} -2\mathcal{Q}(\theta,\theta') = \log\det P_1 + \mathrm{Tr}\left\{P_1^{-1}\mathbf{E}_{\theta'}\{(x_1-\mu)(x_1-\mu)^T \mid Y_N, U_N\}\right\}\\ +N\log\det\Pi + N\mathrm{Tr}\left\{\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\right\}. \end{aligned}$$

Now Lemma C.1 and the product rule provide

$$\begin{aligned} \frac{\partial\mathcal{Q}(\theta,\theta^\star)}{\partial\Gamma} &= -\frac{N}{2}\frac{\partial}{\partial\Gamma}\mathrm{Tr}\left\{\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\right\}\\ &= N\Pi^{-1}(\Psi - \Gamma\Sigma). \qquad (75)\end{aligned}$$

Using the identities (see [5])

$$\mathrm{vec}\{W_1W_2W_3\} = (W_3^T\otimes W_1)\mathrm{vec}\{W_2\}$$

and

$$(W_1\otimes W_2)(W_3\otimes W_4) = (W_1W_3)\otimes(W_2W_4),$$

where $\{W_i\}$ are appropriately sized matrices, we obtain from equation (75),

$$\begin{aligned} \frac{\partial\mathcal{Q}(\theta,\widehat{\theta}_k)}{\partial\mathrm{vec}\{\Gamma\}} &= N\mathrm{vec}\left\{\Pi^{-1}(\Psi-\Gamma\Sigma)\right\}\\ &= N\mathrm{vec}\left\{\Pi^{-1}\Psi\right\} - N(\Sigma\otimes\Pi^{-1})\mathrm{vec}\{\Gamma\} \qquad (76)\\ &= N\left[(\Psi-\Gamma\Sigma)^T\otimes I\right]\mathrm{vec}\left\{\Pi^{-1}\right\}. \qquad (77)\end{aligned}$$

Clearly, the partial derivative of equation (77) with respect to the parameters in μ and the P_1 matrix will equate to zero.

Straightforwardly from (76),

$$\left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \text{vec}\left\{\Gamma\right\} \partial \text{vec}\left\{\Gamma\right\}^T}\right|_{\theta=\widehat{\theta}_{k+1}} = -N\Sigma \otimes \Pi^{-1},$$

while equations (43) and (77) yield

$$\left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \text{vec}\left\{\Gamma\right\} \partial \text{vec}\left\{\Pi\right\}^T}\right|_{\theta=\widehat{\theta}_{k+1}} = (\Psi - \Psi\Sigma^{-1}\Sigma) \times \left.\frac{\partial \text{vec}\left\{\Pi^{-1}\right\}}{\partial \text{vec}\left\{\Pi\right\}^T}\right|_{\theta=\widehat{\theta}_{k+1}} = 0.$$

We now turn to calculating the partial derivatives of $\mathcal{Q}(\theta, \widehat{\theta}_k)$ with respect to the parameters of Π. Lemma C.1 proves that

$$\frac{\partial \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \Pi} = \frac{N}{2}\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\Pi^{-1} - \frac{N}{2}\Pi^{-1}$$

so that

$$\frac{\partial \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \text{vec}\left\{\Pi\right\}} = \frac{N}{2}\text{vec}\left\{\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\Pi^{-1} - \Pi^{-1}\right\}. \tag{78}$$

Lemmas C.1 and C.6, and the product rule then yield

$$\begin{aligned}
&\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \text{vec}\left\{\Pi\right\} \partial \text{vec}\left\{\Pi\right\}^T} \\
= \;& \frac{N}{2}\left(\left(\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\Pi^{-1}\right)^T \otimes I\right)\frac{\partial \text{vec}\left\{\Pi^{-1}\right\}}{\partial \text{vec}\left\{\Pi\right\}^T} + \\
& \frac{N}{2}\left(I \otimes \left(\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\right)\right)\frac{\partial \text{vec}\left\{\Pi^{-1}\right\}}{\partial \text{vec}\left\{\Pi\right\}^T} - \\
& \frac{N}{2}\frac{\partial \text{vec}\left\{\Pi^{-1}\right\}}{\partial \text{vec}\left\{\Pi\right\}^T} \\
= \;& \frac{N}{2}\Pi^{-1} \otimes \Pi^{-1} - \\
& \frac{N}{2}\left(\left(\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\Pi^{-1}\right)^T \otimes I\right)\left(\Pi^{-1} \otimes \Pi^{-1}\right) - \\
& \frac{N}{2}\left(I \otimes \left(\Pi^{-1}\left[\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T\right]\right)\right)\left(\Pi^{-1} \otimes \Pi^{-1}\right).
\end{aligned}$$

Finally, using equation (43) we obtain the identity

$$\begin{aligned}
\Phi - \Psi\Gamma^T - \Gamma\Psi^T + \Gamma\Sigma\Gamma^T &= \Phi - \Psi\Sigma^{-1}\Psi^T - \Psi\Sigma^{-1}\Psi^T + \Psi\Sigma^{-1}\Sigma\Sigma^{-1}\Psi^T \\
&= \Phi - \Psi\Sigma^{-1}\Psi^T \\
&= \Pi
\end{aligned}$$

so that

$$\left.\frac{\partial^2 \mathcal{Q}(\theta, \widehat{\theta}_k)}{\partial \text{vec}\left\{\Pi\right\} \partial \text{vec}\left\{\Pi\right\}^T}\right|_{\theta=\widehat{\theta}_{k+1}} = -\frac{N}{2}\Pi^{-1} \otimes \Pi^{-1}.$$

The elements of $\mathcal{Q}(\theta, \widehat{\theta}_k)\Big|_{\theta=\widehat{\theta}_{k+1}}$ associated with the matrix elements of P_1 and μ are computed in a similar manner.

In order to demonstrate the positivity of $\frac{\partial^2}{\partial\theta\partial\theta^T}\mathcal{Q}(\theta, \widehat{\theta}_k)\Big|_{\theta=\widehat{\theta}_{k+1}}$, notice that Σ is positive definite as a consequence of Theorem 5.1 and Π is positive definite and bounded above by virtue of the compactness assumption in Standing Assumptions 5.1. Finally, Lemma C.4 establishes that $P_{1|N}$ is also positive definite and bounded above. □

Lemma C.3. *Let f and g be non-negative and integrable functions with respect to a measure μ and S be the region in which $f > 0$. If $\int_S (f - g)\mathrm{d}\mu \geq 0$, then*

$$\int_S f \log\left(\frac{f}{g}\right) \mathrm{d}\mu \geq 0, \tag{79}$$

with equality if and only if $f = g$ almost everywhere.

Proof. The following is partially based on an argument presented in [37]. Equation (79) is proved as follows.

$$\begin{aligned}
\int_S f \log\left(\frac{f}{g}\right) \mathrm{d}\mu &= -\int_S f \log\left(\frac{g}{f}\right) \mathrm{d}\mu \\
&\geq -\int_S f \left(\frac{g}{f} - 1\right) \mathrm{d}\mu \\
&= \int_S (f - g)\mathrm{d}\mu \\
&\geq 0.
\end{aligned}$$

The first inequality is due to the well-known fact that $\log x \leq x - 1$.

To prove the second part of the lemma it is necessary to deal with its two components separately. The 'if' proof is trivial - if $f = g$ a.e. then $\log(f/g) = 0$ a.e. and therefore

$$\int_S f \log\left(\frac{f}{g}\right) \mathrm{d}\mu = 0.$$

In order to prove the 'only if' part note that $\log(x)$ may be rewritten as [37]

$$\log x = (x - 1) - \frac{(x - 1)^2}{2\lambda^2} \qquad \text{with} \qquad \lambda \in [1, x],$$

for any $x > 0$. Therefore, we may write

$$\log\left(\frac{g}{f}\right) = \left(\frac{g}{f} - 1\right) - \left(\frac{g}{f} - 1\right)^2 (2\lambda^2)^{-1},$$

and thus

$$\int_S f \log\left(\frac{g}{f}\right) \mathrm{d}\mu = \int_S (g - f)\mathrm{d}\mu - \int_S f \left(\frac{g}{f} - 1\right)^2 (2\lambda^2)^{-1}\mathrm{d}\mu. \tag{80}$$

By assumption the first term on the right hand side of (80) is less than or equal to zero. Similarly the second term can be no greater than zero. Since the whole expression is equal to zero it is necessary to choose f and g so that both terms are uniformly zero.

Noting that $f > 0$ and that $\left(\frac{g}{f} - 1\right)^2 \geq 0$, this implies that

$$\int_S f\left(\frac{g}{f} - 1\right)^2 (2\lambda^2)^{-1} \mathrm{d}\mu = 0$$

only when $f = g$. Since this also makes the first term of (80) equal to zero the proof is complete. □

Lemma C.4. *Consider the system* (11), (14)*. Let the pair* $(\overline{A}, \overline{Q}^{1/2})$ *(defined by equation* (36)*) be controllable, the pair* $(C, \overline{A})$ *be observable, and the matrices* Π *and* P_1 *positive definite. Then there exist constants* $\beta_1, \beta_2 > 0$ *such that*

$$\beta_1 I \geq P_{t|N} \geq \beta_2 I \qquad \forall t \geq 1. \tag{81}$$

Proof. According to Lemmas 7.2 and 7.3 of [22], there exist positive constants α_1 and α_2 such that

$$\alpha_1 I \leq P_{t|t} \leq \alpha_2 I \tag{82}$$

for all $t \geq 1$. We shall now prove that $P_{t|N}$ is bounded below.

Via equations (38), (39) and (82),

$$\begin{aligned} x^T P_{t|t-1} x &= x^T P_{t|t} x + x^T P_{t|t-1} C^T (C P_{t|t-1} C^T + R)^{-1} C P_{t|t-1} x \\ &\geq x^T P_{t|t} x \\ &\geq \alpha_1 \end{aligned} \tag{83}$$

for all $t \geq 1$.

Using the well-known Matrix Inversion Lemma and equations (32) and (37), equation (34) yields

$$\begin{aligned} P_{t|N} &= P_{t|t} + J_t (P_{t+1|N} - P_{t+1|t}) J_t^T \\ &= P_{t|t} - P_{t|t}\overline{A}^T (P_{t+1|t})^{-1} \overline{A} P_{t|t} + P_{t|t}\overline{A}^T P_{t+1|t}^{-1} P_{t+1|N} P_{t+1|t}^{-1} \overline{A} P_{t|t} \\ &= P_{t|t} - P_{t|t}\overline{A}^T (\overline{A} P_{t|t} \overline{A}^T + \overline{Q})^{-1} \overline{A} P_{t|t} + P_{t|t}\overline{A}^T P_{t+1|t}^{-1} P_{t+1|N} P_{t+1|t}^{-1} \overline{A} P_{t|t} \\ &= \left(P_{t|t}^{-1} + \overline{AQ}^{-1} \overline{A}^T\right)^{-1} + P_{t|t}\overline{A}^T P_{t+1|t}^{-1} P_{t+1|N} P_{t+1|t}^{-1} \overline{A} P_{t|t}. \end{aligned} \tag{84}$$

Since $\overline{A}$ is bounded and $\overline{Q} > 0$ it follows from equation (82) that the first term on the right-hand side of equation (84) is positive definite. The second term is positive semi-definite by construction so that, for an arbitrary vector x satisfying $\|x\| = 1$,

$$x^T P_{t|N} x \geq x^T \left(P_{t|t}^{-1} + \overline{AQ}^{-1} \overline{A}^T\right)^{-1} x \geq \alpha_3 > 0 \qquad \forall t \geq 1.$$

We turn now to proving that $P_{t|N}$ is bounded above. Notice that if $P_{t+1|N} \leq P_{t+1|t}$ then via equations (34) and (83)

$$P_{t|N} = P_{t|t} - J_t \underbrace{(P_{t+1|t} - P_{t+1|N})}_{\geq 0} J_t^T \leq P_{t|t} \leq P_{t|t-1}.$$

The result is completed on noting that, via equation (83), $P_{N|N} \leq P_{N|N-1}$. □

Lemma C.5 (Fisher's Identity). *Let $p_\theta(X,Y)$ denote the probability density function of X and Y conditional upon a parameter vector θ, and let $p_\theta(Y)$ refer to the probability density function of the Y conditional upon that same parameter vector θ. Then*

$$\frac{\partial}{\partial\theta}\log p_\theta(Y) = \mathbf{E}_\psi\left\{\frac{\partial}{\partial\theta}\log p_\theta(X,Y)\middle| Y\right\}\Big|_{\psi=\theta} = \frac{\partial}{\partial\theta}\mathcal{Q}(\theta,\psi)\Big|_{\psi=\theta}$$

where $\mathbf{E}_\theta\{\cdot \mid Y\}$ is the conditional expectation operator given Y and θ.

Proof. Let

$$\begin{aligned}\frac{\partial}{\partial\theta}\mathcal{Q}(\theta,\psi) &= \mathbf{E}_\psi\left\{\frac{\partial}{\partial\theta}\log p_\theta(X,Y)\middle| Y\right\}\\ &= \int p_\psi(X\mid Y)\frac{\partial}{\partial\theta}\log p_\theta(X,Y)\mathrm{d}X\\ &= \int p_\psi(X\mid Y)\frac{1}{p_\theta(X,Y)}\frac{\partial}{\partial\theta}p_\theta(X,Y)\mathrm{d}X.\end{aligned}$$

Therefore, when $\psi=\theta$,

$$\begin{aligned}\frac{\partial}{\partial\theta}\mathcal{Q}(\theta,\psi)\Big|_{\psi=\theta} &= \int\frac{1}{p_\theta(Y)}\frac{\partial}{\partial\theta}(p_\theta(X\mid Y)p_\theta(Y))\mathrm{d}X\\ &= \int\frac{1}{p_\theta(Y)}p_\theta(X\mid Y)\frac{\partial}{\partial\theta}p_\theta(Y)\mathrm{d}X + \int\frac{\partial}{\partial\theta}p_\theta(X\mid Y)\mathrm{d}X\\ &= \frac{1}{p_\theta(Y)}\frac{\partial}{\partial\theta}p_\theta(Y)\int p_\theta(X\mid Y)\mathrm{d}X + 0\\ &= \frac{\partial}{\partial\theta}\log p_\theta(Y).\end{aligned}$$

□

Lemma C.6. *Suppose $M\in\mathbf{R}^{n\times n}$ is an invertible matrix. Then*

$$\frac{\mathrm{d}vec\left\{M^{-1}\right\}}{\mathrm{d}vec\left\{M\right\}^T} = -M^{-T}\otimes M^{-1}.$$

Proof.

$$M^{-1}M = I$$

Therefore, denoting $m_{i,j}$ as the i,j^{th} of M,

$$\frac{\partial M^{-1}}{\partial m_{i,j}}M + M^{-1}\frac{\partial M}{\partial m_{i,j}} = 0.$$

Hence (e_i is the i^{th} column of an identity matrix)

$$\frac{\partial M^{-1}}{\partial m_{i,j}} = -M^{-1}\frac{\partial M}{\partial m_{i,j}}M^{-1} = -M^{-1}e_ie_j^TM^{-1}.$$

It then follows directly from the properties of the Kronecker product operator [5] that

$$\frac{\partial\mathrm{vec}\left\{M^{-1}\right\}}{\partial m_{i,j}} = -N\mathrm{vec}\left\{e_ie_j^T\right\}, \tag{85}$$

where $N \triangleq M^{-T} \otimes M^{-1}$. Equation (85) may be written as

$$\frac{\partial \text{vec}\left\{M^{-1}\right\}}{\partial m_{i,j}} = -N(:, n \times (j-1) + i).$$

Finally, recognising that $m_{i,j} = [\text{vec}\{M\}]_{n\times(j-1)+i}$,

$$\frac{\partial \text{vec}\left\{M^{-1}\right\}}{\partial \text{vec}\{M\}^T} = -N = M^{-T} \otimes M^{-1}.$$

□

References

[1] L. BAUM, T. PETRIE, G. SOULES, AND N. WEISS, *A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains*, 41 (1970), pp. 164–171.

[2] N. BERGBOER, V. VERDULT, AND M. VERHAEGEN, *An efficient implementation of Maximum Likelihood identification of LTI state-space models by local gradient search*, in Proceedings of the 41st IEEE CDC, Las Vegas, USA, December 2002.

[3] M. BORRAN AND B. AAZHANG, *EM-based multiuser detection in fast fading multipath environments*, in Proc. EURASIP, vol. 8, 2002, pp. 787–796.

[4] R. BOYLES, *On the convergence of the EM algorithm*, Journal of the Royal Statistical Society, Series B, 45 (1983), pp. 47–50.

[5] J. W. BREWER, *Kronecker products and matrix calculus in system theory*, IEEE Transactions on Circuits and Systems, 25 (1978), pp. 772–781.

[6] P. CAINES, *Linear Stochastic Systems*, John Wiley and Sons, New York, 1988.

[7] C. CHARALAMBOUS AND A. LOGOTHETIS, *Maximum likelihood parameter estimation from incomplete data via the sensitivity equations:The continuous time case*, IEEE Transactions on Automatic Control, 45 (2000), pp. 928–934.

[8] M. CROUSE, R. NOWAK, AND R. BARANIUK, *Wavelet-based statistical signal processing using hidden Markov models*, IEEE Transactions on Signal Processing, 46 (1998), pp. 886–902.

[9] M. DEISTLER, *Model Identification and Adaptive Control*, Springer-Verlag, 2000, ch. System Identification - General Aspects and Structure.

[10] M. DEISTLER, K. PETERNELL, AND W. SCHERRER, *Consistency and relative efficiency of subspace methods*, Automatica, 31 (1995), pp. 1865–1875.

[11] A. DEMPSTER, N. LAIRD, AND D. RUBIN, *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society, Series B, 39 (1977), pp. 1–38.

[12] J. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall, 1983.

[13] R. ELLIOTT AND J. MOORE, *A Martingale Kronekcer Lemma and parameter estimation for linear systems*, IEEE Transactions on Automatic Control, 43 (1998), pp. 1263–1265.

[14] J. GILES, *Introduction to the Analysis of Metric Spaces*, Cambridge University Press, 1987.

[15] ——, *Introduction to the Analysis of Normed Linear Spaces*, Cambridge University Press, 2000.

[16] G. GOLUB AND C. V. LOAN, *Matrix Computations*, Johns Hopkins University Press, 1989.

[17] G. GOODWIN AND A. FEUER, *Estimation with missing data*, Mathematical and Computer Modelling of Dynamical Systems, 5 (1999), pp. 220–244.

[18] G. GOODWIN AND R. PAYNE, *Dynamic System Identification*, Academic Press, 1977.

[19] E. HANNAN AND M. DEISTLER, *The Statistical Theory of Linear Systems*, John Wiley and Sons, New York, 1988.

[20] A. ISAKSSON, *Identification of ARX models subject to missing data*, IEEE Transactions on Automatic Control, 38 (1993), pp. 813–819.

[21] M. JANSSON AND B. WAHLBERG, *On weighting in state-space subspace system identification*, in Proceedings of the European Control Conference, Rome, 1995, pp. 435–440.

[22] A. H. JAZWINSKI, *Stochastic Processes and Filtering Theory*, Academic Press, 1970.

[23] T. KAILATH, A. SAYED, AND B. HASSABI, *Linear Estimation*, Prentice Hall, Upper Saddle River, New Jersey, 2000.

[24] W. LARIMORE, *Canonical variate analysis in identification, filtering and adaptive control*, in Proceedings of the 29th IEEE Conference on Decision and Control, Hawaii, 1990, pp. 596–604.

[25] L. LJUNG, *System Identification: Theory for the User, (2nd edition)*, Prentice-Hall, Inc., New Jersey, 1999.

[26] ——, *Version 5 of the system identification toolbox for use with MATLAB - with object orientation*, in Proc IFAC Symposium SYSID 2000, Santa Barbara, 2000.

[27] L. LJUNG, *Aspects and experiences of user choices in subspace identification methods*, in Proceedings of the 13th IFAC Symposium on System Identification, Rotterdam, The Netherlands, August 2003, pp. 1802–1807.

[28] T. MCKELVEY, *Discussion: 'on the use of minimal parametrizations in multivariable ARMAX identification' by R.P. Guidorzi*, European Journal of Control, 4 (1998), pp. 93–98.

[29] T. MCKELVEY AND A. HELMERSSON, *A dynamical minimal parametrization of multivariable linear systems and its application to optimization and system identification*, in Proc. of the 14th World Congress of IFAC, vol. H, Beijing, P. R. China, 1999, pp. 7–12.

[30] T. MCKELVEY, A. HELMERSSON, AND T. RIBARITS, *A dynamic minimal parametrization of multivariable linear systems and its application to system identification*, Automatica, 40 (2004).

[31] G. MCLACHLAN AND T. KRISHNAN, *The EM Algorithm and Extensions*, John Wiley and Sons, 1996.

[32] X.-L. MENG AND D. RUBIN, *On the global and componentwise rates of convergence of the EM algorithm*, Lin. Alg. Applic., 199 (1994), pp. 413–425.

[33] X.-L. MENG AND D. VAN DYK, *The EM algorithm - an old folk-song sung to a fast new tune*, Journal of the Royal Statistical Society, 59 (1997), pp. 511–567.

[34] R. PINTELON AND J. SCHOUKENS, *System Identification: A Frequency Domain Approach*, IEEE Press, 2001.

[35] R. PINTELON, J. SCHOUKENS, T. MCKELVEY, AND Y. ROLAIN, *Minimum variance bounds for overparameterized models*, IEEE Trans. on Automatic Control, 41 (1996), pp. 719–720.

[36] L. RABINER, *A tutorial on hidden Markov models and selected applications in speech recognition*, Proceedings of the IEEE, 77 (1989), pp. 257–285.

[37] C. RAO, *Linear statistical inference and its application*, Wiley, New York, 1965.

[38] T. RIBARITS, M. DEISTLER, AND B. HANZON, *On new parametrization methods for the estimation of linear state-space models*, International Journal of Adaptive Control and Signal Processing, (2004).

[39] R. SHUMWAY, *An approach to time series smoothing and forecasting using the EM algorithm*, Journal of Time Series Analysis, 3 (1982), pp. 253–264.

[40] R. H. SHUMWAY AND D. S. STOFFER, *Time Series Analysis and its Applications*, Springer–Verlag, 2000.

[41] J.-L. STARCK, F. MURTAGH, AND A. BIJAOUI, *Image processing and data analysis*, Cambridge University Press, Cambridge, 1998. The multiscale approach.

[42] T.SÖDERSTRÖM AND P.STOICA, *System Identification*, Prentice Hall, New York, 1989.

[43] P. VAN OVERSCHEE AND B. D. MOOR, *Subspace Identification for Linear Systems*, Kluwer Academic Publishers, 1996.

[44] M. VERHAEGEN, *Identification of the deterministic part of MIMO state space models in innovations form from input-output data*, Automatica, 30 (1994), pp. 61–74.

[45] S. WRIGHT, *Modified cholesky factorizations in interior-point algorithms for linear programming*, SIAM Journal on Optimization, 9 (1999), pp. 1159–1191.

[46] C. F. WU, *On the convergence properties of the EM algorithm*, The Annals of Statistics, 11 (1983), pp. 95–103.