Offline Bayesian Identification of Jump Markov Nonlinear Systems

Miguel Barão * **Jorge S. Marques** **

* *U. Évora/INESC-ID, Portugal*
** *ISR/IST, Portugal*

Abstract: This paper presents a framework for the offline identification of nonlinear switched systems with unknown model structure. Given a set of sampled trajectories, and under the assumption that they were generated by switching among a number of models, we estimate a set of vector fields and a stochastic switching mechanism that best describes the observed data. The switching mechanism is described by a position dependent hidden Markov model that provides the probabilities of the next active model given the current active model and the state vector. The vector fields and the stochastic matrix is obtained by interpolating a set of nodes distributed over a relevant region in the state space. The work follows a Bayesian formulation where the EM-algorithm is used for optimization.

1. INTRODUCTION

While linear systems identification is an area where solutions based on least squares are common, nonlinear systems are too vast an area to be covered by a single technique. Several approaches have been proposed to deal with this kinds of problems. Some assume a separation between the linear and nonlinear parts, such as in Hammerstein systems, while others assume black-box models. The current work relates to the latter.

Black-box models are universal in the sense that they are able to describe any kind of nonlinear dynamics. Techniques commonly employed are usually based on neural networks, radially basis functions, wavelets, fuzzy sets and Volterra expansions (see Sjöberg et al. [1995] for an overview). In addition to the nonlinearities governing the state evolution, some systems can abruptly switch the active model. These systems are usually known as jump Markov systems and a vast literature exists concerning them. Several different problems are usually dealt with: state estimation problems (Liu and Zhang; [2009]), identification of the dynamical model (Cinquemani et al. [2007]), identification of transition probabilities (Orguner [2008]) and both transition probabilities and state estimation (Tugnait [1982]).

This paper deals with jump Markov nonlinear systems, where the Markov part depends simultaneously on the continuous and discrete states. It follows closely the framework developed in Nascimento et al. [2009], originally intended for computer vision, and improvements introduced in Barão et al. [2010]. These works are further extended here to work for higher dimensional dynamic systems.

The main improvements introduced in this paper are: the extension to an arbitrary D-dimensional space; the interpolation used to define the vector and Markov matrix fields; and the support for irregular grids to better approximate the nonlinear behavior in certain regions of the state space. The extension to a D-dimensional space requires a new metric in the state space, which was not required in the referred papers, and changes in the algorithms to avoid numerical problems. The use of an irregular grid also constitutes a challenge since proper node weighting has to be selected to obtain a good interpolation. Node selection is not considered here.

The paper is organized as follows. Section 2 formulates the problem more precisely; Section 3 introduces some assumptions used to define prior distributions for Bayesian estimation; Section 4 presents the EM-algorithm along with formulas for vector field estimation; Section 4.2.2 focus on the optimization of switching probabilities; Section 5 presents an example showing two coupled Lorenz strange attractors with Markov jumps between them, and the estimated fields obtained with the algorithms developed; Finally section 6 draws conclusions.

2. PROBLEM FORMULATION

The problem under consideration tries to estimate a set of models that best describe observed state trajectories of an unknown nonlinear dynamic system. It is assumed that the complete state $x_t \in \mathbb{R}^d$ is accessible at discrete time steps. The discretized nonlinear systems are assumed to be given by

$$x_t = x_{t-1} + T_{k_t}(x_{t-1}) + w_t \tag{1}$$

where $T_{k_t}(\cdot)$ is a vector field describing the current active model. It is indexed at time t by $k_t \in \{1, \ldots, K\}$, K being the number of models. The additive disturbance is a zero mean multivariable Gaussian distribution $w_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_k)$ where a different covariance matrix $\mathbf{\Sigma}_k$ is assigned to each model k.

The active model can change from one time step to the next according to a Markov model described by a space dependent matrix $B(x) = [b_{ij}(x)]$, where $b_{ij}(x)$ is the

⋆ This work was supported by FCT (INESC-ID and ISR multiannual funding) through the PIDDAC Program funds and project "ARGUS - Activity Recognition and Object Tracking Based on Multiple Models", PTDC/EEA-CRO/098550/2008.

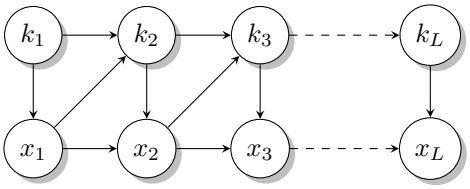

Fig. 1. Markov diagram showing the state variable x_t and active model k_t updates.

probability from switching from model i to model j when the state equals x, *i.e.*,

$$b_{ij}(x_{t-1}) \triangleq \Pr\{k_t = j | k_{t-1} = i, x_{t-1}\}. \quad (2)$$

Matrix $B(x)$ is a stochastic matrix and switching probabilities $b_{ij}(x)$ should obey the following conditions:

$$b_{ij}(x) \geq 0, \quad \sum_{m=1}^{K} b_{im}(x) = 1, \quad \forall\, i,j \in \{1,\ldots,K\}, \quad x \in \mathbb{R}^d. \quad (3)$$

Switching probabilities $b_{ij}(x)$ are also to be estimated from the observed state trajectories. The active model k_t at each time instant is not observed and is unknown.

The identification is performed offline using a collection of recorded trajectories. Each trajectory, denoted by x^s, is an ordered set of points $(x_1^s, x_2^s, \ldots, x_{L_s}^s)$, where $x_t^s \in \mathbb{R}^D$ and L_s is the length of the trajectory s.

The model described above has an hybrid state (x_t, k_t) which is updated using equations (1) and (2). The hybrid state has an observed part x_t and an hidden part k_t.

Figure 1 shows the dependencies among the variables x_t and k_t as the time evolves.

2.1 Space discretization

To obtain a smooth vector field in the state space $\mathbb{R}^D$, the space is discretized into a set of N nodes distributed over the region where data is available. This nodes have coordinates $g_n \in \mathbb{R}^D$, $n \in \{1,\ldots,N\}$.

A set of K vectors are estimated for each node n. Each of these vectors T_k^n represent the vector field $T_k(x)$ when x is over the node n, *i.e.*, when $x = g_n$. Similarly, a stochastic matrix B^n is estimated for each node. Having estimated vectors T_k^n and stochastic matrices B^n for every node of the grid, vector fields $T_k(x)$ and matrix fields $B(x)$ defined over the entire state space are obtained by multivariate interpolation as follows:

$$T_k(x) \triangleq \sum_{n=1}^{N} T_k^n \phi_n(x) \quad (4)$$

and

$$B(x) \triangleq \sum_{n=1}^{N} B^n \phi_n(x), \quad (5)$$

where $\phi_n(x)$ is an interpolation function satisfying the following convexity constraints:

$$\phi_n(x) > 0, \quad \sum_{n=1}^{N} \phi_n(x) = 1, \quad \forall x \in \mathbb{R}^D. \quad (6)$$

While these constraints could be relaxed for the interpolation of the vector field alone, they are required for the interpolated stochastic matrix $B(x)$ to be a valid stochastic matrix.

An interpolating function that satisfies the above constraints is the *Inverse Distance Weighting* defined by Shepard Shepard [1968]. It is defined by

$$\phi_n(x) \triangleq \begin{cases} \dfrac{d(x,g_n)^{-p}}{\sum_{m=1}^{N} d(x,g_m)^{-p}} & \text{if } x \neq g_n \\ 1 & \text{if } x = g_n \end{cases} \quad (7)$$

where $d(\cdot,\cdot)$ is a distance function (metric) in the state space and the parameter p adjusts the influence of neighboring points with respect to a given node.

2.2 Parameter estimation

The discretized space allows the description of the vector fields and stochastic matrices using a finite set of parameters. Let $\mathcal{T} \triangleq \{T_k^n\}$ denote the set of vectors to be estimated and $\mathcal{B} \triangleq \{B^n\}$ the set stochastic matrices. The complete set of parameters is denoted by $\theta \triangleq (\mathcal{T}, \mathcal{B})$.

The parameter estimation then amounts to find θ from the collection of trajectories $\mathcal{X} \triangleq \{x^1, \ldots, x^S\}$, where S is the number of trajectories observed. The methodology follows a Bayesian viewpoint: estimated parameters are given by the maximum *a posteriori*

$$\hat{\theta} = \arg\max_{\theta} p(\theta|\mathcal{X}). \quad (8)$$

Applying the Bayes rule and exposing the dependence on the hidden variables k_t yields

$$\hat{\theta} = \arg\max_{\theta} p(\theta) \sum_{\mathcal{K}} p(\mathcal{X}, \mathcal{K}|\theta). \quad (9)$$

The solution depends on the selection of an appropriate prior $p(\theta)$ and the computation of the probability $p(\mathcal{X}, \mathcal{K}|\theta)$. The marginalization with respect to $\mathcal{K}$ requires the summation over the set of all possible sequences of active models along every observed trajectory. This summation is unfeasible which renders the direct optimization of the above equation impossible. A version of the Expectation-Maximization algorithm Dempster et al. [1977] is employed to deal with this limitation. The following two sections deal with the prior selection and EM-algorithm.

3. PRIOR SELECTION

The selection of a prior $p(\theta)$ has an essential contribution in the quality of the estimated model. Namely, for the portions of the space where poor or no data is available, the resulting model will depend heavily on the prior information. This is specially important for the vector field estimation since, in many occasions, data is scarce at some state space regions. At the same time, some independence assumptions have to be considered to simplify and separate the overall problem into simpler and smaller subproblems.

The first independence assumption regards a separation between the vector field estimation and the stochastic matrix field estimation problems. It is assumed that

$$p(\theta) = p(\mathcal{T}, \mathcal{B}) = p(\mathcal{T})p(\mathcal{B}). \quad (10)$$

Regarding the vector field estimation, the following additional assumptions are made:

- Vector fields of different models are independent from each other, *i.e.*,

$$p(\mathcal{T}) = \prod_{k=1}^{K} p(T_k^1, \ldots, T_k^N). \tag{11}$$

- For a given model, having index k, the vectors T_k^n at its nodes are not mutually independent. Their dependence is introduced so that if there is data near one node but not near a neighbor node, the latter inherits some information from the former. This "generalization" capability is introduced by a careful selection of a prior distribution $p(T_k^1, \ldots, T_k^N)$. The selected prior is a multivariable Gaussian distribution assigning an higher density when two neighbor nodes have similar vectors. This is accomplished by first defining the set of neighbor nodes

$$\mathcal{I} \triangleq \big\{(i,j) \mid d(g_i, g_j) < d_{\max},\ i \neq j\big\}. \tag{12}$$

This set is used to define a sparse matrix Δ having, at each column, a pair $(1, -1)$ at positions corresponding to neighbor nodes. Then, collecting the vectors $T_k^n \in \mathbb{R}^d$ in a single matrix $\mathbf{T_k} \triangleq \big[T_k^1 \ \cdots \ T_k^N\big] \in \mathbb{R}^{d \times N}$, the product $\mathbf{T}_k\mathbf{\Delta}$ produces a new matrix where each column is the difference between two neighbor nodes. Finally, the prior can be defined to be

$$p(T_k^1, \ldots, T_k^N) \propto e^{-\frac{1}{2}\operatorname{Tr}(\mathbf{T}_k\mathbf{\Delta}\mathbf{\Delta}^T\mathbf{T}_k^T)}. \tag{13}$$

The previous definition may not work in all instances since the sparsity of $\mathbf{\Delta}$ may lead to an improper distribution. To ensure positive definiteness of the covariance matrix a small additional term $\epsilon\mathbf{Q}$ is added, where $\mathbf{Q} \in \mathbb{R}^{d \times d}$ is a positive definite metric matrix on the state space. The prior can finally be written as the multivariable Gaussian

$$p(T_k^1, \ldots, T_k^N) \propto e^{-\frac{1}{2}\operatorname{Tr}(\mathbf{T}_k\mathbf{\Lambda}^{-1}\mathbf{T}_k^T)}, \tag{14}$$

where

$$\mathbf{\Lambda}^{-1} = \epsilon\mathbf{Q} + \mathbf{\Delta}\mathbf{Q}\mathbf{\Delta}^T. \tag{15}$$

Regarding the stochastic matrices estimation, a prior $p(\mathcal{B})$ is selected. Currently, it is set to a constant density in the $K-1$ simplex defined by its parameters. The main reason this was chosen is one of simplicity. Also, it is unknown if a different prior, like *e.g.* the Jeffreys non-informative prior, has a noticeable advantage in the considered problem.

4. EXPECTATION-MAXIMIZATION ALGORITHM

Given the priors $p(\mathcal{T})$ and $p(\mathcal{B})$, and a set of recorded trajectories $\mathcal{X} = \{x^1, \ldots, x^S\}$, the maximum *a posteriori* estimate $\hat{\theta}$ of the parameters θ is given by

$$\hat{\theta} = \arg\max_{\theta} p(\theta) \sum_{\mathcal{K}} p(\mathcal{X}, \mathcal{K} | \theta). \tag{16}$$

As referred earlier, the summation over all the sequences of active models $\mathcal{K}$ is unfeasible since it grows exponentially with the amount of data available. To deal with this problem the Expectation-Maximization algorithm from Dempster et al. [1977] is employed on the complete joint probability $p(\mathcal{X}, \mathcal{K}, \theta)$. The problem is formulated as follows: First, take the logarithm of the joint distribution to get

$$\log p(\mathcal{X}, \mathcal{K}, \theta) = \log p(\mathcal{K}|\mathcal{X}, \theta) + \log p(\mathcal{X}|\theta) + \log p(\theta). \tag{17}$$

Then, it can be checked that the maximization of the two right terms $\log p(\mathcal{X}|\theta)$ and $\log p(\theta)$ is equivalent to maximizing the difference

$$\log p(\mathcal{X}, \mathcal{K}, \theta) - \log p(\mathcal{K}|\mathcal{X}, \theta) \tag{18}$$

Since this maximization is still unfeasible, the EM method suggests instead the maximization of its expected value with respect to a previously estimated $\hat{\theta}$, *i.e.*

$$\hat{\theta}^{\text{new}} = \arg\max_{\theta} \Big(U(\theta, \hat{\theta}) - V(\theta, \hat{\theta})\Big), \tag{19}$$

where

$$U(\theta, \hat{\theta}) \triangleq E\big[\log p(\mathcal{X}, \mathcal{K}, \theta) \big| \mathcal{X}, \hat{\theta}\big] \tag{20}$$

$$V(\theta, \hat{\theta}) \triangleq E\big[\log p(\mathcal{K}|\mathcal{X}, \theta) \big| \mathcal{X}, \hat{\theta}\big]. \tag{21}$$

It can be proved that $V(\theta, \hat{\theta}) \leq 0$ for all $\theta \neq \hat{\theta}$. Thus, this term can only improve the difference in (19) and the maximization can be performed for $U(\theta, \hat{\theta})$ alone.

The EM algorithm iterates two steps: Computation of $U(\theta, \hat{\theta})$, called the E-step, and its maximization with respect to θ while keeping $\hat{\theta}$ fixed, called the M-step. These two steps are iterated until convergence to a local *maxima* is attained.

4.1 E-step

In the E-step part of the algorithm, the function $U(\theta, \hat{\theta})$ is found to be given by

$$\begin{aligned} U(\theta,\hat{\theta}) = C &- \frac{1}{2}\sum_{k=1}^{K} \operatorname{Tr}(\mathbf{T}_k\mathbf{\Lambda}^{-1}\mathbf{T}_k^T) + \\ &- \frac{1}{2}\sum_{s,t,k} w_k^s(t) \Big\| x_t^s - x_{t-1}^s - \sum_{n=1}^{N} T_k^n \phi_n(x_{t-1}^s) \Big\|^2_{\Sigma_k^{-1}} + \\ &+ \sum_{s,t}\sum_{i,j=1}^{K} w_{ij}^s(t) \log\Big(\sum_{n=1}^{N} b_{ij}^n \phi_n(x_{t-1}^s)\Big). \end{aligned} \tag{22}$$

where the symbols w_i and w_{ij} are defined by

$$w_j^s(t) \triangleq \Pr\big\{k_t^s = j | x^s, \hat{\theta}\big\}, \tag{23}$$

$$w_{ij}^s(t) \triangleq \Pr\big\{k_{t-1}^s = i, k_t^s = j | x^s, \hat{\theta}\big\}, \tag{24}$$

and calculated using the forward-backward algorithm (see Rabiner [1990]) that is described next.

In the following expressions, the symbol $x_{1:t}^s \triangleq (x_1^s, \ldots, x_t^s)$ represents a trajectory from 1 to t and is introduced to simplify the notation.

The forward part of the forward-backward algorithm works as follows: consider the forward variable

$$\alpha_t^s(i) \triangleq \Pr\big\{x_{1:t}^s, k_t^s = i | \hat{\theta}\big\}. \tag{25}$$

For any time $t \in \{1, \ldots, L_s\}$, the computation of $\alpha_t^s(i)$ can be done with the following algorithm:

(1) Initialization: since we do not have information from time $t < 1$, initialization depends only on the probability of the active model at $t = 1$:

$$\alpha_1^s(i) = \Pr\big\{k_1^s = i | \hat{\theta}\big\}. \tag{26}$$

(2) Induction:

$$\begin{aligned}\alpha_t^s(i) &= \Pr\{x_t^s | x_{t-1}^s, k_t^s = i, \hat{\theta}\} \Pr\{x_{1:t-1}^s, k_t^s = i | \hat{\theta}\} \\ &= \mathcal{N}\big(x_t^s \big| x_{t-1}^s + T_i(x_{t-1}^s), \mathbf{\Sigma}_{k_t^s}\big) \cdot \\ &\quad \cdot \sum_{j=1}^{K} b_{ij}(x_{t-1}^s) \alpha_{t-1}^s(j).\end{aligned} \tag{27}$$

The backward part of the algorithm works similarly, but from the other end backwards in time. It is as follows: consider the backward variable

$$\beta_t^s(i) \triangleq \Pr\{x_{t+1:L_s}^s | k_t^s = i, x_t, \hat{\theta}\}. \tag{28}$$

It is computed using the following algorithm:

(1) Initialization:

$$\beta_{L_s}^s(i) = 1. \tag{29}$$

(2) Induction: since x_{t+1} is conditionally independent of the remaining trajectory $x_{t+2:L_s}$ given x_t and k_{t+1}, then $\beta_t^s(i)$ is given by

$$\begin{aligned}\beta_t^s(i) &= \sum_{j=1}^{K} \Pr\{x_{t+1:L_s}^s | k_{t+1}^s = j, x_t^s\} \cdot \\ &\quad \cdot \Pr\{k_{t+1}^s = j | k_t^s = i, x_t\} \\ &= \sum_{j=1}^{K} \beta_{t+1}^s(j)\, \mathcal{N}\big(x_{t+1}^s \big| x_t^s + T_j(x_t), \mathbf{\Sigma}_j\big)\, b_{ij}(x_t).\end{aligned} \tag{30}$$

The variables $\alpha_t^s(i)$ and $\beta_t^s(i)$ can be combined together to produce the weights $w_i^s(t)$ and $w_{ij}^s(t)$ as shown in the following equations:

$$w_i^s(t) = \frac{\alpha_t^s(i)\beta_t^s(i)}{\sum_{j=1}^{K} \alpha_t^s(j)\beta_t^s(j)}. \tag{31}$$

$$w_{ij}^s(t) = \frac{\alpha_{t-1}^s(i) b_{ij}(x_{t-1}^s) \mathcal{N}\big(x_t^s \big| x_{t-1}^s + T_j(x_{t-1}^s), \mathbf{\Sigma}_j\big) \beta_t^s(j)}{\Pr\{x^s | \hat{\theta}\}} \tag{32}$$

where the normalization constant $\Pr\{x^s|\hat{\theta}\}$ is obtained by summing the numerator over all $1 \le i, j \le K$.

Having computed the expressions (22)–(24), the maximization step can be performed.

4.2 M-step

In the M-step part of the algorithm, equation (22) is maximized with respect to the vectors T_k^n and transition probabilities b_{ij}^n. The vector field maximization can be done explicitly, while the switching probabilities are computed using an iterative algorithm.

Computation of the vector field. The stationarity points are the solutions of the linear matrix equation

$$\frac{\partial U(\theta, \hat{\theta})}{\partial \mathbf{T}_\alpha} = -\mathbf{\Lambda}^{-1}\mathbf{T}_\alpha^T - \mathsf{A}_\alpha \mathbf{T}_\alpha^T \Sigma_\alpha^{-1} + \mathsf{B}_\alpha = \mathbf{0}, \tag{33}$$

where matrices A_α and B_α are defined by

$$\mathsf{A}_\alpha \triangleq \sum_{s=1}^{S} \sum_{t=1}^{L_s} w_\alpha^s(t) \Phi(x_{t-1}^s) \Phi(x_{t-1}^s)^T, \tag{34}$$

$$\mathsf{B}_\alpha \triangleq \sum_{s=1}^{S} \sum_{t=1}^{L_s} w_\alpha^s(t) \Phi(x_{t-1}^s) (x_t^s - x_{t-1}^s)^T \Sigma_\alpha^{-1} \tag{35}$$

and

$$\Phi(x) \triangleq [\phi_1(x) \ \cdots \ \phi_N(x)]^T. \tag{36}$$

Premultiplying (33) by the positive definite matrix $\mathbf{\Lambda}$ yields an equivalent form of the Sylvester equation

$$(-\mathbf{\Lambda}\mathsf{A}_\alpha)\mathbf{T}_\alpha^T \Sigma_\alpha^{-1} - \mathbf{T}_\alpha^T + \mathbf{\Lambda}\mathsf{B}_\alpha = \mathbf{0}, \tag{37}$$

which can be readily solved for $\mathbf{T}_\alpha$ using standard numerical packages.

Computation of the switching probabilities. When differentiating $U(\theta, \hat{\theta})$ with respect to the transition probabilities b_{ij}^n care has to be taken since they satisfy the constraint $\sum_{j=1}^{K} b_{ij}^n = 1$ and, therefore, parameters are not independent. For differentiation, only independent probabilities b_{ij}^n where $j = 2, \ldots, K$ are selected. This differentiation yields

$$\begin{aligned}\frac{\partial U(\theta, \hat{\theta})}{\partial b_{\alpha\beta}^\gamma} = \sum_{s=1}^{S} \sum_{t=1}^{L_s} \Big(&\frac{w_{\alpha\beta}^s(t)}{\sum_{n=1}^{N} b_{\alpha\beta}^n \phi_n(x_{t-1}^s)} + \\ &- \frac{w_{\alpha 1}^s(t)}{\sum_{n=1}^{N} b_{\alpha 1}^n \phi_n(x_{t-1}^s)} \Big) \phi_\gamma(x_{t-1}^s).\end{aligned} \tag{38}$$

Unfortunately, the stationarity condition $\partial U(\theta, \hat{\theta}) / \partial b_{\alpha\beta}^\gamma = 0$ doesn't appear to have a simple explicit solution. A natural gradient iterative method is then used to estimate the parameters b_{ij}^n (see Barão et al. [2010] for details of this approach).

5. SIMULATION RESULTS

As an illustrative example, the algorithm was applied to two interwound Lorenz strange attractors in $\mathbb{R}^3$, randomly switching between the two with nonzero but low probability.

One Lorenz attractor is described by the following differential equations:

$$\begin{aligned}\dot{x}_1 &= 10(x_2 - x_1) \\ \dot{x}_2 &= x_1(28 - x_3) - x_2 \\ \dot{x}_3 &= x_1 x_2 - \tfrac{8}{3} x_3\end{aligned}$$

A second Lorenz attractor was generated with the variables x_1 and x_2 interchanged, producing a mirror effect. Figure 2 shows a set of sampled trajectories from a simulation with stochastic switching. The trajectory color indicates the active model used at each time instant.

For the estimation procedure, a regularly spaced $5 \times 5 \times 5$ grid was used. Two vector fields and corresponding transition probabilities were estimated. Figure 3 shows the results obtained for the vector fields after 50 interations of the EM algorithm.

The proposed benchmark is a difficult problem for several reasons: the simulated systems are strongly nonlinear; there is no data in many regions of the state space; and the two attractors are sufficiently close so that the switching can in some instances be simply justified by the estimation algorithm as a state disturbance instead of a jump.

To visually check the obtained model, a stochastic simulation was performed using the estimated vector fields and switching probabilities. Figure 4 shows the results obtained. It can be seen that while the model does not

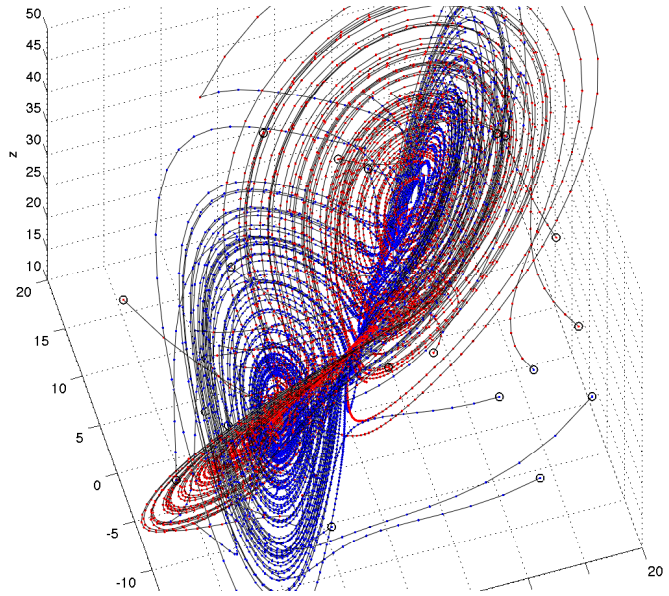

Fig. 2. Trajectories generated by two Lorenz strange attractors with stochastic switching between them.

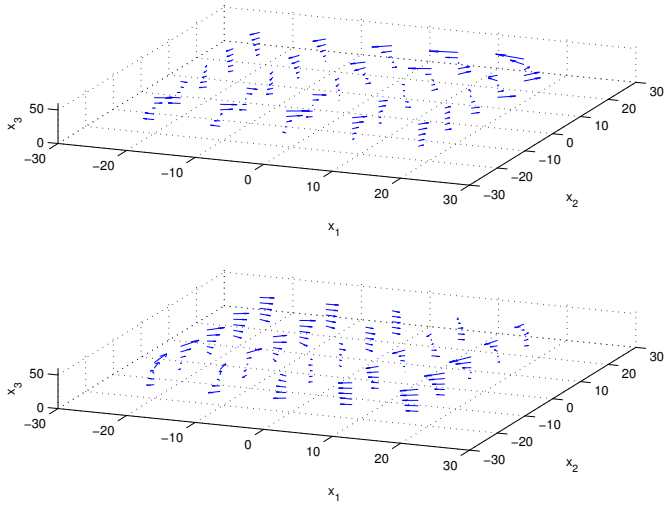

Fig. 3. Estimated vector fields using the EM-method.

precisely replicate the original trajectories, it can reasonably resemble the original systems behavior. It should be emphasized that this results were obtained with a very coarse grid having only 5 nodes for each dimension. A quantitative measure of fitness for the estimated model can be obtained for these kind of problems using the fordward part of the forward-backward algorithm described in section 4.1.

6. CONCLUSIONS AND FUTURE DIRECTIONS

This paper attempts to provide a framework for black box system identification of jump Markov nonlinear systems following a Bayesian approach. The proposed dynamical system is based on the multivariable interpolation of vectors and stochastic matrices in the state space. The interpolation defines a set of vector fields that describe the observed system behavior. For the Bayesian estimation procedure several prior assumptions were used which proved to be determinant in the quality of the resulting

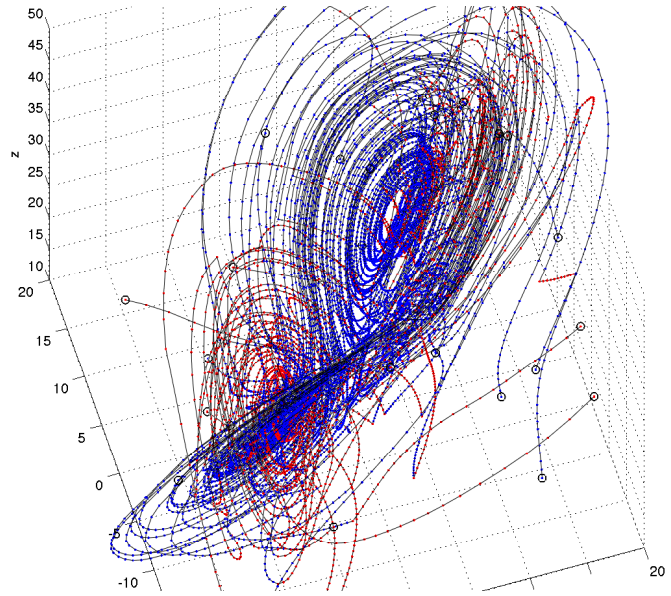

Fig. 4. Stochastic simulation using the estimated vector fields.

vector fields. Namely the prior for the vectors T_k^n allows the algorithm to have a "generalization" capability and smoothly extend the vector field from the regions where data is abundant to regions where no observations are available. The estimation procedure makes use of the Expectation-Maximization algorithm and is demonstrated in a difficult synthetic example.

Several problems are open to further enhancements and are currently under active research. One such problem is the grid selection that should be generated optimally for the data instead of using an *a priori* fixed (possibly irregular) grid. Also, a few parameters are still requiring manual tuning, namely the regularization imposed on the prior $p(\mathcal{T})$, the distance metric and the power p used in the interpolation.

REFERENCES

M Barão, J S Marques, and J M Lemos. An improved EM-method for the estimation of transition probabilities in multiple model switching systems. *NOLCOS 2010 Conference*, September 1–3 2010.

E Cinquemani, R Porreca, G Ferrari-Trecate, and J Lygeros. A general framework for the identification of jump markov linear systems. *Decision and Control, 2007 46th IEEE Conference on*, pages 5737 – 5742, 2007. doi: 10.1109/CDC.2007.4434378.

AP Dempster, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

W Liu and H Zhang;. On state estimation of discrete-time markov jump linear systems. *Control and Decision Conference, 2009. CCDC '09. Chinese*, pages 1110 – 1115, 2009. doi: 10.1109/CCDC.2009.5191528.

JC Nascimento, MAT Figueiredo, and JS Marques. Trajectory analysis in natural images using mixtures of vector fields. *IEEE International Conference on Image Processing*, pages 4353—4356, 2009.

Orguner. Maximum likelihood estimation of transition probabilities of jump markov linear systems. *Signal Processing, IEEE Transactions on*, 56(10):5093 – 5108, 2008. doi: 10.1109/TSP.2008.928936.

LR Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Readings in speech recognition*, 53(3):267–296, 1990.

D Shepard. A two-dimensional interpolation function for irregularly-spaced data. *Proceedings of the 1968 23rd ACM national ...*, Jan 1968.

J Sjöberg, Q Zhang, L Ljung, A Benveniste, and B Delyon. Nonlinear black-box modeling in system identification: a unified overview* 1. *Automatica*, Jan 1995.

J Tugnait. Adaptive estimation and identification for discrete systems with markov jump parameters. *Automatic Control, IEEE Transactions on*, 27(5):1054 – 1065, 1982. doi: 10.1109/TAC.1982.1103061.