Linear Time-Varying Parameter Estimation: Maximum A Posteriori Approach via Semidefinite Programming

Sasan Vakili, Mohammad Khosravi, Peyman Mohajerin Esfahani and Manuel Mazo Jr.

***Abstract*— We study the problem of identifying a linear time-varying output map from measurements and linear time-varying system states, which are perturbed with Gaussian observation noise and process uncertainty, respectively. Employing a stochastic model as *prior* knowledge for the parameters of the unknown output map, we reconstruct their estimates from input/output pairs via a Bayesian approach to optimize the *posterior* probability density of the output map parameters. The resulting problem is a non-convex optimization, for which we propose a tractable linear matrix inequalities approximation to warm-start a first-order subsequent method. The efficacy of our algorithm is shown experimentally against classical Expectation Maximization and Dual Kalman Smoother approaches.**

I. Introduction

Bayesian approaches for estimating characteristics of dynamical systems have been a subject of studies for decades and have recently received extensive attention [1], [2]. In systems theory, the significance of the Bayesian approach is highlighted in state estimation frameworks for dynamical systems [3], [4], e.g., through the celebrated Kalman *filter* which is a recursive causal filter. *Smoother* counterparts, as the Rauch-Tung-Striebel (RTS) [4], on the other hand, are (offline) iterative non-causal algorithms incorporating also future measurements into the current state estimation. An alternative to Bayesian estimation, which requires a prior distribution of the parameters of interest, are minimax estimation approaches assuming instead knowledge of ambiguity sets. The *least favorable* uncertainty model from this ambiguity set is then used for estimation [5]–[8]. Here, we focus instead on designing a classical smoother for a different problem: estimation of system parameters from input/output measurements via Bayesian estimation. This problem arises in, e.g., robot mapping in unknown environments, such as Autonomous Underwater Vehicles (AUVs) operating in the deep sea where global positioning is expensive due to low visibility and lack of radio communications.

Given the parameters are unknown and the states are random variables, applying a Bayesian framework leads to severe non-convexities in the resulting estimation problem. To overcome the non-convexity of these optimization problems, typically iterative schemes are employed. Assuming the parameters also follow a statistical formulation, two main types of smoother approaches: *Dual Kalman Smoother* (DKS) and *Expectation Maximization* (EM) are available in the literature [9]. Dual Kalman Smoothers (and filters) attempt to maximize the joint probability space of parameters and state (conditioned on input and output observations), iterating between estimating the system states using the last parameters' estimates followed by estimating the parameters from the currently estimated states. While DKS is computationally efficient according to its recursive structure, its estimation performance can be significantly suboptimal due to bilinearity between the parameters and the states. The *Expectation Maximization* maximizes the posterior distribution of the parameters from the observed data and their prior density function when incomplete data or hidden variables exist [10]–[12]. This method has been used to estimate parameters of dynamical systems, considering the states as hidden variables [13]–[15] by integrating all possible values of the states in which the model could have generated the observations. The distribution over hidden variables is maximized in the *E-Step* using the parameters estimates from the previous iteration. The *M-Step* maximizes a lower-bound of the original cost by fixing the distribution to the one optimized in the *E-Step*. A closed-form solution of the *M-Step* is provided in [14] for estimating the parameters of linear time-invariant dynamical systems and in [9, Chapter 6] for estimating the parameters of a Gaussian radial basis function (RBF) approximator. Both solutions consider the maximum likelihood case, where no prior exists for the parameters. Finding a closed-form expression for the parameters update in the *M-Step* of a MAP smoothing problem when the parameters are time-varying and in the presence of a-priori knowledge is non-trivial. This challenge leads to a slow convergence of the EM algorithm utilizing computationally demanding approaches to solve the optimization in *M-step*, e.g., first-order methods. The slow convergence of EM is shown experimentally in [16], and further analyses in [17], [18] demonstrate the slow convergence rate of the gradient variant of the EM algorithm for Gaussian Mixture Models.

Alternatives to these iterative schemes can be found in the parameter estimation problem of an elliptically contoured distribution [19, Page 107], employing recent Conic Geometric Optimization methods [20]. However, they require reformulating the MAP problem employing techniques, such as those proposed in [21, Section 3], which result in losing the output map's original structure. We are interested in retaining such a structure in order to leverage the available a-priori knowledge.

In this work, we focus on systems with known linear time-varying dynamics affected by process and measure-

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 956200.

The authors are with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. S.Vakili@tudelft.nl, Mohammad.Khosravi@tudelft.nl, P.MohajerinEsfahani@tudelft.nl, M.Mazo@tudelft.nl

ment Gaussian noise but with unknown time-varying output maps. We propose a method to estimate the parameters of the unknown output map having as *a priori* information a linear stochastic system encoding the evolution of the parameters. We derive an optimization problem applying a fully Bayesian approach, maximizing the exact posterior distribution of the parameters when unfolded over the whole time horizon. A tractable conservative approximation to the resulting optimization problem is derived via a linear matrix inequalities (LMIs), providing a warm-start for a first-order quasi-Newton algorithm that enjoys a locally superlinear convergence rate. This combination allows us to enjoy both the computational advantage of DKS and outperform the statistical performance of EM. We illustrate the efficacy and performance of our proposed method in comparison with DKS and EM through a Monte Carlo experiment with different signal-to-noise ratios (SNR) in Section V.

Notation*:* Throughout this paper $\mathbb{Z}_+$, $\mathbb{R}$, and $\mathbb{R}^{n\times m}$ denote the set of positive integers, real numbers, and n by m real matrices, respectively. Given matrices $\mathrm{A}_1,\dots,\mathrm{A}_k$, we denote by $\mathrm{diag}(\mathrm{A}_1,\dots,\mathrm{A}_k)$ as the block diagonal matrix with diagonal entries $\mathrm{A}_1,\dots,\mathrm{A}_k$. The symbol $\mathbb{I}$ denotes the identity matrix, and tr is the trace operator. Given $\mathrm{A}\in\mathbb{R}^{m\times n}$, a matrix with columns $\mathrm{a}_1,\dots,\mathrm{a}_n\in\mathbb{R}^m$, we define $\mathrm{vec}(\mathrm{A})$ as the vector $[\mathrm{a}_1^\mathsf{T},\dots,\mathrm{a}_n^\mathsf{T}]^\mathsf{T}\in\mathbb{R}^{mn}$. For a positive symmetric matrix $\mathrm{A}\in\mathbb{R}^{n\times n}$, $\Lambda(\mathrm{A}):=\big(\lambda_i(\mathrm{A})\big)_{i=1}^n$ denotes the vector of eigenvalues of A in a descending order, i.e., $\lambda_i(\mathrm{A})$ is the i^{th} largest eigenvalue of A. A multivariate normal (Gaussian) distribution with mean μ and covariance matrix Σ is denoted by $\mathcal{N}(\mu,\Sigma)$, and the symbol $\sim$ stands for *"distributed according to"*.

II. Problem Definition

Consider a discrete-time linear time-varying dynamical system described by the process model:

$$\mathrm{x}_{k+1}=\mathrm{A}_k\mathrm{x}_k+\mathrm{B}_k\mathrm{u}_k+\mathrm{w}_k,\qquad k\in\mathbb{Z}_+,\tag{1}$$

where k denotes the time index, $\mathrm{x}_k\in\mathbb{R}^{n_\mathrm{x}}$ is the vector of state variables, $\mathrm{A}_k\in\mathbb{R}^{n_\mathrm{x}\times n_\mathrm{x}}$ is the state transition matrix, $\mathrm{u}_k\in\mathbb{R}^{n_\mathrm{u}}$ is the vector of inputs, $\mathrm{B}_k\in\mathbb{R}^{n_\mathrm{x}\times n_\mathrm{u}}$ is the input matrix, and $\mathrm{w}_k\in\mathbb{R}^{n_\mathrm{x}}$ is an independent realization at time k of the process noise with Gaussian distribution $\mathcal{N}(0,\Sigma_{\mathrm{w}_k})$. The initial state of system (1), denoted by x_0, is also assumed to be drawn from a Gaussian distribution $\mathcal{N}(\mu_{\mathrm{x}_0},\Sigma_{\mathrm{x}_0})$. For $k\in\mathbb{Z}_+$, the state of the system is observed at time instant k through a perturbed linear time-varying map:

$$\mathrm{y}_k=\mathrm{C}_k\mathrm{x}_k+\mathrm{v}_k,\qquad k\in\mathbb{Z}_+,\tag{2}$$

where $\mathrm{y}_k\in\mathbb{R}^{n_\mathrm{y}}$ denotes the output measurements, $\mathrm{C}_k\in\mathbb{R}^{n_\mathrm{y}\times n_\mathrm{x}}$ is an *unknown time-varying* observation matrix, and $\mathrm{v}_k\in\mathbb{R}^{n_\mathrm{y}}$ is the vector of measurement noise signals with Gaussian distribution $\mathcal{N}(0,\Sigma_{\mathrm{v}_k})$. Let θ_k be the vector of all parameters at each time index k:

$$\theta_k:=\mathrm{vec}(\mathrm{C}_k^\mathsf{T}),\tag{3}$$

which implies that C_k and θ_k uniquely characterize each other. We introduce the following assumption, providing a form of *a priori* information. This plays a role akin to that of a regularizer in non-Bayesian techniques, such as in *Supervised Learning*, where algorithms without such regularizers are prone to overfitting.

Assumption 1 (Output map dynamics)**.** *The dynamics of the output map are governed by the difference equation*

$$\theta_{k+1}=\theta_k+\eta_k,\qquad k\in\mathbb{Z}_+,\tag{4}$$

where k denotes the time index, $\theta_k\in\mathbb{R}^{n_\mathrm{y}n_\mathrm{x}}$ is the vector of parameters driven by the vector of process noise $\eta_k\in\mathbb{R}^{n_\mathrm{y}n_\mathrm{x}}$ with Gaussian distribution $\mathcal{N}(\mu_{\eta_k},\Sigma_{\eta_k})$. Further, assume that the initial parameter of system (4), denoted by θ_0, is drawn from the normal distribution $\mathcal{N}(\mu_{\theta_0},\Sigma_{\theta_0})$.

Assumption 1 imposes a Gaussian random walk dynamics on the evolution of the parameters, which is the minimal structure and assumption on the variations of the parameters because of the maximum entropy feature of the Gaussian distributions. This allows us to employ a stochastic belief of a deterministic reality in the *Bayesian* viewpoint. Let the inputs and outputs of system (1)-(2) be measured for $k=0,\dots,n_\mathcal{T}$, where $(n_\mathcal{T}+1)\in\mathbb{Z}_+$ denotes the length of the measurement data. More precisely, the input-output trajectory data is given by $\mathcal{D}=\{(\mathrm{u}_k,\mathrm{y}_k)\,|\,k=0,\dots,n_\mathcal{T}\}$. Additionally, we assume:

Assumption 2 (Noise)**.** *The process, measurement, and output map noise realizations, w_k, v_k, and η_k respectively, for all $k\in\{0,\dots,n_\mathcal{T}\}$, are independent. Furthermore, the means μ_{x_0}, μ_{θ_0}, μ_{η_k}, and covariance matrices Σ_{x_0}, Σ_{w_k}, Σ_{v_i}, Σ_{θ_0}, and Σ_{η_k}, for $k\in\{0,\dots,n_\mathcal{T}\}$, are known.*

Remark 1 (A priori knowledge)**.** While we assume μ_{θ_0}, μ_{η_k}, Σ_{θ_0}, and Σ_{η_k} to be readily known, in practical applications, these parameters can be obtained through various means depending on the context, e.g.: employing prior knowledge of the nominal model, empirically from previous experiments' data, or if one may assume that $\mu_{\eta_k}=\mu_{\theta_0}$ and $\Sigma_{\eta_k}=\Sigma_{\theta_0}$, for $k\in\mathbb{Z}_+$ by employing a suitable hyperparameter estimation method.

Ultimately, the question is whether the observation model (2) could be estimated. More precisely, we would like to address the following problem:

Problem 1. *Given the process and observation models (1) and (2), input-output measurement data $\mathcal{D}$, and under Assumptions 1 and 2, estimate the* unknown time-varying *observation matrices C_k in an efficient and tractable way.*

To address problem 1, we develop a *Maximum A Posteriori* approach in the next section, followed by a tractable reformulation via LMIs in Section IV.

III. Maximum A Posteriori Estimation

In this section, we propose a Bayesian method for estimating the unknown observation matrices $\mathrm{C}_0,\dots,\mathrm{C}_{n_\mathcal{T}}$. The three main elements in Bayesian estimation methods are a prior density function, an observation model, and a loss

function, which we briefly explain for solving our problem with the *Maximum a Posteriori* (MAP) approach.

A. Lifted Process and Observation Model

First, let us represent the process model (1) in the following *lifted matrix form*:

$$\mathrm{x} = \mathrm{A}(\mathrm{u} + \mathrm{w_x}), \tag{5}$$

where $\mathrm{x} = \left[\mathrm{x}_0^\mathsf{T}, \dots, \mathrm{x}_{n_\mathcal{T}}^\mathsf{T}\right]^\mathsf{T}$ includes the system states over the entire horizon up to time $n_\mathcal{T}$, while the input vector is modified to include the initial state $\mathrm{u} = \left[\mu_{\mathrm{x}_0}^\mathsf{T}, (\mathrm{B}_0\mathrm{u}_0)^\mathsf{T}, \dots, (\mathrm{B}_{n_\mathcal{T}-1}\mathrm{u}_{n_\mathcal{T}-1})^\mathsf{T}\right]^\mathsf{T}$, and the noise vector consists of the uncertainty of the initial state and process noises $\mathrm{w_x} = \left[\mathrm{w}_{\mathrm{x}_0}^\mathsf{T}, \mathrm{w}_0^\mathsf{T}, \dots, \mathrm{w}_{n_\mathcal{T}-1}^\mathsf{T}\right]^\mathsf{T}$ with $\mathrm{w}_{\mathrm{x}_0} \sim \mathcal{N}(0, \Sigma_{\mathrm{x}_0})$. Given that the process noise and initial state uncertainty are assumed to be uncorrelated, we can specify $\mathrm{w_x}$ in terms of a multivariate normal distribution $\mathcal{N}(0, \Sigma_{\mathrm{w_x}})$ in which $\Sigma_{\mathrm{w_x}} = \mathrm{diag}(\Sigma_{\mathrm{x}_0}, \Sigma_{\mathrm{w}_0}, \dots, \Sigma_{\mathrm{w}_{n_\mathcal{T}-1}})$. The *lifted* transition matrix A has the lower triangular form:

$$\mathrm{A} = \begin{bmatrix} \mathbb{I} & & & & \\ A_0 & \mathbb{I} & & & \\ A_1A_0 & A_1 & \mathbb{I} & & \\ \vdots & \vdots & \ddots & \ddots & \\ A_{n_\mathcal{T}-1}\dots A_0 & A_{n_\mathcal{T}-1}\dots A_1 & \dots & A_{n_\mathcal{T}-1} & \mathbb{I} \end{bmatrix}.$$

Similarly, the observation model (2) for the entire trajectory can be expressed as:

$$\mathrm{y} = \mathrm{Cx} + \mathrm{v}, \tag{6}$$

where $\mathrm{y} = \left[\mathrm{y}_0^\mathsf{T}, \dots, \mathrm{y}_{n_\mathcal{T}}^\mathsf{T}\right]^\mathsf{T}$ is the vector of all measurements, $\mathrm{v} \sim \mathcal{N}(0, \Sigma_\mathrm{v})$ is the vector of all measurement noise realizations with $\Sigma_\mathrm{v} = \mathrm{diag}(\Sigma_{\mathrm{v}_0}, \Sigma_{\mathrm{v}_1}, \dots, \Sigma_{\mathrm{v}_{n_\mathcal{T}}})$, and C is the lifted observation matrix:

$$\mathrm{C} = \mathrm{diag}(\mathrm{C}_0, \mathrm{C}_1, \dots, \mathrm{C}_{n_\mathcal{T}}).$$

Finally, we describe the dynamics of the output map parameters θ for the entire trajectory as:

$$\theta = \mu_\theta + \mathrm{w}_\theta, \tag{7}$$

where $\theta = \left[\theta_0^\mathsf{T}, \dots, \theta_{n_\mathcal{T}}^\mathsf{T}\right]^\mathsf{T}$, and μ_θ results from the summation of the biases of the initial parameter and the noise $\mu_\theta = \left[\mu_{\theta_0}^\mathsf{T}, \mu_{\theta_0}^\mathsf{T} + \mu_{\eta_0}^\mathsf{T}, \dots, \mu_{\theta_0}^\mathsf{T} + \sum_{i=0}^{n_\mathcal{T}-1} \mu_{\eta_i}^\mathsf{T}\right]^\mathsf{T}$. Similarly, the noise signal w_θ results from the integration over the entire horizon of η_k including the uncertainty of the initial parameter as $\mathrm{w}_\theta = D\eta$, where $\eta = \left[\eta_{\theta_0}^\mathsf{T}, \eta_0^\mathsf{T}, \dots, \eta_{n_\mathcal{T}-1}^\mathsf{T}\right]^\mathsf{T}$ and $\eta_{\theta_0} \sim \mathcal{N}(0, \Sigma_{\theta_0})$, with

$$D = \begin{bmatrix} \mathbb{I} & & & \\ \mathbb{I} & \mathbb{I} & & \\ \vdots & \vdots & \ddots & \\ \mathbb{I} & \mathbb{I} & \dots & \mathbb{I} \end{bmatrix}.$$

Since the parameters are assumed to be independent, we have $\mathrm{w}_\theta \sim \mathcal{N}(0, \Sigma_{\mathrm{w}_\theta})$, $\Sigma_{\mathrm{w}_\theta} = \mathrm{D}\Sigma_\eta\mathrm{D}^\mathsf{T}$, with $\Sigma_\eta = \mathrm{diag}(\Sigma_{\theta_0}, \Sigma_{\eta_0}, \dots, \Sigma_{\eta_{n_\mathcal{T}-1}})$. Ultimately, the model (7) is used to specify the prior density function. In what follows, we represent C as $\mathrm{C}(\theta)$ to emphasize the dependence of C on θ according to (3). Finally, substituting x in (6) with the expression from (5) results in the observation model, with unknown $\mathrm{C}(\theta)$, describing the measurements y as a function of the applied inputs u:

$$\mathrm{y} = \mathrm{C}(\theta)\mathrm{Au} + \mathrm{w_y}(\theta), \tag{8}$$

where $\mathrm{w_y}(\theta) = \mathrm{C}(\theta)\mathrm{Aw_x} + \mathrm{v}$. Also, from Assumption 2:

$$\mathrm{w_y}(\theta)|\theta \sim \mathcal{N}(0, \Sigma_{\mathrm{w_y}}(\theta)),$$

where $\Sigma_{\mathrm{w_y}}(\theta) = \mathrm{C}(\theta)\mathrm{A}\Sigma_{\mathrm{w_x}}\mathrm{A}^\mathsf{T}\mathrm{C}(\theta)^\mathsf{T} + \Sigma_\mathrm{v}$. This model is used later to specify the conditional probability density function of the measurements. Note that $\mathrm{w_y}(\theta)|\theta$ remains Gaussian with the derived covariance since both noise sources, $\mathrm{w_x}$ and v, are Gaussian and independent.

B. MAP Loss Function

In *maximum a posteriori* estimation one aims to find an estimate $\hat{\theta}$ for the parameters by minimizing the cost function [22]: $\mathbb{E}\left[1 - \mathbb{1}_{\theta:\left\|\theta-\hat{\theta}\right\|_\infty \le \epsilon}(\theta)\right]$, where θ is the vector of random variables, $\mathbb{1}(.)$ is an indicator function, and ϵ a small scalar. It is further shown in [23, Chapter 4] that minimizing this loss function implies maximizing the conditional probability density function of θ given the vector of observations and inputs, i.e.

$$\hat{\theta} = \underset{\theta}{\mathrm{argmax}}\ p(\theta|\mathrm{y}, \mathrm{u}), \tag{9}$$

where $\hat{\theta}$ is the estimate of the true parameter θ. The following lemma formalizes this step and serves as the first step to compute the proposed estimator (9).

Lemma 1 (MAP Optimization Problem)**.** *Let us define the function* $\mathcal{J} : \mathbb{R}^{n_\mathrm{y} n_\mathrm{x}(n_\mathcal{T}+1)} \to \mathbb{R}$ *as*

$$\begin{aligned}\mathcal{J}(\theta) := \ & \mathrm{logdet}(\Sigma_{\mathrm{w_y}}(\theta)) \\ & + \left\|\mathrm{y} - \mathrm{C}(\theta)\mathrm{Au}\right\|^2_{\Sigma^{-1}_{\mathrm{w_y}}(\theta)} + \left\|\theta - \mu_\theta\right\|^2_{\Sigma^{-1}_{\mathrm{w}_\theta}}.\end{aligned} \tag{10}$$

The estimation problem (9) *is equivalent to*

$$\hat{\theta} = \underset{\theta}{\mathrm{argmin}}\ \mathcal{J}(\theta). \tag{11}$$

Proof. Using Bayes' rule we can reformulate the optimization problem (9) as

$$\max_\theta\ p(\theta|\mathrm{y}, \mathrm{u}) = \max_\theta\ \frac{p(\mathrm{y}|\theta, \mathrm{u})p(\theta|\mathrm{u})}{p(\mathrm{y}|\mathrm{u})}. \tag{12}$$

We first note that the denominator of (12) does not depend on θ, and hence can be neglected without changing the optimizer. Moreover, we note that the dynamics of θ in (4) (or equivalently in the lifted form in (7)) does not depend on the input sequence of u (i.e., $p(\theta|\mathrm{u}) = p(\theta)$). Next, using a straightforward computation, one can derive the probability density functions $p(\theta)$ and $p(\mathrm{y}|\theta, \mathrm{u})$. Specifically, from (7) we know that the variable θ is Gaussian with the probability density function

$$p(\theta) = \frac{\exp\left(-\frac{1}{2}(\theta - \mu_\theta)^\mathsf{T}\Sigma^{-1}_{\mathrm{w}_\theta}(\theta - \mu_\theta)\right)}{\sqrt{(2\pi)^{(n_\mathcal{T}+1)n_\mathrm{y}n_\mathrm{x}}\det(\Sigma_{\mathrm{w}_\theta})}}.$$

Similarly, from (8) we know that given θ and the input sequence u, the output sequence y is also Gaussian with the conditional probability density function

$$p(\mathrm{y}|\theta,\mathrm{u}) = \frac{\exp\Big(-\frac{1}{2}(\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u})^{\mathsf{T}}\Sigma_{\mathrm{w_y}}^{-1}(\theta)(\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u})\Big)}{\sqrt{(2\pi)^{(n_\mathcal{T}+1)n_\mathrm{y}}\det(\Sigma_{\mathrm{w_y}}(\theta))}}.$$

Finally, applying the monotonically increasing function $\log$, and observing that all terms in the denominators except $\det(\Sigma_{\mathrm{w_y}}(\theta))$ are constant, we arrive at the minimization problem of the function $\mathcal{J}$ defined in (11). ■

In the next section, we propose a tractable conservative approximation using Semidefinite programming to tackle the non-convex objective function $\mathcal{J}(\theta)$ defined in (10).

Remark 2 (Robust estimation)**.** Alternatively a robust minimax estimation formulation similar to [8] could be employed. This approach, however, requires finding an ambiguity set to approximate non-Gaussian observation uncertainties due to the multiplication of Gaussian variables in $\mathrm{w_y}(\theta)$.

IV. Proposed Solution

The optimization problem (11) is non-convex not only because of the weight $\Sigma_{\mathrm{w_y}}^{-1}(\theta)$, quadratic in the parameters θ, in the second term but also because of the log-determinant operator on the first term. A typical approach is to use first-order algorithms to find a solution due to the mentioned non-convexities. These algorithms, however, only guarantee convergence to a local optimum. Therefore, selecting an appropriate initial starting point is crucial to the obtained quality of the solution. We propose to solve the problem in two steps: first we perform a convex relaxation of (11) into a set of LMIs, which we use to compute an initial approximate minimizer; next, we employ this approximate minimizer to initialize (warm-start) a first-order optimization method, e.g., steepest descent [24] or quasi-Newton algorithms [25], to solve (11) thus refining our initial minimizer estimate.

Theorem 2 (LMI conservative approximation)**.** *Consider the following LMIs:*

$$\begin{aligned}
\min_{\mathrm{S},\theta,\gamma,\beta}\quad & \mathrm{tr}(\mathrm{S}-\mathbb{I})+\gamma+\beta \\
\text{s.t.}\quad & \begin{bmatrix} -\Sigma_{\mathrm{w_x}}^{-1} & \mathrm{A}^{\mathsf{T}}\mathrm{C}(\theta)^{\mathsf{T}} \\ \mathrm{C}(\theta)\mathrm{A} & \Sigma_\mathrm{v}-\mathrm{S}\end{bmatrix} \succeq 0, \\
& \begin{bmatrix} -\mathrm{S} & (\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u}) \\ (\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u})^{\mathsf{T}} & -\gamma\end{bmatrix} \preceq 0, \\
& \begin{bmatrix} -\Sigma_{\mathrm{w}_\theta} & (\theta-\mu_\theta) \\ (\theta-\mu_\theta)^{\mathsf{T}} & -\beta\end{bmatrix} \preceq 0.
\end{aligned} \tag{13}$$

Then, the optimal value of the nonlinear program (11) *is upper bounded by* $J^*+\big\|\mathrm{y}-\mathrm{C}(\theta^*)\mathrm{A}\mathrm{u}\big\|^2_{\Sigma_{\mathrm{w_y}}^{-1}(\theta^*)-\mathrm{S}^{*-1}}$*, where* J^* *and* (S^*,θ^*) *are the optimal value and the optimizer of* (13)*, respectively.*

Proof. Consider a matrix $\mathrm{S}\succ 0$ upper bounding the covariance matrix $\Sigma_{\mathrm{w_y}}(\theta)\succ 0$ as

$$\Sigma_{\mathrm{w_y}}(\theta) = \mathrm{C}(\theta)\mathrm{A}\Sigma_{\mathrm{w_x}}\mathrm{A}^{\mathsf{T}}\mathrm{C}(\theta)^{\mathsf{T}}+\Sigma_\mathrm{v} \preceq \mathrm{S}. \tag{14}$$

Thus, $\lambda_i(\Sigma_{\mathrm{w_y}}(\theta)) \le \lambda_i(\mathrm{S})$, for $i=1,\dots,n_\mathrm{y}(n_\mathcal{T}+1)$, which implies that

$$\mathrm{logdet}(\Sigma_{\mathrm{w_y}}(\theta)) \le \mathrm{logdet}(\mathrm{S}).$$

Since $\mathrm{logdet}(\mathrm{S}) = \sum_{i=1}^{n_\mathrm{y}(n_\mathcal{T}+1)}\log\lambda_i(\mathrm{S})$ and $\mathrm{tr}(\mathrm{S}) = \sum_{i=1}^{n_\mathrm{y}(n_\mathcal{T}+1)}\lambda_i(\mathrm{S})$, we also have

$$\mathrm{logdet}(\Sigma_{\mathrm{w_y}}(\theta)) \le \mathrm{logdet}(\mathrm{S}) \le \mathrm{tr}(\mathrm{S}-\mathbb{I}).$$

Using the *Schur complement*, one can see that (14) is equivalent to the following linear matrix inequality

$$\begin{bmatrix} -\Sigma_{\mathrm{w_x}}^{-1} & \mathrm{A}^{\mathsf{T}}\mathrm{C}(\theta)^{\mathsf{T}} \\ \mathrm{C}(\theta)\mathrm{A} & \Sigma_\mathrm{v}-\mathrm{S}\end{bmatrix} \preceq 0.$$

Similarly, considering $\gamma\ge 0$ and $\beta\ge 0$ such that

$$\begin{aligned}(\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u})^{\mathsf{T}}\mathrm{S}^{-1}(\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u}) \le \\ (\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u})^{\mathsf{T}}\Sigma_{\mathrm{w_y}}(\theta)^{-1}(\mathrm{y}-\mathrm{C}(\theta)\mathrm{A}\mathrm{u}) \le \gamma,\end{aligned} \tag{15}$$

and

$$(\theta-\mu_\theta)^{\mathsf{T}}\Sigma_{\mathrm{w}_\theta}^{-1}(\theta-\mu_\theta)\le\beta, \tag{16}$$

we can apply again the *Schur complement* to the inequalities in (15) and (16), and obtain the last two LMIs in (13). Finally, replacing the terms in the cost function $\mathcal{J}(\theta)$ in (10) with their bounds and including the corresponding LMIs as constraints arrives at the LMI (16). Note further that by definition we have

$$\begin{aligned}J^*+\big\|\mathrm{y}-\mathrm{C}(\theta^*)\mathrm{A}\mathrm{u}\big\|^2_{\Sigma_{\mathrm{w_y}}^{-1}(\theta^*)-\mathrm{S}^{*-1}} = \\ \mathcal{J}(\theta^*)+\mathrm{tr}(\mathrm{S}^*-\mathbb{I})-\mathrm{logdet}(\Sigma_{\mathrm{w_y}}(\theta^*)) \ge \mathcal{J}(\theta^*),\end{aligned} \tag{17}$$

where the function $\mathcal{J}(\theta^*)$ is defined in (10), and the last inequality follows from (14). ■

The tightness of the inequality in (17) mainly depends on the gap between $\log\det(\mathrm{S})$ and $\mathrm{tr}(\mathrm{S}-\mathbb{I})$ since $\log\det(\mathrm{S})$ is bounded from above by $\mathrm{tr}(\mathrm{S}-\mathbb{I})$, which is negligible when $\lambda_i(\mathrm{S})\approx 1$, for $i=1,\dots,n_\mathrm{y}(n_\mathcal{T}+1)$. One may employ a suitable matrix W to scale the eigenvalues of S, replace $\log\det(\mathrm{S})$ with $\log\det(\mathrm{WSW})-2\log\det(\mathrm{W})$ and approximate $\log\det(\mathrm{WSW})$ with $\mathrm{tr}(\mathrm{WSW}-\mathbb{I})$. Furthermore, the closeness of J^* and $\mathcal{J}(\theta^*)$ in (17) is proportional to the fitness quality of the measurements and whether S^* is close to the covariance matrix accordingly.

In addition, Theorem 2 provides an approximation of (11) producing an initial near-optimal solution. As already indicated, we propose to employ this solution to warm-start a local (non-convex) optimizer. Due to its fast convergence, we propose to employ as refining optimizer the BFGS algorithm [25, Chapter 6], a variant of quasi-Newton methods. The BFGS algorithm approximates the Hessian matrix for its search directions relying on an analytical expression of the gradient $\nabla_\theta\mathcal{J}(\theta)$. The gradient of the cost function (10) with respect to the parameters θ

$$\nabla_\theta\mathcal{J}(\theta) = \Big[\frac{\partial\mathcal{J}}{\partial\theta_1},\dots,\frac{\partial\mathcal{J}}{\partial\theta_{n_\mathrm{y}n_\mathrm{x}(n_\mathcal{T}+1)}}\Big]^{\mathsf{T}} \tag{18}$$

can be easily derived applying the chain rule:

$$\frac{\partial \mathcal{J}}{\partial \theta_{ijk}} = 2\mathrm{tr}\Big[\Big(\mathrm{A}\Sigma_{\mathrm{w_x}}\mathrm{A}^\intercal \mathrm{C}(\theta)^\intercal \Sigma_{\mathrm{w_y}}(\theta)^{-1}\Big)\mathrm{C}_k^{ij}(\theta) - \Big(\mathrm{A}\Sigma_{\mathrm{w_x}}\mathrm{A}^\intercal \mathrm{C}(\theta)^\intercal \Sigma_{\mathrm{w_y}}(\theta)^{-1} \times (\mathrm{y} - \mathrm{C}(\theta)\mathrm{Au})(\mathrm{y} - \mathrm{C}(\theta)\mathrm{Au})^\intercal \Sigma_{\mathrm{w_y}}(\theta)^{-1}\Big)\mathrm{C}_k^{ij}(\theta) - \Big(\mathrm{Au}(\mathrm{y} - \mathrm{C}(\theta)\mathrm{Au})^\intercal \Sigma_{\mathrm{w_y}}(\theta)^{-1}\Big)\mathrm{C}_k^{ij}(\theta) + \Big((\theta - \mu_\theta)^\intercal \Sigma_{\mathrm{w}_\theta}^{-1}\Big)\theta^{ijk}\Big],$$

where $\mathrm{C}_k^{ij}(\theta)$ is the single-entry matrix of $\mathrm{C}(\theta)$ with the block matrix of $\mathrm{C}_k(\theta)$ having 1 at index (i, j) and zero elsewhere, and θ^{ijk} is the single-entry vector of θ with 1 at index ijk and zero elsewhere.

The LMIs (13) initializes the original non-convex problem with a locally optimal solution. Thus, the computational complexity of the proposed method consists of the well-known computational complexity of solving the SDP problems [26], i.e., a one-time solution of (13), and the computation of the gradient (18) per iteration of the first-order method:

$$\mathcal{O}\Big(\frac{n_{\mathrm{x}}^3(n_\mathcal{T}+1)^3 + n_{\mathrm{x}}^2(n_\mathcal{T}+1)^2}{2} + n_{\mathrm{y}}n_{\mathrm{x}}^2(n_\mathcal{T}+1)^2 + n_{\mathrm{x}}n_{\mathrm{y}}^2(n_\mathcal{T}+1)^3 + (n_{\mathrm{y}})^3(n_\mathcal{T}+1)^3\Big),$$

which is $\mathcal{O}(n_\mathcal{T}^3)$ when $n_{\mathrm{x}}, n_{\mathrm{y}} \ll n_\mathcal{T}$.

V. Numerical Example

In this section, we provide a numerical example to verify the efficacy and performance of the proposed method: employing the LMIs (13) to warm-start the solution of (11) via the BFGS optimizer. Additionally, we compare the resulting solution with the estimates obtained from EM and DKS algorithms. To obtain a fair comparison, we also employ the same BFGS optimizer for the *M-Step* of the EM estimation.

We demonstrate our solution on a three-dimensional, i.e., $n_{\mathrm{x}} = 3$, linear time-invariant process model

$$\mathrm{x}_{k+1} = \begin{bmatrix} 0.7 & 0.25 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0.25 & 0.7 \end{bmatrix} \mathrm{x}_k + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} (3.5 + \cos(2k)) + \mathrm{w}_k,$$

with $\mu_{\mathrm{x}_0} = [1, 0.5, 2]^\intercal$. The observation is a two-dimensional model, i.e., the number of measurements per time instant is $n_{\mathrm{y}} = 2$. The system has sampled input and measurement pairs in $\mathcal{D}$ every 100 milliseconds for 10 seconds, i.e., $n_\mathcal{T} = 100$. As such, the number of parameters to be estimated is $n_{\mathrm{y}}n_{\mathrm{x}}(n_\mathcal{T}+1) = 606$. The noise covariance of process, observation and output map dynamics, Σ_{w_k}, Σ_{v_k} and Σ_{η_k}, are assumed to remain constant across the entire horizon. The initial state and parameter noise covariance are also assumed similar to the noise covariance of process and output map dynamics, respectively (i.e., $\Sigma_{\mathrm{x}_0} = \Sigma_{\mathrm{w}_k}$ and $\Sigma_{\theta_0} = \Sigma_{\eta_k}$). The output map noise biases μ_{η_k} are generated such that $\mu_{1,\eta_k} = 5 + \mathrm{e}^{-0.6k}\cos(0.4k)$, $\mu_{2,\eta_k} = 1.5 + \mathrm{e}^{-0.6k}\sin(0.025k)$, $\mu_{3,\eta_k} = 2$, $\mu_{4,\eta_k} = 5 + \mathrm{e}^{-0.6k}\cos(0.4k)$, $\mu_{5,\eta_k} = 1.5 + \mathrm{e}^{-0.6k}\sin(0.025k)$, and $\mu_{6,\eta_k} = 2$. The initial parameter bias, μ_{θ_0}, is derived from μ_{η_0} by setting $k = 0$.

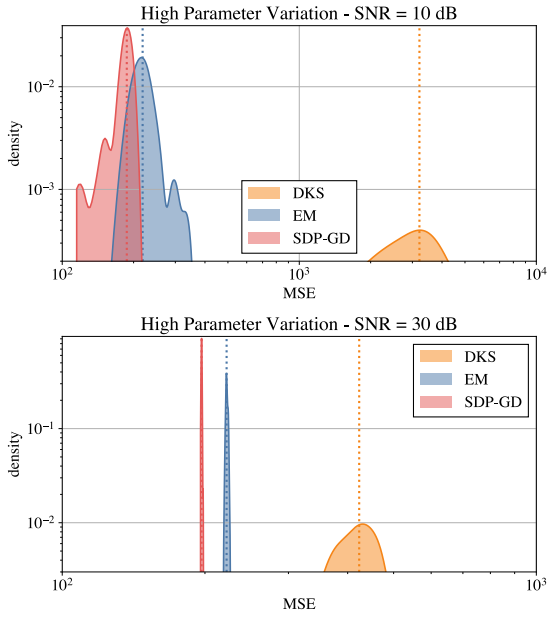

Fig. 1: The Mean Squared Error of the three methods in high noise of Σ_η and two different SNRs for 100 experiments.

The DKS and EM algorithms are initialized with these noise bias values. We examine the performance of our algorithm, SDP-GD, compared with EM and DKS on four different scenarios generated by employing High/Low SNR for the process and observation noise, specifically 30 and 10 dB, and *High/Low* parameter variation of:

$$\text{High:} \quad \Sigma_{\eta_k} = \mathrm{diag}(2.17,\ 0.076,\ 1.19,\ 1.38,\ 0.87,\ 1.27)$$
$$\text{Low:} \quad \Sigma_{\eta_k} = \mathrm{diag}(6.9,\ 0.2,\ 3.8,\ 4.4,\ 2.8,\ 4) \cdot 10^{-2}.$$

Combined results from 100 experiments for each of the four scenarios, keeping the same ground-truth realization in each of the scenarios, are illustrated in Figures 1 and 2. In the figures, we illustrate the median (vertical dotted lines) and distribution across experiments of the mean squared error (MSE) of the predicted parameters, i.e. $\mathrm{MSE} = \frac{1}{n_\mathcal{T}+1}\sum_{k=0}^{n_\mathcal{T}} \|\theta_k - \hat{\theta}_k\|_2^2$. One can observe in the figures how the DKS underperforms compared to the EM and our SDP-GD solutions in more challenging scenarios where the process-observation model noise is high or in the presence of *High* parameter variation.

The average and standard deviation of the computation time of each method across 100 experiments are reported in Table I. We performed all the experiments on a cluster node with 384G memory and 40 CPU cores (2 Intel Xeon Gold 6148 @ 2.40GHz). The elapsed execution times confirm our hypothesis that EM is computationally more expensive than the other alternatives. The performance of EM and DKS algorithms highly depends on the initialization, while in contrast, our proposed solution takes advantage of a warm-start initializer obtained from solving a convexified approximation of the original optimization problem. This initialization helps converge to a better local optimum faster than the EM algorithm. In addition, the *M-step* of the EM algorithm, in

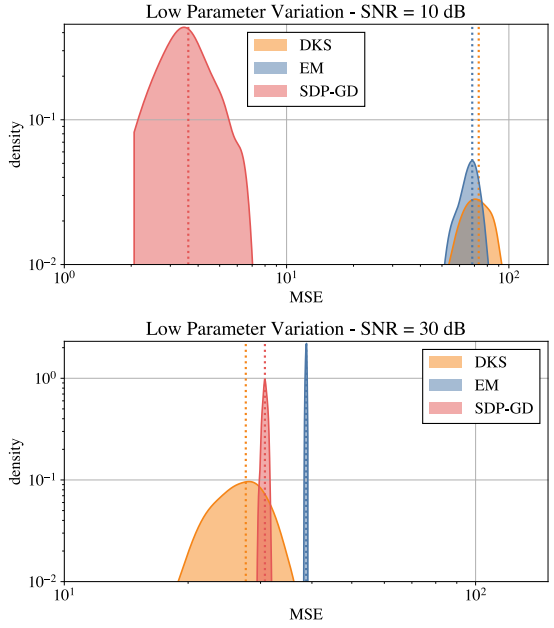

Fig. 2: The Mean Squared Error of the three methods in low noise of Σ_η and two different SNRs for 100 experiments.

Elapsed time per seconds (mean $\pm$ std)

	Experiment Scenario	DKS	EM	SDP-GD
10 dB	*Low* Parameter Variation	18 $\pm$ 4	14265 $\pm$ 3112	1265 $\pm$ 109
	High Parameter Variation	27 $\pm$ 13	22547 $\pm$ 6768	1542 $\pm$ 178
30 dB	*Low* Parameter Variation	9 $\pm$ 3	7999 $\pm$ 802	1543 $\pm$ 132
	High Parameter Variation	21 $\pm$ 13	3796 $\pm$ 412	1605 $\pm$ 128

TABLE I: The Average computation performance on all scenarios for 100 experiments.

this problem, does not hold a closed-form solution, which results in utilizing a first-order method. This gradient *M-Step* also plays a part in the general slowness of the EM algorithm. Our algorithm, however, requires a one-time execution of the set of LMIs followed by an iterative quasi-Newton method with a superlinear convergence rate. Hence, it provides the best of both worlds, i.e., better estimations than EM and DKS with less computation time than EM.

VI. Conclusion

We have introduced a method for the estimation of an unknown output map of a linear time-varying system. We employed a stochastic characterization of the evolution of the output map parameters, which serves as *a priori* information on a MAP optimization to solve the estimation problem. The MAP optimization is solved by relaxing the optimization as an SDP, whose solution serves as warm-start for a gradient descent algorithm. Comparing with standard approaches to solve this problem, namely EM and DKS, we showed experimentally the superiority of our method in estimation performance, and lower computational demands compared to EM. Future work will explore the incorporation of other types of a priori knowledge on the output map, the development of efficient causal filters following similar approaches, the minimax formulation for robust estimation, considering noise models with more general structures, and introducing methods for efficient design of the control input.

References

[1] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017-07-31.

[2] R. van de Schoot, S. Depaoli, R. King, B. Kramer, K. Märtens, M. G. Tadesse, M. Vannucci, A. Gelman, D. Veen, J. Willemsen, *et al.*, "Bayesian statistics and modelling," *Nature Reviews Methods Primers*, vol. 1, no. 1, p. 1, 2021.

[3] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Journal of Basic Engineering*, vol. 83, no. 1, pp. 95–108, 1961.

[4] H. Rauch, "Solutions to the linear smoothing problem," *IEEE Transactions on Automatic Control*, vol. 8, no. 4, pp. 371–372, 1963.

[5] S. Yi and M. Zorzi, "Robust kalman filtering under model uncertainty: The case of degenerate densities," *IEEE Transactions on Automatic Control*, vol. 67, no. 7, pp. 3458–3471, 2021.

[6] S. Shafieezadeh Abadeh, V. A. Nguyen, D. Kuhn, and P. M. Mohajerin Esfahani, "Wasserstein distributionally robust kalman filtering," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[7] V. A. Nguyen, S. Shafieezadeh-Abadeh, D. Kuhn, and P. Mohajerin Esfahani, "Bridging bayesian and minimax mean square error estimation via wasserstein distributionally robust optimization," *Mathematics of Operations Research*, vol. 48, no. 1, pp. 1–37, 2023.

[8] S. Yi and M. Zorzi, "Robust fixed-lag smoothing under model perturbations," *Journal of the Franklin Institute*, vol. 360, no. 1, 2023.

[9] S. Haykin, *Kalman filtering and neural networks*. John Wiley & Sons, 2004.

[10] L. E. Baum and J. A. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology," *Bulletin of the American Mathematical Society*, vol. 73, no. 3, pp. 360–363, 1967.

[11] S. Liu, X. Zhang, L. Xu, and F. Ding, "Expectation–maximization algorithm for bilinear systems by using the rauch–tung–striebel smoother," *Automatica*, vol. 142, p. 110365, 2022.

[12] M. Zheng and Y. Ohta, "Bayesian positive system identification: Truncated Gaussian prior and hyperparameter estimation," *Systems & Control Letters*, vol. 148, p. 104857, 2021.

[13] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," *Automatica*, vol. 47, no. 1, 2011.

[14] Z. Ghahramani and G. E. Hinton, "Parameter estimation for linear dynamical systems," *Technical Report CRG-TR-96-2*, 1996.

[15] N. Sammaknejad, Y. Zhao, and B. Huang, "A review of the expectation maximization algorithm in data-driven process identification," *Journal of process control*, vol. 73, pp. 123–136, 2019.

[16] I. Naim and D. Gildea, "Convergence of the EM algorithm for Gaussian mixtures with unbalanced mixing coefficients," *arXiv preprint arXiv:1206.6427*, 2012.

[17] S. Balakrishnan, M. J. Wainwright, and B. Yu, "Statistical guarantees for the em algorithm: From population to sample-based analysis," *The Annals of Statistics*, 2017.

[18] B. Yan, M. Yin, and P. Sarkar, "Convergence of gradient EM on multi-component mixture of Gaussians," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[19] M. E. Johnson, *Multivariate statistical simulation: A guide to selecting and generating continuous multivariate distributions*. John Wiley & Sons, 1987, vol. 192.

[20] S. Sra and R. Hosseini, "Conic geometric optimization on the manifold of positive definite matrices," *SIAM Journal on Optimization*, vol. 25, no. 1, pp. 713–739, 2015.

[21] R. Hosseini and S. Sra, "An alternative to EM for Gaussian mixture models: Batch and stochastic Riemannian optimization," *Mathematical programming*, vol. 181, no. 1, pp. 187–223, 2020.

[22] Z. Chen *et al.*, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[23] B. C. Levy, *Principles of signal detection and parameter estimation*. Springer Science & Business Media, 2008.

[24] D. G. Luenberger, *Optimization by vector space methods*. John Wiley & Sons, 1997.

[25] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.

[26] Y. Nesterov and A. Nemirovskii, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.