Estimating State-Space Models in Innovations Form using the Expectation Maximisation Algorithm

Adrian Wills, Thomas B. Schön and Brett Ninness

***Abstract*— The expectation maximisation (EM) algorithm has proven to be effective for a range of identification problems. Unfortunately, the way in which the EM algorithm has previously been applied has proven unsuitable for the commonly employed innovations form model structure. This paper addresses this problem, and presents a previously unexamined method of EM algorithm employment. The results are profiled, which indicate that a hybrid EM/gradient-search technique may in some cases outperform either a pure EM or a pure gradient-based search approach.**

I. Introduction

The expectation maximisation (EM) algorithm is an iterative search technique for solving maximum likelihood estimation problems. It is an alternative to the more common gradient-based search approaches and it has proven useful in applications where gradients are difficult to compute. It is also well regarded for its numerical stability [1].

The method has its origins in the statistics literature [2], but has been widely applied in very many other areas such as image processing [3], econometrics [4], epidemiology [5, 6] and speech recognition [7], just to mention a few.

It has also be employed in the context of this paper, which is system identification. This includes work on time-series modeling [8], ARX modeling with censored data [9, 10], estimation of linear and bilinear state-space systems [11, 12], frequency domain estimation [13], and non-linear system estimation [14–16, 18].

A fundamental step in designing an EM algorithm is the choice of the so-called "missing data". In some cases, it literally is missing, in that measurements are censored [9, 10]. However, in most applications of the EM algorithm it is the "wished-for" data that if it were available, would make the estimation problem more straightforward. For example, in all of the following previous works [8, 11–16, 18] the missing data is chosen as the underlying system state sequence.

The essence of the EM algorithm is to then replace this wished-for data with estimates formed by an appropriate smoothing algorithm. For example, in the linear state-space case [11] Kalman smoothed state estimates are used.

Unfortunately, this creates a difficulty in employing the EM algorithm when the state-space system is in innovations form. This arises due to the associated deterministic relationship between states and measurements. This is significant, since it precludes the application of the EM algorithm to a range of transfer function structures and continuous time models accommodating non-regularly sampled data. This difficulty was first recognised in [19].

The contribution of this paper is to develop and illustrate a solution to this problem, whose foundation is the use of just the initial state as the missing data. The utility of the resulting EM-based estimation algorithm that ensues is profiled via a simulation study.

A. Wills and B. Ninness are with the School of Electrical Engineering and Computer Science, University of Newcastle, Callaghan, NSW 2308, Australia {Adrian.Wills, Brett.Ninness}@newcastle.edu.au

T. B. Schön is with the Division of Automatic Control, Linköping University, SE-581 83 Linköping, Sweden, E-mail: schon@isy.liu.se

This work was partly supported by the Australian Research Council (ARC) and partly supported by the strategic research center MOVIII, funded by the Swedish Foundation for Strategic Research (SSF) and CADICS, a Linneaus Center funded by be Swedish Research Council.

II. Problem Formulation

The very well known general discrete-time linear time-invariant state-space model is

$$x_{t+1} = Ax_t + Bu_t + w_t, \tag{1a}$$
$$y_t = Cx_t + Du_t + \epsilon_t, \tag{1b}$$

where $x_t \in \mathbb{R}^n$ is the system state, $y_t \in \mathbb{R}^p$ is the output, $u_t \in \mathbb{R}^m$ is the input, and t is the time index. The terms w_t and ϵ_t are sequences of i.i.d. random variables.

As is widely appreciated [20], this model has an associated "innovations form" representation

$$x_{t+1} = Ax_t + Bu_t + Ke_t, \tag{2a}$$
$$y_t = Cx_t + Du_t + e_t, \tag{2b}$$

where e_t is again an i.i.d. random variable sequence. Both e_t and the initial state x_1 are assumed to be Gaussian distributed according to

$$e_t \sim \mathcal{N}(0, R), \qquad R > 0, \tag{3a}$$
$$x_1 \sim \mathcal{N}(\mu, P_1), \qquad P_1 > 0. \tag{3b}$$

For future reference, we note that an alternate formulation of (2) is

$$y_t = \widehat{y}_{t|t-1} + e_t, \tag{4a}$$
$$\widehat{y}_{t|t-1} = Cx_t + Du_t, \tag{4b}$$
$$x_{t+1} = (A - KC)x_t + (B - KD)u_t + Ky_t, \tag{4c}$$

where $\widehat{y}_{t|t-1} \triangleq E\{y_t \mid Y_{t-1}\}$ is the mean square optimal one-step ahead predictor of y_t given the past observations $Y_{t-1} \triangleq [y_1, \cdots, y_{t-1}]$, which implies that this predictor depends on the initial state x_1.

With this in mind, the work here considers the estimation of the system quantities

$$A, B, C, D, K, R, P_1, \mu \tag{5}$$

in this innovations form structure on the basis of measurements

$$U_N \triangleq [u_1, \cdots, u_N], \quad Y_N \triangleq [y_1, \cdots, y_N], \tag{6}$$

of observed input-output responses. Again, for future reference, the non-zero terms that define the system matrices (5) will be assumed collected into a vector θ of parameters that define the system (2).

This is a very well studied problem for which a standard solution methodology has evolved. This involves gradient-based search techniques to determine the minima of the prediction error or the negative log-likelihood criteria. This is a generally reliable approach, although difficulties can arise with capture in local minima, and computational load for systems of high state and input-output dimension.

In relation to these latter problems, the work [11] has established that when employing the model structure (1), these problems can be lessened by employing the EM algorithm as an alternative to a gradient-based search.

The topic of this paper is to extend that work to the innovations form structure (2). One motivation for this is to allow the EM algorithm to be applied to the popular transfer function model structure

$$y_t = G(q, \theta)u_t + H(q, \theta)e_t. \tag{7}$$

Here θ represents a vector parametrizing the functions G and H which are rational in the shift operator q.

A further motivation is to allow the EM algorithm to be applied to continuous-time modeling accommodating measurements that are obtained at irregularly spaced time intervals [21].

Unfortunately, the approach used in [11] for implementing the EM algorithm is not directly applicable to the innovations form model (2). This is due to the fact that according to (4c), the state is a deterministic function of the observations y_t, u_t and the initial state x_1. Key to the work in [11] and related approaches to employing the EM algorithm [8, 12–16, 18] is the consideration of the joint log-likelihood of the observations and the state sequence. Due to the determinism present in (4c), this likelihood will have a Dirac delta form, and hence be unsuitable for the purposes of parameter estimation.

The remainder of this paper is devoted to addressing this problem and profiling the performance of the solution obtained.

III. The Expectation Maximisation (EM) Algorithm

This paper employs the maximum likelihood framework, wherein an estimate $\widehat{\theta}^{\mathrm{ML}}$ of an unknown parameter vector θ is obtained by solving a particular optimisation problem

$$\widehat{\theta}^{\mathrm{ML}} = \arg\max_{\theta} \; L_\theta(Y_N). \tag{8}$$

The cost $L_\theta(Y_N)$ to be maximised is the log-likelihood function defined as

$$L_\theta(Y_N) \triangleq \log p_\theta(Y_N), \tag{9}$$

where $p_\theta(Y_N)$ is the joint probability density function for the observed stochastic observations Y_N.

The EM algorithm is a method for computing $\widehat{\theta}^{\mathrm{ML}}$ that is very general and addresses a wide range of applications. Key to both its implementation and theoretical underpinnings is the consideration of a joint log-likelihood function of both the measurements Y_N and the "missing data" Z

$$L_\theta(Y_N, Z) = \log p_\theta(Y_N, Z). \tag{10}$$

The missing data Z consist of measurements that while not available, would be useful to the estimation problem. The choice of Z is a design variable in the deployment of the EM algorithm.

Importantly, by the definition of conditional probability, the likelihood (9) and the "complete data" likelihood (10) are related according to

$$\log p_\theta(Y_N) = \log p_\theta(Z, Y_N) - \log p_\theta(Z \mid Y_N). \tag{11}$$

Denote by θ_k an estimate of the likelihood maximiser $\widehat{\theta}^{\mathrm{ML}}$, and therefore denote by $p_{\theta_k}(Z \mid Y_N)$ the conditional density of the missing data Z, given observations of the available data Y_N and depending on the choice θ_k.

This permits the following expression eventuating from taking conditional expectations relative to $p_{\theta_k}(Z \mid Y_N)$ of both sides of (11).

$$\begin{aligned} \log\, p_\theta(Y_N) &= \int \log p_\theta(Z, Y_N) p_{\theta_k}(Z \mid Y_N) dZ \\ &\quad - \int \log p_\theta(Z \mid Y_N) p_{\theta_k}(Z \mid Y_N) dZ \\ &= \underbrace{\mathrm{E}_{\theta_k}\left\{\log p_\theta(Z, Y_N) \mid Y_N\right\}}_{\triangleq \mathcal{Q}(\theta, \theta_k)} \\ &\quad - \underbrace{\mathrm{E}_{\theta_k}\left\{\log p_\theta(Z \mid Y_N) \mid Y_N\right\}}_{\triangleq \mathcal{V}(\theta, \theta_k)}. \end{aligned} \tag{12}$$

The difference between the likelihood $L_{\theta_k}(Y_N)$ at the estimate θ_k and the likelihood $L_\theta(Y_N)$ at an arbitrary value of θ is then expressible in terms of these newly defined $\mathcal{Q}$ and $\mathcal{V}$ functions as

$$\begin{aligned} L_\theta(Y_N) - L_{\theta_k}(Y_N) &= (\mathcal{Q}(\theta, \theta_k) - \mathcal{Q}(\theta_k, \theta_k)) \\ &\quad + (\mathcal{V}(\theta_k, \theta_k) - \mathcal{V}(\theta, \theta_k)). \end{aligned} \tag{13}$$

It can be simply established that

$$\mathcal{V}(\theta_k, \theta_k) - \mathcal{V}(\theta, \theta_k) \geq 0, \tag{14}$$

since it is the Kullback–Leibler divergence metric between two densities [22].

As a result, if a new estimate θ_{k+1} of $\widehat{\theta}^{\mathrm{ML}}$ is obtained such that relative to the previous estimate θ_k, it holds that $\mathcal{Q}(\theta_{k+1}, \theta_k) > \mathcal{Q}(\theta_k, \theta_k)$, then necessarily via (14) $L_{\theta_{k+1}}(Y_N) > L_{\theta_k}(Y_N)$.

This observation leads to the EM algorithm, which iterates between forming $\mathcal{Q}(\theta, \theta_k)$ using an estimate θ_k of $\widehat{\theta}^{\mathrm{ML}}$ and then maximising $\mathcal{Q}(\theta, \theta_k)$ with respect to θ to obtain a new better estimate θ_{k+1}.

Algorithm 3.1: Expectation Maximisation Algorithm

1) Set $k = 0$ and initialize θ_0 such that $L_{\theta_0}(Y_N)$ is finite.
2) **Expectation (E) step:** Compute

$$\mathcal{Q}(\theta, \theta_k) = \mathrm{E}_{\theta_k} \left\{ \log p_\theta(Z, Y_N) \mid Y_N \right\}. \tag{15}$$

3) **Maximisation (M) step:** Compute

$$\theta_{k+1} = \arg\max_\theta \ \mathcal{Q}(\theta, \theta_k). \tag{16}$$

4) If not converged, update $k := k + 1$ and return to step 2.

The text [23] is an excellent reference for readers seeking more information about the EM algorithm and its properties.

IV. An EM Algorithm for Innovations Form Models

In the many previous works [8, 12–16, 18], the EM algorithm has been applied by choosing the missing data as the unmeasured state sequence, i.e.,

$$Z = \{x_1, \cdots, x_N\}. \tag{17}$$

As already mentioned, due to the deterministic relationship (4c), if this strategy is applied to the innovations form model (2), the joint density $p_\theta(Y_N, Z)$ will have a Dirac-delta form, and hence be unsuitable.

A main contribution of this paper is to establish that this difficulty can be overcome by choosing the missing data simply as the unobserved initial state value $Z = x_1$. The so-called "E-step" in which the required expectation (15) is evaluated may then be achieved by the following lemma.

Lemma 4.1: With regard to the innovations form model structure (2), and with the choice $Z = x_1$, the function $\mathcal{Q}(\theta, \theta_k)$ defined by (15) is given as (with constants and common factors unimportant to the estimation process ignored)

$$\begin{aligned}\mathcal{Q}(\theta, \theta_k) = &- \log\det P_1 - N \log\det R \\ &- \mathrm{Tr}\left\{ P_1^{-1} \left((\hat{x}_{1|N} - \mu)(\hat{x}_{1|N} - \mu)^T + P_{1|N} \right) \right\} \\ &- \mathrm{Tr}\left\{ R^{-1} \sum_{t=1}^{N} \varepsilon_t \varepsilon_t^T \right\} - \mathrm{Tr}\left\{ R^{-1} \sum_{t=1}^{N} C P_t C^T \right\}\end{aligned} \tag{18}$$

where

$$\widehat{x}_{1|N} \triangleq \mathrm{E}_{\theta_k} \left\{ x_1 \mid Y_N \right\} \tag{19a}$$

$$P_{1|N} \triangleq \mathrm{Cov}_{\theta_k} \left\{ x_1 \mid Y_N \right\} \tag{19b}$$

$$\varepsilon_t \triangleq y_t - \widehat{y}_{t|t-1} \tag{19c}$$

$$\widehat{y}_{t|t-1} = \mathrm{E}_{\theta_k} \left\{ y_t \mid Y_{t-1} \right\} \tag{19d}$$

$$P_t \triangleq \mathrm{Cov}_{\theta_k} \left\{ x_t \mid Y_{t-1} \right\} \tag{19e}$$

Proof: By repeated application of Bayes' rule

$$p_\theta(x_1, Y_N) = p_\theta(x_1) \prod_{t=1}^{N} p_\theta(y_t \mid Y_{t-1}, x_1). \tag{20}$$

Therefore by the definition (12), the choice $Z = x_1$ implies

$$\begin{aligned}\mathcal{Q}(\theta, \theta_k) &= \int \log p_\theta(x_1, Y_N) p_{\theta_k}(x_1 | Y_N) \, \mathrm{d}x_1 \\ &= \int \log p_\theta(x_1) p_{\theta_k}(x_1 | Y_N) \, \mathrm{d}x_1 + \\ &\quad \sum_{t=1}^{N} \int \log p_\theta(y_t \mid Y_{t-1}, x_1) p_{\theta_k}(x_1 | Y_N) \, \mathrm{d}x_1.\end{aligned} \tag{21}$$

Since the innovations e_t and the initial state x_1 are assumed to have the Gaussian distribution (3), the densities above will also be Gaussian. This allows the explicit evaluation (neglecting constant terms)

$$\begin{aligned}\int \log p_\theta(x_1) p_{\theta_k}(x_1 | Y_N) \, \mathrm{d}x_1 = &-\frac{1}{2} \log\det P_1 \\ &- \frac{1}{2} \int \|x_1 - \mu\|^2_{P_1^{-1}} \, p_{\theta_k}(x_1 \mid Y_N) \, \mathrm{d}x_1.\end{aligned} \tag{22}$$

Similarly, using the representation (4)

$$\begin{aligned}\int \log p_\theta(y_t \mid Y_{t-1}, x_1) \, p_{\theta_k}(x_1 | Y_N) \, \mathrm{d}x_1 = &-\frac{1}{2} \log\det R \\ &- \frac{1}{2} \int \left\| y_t - \widehat{y}_{t|t-1} \right\|^2_{R^{-1}} p_{\theta_k}(x_1 \mid Y_N) \, \mathrm{d}x_1.\end{aligned} \tag{23}$$

Using the fact that $x^T A x = \mathrm{Tr}\{A x x^T\}$ allows (22) to be re-expressed according to

$$\begin{aligned}&\int \|x_1 - \mu\|^2_{P_1^{-1}} \, p_{\theta_k}(x_1 \mid Y_N) dx_1 \\ &\quad = \mathrm{Tr}\left\{ P_1^{-1} \left((\hat{x}_{1|N} - \mu)(\hat{x}_{1|N} - \mu)^T + P_{1|N} \right) \right\}.\end{aligned} \tag{24}$$

Similarly, (23) may be reformulated as

$$\begin{aligned}&\int \left\| y_t - \widehat{y}_{t|t-1} \right\|^2_{R^{-1}} p_{\theta_k}(x_1 \mid Y_N) \, \mathrm{d}x_1 \\ &\quad = \mathrm{Tr}\left\{ R^{-1} \varepsilon_t \varepsilon_t^T \right\} - \mathrm{Tr}\left\{ R^{-1} C P_t \, C^T \right\}.\end{aligned} \tag{25}$$

■

The required terms (19a) and (19b) can be obtained by a Kalman smoother [24]. The term (19d) may be computed by the formulation (4b) and (4c). Finally, according to (4c), the term (19e) may evaluated recursively according to

$$P_{t+1}^{1/2} = (A - KC) P_t^{1/2}, \quad P_t = P_t^{1/2} P_t^{T/2}. \tag{26}$$

With the computation of the E-step (15) delivering $\mathcal{Q}(\theta, \theta_k)$ addressed, our attention now turns to the M-step (16), where its maximising argument must be found to deliver the next iterate θ_{k+1}.

In the previous work [8, 11] employing the model structure (1), the system matrices fully parametrized, and missing data choice (17), the M-step has been demonstrated to involve a linear regression, and hence solvable in closed form.

It has already been mentioned how this work differs in that it addresses the innovations form (2). Another important difference is that in order to be relevant to transfer function and continuous-time modeling applications, it allows for constraints in the formulation of the system matrices (5). These two factors combine to complicate the M-step so that

a closed form expression for the maximiser θ_{k+1} does not exist.

To address this, let θ be partitioned as

$$\theta^T = [\eta^T, \beta^T]^T, \tag{27}$$

where η parametrizes P_1, R and μ, and β parametrizes A, B, C, D, K. Typically, while there may be constraints on how β parametrizes the associated system matrices, none exist for the formulation of P_1, R (except for the fact that they have to be positive semi-definite matrices) and μ. This can be exploited by noting that in this case, (18) is clearly globally maximised with respect to μ by the choice

$$\mu = \widehat{x}_{1|N}. \tag{28}$$

With this value substituted into (18), the terms involving P_1 become

$$-\log\det P_1 - \operatorname{Tr}\left\{P_1^{-1} P_{1|N}\right\}. \tag{29}$$

Furthermore, by basic matrix calculus [25]

$$-\frac{\partial}{\partial P_1}\left(\log\det P_1 - \operatorname{Tr}\left\{P_1^{-1}P_{1|N}\right\}\right) = -P_1^{-1} - P_1^{-1}P_{1|N}P_1^{-1} \tag{30}$$

which is clearly zero for the choice

$$P_1 = P_{1|N} \tag{31}$$

which is then a stationary point of (18). By an identical argument

$$R = \frac{1}{N}\sum_{t=1}^{N} \varepsilon_t \varepsilon_t^T + C P_t C^T \tag{32}$$

is also a stationary point of (18). These values (28), (31) and (32) substituted into (18) deliver a "concentrated" form $\widetilde{\mathcal{Q}}(\beta, \theta_k)$ that depends only on β as follows

$$\widetilde{\mathcal{Q}}(\beta, \theta_k) = -\log\det\left(\frac{1}{N}\sum_{t=1}^{N}\varepsilon_t\varepsilon_t^T + C P_t C^T\right). \tag{33}$$

Unfortunately, it is not possible to determine stationary points of this function in closed form. The solution to this difficulty, proposed here, is the employment of a gradient-based search technique which has the general quasi-Newton, form whereby an estimate β^i of the maximiser of $\widetilde{\mathcal{Q}}(\beta, \theta_k)$ is updated to a better one β^{i+1}according to

$$\beta^{i+1} = \beta^i + \mu^i p^i, \quad p^i = H^i g^i. \tag{34a}$$

$$g^i = \widetilde{\mathcal{Q}}'(\beta^i, \theta_k) \triangleq \frac{\partial}{\partial\beta}\widetilde{\mathcal{Q}}(\beta, \theta_k)\bigg|_{\beta=\beta^i}. \tag{34b}$$

Here H^i is a positive definite matrix that delivers a search direction p^i by modifying the gradient direction, and μ^i is a step length. The authors have found that a BFGS formulation for H^i with back-stepping line-search for μ^i is effective [26].

In order to implement this, it is necessary to develop an expression for the gradient (34b). This is established by the following lemma.

Lemma 4.2: The gradients of the $\widetilde{\mathcal{Q}}(\beta, \theta_k)$ with respect to β are given by

$$\frac{\partial\widetilde{\mathcal{Q}}(\beta)}{\partial\beta_i} = -2\sum_{t=1}^{N}\varepsilon_t^T R(\beta)^{-1}\frac{\partial\varepsilon_t}{\partial\beta_i} \tag{35}$$

$$-\sum_{t=1}^{N}\operatorname{Tr}\left\{R(\beta)^{-1}\frac{\partial C P_t C^T}{\partial\beta_i}\right\}, \tag{36}$$

where $R(\beta)$ is given by

$$R(\beta) \triangleq \frac{1}{N}\sum_{t=1}^{N}\varepsilon_t\varepsilon_t^T + C P_t C^T \tag{37}$$

and the terms in these expressions may be computed recursively according to

$$\frac{\partial\varepsilon_t}{\partial\beta_i} = -\frac{\partial C}{\partial\beta_i}\widehat{x}_t - C\frac{\partial\widehat{x}_t}{\partial\beta_i} - \frac{\partial B}{\partial\beta_i}, \tag{38a}$$

$$\frac{\partial\widehat{x}_{t+1}}{\partial\beta_i} = \frac{\partial A}{\partial\beta_i}\widehat{x}_t + A\frac{\partial\widehat{x}_t}{\partial\beta_i} + \frac{\partial B}{\partial\beta_i}u_t + \frac{\partial K}{\partial\beta_i}\varepsilon_t + K\frac{\partial\varepsilon_t}{\partial\beta_i}, \tag{38b}$$

$$\frac{\partial C P_t C^T}{\partial\beta_i} = \frac{\partial C}{\partial\beta_i}P_t C^T + C\frac{\partial P_t}{\partial\beta_i}C^T + C P_t\frac{\partial C^T}{\partial\beta_i}, \tag{38c}$$

$$\begin{aligned}\frac{\partial P_{t+1}}{\partial\beta_i} = &\left(\frac{\partial A}{\partial\beta_i} - \frac{\partial K}{\partial\beta_i}C - K\frac{\partial C}{\partial\beta_i}\right)P_t\left(A - KC\right)^T \\ &+ \left(A - KC\right)\frac{\partial P_t}{\partial\beta_i}\left(A - KC\right)^T \\ &+ \left(A - KC\right)P_t\left(\frac{\partial A}{\partial\beta_i} - \frac{\partial K}{\partial\beta_i}C - K\frac{\partial C}{\partial\beta_i}\right)^T,\end{aligned} \tag{38d}$$

$$\frac{\partial\widehat{x}_1}{\partial\beta_i} = 0, \quad \frac{\partial P_1}{\partial\beta_i} = 0. \tag{38e}$$

Proof: The result follows by standard application of the product rule for differentiation, and basic results of matrix calculus. ■

A. Final Algorithm

Combining the above E and M steps results in the following EM algorithm for identifying state-space models in innovations form.

Algorithm 4.1: EM for identification of innovation models

1) Set $k = 0$ and initialize θ_0 such that $L_{\theta_0}(Y_N)$ is finite.
2) **Expectation (E) step:**
   Based on θ_k and its associated $A, B, C, D, K, R, \mu, P_1$ system parameters, run a Kalman smoother to obtain $\widehat{x}_{1|N}$ and $P_{1|N}$.
3) **Maximisation (M) step:**
   Use a quasi-Newton search algorithm to maximise $Q(\theta_{k+1}, \theta_k)$ over θ_{k+1}.
4) If not converged, update $k := k + 1$ and return to step 2.

V. Numerical Examples

In order to demonstrate the efficacy of the above algorithm, it is applied to two examples in this section. The first considers an output-error (OE) system in the form of (2)

where $K = 0$. The second example considers a second-order innovations model in the form of (2) with $K \neq 0$.

To show that the algorithm is relatively insensitive to the choice of initial value, a Monte-Carlo simulation is performed over randomized initial parameter values. In each run, the measured input/output data remains the same, and only the initial parameter values for the algorithm are randomized. In this way, it is expected that the algorithm should produce very similar parameter estimates for each run.

A. Output-error model

Consider the following output-error model

$$x_{t+1} = ax_t + bu_t, \tag{39a}$$
$$y_t = x_t + du_t + e_t, \tag{39b}$$

where the noise process is given by $e_t \sim \mathcal{N}(0, r)$ and the initial state is assumed distributed according to $x_1 \sim \mathcal{N}(\mu, p_1)$. For the purposes of the simulation here, the true parameters are given by

$$\theta^\star = \begin{bmatrix} a^\star & b^\star & d^\star & r^\star & \mu^\star & p_1^\star \end{bmatrix} = \begin{bmatrix} 0.5 & 1 & 0 & 0.1 & 0 & 1 \end{bmatrix} \tag{40}$$

The above system was used to generate $N = 100$ output measurements Y_N for a Gaussian distributed input signal $u_t \sim \mathcal{N}(0, 1)$.

Based on the data Y_N and U_N, a Monte–Carlo simulation was performed with $M = 100$ runs; in each run the initial parameter value θ_0 was constructed by drawing the individual elements from a uniform random number generator between 0 and 1. That is, for each run, the same data is used, but the initial value is random.

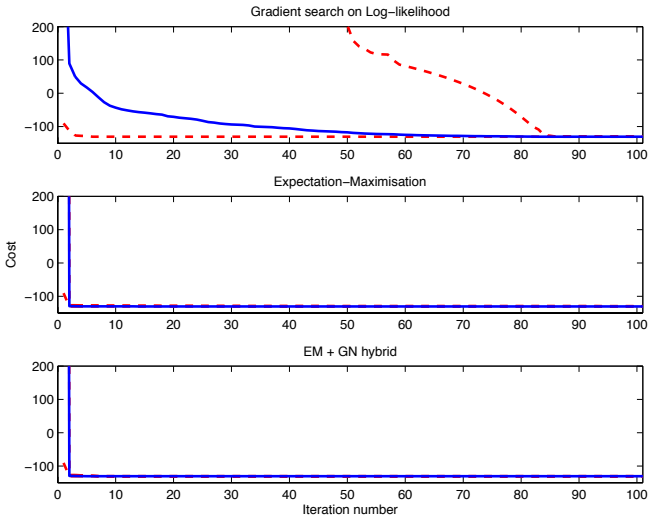
In the first instance, the EM Algorithm 4.1 was used to generate parameter estimates. To gauge its utility, the average negative log-likelihood cost is shown in Figure 1 as a function of iteration number. Also shown are the best and the worst case cost trajectories out of all the runs. Note that instances where the algorithm failed to reduce the cost to within 10% of the minimum value are not shown. The number of failures is provided in Table I.

I: Number of failed runs for the three algorithms.

	DGS	EM	EM+GS
Output-Error	10	0	0
Innovations	70	1	0

On the one hand these results look promising, and on the other hand it is difficult to gauge the performance of Algorithm 4.1 in isolation. Therefore, by way of comparison, a standard method of minimizing the negative log-likelihood $-L_\theta(Y_N)$ was also employed. Specifically, the same BFGS quasi-Newton algorithm that was used for EM was also used to minimise $-L_\theta(Y_N)$ directly. In this case the gradient was obtained from the Kalman Filter.

The results for directly minimizing $-L_\theta(Y_N)$ are denoted by DGS (for Direct Gradient Search). Again, the average and worst/best case results are shown in Figure 1, where



1: *Negative log-likelihood cost against iteration number for the Output-Error model* (39). *Top: direct gradient search on log-likelihood cost; Middle: EM algorithm 4.1; Bottom: hybrid EM + gradient search. In each plot, the red-dashed lines indicate the best and worst case search performance and the blue-solid line indicates average search performance.*

failed runs have been removed. The number of failed runs are provided in Table I.

The number of failures for direct gradient search (DGS) is significantly more than for EM in this case. At the same time, it was observed that (when successful) the direct gradient search often provides rapid convergence to the minimum.

This observation is well recognised within the statistics literature. For example, the work in [27] studies a number of accelerated algorithms that include hybrid combinations of direct gradient search on $-L_\theta(Y_N)$ and EM iterations. Motivated by their findings, a third algorithm is trialed here that combines both the EM and the DGS variants.

In particular, the EM algorithm is employed until $|L_{\theta_{k+1}} - L_{\theta_k}| \leq \epsilon |L_{\theta_k} - L_{\theta_0}|$, where $\epsilon = 0.1$ in this case. At this point the direct gradient search algorithm is used to further refine the estimate. Results from this hybrid EM and DGS algorithm are shown in Figure 1. It is worth noting that there were no failed runs for this case.

B. Innovations model

Consider a second-order innovations model in the form of (2) where the system matrices are parametrized via

$$A = \begin{bmatrix} \theta_1 & 1 \\ 0 & \theta_2 \end{bmatrix}, \quad B = \begin{bmatrix} \theta_3 \\ \theta_4 \end{bmatrix}, C = \begin{bmatrix} 1 & 1 \end{bmatrix}, \quad D = \theta_5,$$
$$K = \begin{bmatrix} \theta_6 \\ \theta_7 \end{bmatrix}, \quad R = \theta_8^2, \quad x_1 = \begin{bmatrix} \theta_9 \\ \theta_{10} \end{bmatrix},$$
$$P_1 = \begin{bmatrix} \theta_{11} & 0 \\ \theta_{12} & \theta_{13} \end{bmatrix} \begin{bmatrix} \theta_{11} & 0 \\ \theta_{12} & \theta_{13} \end{bmatrix}^T$$

and where the true parameter values for θ are given by

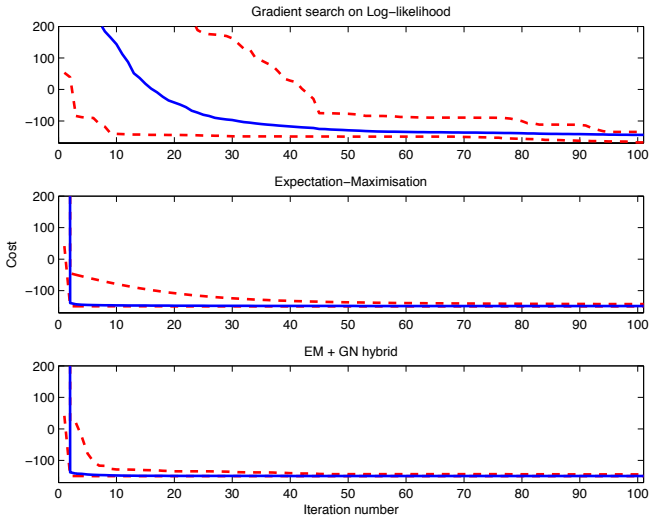
$$\begin{array}{lllll} \theta_1^\star = 0.9 & \theta_2^\star = 0.1 & \theta_3^\star = 1 & \theta_4^\star = 1 & \theta_5^\star = 0.1 \\ \theta_6^\star = 0.2 & \theta_7^\star = 0.3 & \theta_8^\star = \sqrt{0.1} & \theta_9^\star = 0 & \theta_{10}^\star = 0 \\ \theta_{11}^\star = 1 & \theta_{12}^\star = 0 & \theta_{13}^\star = 1 & & \end{array}$$

In a similar manner to the previous Output-Error example, the above system was used to generate $N = 100$ output measurements Y_N for a Gaussian distributed input signal $u_t \sim \mathcal{N}(0, 1)$.

Using this data, and again employing random initial values for θ_0 in the same way as for the Output-Error case, a Monte-Carlo simulation was performed over $M = 100$ runs.

For the same reasons mentioned above, three algorithms are trialed for this case. Namely, the expectation maximisation algorithm (EM), the direct gradient search for the likelihood function (DGS), and a hybrid of the two algorithms in accordance with the findings of [27].

The results are shown in Figure 2. Again any failures of the algorithms to achieve a cost within 10% of the minimum value have been removed from the plots. The number of failures for each algorithm are provided in Table I.



2: *Negative log-likelihood cost against iteration number for Innovations model. Top: direct gradient search on log-likelihood cost; Middle: EM algorithm in 4.1; Bottom: hybrid EM + gradient search. In each plot the red-dashed lines indicate the best and worst case search performance and the blue-solid line indicates average search performance.*

VI. Conclusion

This paper has derived a new approach to the employment of the EM algorithm for estimating innovations form model structures. A profile of the method in two simple simulation examples illustrates promising performance. Additionally, the use of a hybrid approach of handing over from EM-based iterations to gradient-based search iterations appears to provide both enhanced robustness and convergence rate.

Further theoretical analysis and more detailed simulation studies are required before any general conclusions on this topic can be made. Nevertheless, the initial results presented here indicate that further study may indeed be warranted.

References

[1] K. Lange, "A gradient algorithm locally equivalent to the EM algorithm," *Journal of the Royal Statistical Society*, vol. 57, no. 2, pp. 425–437, 1995.

[2] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1–38, 1977.

[3] J.-L. Starck, F. Murtagh, and A. Bijaoui, *Image processing and data analysis*. Cambridge: Cambridge University Press, 1998, the multiscale approach.

[4] P. A. Ruud, "Extensions of estimation methods using the EM algorithm," *Journal of Econometrics*, vol. 49, pp. 305–341, 1991.

[5] O. E. Barndorff–Nielsen, D. R. Cox, and C. Klüppelberg, *Complex Stochastic Systems*. Chapman and Hall, 1999.

[6] S. Duncan and M. Gyöngy, "Uing the EM algorithm to estimate the disease parameters for smallpox in 17th century London," in *Proceedings of the IEEE international conference on control applications*, Munich, Germany, Oct. 2006, pp. 3312–3317.

[7] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–285, 1989.

[8] R. Shumway, "An approach to time series smoothing and forecasting using the EM algorithm," *Journal of Time Series Analysis*, vol. 3, no. 4, pp. 253–264, 1982.

[9] A. Isaksson, "Identification of ARX models subject to missing data," *IEEE Trans. Auto. Control*, vol. 38, no. 5, pp. 813–819, 1993.

[10] G. Goodwin and A. Feuer, "Estimation with missing data," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 5, no. 3, pp. 220–244, 1999.

[11] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, 2005.

[12] S. Gibson, A. Wills, and B. Ninness, "Maximum-likelihood parameter estimation of bilinear systems," *IEEE Trans. Automat. Control*, vol. 50, no. 10, pp. 1581–1596, 2005.

[13] A. Wills, B. Ninness, and S. Gibson, "Maximum likelihood estimation of state space models from frequency domain data," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 19–33, 2009.

[14] R. B. Gopaluni, "A particle filter approach to identification of nonlinear processes under missing observations," *The Canadian Journal of Chemical Engineering*, vol. 86, no. 6, pp. 1081–1092, Dec. 2008.

[15] G. C. Goodwin and J. C. Agüero, "Approximate EM algorithms for parameter and state estimation in nonlinear stochastic models," in *Proceedings of the 44th IEEE Conf. Dec. & Cont. (CDC)*, Seville, Spain, Dec. 2005, pp. 368–373.

[16] Z. Ghaharamani and S. T. Roweis, "Learning nonlinear dynamical systems using an EM algorithm," in *Advances in Neural Information Processing Systems*, vol. 11. MIT Press, 1999, pp. 599–605.

[17] A. Wills, T. Schön, and B. Ninness, "Parameter estimation for discrete-time nonlinear systems using EM," in *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, Jul. 2008.

[18] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," in *Automatica*, 2010. Accepted for publication.

[19] V. Solo, "An EM algorithm for singular state space models: II," in *Proceedings of the 43rd IEEE Conference on Decision and Control (CDC)*, Nassau, Bahamas, Dec. 2004, pp. 3611–3612.

[20] B. Anderson and J. Moore, *Optimal Filtering*. Prentice Hall, 1979.

[21] L. Ljung and A. Wills, "Issues in sampling and estimating continuous-time models with stochastic disturbances," *Automatica. Available online 15th March 2010 doi:10.1016/j.automatica.2010.02.011.*, 2010.

[22] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, 2005.

[23] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions (2nd Edition)*. John Wiley and Sons, 2008.

[24] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Academic Press, 1970.

[25] D. S. Bernstein, *Matrix Mathematics*. Princeton Uni. Press, 2005.

[26] J. Nocedal and S. Wright, *Numerical Optimization*. Springer-Verlag, New York, 1999.

[27] M. Jamshidian and R. I. Jennrich, "Acceleration of the EM algorithm by using quasi-newton methods," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 59, no. 3, pp. 569–587, 1997.