2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

An Efficient Method for the Joint Estimation of System Parameters and Noise Covariances for Linear Time-Variant Systems

Léo Simpson[1], Andrea Ghezzi[2], Jonas Asprion[1], Moritz Diehl[2,3]

***Abstract*— We present an optimization-based method for the joint estimation of system parameters and noise covariances of linear time-variant systems. Given measured data, this method maximizes the likelihood of the parameters. We solve the optimization problem of interest via a novel structure-exploiting solver. We present the advantages of the proposed approach over commonly used methods in the framework of Moving Horizon Estimation. Finally, we show the performance of the method through numerical simulations on a realistic example of a thermal system. In this example, the method can successfully estimate the model parameters in a short computational time.**

I. Introduction

System identification and estimation enable us to build accurate models which is a fundamental prerequisite for successfully solving control tasks. Having precise models also allow for reliable predictions about the system behavior which are essential for the deployment of Model Predictive Control (MPC) [1].

In the context of system identification, subspace methods are widely used for identifying linear systems [2]–[4]. However, these methods cannot enforce any particular structure, which is often given by the laws of physics. Parametric system identification overcomes this limitation [4]. For online state estimation of linear systems, several methods exists such as the Kalman filter (KF) [5]. To apply one of these state estimation methods, it is often necessary to estimate the covariances of the noise model using the available data, and one could use, e.g., covariance matching [6], or correlation techniques [7].

The Maximum Likelihood Estimation (MLE) problem for parametric linear dynamical systems has been formulated in [8], [9], or more recently in [10]. Approximate versions of the MLE problem have also been studied. These fall into the class of prediction error methods, and they have the advantage of being more computationally tractable compared to the exact MLE problem. Nevertheless, when the number of parameters to estimate grows, the resulting optimization problem becomes difficult to solve, limiting the actual use of methods based on MLE. To get through this limitation, typically two separate tasks are considered, first, the system parameters are identified, and secondly, the estimation of the process and measurement noise is carried out [10].

[1] Research and Development, Tool-Temp AG, Switzerland, leo.simpson@tool-temp.ch.

[2] Department of Microsystems Engineering (IMTEK), University of Freiburg, 79110 Freiburg, Germany {andrea.ghezzi, moritz.diehl}@imtek.uni-freiburg.de

[3] Department of Mathematics, University of Freiburg, 79104 Freiburg, Germany

This research was supported by the EU via ELO-X 953348.

Contributions: In this paper, we study the MLE problem for linear time-variant systems and provide the following contributions

- we introduce a framework in which the MLE formulation can be stated and used for the joint estimation of parameters in the deterministic part of the model and parameters in the covariance matrices of the process and measurement noise;
- we discuss and motivate with a counterexample why this method might provide generally a better parameter estimation than Trajectory Optimization (TO) methods, which are widely used in the context of Moving Horizon Estimation (MHE) [11], [12];
- we propose a tailored optimization algorithm to efficiently solve the optimization problem resulting from the MLE approach, and compare it with a state-of-the-art solver.

The combination of the MLE formulation with the proposed optimization algorithm constitutes a novel parameter estimation method for which performance, in terms of prediction accuracy, and efficiency, in terms of runtime, is ultimately proven on a realistic example of thermal control system.

Outline: In Section II we introduce the considered class of systems, the estimation task, and we provide relevant examples that fall into this class. Section III introduces the MLE method for parameter identification. In Section IV we compare the MLE method against TO, another common method for parameter estimation, providing a statistical result and a counterexample for TO. In Section IV-B, we present an optimization algorithm to solve the MLE problem. Section V presents numerical results of the proposed method for a realistic thermal control system.

Notation: We denote by S_n^{++}, the set of symmetric Positive Definite (PD) matrices of $\mathbb{R}^{n\times n}$. For $M \in S_n^{++}$ and $e \in \mathbb{R}^n$, we write $\|e\|_M^2 \coloneqq e^\top M e$ for $e \in \mathbb{R}^n$, and $|M|$ the determinant of M. For the unweighted L_2 norm, we omit the index: $\|e\|^2 \coloneqq e^\top e$. The Gaussian distribution with mean $\mu \in \mathbb{R}^n$ and covariance matrix $\Sigma \in S_n^{++}$ is $\mathcal{N}(\mu, \Sigma)$, and $f_{\text{gauss}}(\cdot, \mu, \Sigma)$ is its density function. The uniform probability distribution on the interval $[a, b]$ is denoted by $\mathcal{U}(a, b)$. The symbol I_n stands for the identity matrix. Throughout the paper, we use hat symbols for estimates, e.g., $\hat{y}_k$.

II. Problem Statement

In this work, we consider the class of parametric discrete-time and time-variant linear systems affected by state and output stochastic noise, defined by the following equations,

valid for $k = 0, \dots, N$

$$\begin{aligned} x_{k+1} &= A_k(\alpha)x_k + b_k(\alpha) + w_k, \\ y_k &= C_k(\alpha)x_k + v_k, \\ w_k &\sim \mathcal{N}\left(0, Q_k(\alpha)\right), \\ v_k &\sim \mathcal{N}\left(0, R_k(\alpha)\right), \end{aligned} \tag{1}$$

where $x_k \in \mathbb{R}^{n_x}$, $y_k \in \mathbb{R}^{n_y}$ are the states and the measurements while $\alpha \in \mathbb{R}^{n_\alpha}$ stacks the unknown parameters of the dynamical model and of the noise covariance model. The functions $A_k(\cdot), b_k(\cdot), C_k(\cdot), Q_k(\cdot)$ and $R_k(\cdot)$ are of appropriate dimensions and are assumed to be known. We assume that the random variables $w_0, \dots, w_{N-1}$ and $v_0, \dots, v_N$ are drawn independently. Additionally, we consider that the initial state comes from the following distribution

$$x_0 \sim \mathcal{N}\left(\hat{x}_0, P_0\right), \tag{2}$$

with $\hat{x}_0 \in \mathbb{R}^n$ and P_0 a fixed positive semi-definite matrix. Note that this assumption does not lead to any loss of generality, because choosing $A_0(\alpha)$ and $b_0(\alpha)$ is equivalent to choosing the Gaussian distribution of the state x_1.

The set of possible parameters α is denoted by $\mathcal{A}$ and is assumed to be with the following form

$$\mathcal{A} \coloneqq \{\alpha \in \mathbb{R}^{n_\alpha} \mid h(\alpha) \leq 0\}, \tag{3}$$

where the function $h : \mathbb{R}^{n_\alpha} \to \mathbb{R}^{n_h}$ is continuously differentiable. This function might express prior knowledge about the parameters. For instance, it can specify the ranges in which the parameters can take value. It is also necessary to ensure that for any $\alpha \in \mathcal{A}$, the matrices $Q_k(\alpha)$ and $R_k(\alpha)$ are PD.

We assume that measurements are available, i.e., we know the sequence $y_0, \dots, y_N$. We denote by $\mathcal{Y}_k$ the information set up to time k as $\mathcal{Y}_k \coloneqq (y_0, \dots, y_k)$. The task is to find the parameter α which makes measurements as likely as possible.

Remark 1. *The equations* (1) *notably model the case where the dynamical equations contain inputs u_k which have already been chosen and are assumed to be known. Even if the inputs u_k act in a nonlinear way, the estimation problem still falls into the general class described by equations* (1)

Remark 2. *One important application of this setting is the estimation of a disturbance model which can be used to achieve offset-free MPC [13]. When such models are used, the process noise w_k now contains two components with a different meaning, which need to be scaled [14]. Generally, this problem is difficult, and it also falls into the class of estimation problems described in this paper.*

III. Maximum Likelihood formulation

In the following, we formulate an optimization problem to estimate α from the data $\mathcal{Y}_N = (y_0, \dots, y_N)$. More precisely, we formulate the Maximum Likelihood Estimation (MLE) problem for identifying α given the probabilistic model (1). These formulations have been already derived in [9] to estimate model parameters or in [15] to estimate the matrices Q and R. Before diving into the MLE problem, we briefly recall the Kalman filter, a central tool for the formulation of the MLE problem.

A. The Kalman filter

For given parameters α and past measurements $\mathcal{Y}_{k-1}$, the Kalman filter (KF), introduced in [5], yields a Gaussian probability density of the state x_k given past measurements, which is defined by its mean and its covariance, usually referred to as $\hat{x}_{k|k-1}$ and $P_{k|k-1}$, but in this paper we will write them $\hat{x}_k$ and P_k. These are defined with the initial conditions $(\hat{x}_0, P_0)$ and the following recursive equations, valid for $k = 0, \dots, N$

$$\begin{aligned} S_k &= C_k P_k C_k^\top + R_k, \\ e_k &= y_k - C_k \hat{x}_k, \\ \hat{x}_{k+1} &= A_k\left(\hat{x}_k + P_k C_k^\top S_k^{-1} e_k\right) + b_k, \\ P_{k+1} &= A_k\left(P_k - P_k C_k^\top S_k^{-1} C_k P_k\right) A_k^\top + Q_k, \end{aligned} \tag{4}$$

where the dependency on α has been omitted for simplicity.

Specifically, the function that maps past data and parameters to the prediction of the next measurement and its covariance is given by

$$\begin{aligned} \hat{y}_k(\alpha, \mathcal{Y}_{k-1}) &\coloneqq C_k \hat{x}_k, \\ S_k(\alpha) &\coloneqq C_k P_k C_k^\top + R_k. \end{aligned} \tag{5}$$

Note that $S_k(\alpha) \in S_{n_y}^{++}$ for any $\alpha \in \mathcal{A}$. Finally, the probability density function of y_k given the probabilistic model (1) for some α, and the measurements $\mathcal{Y}_{k-1}$ is

$$p\left(y_k \mid \mathcal{Y}_{k-1}, \alpha\right) = f_{\text{gauss}}(y_k, \hat{y}_k(\alpha, \mathcal{Y}_{k-1}), S_k(\alpha)). \tag{6}$$

B. Maximum Likelihood problem

We define the Maximum Likelihood (ML) estimation problem as

$$\underset{\alpha \in \mathcal{A}}{\text{maximize}}\ p\left(\mathcal{Y}_N \mid \alpha\right), \tag{7}$$

where $p(\mathcal{Y}_N \mid \alpha)$ stands for the value of the probability density function of the measurements $y_0, \dots, y_N$ given the probabilistic model (1). In previous works, this problem has been derived explicitly [9], we recall this result in the following proposition.

Proposition 1. *The ML formulation* (7) *is equivalent to the following optimization problem*

$$\underset{\alpha \in \mathcal{A}}{\text{minimize}} \sum_{k=0}^{N} \|y_k - \hat{y}_k(\alpha, \mathcal{Y}_{k-1})\|^2_{S_k(\alpha)^{-1}} + \log\left|S_k(\alpha)\right|, \tag{8}$$

where $\hat{y}_k(\alpha, \mathcal{Y}_{k-1})$ and $S_k(\alpha)$ are defined in (5)*.*

Proof. Using basic probability rules, it is easy to derive the following formula

$$p(\mathcal{Y}_N \mid \alpha) = \prod_{k=0}^{N} p\left(y_k \mid \mathcal{Y}_{k-1}, \alpha\right), \tag{9}$$

where $p\left(y_k \mid \mathcal{Y}_{k-1}, \alpha\right)$ is defined in the previous section. Combining equations (9) and (6), the likelihood in (7) can be written explicitly

$$\begin{aligned} p\left(\mathcal{Y}_N \mid \alpha\right) &= \prod_{k=0}^{N} f_{\text{gauss}}(y_k, \hat{y}_k(\alpha, \mathcal{Y}_{k-1}), S_k(\alpha)), \\ &= \prod_{k=0}^{N} (|2\pi S_k(\alpha)|)^{-\frac{1}{2}} \, e^{-\frac{1}{2}\|y_k - \hat{y}_k(\alpha, \mathcal{Y}_{k-1})\|^2_{S_k(\alpha)^{-1}}} \end{aligned}$$

Finally, we apply the decreasing function $p \mapsto -2\log(p)$, then disregard the additive constant $n_y \log(2\pi)$, which leads to the desired form (8).

□

Remark 3. *This ML formulation can be under-determined depending on the choice of the uncertain parameters α. Indeed, some parameters may be impossible to estimate from the available data when the system is over parameterized, or when it is not excited enough. In this paper, we simply assume that the parameterization and the measured data are such that there is a unique parameter that maximizes the likelihood in* (7)*. In practice, expert knowledge about the system at hand usually allows one to formulate valid parameterization and design experiments to collect sufficiently information-rich data.*

It has been shown, under some additional assumption, that this MLE formulation provides an asymptotically unbiased estimate, and that it converges almost surely to the true parameters when the number of data points goes to infinity [4], [8]. Here, we simply state a statistical result that states that if the data is generated through the model (1), the true parameters minimize the expected value of the objective function in (8). This result can easily be proven by the fact that the objective function is the negative log-likelihood.

Proposition 2. *If $\alpha^\star \in \mathcal{A}$ is the true parameter and $\Psi(\cdot, \mathcal{Y}_N)$ is the objective function in* (8)*, then the following holds*

$$\alpha^\star \in \arg\min_{\alpha \in \mathcal{A}} \mathop{\mathbb{E}}_{\mathcal{Y}_N} \left[\Psi(\alpha, \mathcal{Y}_N)\right]. \tag{10}$$

IV. Comparison with Trajectory Optimization

In this section, we compare the presented formulation with another one, namely, Trajectory Optimization for parameter estimation.

A. Trajectory Optimization

The formulation stated so far falls into the class of *prediction error estimation methods* [16]. Another class of methods widely used for parameter estimation is Trajectory Optimization (TO) [11], [12]. These methods are typically used in Moving Horizon Estimation (MHE) settings for jointly estimating the state and the parameters of a model. In this section, we show that these methods are in general suboptimal compared to the one presented and they might fail to estimate some parameters even for an arbitrarily large number of data points N.

In TO methods, when the matrices Q_k and R_k are fixed, the parameters are found by solving the following problem

$$\begin{aligned} \underset{\alpha, x_0, \ldots, x_N}{\text{minimize}} \; & \sum_{k=0}^{N-1} \|x_{k+1} - A_k(\alpha)x_k - b_k(\alpha)\|^2_{Q_k^{-1}} \\ & + \sum_{k=0}^{N} \|C_k(\alpha)x_k - y_k\|^2_{R_k^{-1}} + \|x_0 - \hat{x}_0\|^2_{P_0^{-1}}. \end{aligned} \tag{11}$$

This formulation can also be stated in a likelihood formalism: if $\mathcal{X}_N \coloneqq (x_0, \ldots, x_N)$ stands for the trajectories, (11) is equivalent to solving the following problem

$$\underset{\mathcal{X}_N \in \mathbb{R}^{(N+1)n_x}, \alpha \in \mathcal{A}}{\text{maximize}} \; p\left(\mathcal{X}_N, \mathcal{Y}_N \mid \alpha\right) =: \Phi(\mathcal{X}_N, \alpha) \tag{12}$$

Indeed, the following holds

$$\begin{aligned} p\left(\mathcal{X}_N, \mathcal{Y}_N \mid \alpha\right) &= p\left(\mathcal{X}_N \mid \alpha\right) \cdot p\left(\mathcal{Y}_N \mid \alpha, \mathcal{X}_N\right), \\ &= \prod_{k=0}^{N-1} f_{\text{gauss}}\left(x_{k+1}, A_k(\alpha)x_k + b_k(\alpha), Q_k\right) \\ &\quad \times \prod_{k=0}^{N} f_{\text{gauss}}\left(y_k, C_k(\alpha)x_k, R_k\right), \end{aligned}$$

which is proportional to the exponential of half the negative objective in (11), when the covariance matrices Q_k and R_k are independent of α. In addition, using the law of total probability, the likelihood used in (7) can also be written as

$$p\left(\mathcal{Y}_N \mid \alpha\right) \propto \int_{\mathbb{R}^{(N+1)n_x}} \Phi(\mathcal{X}_N, \alpha) \mathrm{d}\mathcal{X}_N. \tag{13}$$

This formula shows a new perspective on TO for parameter estimation. Indeed, TO could be interpreted as an approximation to MLE which relies on

$$\operatorname*{argmax}_{\alpha \in \mathcal{A}} \int \Phi(\mathcal{X}_N, \alpha) \mathrm{d}\mathcal{X}_N \approx \operatorname*{argmax}_{\alpha \in \mathcal{A}} \left(\max_{\mathcal{X}_N} \Phi(\mathcal{X}_N, \alpha) \right).$$

In the next part, we highlight the superiority of the exact MLE over TO through an illustrative example.

B. An illustrative example

While the approximation above can sometimes give decent results, it fails, in general, to give an unbiased estimation of α as we see in the example below.

Example 1. *Let us consider the following probabilistic model, where only one parameter α needs to be estimated*

$$\begin{aligned} x_{k+1} &= x_k + w_k, && k = 0, \ldots, N-1, \\ y_k &= \alpha x_k + v_k, && k = 0, \ldots, N, \\ \begin{bmatrix} w_k \\ v_k \end{bmatrix} &\sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right), && k = 0, \ldots, N, \\ x_0 &= 0. \end{aligned} \tag{14}$$

The task is to estimate $\alpha \geq 0$ from measurements $y_0, \ldots, y_N$.

The TO formulation for the problem (14) reads

$$\begin{aligned} \underset{\alpha, x_1, \ldots, x_N}{\text{minimize}} \; & \sum_{k=0}^{N-1} \left(x_{k+1} - x_k\right)^2 + \sum_{k=0}^{N} \left(\alpha x_k - y_k\right)^2 \\ \text{subject to} \; & \alpha \geq 0. \end{aligned} \tag{15}$$

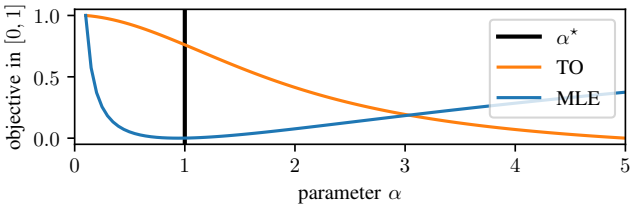

Fig. 1. Objective functions for problems (11) and (7) applied to the Example 1. For TO, the minimum over $\mathcal{X}_N$ for a given α is shown. The data $\mathcal{Y}_N$ is generated from the probabilistic model (14) with $\alpha^\star = 1$ and $N = 1000$. Each objective function is transformed affinely such that its values are between 0 and 1 on the interval $[0, 5]$.

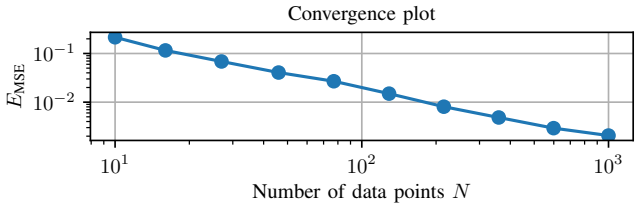

Fig. 2. Mean Squared Error over $m = 200$ samples of the estimates against the length of the measurement time series N, for Example 1.

For any number N, and any sequence $y_0, \dots, y_N$, the solution of problem (15) can only be $\alpha = +\infty$. Indeed, for $x_k = \varepsilon y_k$ and $\alpha = 1/\varepsilon$ with some $\varepsilon > 0$, the objective value of (15) is $\varepsilon^2 \sum_{k=0}^{N-1} (y_{k+1} - y_k)^2$ which is arbitrarily small when ε is close to zero. Hence, the TO method is incapable to estimate α in this example. Figure 1 illustrates the objective functions corresponding to the problem (11), (7) for the Example 1.

In contrast, we can prove and also show experimentally that the MLE formulation (7) provides an asymptotically unbiased estimate for α in this example. For this purpose, we generate measurement time series $\mathcal{Y}_{N,1}, \dots, \mathcal{Y}_{N,m}$ by simulating the system (14) with different parameters $\alpha_1^\star, \dots, \alpha_m^\star \in [0, 2]$. Then, we compute the corresponding estimates $\hat{\alpha}_i$ that solve the problems (7). Since only one parameter is sought, it is enough to use a simple line search to compute the corresponding estimate. To observe the asymptotic behavior of these estimates when N goes to infinity, let us compute the Mean Squared Error (MSE) $E_{\text{MSE}} := \frac{1}{m} \sum_{i=1}^{m} (\hat{\alpha}_i - \alpha_i^\star)^2$ and repeat the same experiment for many values of N. The profile of the MSE as a function of N is depicted in Figure 2. From this experiment, we can observe that the MLE formulation provides good estimates and, as expected, the performance increases with the number of measurements N.

Tailored optimization algorithm

Due to the nonlinearity of the functions $\hat{y}_k(\alpha, \mathcal{Y}_{k-1})$ and $S_k(\alpha)$, the optimization problem (8) is a nonconvex and Nonlinear Programming problem (NLP). Hence, solving this problem to global optimality is very hard. In fact, the computational difficulty of this optimization problem, even to local optimality, has been the main obstacle to the use of MLE methods to estimate parameters in the noise covariances of linear systems. In the following, we discuss two NLP algorithms for solving efficiently the MLE problem (8). Even though such algorithms converge to a local minimum that is not necessarily the global minimum, we assume that this already provides a correct estimate. In the first part, we present how to use a sparse interior point solver for this problem, and in the second we present a hand-tailored Sequential Quadratic Programming (SQP) specific to the optimization problem concerned.

The efficiency of these algorithms will be assessed in Section V on a realistic numerical example.

Optimization using a sparse interior-point solver

We formulate the MLE optimization problem using CasADi [17] via its Python interface and solve the corresponding NLP using IPOPT [18] with the shipped sparse linear solver MUMPS. We promote sparsity in the optimization problem by adopting a multiple shooting formulation. Therefore, we lift the variables involved in the Kalman filter propagation and we impose equations (4) as constraints. The optimization problem (7) takes the following form

$$
\begin{aligned}
&\underset{\alpha, \boldsymbol{e}, \boldsymbol{S}, \hat{\boldsymbol{x}}, \boldsymbol{P}}{\text{minimize}} \ \sum_{k=0}^{N} (e_k)^\top (S_k)^{-1} e_k + \log |S_k| \\
&\text{subject to} \\
&S_k = C_k P_k C_k^\top + R_k(\alpha), \qquad \text{for } k = 0, \dots, N, \\
&e_k = y_k - C_k \hat{x}_k, \\
&\hat{x}_{k+1} = A_k(\alpha) \left(\hat{x}_k + P_k C_k^\top S_k^{-1} e_k \right) + b_k(\alpha), \\
&P_{k+1} = A_k(\alpha) \left(P_k - P_k C_k^\top S^{-1} C_k P_k \right) A_k(\alpha)^\top + Q_k(\alpha), \\
&h(\alpha) \leq 0,
\end{aligned}
\tag{16}
$$

C. A tailored Sequential Quadratic Programming algorithm

Before describing the tailored algorithm, we need to reformulate the optimization problem (8). Thus, we define the functions $e_k(\alpha)$ and $S_k(\alpha)$ that map the parameters α to the solution e_k and S_k of the recursive equations of the Kalman filter defined in (4). Secondly, we define a function $\varphi : \mathbb{R}^{n_y} \times S_{n_y}^{++} \times \mathbb{R} \to \mathbb{R}$, with $\varphi(e, S, \gamma) := e^\top S^{-1} e + \gamma$. With these definitions, problem (8) can be reformulated as follows

$$
\begin{aligned}
&\underset{\alpha \in \mathbb{R}^{n_\alpha}}{\text{minimize}} \ \sum_{k=0}^{N} \varphi\big(e_k(\alpha), S_k(\alpha), \log |S_k(\alpha)| \big) \\
&\text{subject to} \quad h(\alpha) \leq 0.
\end{aligned}
\tag{17}
$$

An important point is that the function $\varphi(\cdot, \cdot, \cdot)$ is convex [19], hence the objective function has a "convex-over-nonlinear" structure, which allows the use of an optimization technique called the Generalized Gauss-Newton (GGN) method [20], [21]. Finally, for compactness and consistency with the notation adopted in [21], we can rewrite the optimization problem (17) as

$$
\begin{aligned}
&\underset{\alpha \in \mathbb{R}^{n_\alpha}}{\text{minimize}} \ \sum_{k=0}^{N} \phi\big(F_k(\alpha)\big) \\
&\text{subject to} \quad h(\alpha) \leq 0,
\end{aligned}
\tag{18}
$$

where F_k are nonlinear functions given by stacking the components of the functions $e_k(\alpha)$, $S_k(\alpha)$, and $\log|S_k(\alpha)|$, and ϕ is the vector-input version of φ.

The GGN method that we develop consists in sequentially solving a Quadratic Program (QP) obtained by the quadratic approximation of (18) around the current solution point $\bar{\alpha}$. Specifically, we linearize the inequality constraints and the functions $F_k(\alpha)$, while we replace $\phi(\cdot)$ by its quadratic approximation $\phi_{\text{quad}}(\cdot)$ defined as follows

$$\phi_{\text{quad}}(\Delta F;\bar{F}) \coloneqq \phi(\bar{F}) + \frac{\mathrm{d}\phi}{\mathrm{d}F}\Delta F + \frac{1}{2}(\Delta F)^\top \frac{\mathrm{d}^2\phi}{\mathrm{d}F^2}\Delta F,$$

which is ensured to be convex. As a result, the QP to solve at each iteration reads

$$\begin{aligned} \underset{\Delta\alpha\in\mathbb{R}^{n_\alpha}}{\text{minimize}} \quad & \sum_{k=0}^{N} \phi_{\text{quad}}\left(\frac{\mathrm{d}F_k}{\mathrm{d}\alpha}(\bar{\alpha})\Delta\alpha; F_k(\bar{\alpha})\right) \\ \text{subject to} \quad & h(\bar{\alpha}) + \frac{\mathrm{d}h}{\mathrm{d}\alpha}(\bar{\alpha})\Delta\alpha \le 0. \end{aligned} \tag{19}$$

Finally, to ensure convergence, the optimization variable α is ultimately updated in the direction found by the QP, using a globalization technique based on back-tracking line-search until the Armijo condition is satisfied [22].

The linearization of the functions $F_k(\cdot)$ is done by propagating the values and derivatives of S_k, e_k, $\hat{x}_{k|k-1}$ and $P_{k|k-1}$ in equations (4). We also use the mathematical formula $\frac{\mathrm{d}\log|S|}{\mathrm{d}S} = S^{-1}$ for the linearization of $\log|S_k(\alpha)|$. Note that the hand-tailored implementation of these derivatives improves efficiency. For instance, the inverse matrices S_k^{-1} are computed only once, while they are used multiple times: in the first equation of (4) or in the derivative of $\log|S_k|$.

Finally, regarding the stopping criterion, the algorithm stops when the cost decreases less than a given relative tolerance, which we set to 10^{-5}. The presented algorithm is implemented in Python using standard libraries for linear algebra, and CVXOPT [23] for solving the QP.

Remark 4. *The significant steps are the propagation of the derivatives of P_k and the solution of the QP. Hence, the complexity of an SQP step is $\mathcal{O}(Nn_x^3n_\alpha + n_\alpha^3)$.*

Remark 5. *Even though the method scales linearly in the horizon length N, for online parameter estimation, where the optimization needs to be performed quickly and the quantity of past data is growing, moving horizons might be considered, as proposed in [10].*

V. Numerical example

In this section, we apply the presented method to a realistic estimation task. We use this task to investigate the performance of the presented method when the dimensions of the system scale up. It is also used to assess the efficiency of the optimization algorithms discussed in Section IV-B.

The present estimation task is inspired by an industrial problem of controlling the temperature of a fluid through mass transport inside a straight pipe. The control inputs are the temperature of the inlet and the position of a valve located in the inlet, which can modify the fluid velocity.

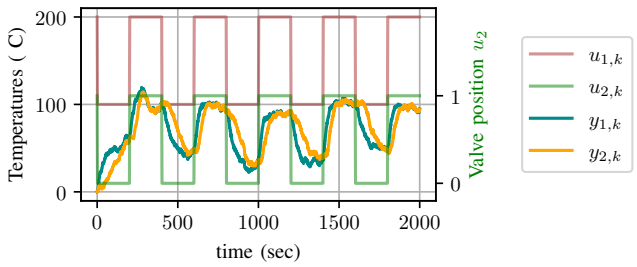

Fig. 3. Example of input and output data generated through the described process, for the parameters $\alpha^\star = (\frac{1}{2},\dots,\frac{1}{2}) \in [0,1]^7$.

The system is also subjected to unknown disturbances and heat losses. The output measurements are obtained by two thermometers placed at two different locations of the pipe. In the context of controlling this system via MPC with a linear state estimator, an accurate knowledge of its parameters is required. Thus, our task is to estimate model parameters, such as the heat losses, or the heat transfer coefficients that depend on the valve position. The precision of each sensor is also a parameter to estimate, jointly with the process noise and the disturbance fluctuations. For this thermal system we propose a linear model given by the following equations, for $k = 0,\dots,N$

$$\begin{aligned} x_{1,k+1} &= (1-a_k)x_{1,k} + a_k\alpha_1 u_{1,k} + w^x_{1,k}, \\ x_{i,k+1} &= (1-a_k)x_{i,k} + a_k x_{i-1,k} + w^x_{i,k}, \quad i = 2,\dots,5, \\ d_{k+1} &= d_k + w^d_k, \\ y_{1,k} &= x_{2,k+1} + d_k + v_{1,k}, \\ y_{2,k} &= x_{5,k+1} + d_k + v_{2,k}, \\ w^x_k &\sim \mathcal{N}(0, \mathrm{diag}(\alpha_4,\varepsilon,\varepsilon,\varepsilon,\varepsilon)), \\ w^d_k &\sim \mathcal{N}(0,\alpha_5), \\ v_k &\sim \mathcal{N}(0, \mathrm{diag}(\alpha_6,\alpha_7)), \end{aligned} \tag{20}$$

where $a_k = \frac{1}{10}(\alpha_2 + \alpha_3 u_{2,k})$ and $\varepsilon = 10^{-6}$. The state $x \in \mathbb{R}^5$ models the temperature of the fluid at different locations along the pipe. The state has been augmented by $d \in \mathbb{R}$ to account for disturbances (cf. Remark 2). The control is given by $u \in \mathbb{R}^2$, where u_1 denotes the inlet temperature and u_2 the valve position. Note that the control acts both linearly and non-linearly on the system, which makes the present system time-variant. The measured temperatures at locations 2 and 5 corresponds to the output $y \in \mathbb{R}^2$. The system has parameters in both the dynamics and the noise covariances, the parameter vector is $\alpha \in \mathcal{A} = [0,1]^7$. The parameter α_1 models heat losses, while α_2 and α_3 model the heat transfer dues to mass transport for the two positions of the valve. The task is to estimate the whole parameter vector α.

We first collect measurements by simulating the system using equations (20), where the inputs alternate every 200 time-steps between two values $u^{\text{low}} = (100, 0)$ and $u^{\text{high}} = (200, 1)$. Figure 3 shows a time series generated via this model. Each parameter is sampled from a uniform probability distribution $\mathcal{U}(0,1)$. We apply the presented MLE method with different sizes of measurement data, from $N = 1000$ to $N = 3000$. The two optimization algorithms described

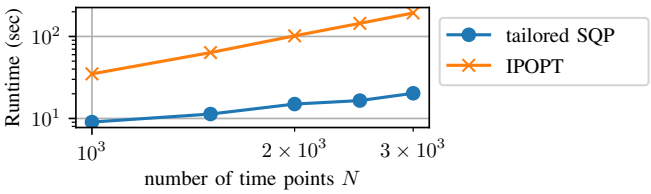

Fig. 4. Comparison of runtime for the algorithm using the tailored SQP and the one using IPOPT when the number of data points grows.

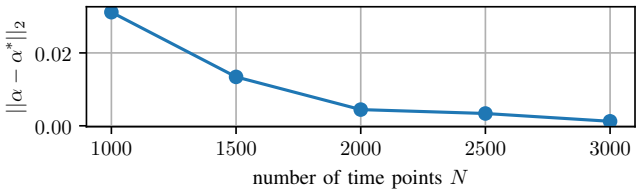

Fig. 5. Difference between the estimated parameters α, and true parameters $\alpha^\star$ when the amount of data grows.

in Section IV-B are used separately. The first observation is that both algorithms, i.e., the one based on the solver IPOPT and the one based on the tailored SQP method, converge to the same point, with a maximum difference between the two solutions smaller than 10^{-3}. This is encouraging because it seems to imply that both algorithms converged to the optimum of the MLE optimization problem.

Runtimes of the two algorithm are compared in Figure 4. This figure confirms that the algorithm complexity scales linearly in the number of data point N, as it was mentioned in Remark 4. Moreover, it shows that the developed SQP method has a runtime 5 times smaller than IPOPT. Even though, our implementation is done in Python using standard libraries while IPOPT runs compiled C code. Hence, we expect that by implementing the proposed SQP method in a compiled language we could reduce its runtime dramatically. As a reference, for the investigated problem with nontrivial dimensions and with a rather difficult estimation task, the algorithm takes about 20 seconds for $N = 3000$ data points.

Regarding the estimation performance, in Figure 5, we compare the sum of squares of the differences between the estimated parameters and the true parameters. The plot shows that both model parameters and noise variances are correctly estimated, and in case of enough data points, the true parameters are recovered.

VI. Conclusion and Outlook

This paper offers a study about parameter estimation for linear dynamical systems in the maximum likelihood framework. We have shown, from a theoretical and a numerical perspective, that through this framework it is possible to jointly estimate parameters in the system dynamics and the noise covariances. Specifically, we presented a tailored optimization algorithm that extends the application of the maximum likelihood framework to systems with realistic dimensions. A fast open-source implementation of the algorithm is left for future research, as well as the case of online estimation.

References

[1] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Nob Hill, 2017.

[2] M. Verhaegen, "Identification of the deterministic part of mimo state space models given in innovations form from input-output data," *Automatica*, vol. 30, no. 1, pp. 61–74, 1994.

[3] P. Van Overschee and B. De Moor, "N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems," *Automatica*, vol. 30, no. 1, pp. 75–93, 1994.

[4] L. Ljung, *System identification: Theory for the User*. Upper Saddle River, N.J.: Prentice Hall, 1999.

[5] R. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[6] K. Myers and B. Tapley, "Adaptive sequential estimation with unknown noise statistics," *IEEE Transactions on Automatic Control*, vol. 21, no. 4, pp. 520–523, 1976.

[7] R. Mehra, "On the identification of variances and adaptive Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 15, no. 2, pp. 175–184, 1970.

[8] R. Kashyap, "Maximum likelihood identification of stochastic linear systems," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 25–34, 1970.

[9] K. Astrom, "Maximum likelihood and prediction error methods," *IFAC Proceedings Volumes*, vol. 12, no. 8, pp. 551–574, 1979.

[10] J. Valluru, P. Lakhmani, S. C. Patwardhan, and L. T. Biegler, "Development of moving window state and parameter estimators under maximum likelihood and Bayesian frameworks," *Journal of Process Control*, vol. 60, pp. 48–67, 2017.

[11] H. Bock, E. Kostina, and J. Schlöder, "Numerical methods for parameter estimation in nonlinear differential algebraic equations," *GAMM Mitteilungen*, vol. 30/2, pp. 376–408, 2007.

[12] P. Kühl, M. Diehl, T. Kraus, J. P. Schlöder, and H. G. Bock, "A real-time algorithm for moving horizon state and parameter estimation," *Computers and Chemical Engineering*, vol. 35, no. 1, pp. 71–83, 2011.

[13] G. Pannocchia and J. Rawlings, "Disturbance Models for Offset-Free Model-Predictive Control," *AIChE Journal*, vol. 49, pp. 426–437, 2003.

[14] S. J. Kuntz and J. B. Rawlings, "Maximum likelihood estimation of linear disturbance models for offset-free model predictive control," *American Control Conference*, pp. 3961–3966, 2022.

[15] P. Abbeel, A. Coates, M. Montemerlo, A. Y. Ng, and S. Thrun, "Discriminative training of Kalman filters." in *Robotics: Science and systems*, vol. 2, 2005, p. 1.

[16] L. Ljung, "Prediction error estimation methods," *Circuits, Systems and Signal Processing*, vol. 21, no. 1, pp. 11–21, 2002.

[17] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.

[18] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[19] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge: University Press, 2004.

[20] N. N. Schraudolph, "Fast curvature matrix-vector products for second-order gradient descent," *Neural Computation*, vol. 14, no. 7, pp. 1723–1738, 2002.

[21] F. Messerer, K. Baumgärtner, and M. Diehl, "Survey of sequential convex programming and generalized Gauss-Newton methods," *ESAIM: Proceedings and Surveys*, vol. 71, pp. 64–88, 2021.

[22] J. Nocedal and S. Wright, *Numerical Optimization*. Springer-Verlag, 2000.

[23] M. S. Andersen, J. Dahl, and L. Vandenberghe, "Cvxopt: A python package for convex optimization, version 1.1.6," Available at cvxopt.org, 2013.