Gradient-Based Recursive Maximum Likelihood Identification of Jump Markov Non-Linear Systems

Andre R. Braga, Carsten Fritsche, Fredrik Gustafsson and Marcelo G. S. Bruno

The self-archived postprint version of this journal article is available at Linköping University Institutional Repository (DiVA):

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-142451>

N.B.: When citing this work, cite the original publication.

Braga, A. R., Fritsche, C., Gustafsson, F., Bruno, M. G. S., (2017), Gradient-Based Recursive Maximum Likelihood Identification of Jump Markov Non-Linear Systems, *2017 20TH INTERNATIONAL CONFERENCE ON INFORMATION FUSION (FUSION)*, , 228-234.
<https://doi.org/10.23919/ICIF.2017.8009651>

Original publication available at:

<https://doi.org/10.23919/ICIF.2017.8009651>

Copyright: IEEE

<http://www.ieee.org/>

©2017 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Gradient-Based Recursive Maximum Likelihood Identification of Jump Markov Non-Linear Systems

André R. Braga

Campus Quixadá

Federal University of Ceará
63902-580 Quixadá CE, Brazil

Carsten Fritsche, Fredrik Gustafsson

Department of Electrical Engineering

Linköping University
SE-581 83 Linköping, Sweden

Marcelo G.S. Bruno

Division of Electronics Engineering

Aeronautics Institute of Technology
12228-900 São José dos Campos SP, Brazil

Abstract—This paper deals with state inference and parameter identification in Jump Markov Non-Linear System. The state inference problem is solved efficiently using a recently proposed Rao-Blackwellized Particle Filter, where the discrete state is integrated out analytically. Within the RBPF framework, Recursive Maximum Likelihood parameter identification is performed using gradient ascent algorithms. The proposed learning method has the advantage over (online) Expectation Maximization methods, that it can be easily applied to cases where the probability density functions defining the Jump Markov Non-Linear System are not members of the exponential family. Two benchmark problems illustrate the parameter identification performance.

I. INTRODUCTION

In recent years, there has been an increased research interest in estimation methods for Jump Markov Non-Linear System. It can be seen as an extension of the class of nonlinear state-space models, where the unknown continuous state may evolve in time according to different state-space formulations (or different parametrization thereof). The switching between different state-space formulations is modeled by a discrete state (or mode) variable, evolving according to a Markov chain, which is latent and has to be estimated together with the continuous state. The popularity of such systems stems from the fact that many real world applications can be described using Jump Markov Non-Linear System (JMNLs) as, for instance, the modeling of economic systems [1], [2], failures in subsystems or components [3], electromagnetic disturbance effects on flight controllers [4], random changes in power system structures [5], or mobile radio propagation channel switching in localization applications [6].

It is well known, that the optimal state estimation problem in systems with Markovian switching structure is NP-hard, as the number of possible discrete state combinations increases exponentially with time [7]. Therefore, many research efforts have been devoted to finding suitable approximations to solve the inference problem, see for instance [8]–[14].

The estimation problem becomes even more challenging, when parameters such as the transition probability matrix modeling the discrete state, or other model parameters such as the process and/or measurement noise (co-)variance are additionally assumed unknown. Parameter estimation within the JMNLs framework can be broadly categorized into Bayesian and non-Bayesian approaches. Both categories can be further subdivided into offline approaches, where a complete history (or

batch) of measurements is processed at every time instance for parameter estimation, and online approaches where only the most recent measurement is processed. In the following, we focus our review on non-Bayesian (i.e. Maximum Likelihood (ML) parameter estimation) approaches.

In the non-Bayesian framework, the author of [15] proposed to estimate the Transition Probability Matrix (TPM) of a jump Markov linear Gaussian system in an offline fashion using the so-called Truncated Maximum Likelihood (TML). Improved estimators for this problem have appeared in [16]. The offline estimation of the TPM and model parameters in JMNLs has appeared in [17] where the Expectation Maximization (EM) algorithm [18], [19] has been utilized within a Particle Filter (PF) framework. More recently, online EM algorithms for transition probability matrix and model parameters estimation have appeared in [14], [20].

This paper proposes to use a gradient-ascent approach to solve the parameter identification problem in JMNLs. It builds up on previous work [14], where the online EM approach embedded within the Rao-Blackwellized Particle Filter (RBPF) framework is replaced by an Recursive Maximum Likelihood (RML) approach. Following [21], there are certain advantages and disadvantages within both methods that further motivate the use of the RML approach.

The EM algorithm is usually simple to implement, provided that the Maximization (M)-Step is available in closed-form, it often implicitly deals with parameter constraints and it is parametrization independent. The RML on the other hand, does not require the M-Step. Hence, it can be also applied in situations where simple expression for the M-Step are not available. Parameter constraints such as for instance the transition probabilities of a Markov chain, have to be dealt with explicitly in RML methods either through reparameterization or by using constrained optimization methods. A reparameterization, however, will change the gradient, and hence the convergence behavior of the RML algorithm. Thus, extra care has to be taken in the algorithm design. Finally, simple expressions for the EM algorithm are obtained when the probability density functions (pdfs) describing the JMNLs are members of the exponential family. If this is not the case, the RML might be an interesting alternative.

The remainder of this paper is organized as follows. In Section II, the system model and notations are presented.

In Section III, the offline and online implementations of the gradient ascent algorithm are discussed. In Section IV, it is shown how the score vector can be computed in an online fashion, followed by Section V, where an Sequential Monte Carlo (SMC) approximation for the score vector computation is proposed. The performance of the proposed solution is investigated on benchmark problems in Section VI. Finally, Section VII concludes the work and provides future research directions.

II. SYSTEM MODEL

Consider the following discrete-time JMNLs

$$r_t \sim \Pi(r_t|r_{t-1}), \quad (1a)$$

$$x_t \sim f_{r_t}(x_t|x_{t-1};\theta_{r_t}), \quad (1b)$$

$$y_t \sim g_{r_t}(y_t|x_t;\theta_{r_t}), \quad (1c)$$

where $x_t \in \mathbb{R}^{n_x}$ is a continuous state variable and r_t is a discrete mode variable $r_t \in \{1, \dots, K\}$, which is assumed to be among K possible modes. The state and mode variables are latent, but observed indirectly through the measurements $y_t \in \mathbb{R}^{n_y}$. The mode switching is assumed to be a finite state Markov process (Markov chain) with transition probabilities

$$\pi_{k\ell} = \Pi(\ell|k) = \mathbb{P}(r_t = \ell|r_{t-1} = k). \quad (2)$$

Such systems are called hybrid, where switching between a finite number of different non-linear dynamical models is possible. For notational brevity, we let Π refer in the following to both the transition kernel for r_t and the $K \times K$ TPM with entries $[\Pi]_{k\ell} = \pi_{k\ell}$. Further, the state transition density and the measurement likelihood for each mode k is given by $f_k(x_t|x_{t-1};\theta_k)$ and $g_k(y_t|x_t;\theta_k)$, where each mode k is parameterized by its own set of parameters θ_k . The unknown model parameters are thus given by

$$\theta = (\{\theta_k\}_{k=1}^K, \Pi). \quad (3)$$

In order to simplify the notation, the initial state of the system (x_0, r_0) is assumed fixed and known. The extension to unknown initial system states is straightforward. The task is now to jointly estimate the system states, i.e. $(x_{1:n}, r_{1:n})$, together with the unknown model parameters θ in an online fashion.

III. GRADIENT ASCENT ALGORITHM

In this work, we are concerned with estimating the unknown model parameters θ in JMNLs using an ML approach. In ML estimation, the estimate $\hat{\theta}^{\text{ML}}$ is the maximizing argument of the log-likelihood function of the observed data, or equivalently

$$\hat{\theta}^{\text{ML}} = \arg \max_{\theta \in \Theta} \log p(y_{1:n}; \theta), \quad (4)$$

where $y_{1:n}$ is the sequence of n observations and Θ is the feasible set of the parameters. For JMNLs, direct optimization of (4) is generally not possible due to the intractability of computing the likelihood $p(y_{1:n}; \theta)$. In the literature, there are two prevalent approaches that address this difficulty. One approach is based on the EM algorithm [18] which can be

implemented either offline for a fixed batch of data $y_{1:n}$, or online where only the most recent measurement y_n is processed at each time step [22]. The other approach, which is further detailed in this paper, calculates the ML estimate by using gradient ascent algorithms. Gradient ascent-based ML estimation can also be implemented offline or online, and the latter is sometimes called RML [23].

A. Offline Implementation

Let $\{\theta_m\}_{m \in N}$ denote the sequence of parameters obtained from the (iterative) gradient algorithm. Then the parameter at iteration $m+1$ can be updated according to

$$\theta_{m+1} = \theta_m + \gamma_{m+1} \nabla_{\theta} \log p(y_{1:n}; \theta)|_{\theta=\theta_m}, \quad (5)$$

where $\nabla_{\theta} \log p(y_{1:n}; \theta)$ is the score vector evaluated at $\theta = \theta_m$, and $\{\gamma_m\}_{m \geq 1}$ is a non-negative scalar that needs to be adjusted at each iteration to ensure, a minima, that the computed likelihood sequence is non-decreasing [21].

For JMNLs, the score vector $\nabla_{\theta} \log p(y_{1:n}; \theta)$ can be obtained by using Fisher's identity [22]

$$\nabla_{\theta} \log p(y_{1:n}; \theta) = \mathbb{E}_{\theta} \{ \nabla_{\theta} \log p(x_{1:n}, r_{1:n}, y_{1:n}; \theta) | y_{1:n} \}, \quad (6)$$

where the expectation is taken w.r.t. the conditional density $p(x_{1:n}, r_{1:n} | y_{1:n}; \theta)$. For JMNLs, the joint density can be factorized as follows (recall that (x_0, r_0) are assumed known)

$$p(x_{1:n}, r_{1:n}, y_{1:n}; \theta) = \prod_{t=1}^n p(x_t, r_t, y_t | x_{t-1}, r_{t-1}; \theta). \quad (7)$$

By plugging (7) into (6) and by making use of the indicator function $\mathbb{1}(\cdot)$, the score can be written as

$$\begin{aligned} \nabla_{\theta} \log p(y_{1:n}; \theta) &= \sum_{t=1}^n \mathbb{E}_{\theta} \{ \nabla_{\theta} \log \Pi(r_t | r_{t-1}) | y_{1:n} \} \\ &\quad + \sum_{t=1}^n \mathbb{E}_{\theta} \{ \nabla_{\theta} \log [f_{r_t}(x_t | x_{t-1}; \theta_{r_t}) g_{r_t}(y_t | x_t; \theta_{r_t})] | y_{1:n} \} \\ &= \sum_{k=1}^K \sum_{\ell=1}^K \sum_{t=1}^n \mathbb{E}_{\theta} \{ \mathbb{1}(r_t = \ell, r_{t-1} = k) \nabla_{\theta} \log \pi_{k\ell} | y_{1:n} \} \\ &\quad + \sum_{k=1}^K \sum_{t=1}^n \mathbb{E}_{\theta} \{ \mathbb{1}(r_t = k) \nabla_{\theta} \log f_k(x_t | x_{t-1}; \theta_k) | y_{1:n} \} \\ &\quad + \sum_{k=1}^K \sum_{t=1}^n \mathbb{E}_{\theta} \{ \mathbb{1}(r_t = k) \nabla_{\theta} \log g_k(y_t | x_t; \theta_k) | y_{1:n} \}. \end{aligned} \quad (8)$$

We further introduce the set of additive functionals

$$\mathcal{S}_{k\ell,n}^{(1)} = \sum_{t=1}^n \mathbb{E}_{\theta} \{ \mathbb{1}(r_t = \ell, r_{t-1} = k) s_{k\ell,t}^{(1)} | y_{1:n} \}, \quad (9a)$$

$$\mathcal{S}_{k,n}^{(2)} = \sum_{t=1}^n \mathbb{E}_{\theta} \{ \mathbb{1}(r_t = k) s_{k,t}^{(2)}(y_t, x_t, x_{t-1}) | y_{1:n} \}. \quad (9b)$$

where

$$s_{k\ell,t}^{(1)} = \nabla_{\theta} \log \pi_{k\ell}, \quad (10a)$$

$$\begin{aligned} s_{k,t}^{(2)}(y_t, x_t, x_{t-1}) &= \nabla_{\theta} \log f_k(x_t | x_{t-1}; \theta_k) \\ &\quad + \nabla_{\theta} \log g_k(y_t | x_t; \theta_k). \end{aligned} \quad (10b)$$

Then, the score can then be finally written as

$$\nabla_{\theta} \log p(y_{1:n}; \theta) = \sum_{k=1}^K \sum_{\ell=1}^K \mathcal{S}_{k\ell,n}^{(1)} + \sum_{k=1}^K \mathcal{S}_{k,n}^{(2)}. \quad (11)$$

B. Online Implementation

For the recursive version, the time index n replaces the iteration index m and the gradient of the predictive likelihood $p(y_n|y_{1:n-1}; \theta)$ is evaluated instead of $p(y_{1:n}; \theta)$ [22]. The parameter estimate at time $n+1$ is thus updated as

$$\theta_{n+1} = \theta_n + \gamma_{n+1} \nabla_{\theta} \log p(y_n|y_{1:n-1}; \theta_{0:n}). \quad (12)$$

Here, the notation $p(y_n|y_{1:n-1}; \theta_{0:n})$ shall indicate the necessary requirement that, for an online implementation, *all* parameter estimates $\theta_{0:n}$ (and not only θ_n) are used in the evaluation of $\nabla_{\theta} \log p(y_n|y_{1:n-1}; \theta)$, because otherwise, one would need to go through the entire history of observations. By making use of Bayes' rule, the predictive likelihood can be expressed as

$$\begin{aligned} \nabla_{\theta} \log p(y_n|y_{1:n-1}; \theta_{0:n}) \\ = \nabla_{\theta} \log p(y_{1:n}; \theta_{0:n}) - \nabla_{\theta} \log p(y_{1:n-1}; \theta_{0:n-1}). \end{aligned} \quad (13)$$

This result states that, for the recursive implementation, the difference between the score vectors of the current and previous time step has to be formed. However, in order to arrive at a true online implementation of the algorithm, the score $\nabla_{\theta} \log p(y_{1:n}; \theta)$ needs to be evaluated in an online fashion.

C. Score Computation for Transition Matrix Elements

In order to compute the gradient w.r.t. the elements of the TPM, we must ensure that the TPM remains stochastic, i.e. the elements remain positive and are properly normalized. This can be achieved by a reparameterization, i.e. we perform the gradient ascent in another set of unconstrained variables $v_{k\ell}$ using the *soft-max* function [24], yielding

$$\pi_{k\ell} = \frac{\exp \{v_{k\ell}\}}{\sum_{\ell'} \exp \{v_{k\ell'}\}}. \quad (14)$$

We then perform gradient ascent according to

$$v_{k\ell}^{n+1} = v_{k\ell}^n + \gamma_{n+1} \frac{\partial}{\partial v_{k\ell}} \log p(y_n|y_{1:n-1}; \theta_{0:n}) \quad (15)$$

and the equivalent update of $\pi_{k\ell}$ will be

$$\pi_{k\ell}^{n+1} = \frac{\pi_{k\ell}^n \exp \left\{ \gamma_{n+1} \frac{\partial}{\partial v_{k\ell}} \log p(y_n|y_{1:n-1}; \theta_{0:n}) \right\}}{\sum_{\ell'} \pi_{k\ell'}^n \exp \left\{ \gamma_{n+1} \frac{\partial}{\partial v_{k\ell'}} \log p(y_n|y_{1:n-1}; \theta_{0:n}) \right\}}. \quad (16)$$

IV. ONLINE COMPUTATION OF THE SCORE

The score $\nabla_{\theta} \log p(y_{1:n}; \theta)$ requires the computation of expectations w.r.t. the joint density $p(x_{1:n}, r_{1:n}|y_{1:n}; \theta)$, see (9), which is also referred as an instance of the problem of smoothed additive functionals estimation [25]. These smoothed estimates can be obtained using standard two-filter or forward/backward smoothers, see for instance [26]. However, the use of those filters will result in an offline

implementation. For smoothed additive functionals, an online computation is possible by using forward-only smoothing techniques [27] as follows:

The structure of (9) allows us to construct a sequence of additive functionals according to

$$S_n(z_{0:n}) = \sum_{t=1}^n s_t(z_{t-1}, z_t), \quad (17)$$

where we have introduced the joint state variable $z_t = (x_t, r_t)$ and

$$s_t(z_{t-1}, z_t) = \begin{pmatrix} s_t^{(1)}(z_{t-1}, z_t) \\ s_t^{(2)}(z_{t-1}, z_t) \end{pmatrix}. \quad (18)$$

The elements in the vector are further defined as

$$s_t^{(1)}(z_{t-1}, z_t) = [\mathbb{1}(r_t = \ell, r_{t-1} = k) s_{k\ell,t}^{(1)}]_{\ell=1}^K, \quad (19a)$$

$$s_t^{(2)}(z_{t-1}, z_t) = [\mathbb{1}(r_t = k) s_{k,t}^{(2)}(x_t, x_{t-1})]_{k=1}^K, \quad (19b)$$

where the dependence on y_t in the notation is omitted, and where $[v_j]_{j=1}^n$ stacks the vectors v_j into a single vector according to $[v_j]_{j=1}^n \triangleq (v_1^T, \dots, v_n^T)^T$. Consequently, (9) can be compactly expressed as

$$\mathcal{S}_n = \mathbb{E}_{\theta} \{S_n(z_{0:n})|y_{1:n}\}. \quad (20)$$

The idea is now to introduce an auxiliary function

$$T_t(z_t) \triangleq \mathbb{E}_{\theta} \{S_t(z_{0:t})|z_t, y_{1:t}\}, \quad (21)$$

so that the smoothed additive functional (9) can be written as a filtered estimate of $T_t(z_t)$ according to

$$\mathcal{S}_t = \mathbb{E}_{\theta} \{T_t(z_t)|y_{1:t}\} = \int T_t(z_t) p(z_t|y_{1:t}) dz_t. \quad (22)$$

Due to the additive structure of (17) it is furthermore possible to compute $T_t(z_t)$ recursively

$$\begin{aligned} T_t(z_t) = \int [T_{t-1}(z_{t-1}) + s_t(z_{t-1}, z_t)] \\ \times p(z_{t-1}|z_t, y_{1:t-1}; \theta) dz_{t-1}, \end{aligned} \quad (23)$$

where the recursion is initialized with $T_0(z_0) = 0$.

V. SMC IMPLEMENTATION

The exact computation of the expectations is, in general, not possible due to the intractability of the integrals in (22) and (23). We propose to approximate these using an SMC technique [25], which is briefly discussed in the following. The state inference problem is solved using an RBPF [14], which makes use of the following decomposition of the joint posterior density

$$p(x_{1:t}, r_t|y_{1:t}; \theta) = p(r_t|x_{1:t}, y_{1:t}; \theta) p(x_{1:t}|y_{1:t}; \theta). \quad (24)$$

Here, the density $p(r_t|x_{1:t}, y_{1:t}; \theta)$ is computed analytically using conditional Hidden Markov Model (HMM) filters, while the density $p(x_{1:t}|y_{1:t}; \theta)$ is approximated numerically using SMC techniques. The RBPF produces as output the set $\{x_{1:t}^{(i)}, \{\alpha_{t|t}^{(i)}(\ell)\}_{\ell=1}^K, w_t^{(i)}\}_{i=1}^N$ where $\{x_{1:t}^{(i)}, w_t^{(i)}\}_{i=1}^N$ is a weighted particle representation of $p(x_{1:t}|y_{1:t}; \theta)$ and where

$\alpha_{t|t}^{(i)}(\ell) \triangleq p(r_t = \ell | x_{1:t}^{(i)}, y_{1:t}; \theta)$ is attached to each particle with $\ell \in \{1, \dots, K\}$. For online parameter identification, we require to compute the auxiliary function $T_t(z_t)$, cf. (23). Following the derivation in [14], this can be accomplished by introducing an SMC approximation of an extended backward kernel $p(x_{1:t-1}, r_{t-1} | x_t, r_t, y_{1:t-1}; \theta)$, which is given by

$$p(x_{1:t-1}, r_{t-1} = k | x_t^{(i)}, r_t = \ell, y_{1:t-1}; \theta) \approx \sum_{j=1}^N \frac{\tilde{w}_t^{(i,j)}(k, \ell)}{\sum_{u=1}^N \sum_{m=1}^K \tilde{w}_t^{(i,u)}(m, \ell)} \delta(x_{1:t-1} - x_{1:t-1}^{(j)}), \quad (25)$$

where

$$\tilde{w}_t^{(i,j)}(k, \ell) = f_\ell(x_t^{(i)} | x_{t-1}^{(j)}; \theta_\ell) \pi_{k\ell} \alpha_{t-1|t-1}^{(j)}(k) w_{t-1}^{(j)}. \quad (26)$$

By using this approximation of the backward kernel, the auxiliary function $\hat{T}_t^{(i)}(\ell) \approx T(x_t^{(i)}, r_t = \ell)$ can be updated recursively according to

$$\hat{T}_t^{(i)}(\ell) = \sum_{j=1}^N \sum_{k=1}^K \left(\frac{\tilde{w}_t^{(i,j)}(k, \ell)}{\sum_{u=1}^N \sum_{m=1}^K \tilde{w}_t^{(i,u)}(m, \ell)} \times \left[\hat{T}_{t-1}^{(j)}(k) + s_t(x_{t-1}^{(j)}, r_{t-1} = k, x_t^{(i)}, r_t = \ell) \right] \right), \quad (27)$$

with $\hat{T}_0^{(i)}(\ell) = 0$.

VI. SIMULATIONS

The performance of the proposed joint state inference and parameter identification algorithm is assessed on the following benchmark model [28]

$$x_t = \frac{1}{2}x_{t-1} + 25 \frac{x_{t-1}}{1 + x_{t-1}^2} + 8 \cos[1.2(t-1)] + v_t, \quad (28a)$$

$$y_t = \frac{x_t^2}{20} + e_{r_t}, \quad (28b)$$

where the initial state x_0 and process noise v_t are distributed according to $x_0 \sim \mathcal{N}(0, 1)$ and $v_t \sim \mathcal{N}(0, 1)$, respectively. It is further assumed that the measurement noise e_{r_t} is mode dependent, and switches between $K = 2$ modes. The parameters θ_{r_t} of the corresponding measurement noise probability density function (pdf) are assumed to be unknown, and are estimated together with the TPM and the states.

From the model given in (28), we simulate a single state and measurement trajectory of length 10 000, and perform 100 Monte Carlo runs of the RBPF (with $N = 150$ particles) on this data [14]. Even though our proposed algorithm jointly estimates the states and the parameters, we will only show results for the parameter identification.

A. Experiment 1: Gaussian + Gaussian

In the first experiment, we assume that the measurement noise pdf is Gaussian with mode-dependent mean and variance. Hence, we can write $g_k(y_t | x_t; \theta_k) = \mathcal{N}(y_t; x_t^2/20 + \mu_k, \sigma_k^2)$, with $\theta_k = [\mu_k, \sigma_k]$, $k = 1, 2$. Note, that for the vector of unknowns θ_k , different parameterizations are possible. For instance, we could replace the standard deviation σ_k with

the variance σ_k^2 or the precision σ_k^{-2} , resulting in different convergence behaviors of the algorithm. For the chosen parameterization, the partial derivatives required for the computation of the score vector are given by

$$\frac{\partial}{\partial \mu_k} \log g_k(y_t | x_t; \theta_k) = \frac{y_t - \frac{x_t^2}{20} - \mu_k}{\sigma_k^2}, \quad (29a)$$

$$\frac{\partial}{\partial \sigma_k} \log g_k(y_t | x_t; \theta_k) = -\frac{1}{\sigma_k} + \frac{(y_t - \frac{x_t^2}{20} - \mu_k)^2}{\sigma_k^3}. \quad (29b)$$

The simulation parameters were chosen as follows: $\pi_{11} = 0.95$, $\pi_{22} = 0.8$, $\theta_1 = \{0, 1\}$ and $\theta_2 = \{3, 2\}$, whereas the RBPF has been initialized with the following parameters: $\hat{\pi}_{11}^0 = 0.5$, $\hat{\pi}_{22}^0 = 0.5$, $\hat{\theta}_1^0 = \{-1, 5\}$ and $\hat{\theta}_2^0 = \{4, 5\}$.

Since all pdfs describing the JMNLs are Gaussian and thus are members of the exponential family, it is natural to compare the performance of the RML to the EM algorithm proposed in [14]. While the EM parameter estimation starts after 50 iterations (burn-in) in order to guarantee that the M-step update is numerically well-behaved [29], the RML parameter updates are started from the very first iteration. For a fair comparison of the convergence behaviour of the RML and the EM algorithm, we show results for two different choices of the step-size γ_t , namely $\gamma_t = t^{-0.5}$ and $\gamma_t = t^{-0.7}$.

The results are presented in Fig. 1 from which the following conclusions can be drawn. We start our interpretation of the results for the case $\gamma_t = t^{-0.5}$. In terms of estimating the mean μ_k we can observe that for $k = 1$, both RML and EM yield a similar performance. For $k = 2$, the RML estimates have much less variance compared to EM. When we look at the σ_1 estimates, the same conclusions can be drawn. For σ_2 , the RML presents a slower convergence compared to EM, but still provides a smaller variance around the average. For the estimation of the TPM elements, the RML approach provides a more well behaved estimation for both π_{11} and π_{22} . The EM solution, although providing larger ripples in the estimation seems to converge faster to the true value of π_{22} and, after convergence, oscillates with larger amplitudes.

The results are somewhat different if we set $\gamma_t = t^{-0.7}$. We can observe that the RML method generally exhibits a slower convergence, but with estimation variance almost unaltered. The EM method, on the other hand, is less affected by slower convergence, but the estimation variance is considerably improved. As an example, the RML estimation of σ_k yields a much slower convergence for both modes and is even not capable of stabilizing without significant bias for $k = 2$. On the other hand, the variance of estimating σ_2 via the EM method is considerably decreased.

Quantitatively, we can observe for both choices of step size parameters that the parameter estimation performance of the first mode is better than that of the second mode. This can be explained by the fact that the transition probability $\pi_{11} = 0.95$ for remaining in the first mode is higher than $\pi_{22} = 0.8$. Thus, on average we will have more measurement data available for estimating the parameters of the first mode than for the second mode.

Figure 1. Results using step size $\gamma_t = t^{-0.5}$ with RML (blue) and EM (red) in the first and second columns, respectively. Results using step size $\gamma_t = t^{-0.7}$ with RML (blue) and EM (red) in the third and fourth columns, respectively. The lines show the averages and the shaded areas show the upper and lower bounds.

B. Experiment 2: Gaussian + Cauchy

In the second experiment, we assume that the first mode of the measurement noise remains Gaussian, i.e. $g_1(y_t|x_t; \theta_1) = \mathcal{N}(y_t; x_t^2/20 + \mu_1, \sigma_1^2)$ and $\theta_1 = [\mu_1, \sigma_1]$, while the second mode is given by a Cauchy distribution, i.e.

$$g_2(y_t|x_t; \theta_2) = \frac{1}{\pi\sigma_2} \left[\frac{\sigma_2^2}{(y_t - x_t^2/20 - \mu_2)^2 + \sigma_2^2} \right], \quad (30)$$

with $\theta_2 = [\mu_2, \sigma_2]$, and where μ_2 and σ_2 are so-called location and scale parameters, respectively. These are generally different to the mean and standard deviation since the Cauchy distribution does not have finite moments [30]. While the partial derivatives corresponding to mode $k = 1$ remain unaltered, see (29), the partial derivatives for the second mode are given as follows

$$\frac{\partial}{\partial \mu_k} \log g_k(y_t|x_t; \theta_k) = \frac{2(y_t - \frac{x_t^2}{20} - \mu_k)}{(y_t - \frac{x_t^2}{20} - \mu_k)^2 + \sigma_k^2}, \quad (31a)$$

$$\frac{\partial}{\partial \sigma_k} \log g_k(y_t|x_t; \theta_k) = \frac{1}{\sigma_k} - \frac{2\sigma_k}{(y_t - \frac{x_t^2}{20} - \mu_k)^2 + \sigma_k^2}. \quad (31b)$$

For this particular chosen model, we only assess the performance of the RML algorithm, since the online EM method would not be appropriate due to the non-exponential Cauchy distribution present in the probabilistic model. The simulation parameters are the same as those given for the first experiment. The same holds for the initial estimates used for the RBPf, except that $\theta_1^0 = \{-2, 3\}$ and $\theta_2^0 = \{4, 3\}$ have been modified. The step-size has been fixed to $\gamma_t = t^{-0.5}$.

The parameter identification results for this model are presented in Fig. 2. It can be observed that the proposed RML is capable of estimating the unknown model parameters with acceptable convergence behavior. The estimation results for the parameters of the Gaussian distribution are very similar to those of the first experiment, but with slightly larger estimation variance. For the estimation of the parameters of the Cauchy distribution, a slower convergence can be observed, but this is due to the long tails of the distribution which makes the estimation more difficult.

VII. CONCLUSION

In this paper, we have proposed a joint inference and parameter identification algorithm for JMNLS. The parameter identification problem was solved by a RML algorithm

Figure 2. Results for Gaussian + Cauchy experiment using RML. The lines show the averages and the shaded areas show the upper and lower bounds.

based on gradient ascent, that can be also employed in cases when the probabilistic models describing the JMNLs are not members of the exponential family. Simulation results on two benchmark models have shown that the performance is comparable to a previously proposed online EM method. Notable differences were the smaller estimation variance of the RML compared to the online EM method and slower convergence of the former. It was further observed that the choice of the step-size in the RML is more sensitive than in the EM method. In order to speed up convergence, it is possible to use quasi-Newton methods [31] where the information from the Hessian is additionally explored, which is subject to future work.

ACKNOWLEDGMENT

André R. Braga was supported by CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico, CISB - Centro de Pesquisa e Inovação Sueco-Brasileiro and Saab AB. Carsten Fritsche was supported by the Vinnova Industry Excellence Center ELLIIT at Linköping University.

REFERENCES

- [1] D. Cajueiro, "Stochastic Optimal Control of Jumping Markov Parameter Process with Applications to Finance," Ph.D. dissertation, Instituto Tecnológico de Aeronáutica, 2002.
- [2] L. E. Svensson and N. Williams, "Optimal Monetary Policy under Uncertainty in DSGE Models: A Markov Jump-Linear-Quadratic Approach," National Bureau of Economic Research, Working Paper 13892, March 2008.
- [3] A. S. Willsky, "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica*, vol. 12, no. 6, pp. 601–611, Nov. 1976.
- [4] W. S. Gray, O. R. Gonzalez, and M. Dogan, "Stability Analysis of Digital Linear Flight Controllers Subject to Electromagnetic Disturbances," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 4, pp. 1204–1218, Oct 2000.
- [5] K. A. Loparo and F. Abdel-Malek, "A Probabilistic Approach to Dynamic Power System Security," *IEEE Transactions on Circuits and Systems*, vol. 37, no. 6, pp. 787–798, Jun 1990.
- [6] J.-F. Liao and B.-S. Chen, "Robust Mobile Location Estimator with NLOS Mitigation using Interacting Multiple Model Algorithm," *IEEE Transactions on Wireless Communications*, vol. 5, no. 11, pp. 3002–3006, November 2006.
- [7] G. Ackerson and K. Fu, "On State Estimation in Switching Environments," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 10–17, Feb 1970.
- [8] H. A. P. Blom and Y. Bar-Shalom, "The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, Aug 1988.
- [9] A. Logothetis and V. Krishnamurthy, "Expectation Maximization Algorithms for MAP Estimation of Jump Markov Linear Systems," *IEEE Transactions on Signal Processing*, vol. 47, no. 8, pp. 2139–2156, Aug 1999.
- [10] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle Filters for State Estimation of Jump Markov Linear Systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 613–624, Mar 2001.
- [11] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: Wiley-Interscience, 2001.
- [12] C. Andrieu, M. Davy, and A. Doucet, "Efficient Particle Filtering for Jump Markov Systems. Application to Time-Varying Autoregressions," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1762–1770, July 2003.
- [13] Y. Boers and J. N. Driessen, "Interacting Multiple Model Particle Filter," *IEEE Proceedings - Radar, Sonar and Navigation*, vol. 150, no. 5, pp. 344–349, Oct 2003.
- [14] E. Özkan, F. Lindsten, C. Fritsche, and F. Gustafsson, "Recursive Maximum Likelihood Identification of Jump Markov Nonlinear Systems," *Signal Processing, IEEE Transactions on*, vol. 63, no. 3, pp. 754–765, Feb 2015.
- [15] J. Tugnait, "Adaptive Estimation and Identification for Discrete Systems with Markov Jump Parameters," *IEEE Transactions on Automatic Control*, vol. 27, no. 5, pp. 1054–1065, Oct 1982.
- [16] U. Orguner and M. Demirekler, "Maximum Likelihood Estimation of Transition Probabilities of Jump Markov Linear Systems," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5093–5108, Oct 2008.
- [17] T. T. Ashley and S. B. Andersson, "A Sequential Monte Carlo Framework for the System Identification of Jump Markov State Space Models," in *2014 American Control Conference*, June 2014, pp. 1144–1149.
- [18] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.
- [19] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, 2nd ed., ser. Wiley series in probability and statistics. Wiley, 2008.
- [20] C. Fritsche, E. Özkan, and F. Gustafsson, "Online EM Algorithm for Jump Markov Systems," in *2012 15th International Conference on Information Fusion*, July 2012, pp. 1941–1946.
- [21] O. Cappé, E. Moulines, and T. Ryden, *Inference in Hidden Markov Models (Springer Series in Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.
- [22] N. Kantas, A. Doucet, S. S. Singh, J. Maciejowski, and N. Chopin, "On Particle Methods for Parameter Estimation in State-Space Models," *Statist. Sci.*, vol. 30, no. 3, pp. 328–351, 08 2015.
- [23] F. LeGland and L. Mevel, "Recursive Estimation in Hidden Markov Models," in *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 4, Dec 1997, pp. 3468–3473 vol.4.

- [24] A. Krogh and S. K. Riis, "Hidden Neural Networks," *Neural Computation*, vol. 11, no. 2, pp. 541–563, 1999.
- [25] G. Poyiadjis, A. Doucet, and S. S. Singh, "Particle Approximations of the Score and Observed Information Matrix in State Space Models with Application to Parameter Estimation," *Biometrika*, vol. 98, no. 1, pp. 65–80, 2011.
- [26] A. Doucet, S. Godsill, and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [27] P. Del Moral, A. Doucet, and S. Singh, "Forward Smoothing using Sequential Monte Carlo," Cambridge University Engineering Department, Tech. Rep. CUED/F-INFENG/TR638, 2010.
- [28] G. Kitagawa, "Monte Carlo Filter and Smoother for Non-Gaussian Non-linear State Space Models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. pp. 1–25, 1996.
- [29] O. Cappe, "Online EM Algorithm for Hidden Markov Models," *Journal of Computational and Graphical Statistics*, vol. 20, no. 3, pp. 728–749, 2011.
- [30] N. Johnson and S. Kotz, *Continuous Univariate Distributions: Distributions in Statistics*, ser. Their Distributions in statistics. Houghton Mifflin, 1970, no. v. 1.
- [31] J. Nocedal and S. J. Wright, *Numerical Optimization*, ser. Springer Series in Operations Research and Financial Engineering. Berlin: Springer, 2006.