Segmental Switching Linear Dynamic Systems

Sang Min Oh James M. Rehg Frank Dellaert
College of Computing, Georgia Institute of Technology
TSRB, 85 5th Street NW, Atlanta, GA 30332
{sangmin, rehg, dellaert}@cc.gatech.edu
GIT-GVU-TechReport

Abstract

We introduce Segmental Switching Linear Dynamic Systems (S-SLDS), which improve on standard SLDSs by explicitly incorporating duration modeling capabilities. We show that S-SLDSs can adopt arbitrary finite-sized duration models that describe data more accurately than the geometric distributions induced by standard SLDSs. We also show that we can convert an S-SLDS to an equivalent standard SLDS with sparse structure in the resulting transition matrix. This insight makes it possible to adopt existing inference and learning algorithms for the standard SLDS models to the S-SLDS framework. As a consequence, the more powerful S-SLDS model can be adopted with only modest additional effort in most cases where an SLDS model can be applied. The experimental results on honeybee dance decoding tasks demonstrate the robust inference capabilities of the proposed S-SLDS model.

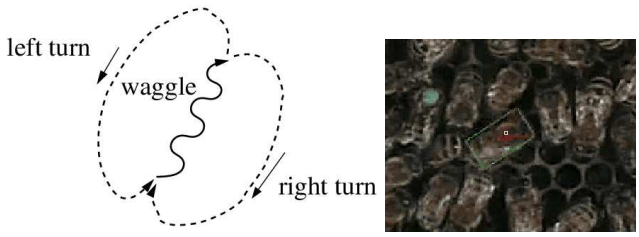
1 Introduction

Switching Linear Dynamical System (SLDS) models have been studied in a variety of problem domains. Representative examples include computer vision [5, 19, 26, 20, 21], computer graphics [33], speech recognition [28], econometrics [14], machine learning [16], and statistics [30]. An SLDS model represents the nonlinear dynamic behavior of a complex system by switching among a set of linear dynamic models over time. In contrast to HMM's, the Markov process in an SLDS selects from a set of continuously evolving linear Gaussian dynamic models, rather than from a fixed Gaussian mixture density. SLDS models have become increasingly popular in the vision and graphics communities as they provide an intuitive framework for describing the continuous but non-linear dynamics of real-world motion.

Nevertheless, the modeling capabilities of a standard SLDS are *limited* by the Markov assumption which is imposed upon the switching process. This process governs the transitions between LDS models and makes it possible for an SLDS to represent nonlinear dynamics. As a consequence of the Markov assumption, however, the probability of remaining in a given switching state follows a geometric distribution with the property that a duration of one time step has the largest probability mass.

Hence, if we perform inference with standard SLDSs, the results suffer from the restriction to geometric distributions. In previous work [20] we used Markov chain Monte Carlo sampling (MCMC) to approximate the true posterior over label sequences. However, the reported results still had several over-segmentations due to the increased importance attached to short durations in the geometric distribution induced in the standard SLDS model, especially in the presence of high levels of noise.

These same problems were previously addressed by the HMM communities, and several extensions to HMM models which provide enhanced duration modeling are described in [10, 17, 23]. The current paper applies some of the same ideas to arrive at SLDS models with enhanced duration modeling capabilities.



(a) (b)

Figure 1: (a) A bee dance is in three patterns : waggle, left turn, and right turn. (b) Green box is a tracked bee.

1.1 Biotracking

The application domain which motivates this work is a new research area which enlists visual tracking and AI modeling techniques in the service of biology [1, 2]. The current state of biological field work is still dominated by manual data interpretation, a time-consuming and error-prone process. Automatic interpretation methods can provide field biologists with new tools for the quantitative study of animal behavior. A classical example of animal behavior and communication is the honey bee dance, depicted in a stylized form in Fig.1(a). Honey bees communicate the location and distance to a food source through a dance that takes place within the hive. The dance is decomposed into three different regimes: "turn left", "turn right" and "waggle". The length (duration) and orientation of the waggle phase corresponds to the distance and the orientation to the food source. Figure 1(b) shows a dancer bee that was tracked by a previously developed vision-based tracker [13]. After tracking, the obtained trajectory of the dancing bee is manually labeled as "turn left" (blue), "turn right" (red) or "waggle" (green) and is shown in Figure 2.

The research goals in this application domain are two-fold. First, we aim to learn the motion patterns of honey bee dances from the obtained training dance sequences. Second, we should be able to automatically segment new dance sequences into three dance modes reliably, i.e., the labeling problem. Note that labels are initially unknown.

Figure 2: An example honey bee dance trajectory. The track is automatically obtained using a vision-based tracker and manually labeled afterward. Key : waggle (green), right-turn (red), left-turn (blue).

1.2 A Model-Based Approach

We take a model-based approach, in which we employ a computational model of behavior in order to interpret the data. In our case the motions are complex, i.e. they are comprised of sub-behaviors. The model we use should be expressive enough to accurately model the individual sub-behaviors, while at the same time able to capture the inter-relationships between them.

Hence, the basic generative model we adopt is the Switching Linear Dynamic System (SLDS) model [25, 24, 26]. In an SLDS model, there are multiple linear dynamic systems (LDS) that underly the motion, one for each behavioral mode that we assume. We can then model the complex behavior of the target by switching within this set of LDSs. In contrast to an HMM, an SLDS provides the possibility to describe complex temporal patterns concisely and accurately. SLDS models have become increasingly popular in the vision and graphics communities as they provide an intuitive framework for describing the continuous but non-linear dynamics of real-world motion. For example, it has been used for human motion classification [25, 24, 26, 27] and motion synthesis [33].

1.3 Background

1.3.1 Linear Dynamic Systems

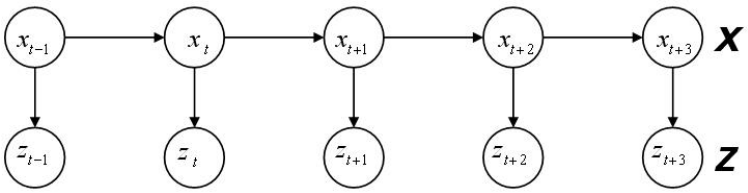


Figure 3: A linear dynamic system (LDS)

An LDS is a time-series state-space model consisting of a linear Gaussian dynamics model and a linear Gaussian observation model. The graphical representation of an LDS is shown in Fig.3. The Markov chain at the top represents the state evolution of the continuous hidden states x_t. The prior density p_1 on the initial state x_1 is assumed to be normal with mean μ_1 and covariance Σ_1, i.e., $x_1 \sim \mathcal{N}(\mu_1, \Sigma_1)$.

The state x_t is obtained by the product of state transition matrix F and the previous state x_{t-1} corrupted by zero-mean white noise w_t with covariance matrix Q:

$$x_t = Fx_{t-1} + w_t \text{ where } w_t \sim \mathcal{N}(0, Q) \tag{1}$$

In addition, the measurement z_t is generated from the current state x_t through the observation matrix H, and corrupted by zero-mean observation noise v_t:

$$z_t = Hx_t + v_t \text{ where } v_t \sim \mathcal{N}(0, V) \tag{2}$$

Thus, an LDS model M is defined by the tuple $M \triangleq \{(\mu_1, \Sigma_1), (F, Q), (H, V)\}$. Exact inference in an LDS can be done exactly using the RTS smoother [3], an efficient belief propagation implementation. For further details on LDSs, the reader is referred to [3, 18, 29].

1.3.2 Switching Linear Dynamic Systems

In an SLDS we assume the existence of n distinct LDS models $M \triangleq \{M_l | 1 \leq l \leq n\}$. The graphical model corresponding to an SLDS is shown in Fig.4. The middle chain, representing the hidden state sequence $X \triangleq \{x_t | 1 \leq t \leq T\}$, together with the observations $Z \triangleq \{z_t | 1 \leq t \leq T\}$ at the bottom, is identical to an LDS in Fig.3. However, we now have an additional discrete Markov chain $L \triangleq \{l_t | 1 \leq t \leq T\}$ that determines which of the n models M_l is used at every time-step. We call $l_t \in M$ the label at time t and L a label sequence.

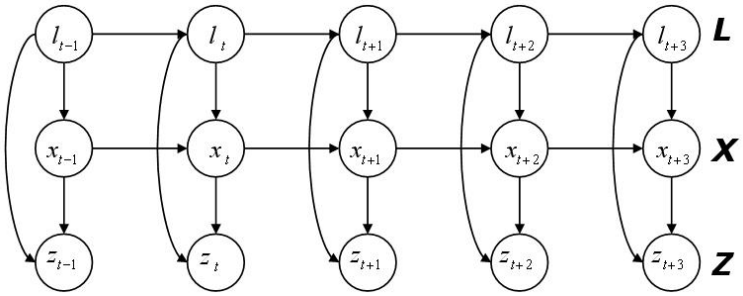


Figure 4: Switching linear dynamic systems (SLDS)

In addition to a set of LDS models M, we specify two additional parameters: a multinomial distribution $\pi(l_1)$ over the initial label l_1 and an $n \times n$ transition matrix B that defines the switching behavior between the n distinct LDS models. In summary, a standard SLDS model is defined by the tuple $\Theta \triangleq \left\{\pi, B, M \triangleq \{M_l | 1 \leq l \leq n\}\right\}$.

Switching linear dynamic system (SLDS) models have been studied in a variety of research communities ranging from computer vision [25, 24, 26, 19, 5, 31], computer graphics [31, 33, 27], tracking [4], signal processing [9, 8] and speech recognition [28], to econometrics [14], visualization [34], machine learning [16, 11, 20, 21, 22, 12], control systems [32] and statistics [30]. While one can find several versions of SLDS in the literature, our work is most closely related to the model structure and extensions described in [25, 24, 26, 20, 21, 22].

1.3.3 Learning and Inference in SLDS

The EM algorithm [6] can be used to obtain the maximum-likelihood parameters $\hat{\Theta}$. The hidden variables in EM are the label sequence L and the state sequence X. Given the observation data Z, EM iterates between the two steps:

- E-step : Inference to obtain the posterior distribution

$$f^i(L, X) \triangleq P(L, X | Z, \Theta^i) \tag{3}$$

  over the hidden variables L and X, using a current guess for the SLDS parameters Θ^i.

- M-step : maximize the expected log-likelihoods with respect to Θ:

$$\Theta^{i+1} \leftarrow \underset{\Theta}{\text{argmax}} \ \langle \log P(L, X, Z | \Theta) \rangle_{f^i(L,X)} \tag{4}$$

Above, $\langle \cdot \rangle_W$ denotes the expectation of a function $(\cdot)$ under a distribution W. Note that the exact E-step in Eq.3 is proved to be intractable [15]. Thus, there have been research efforts to derive efficient approximate inference methods, e.g.,GPB2 [3, 5], pseudo-EM algorithm [30], a variational approximation [11, 24, 26, 22], an approximate Viterbi method [25, 24, 26], expectation propagation [34], iterative Monte Carlo methods [8], sequential Monte Carlo methods [9], Gibbs sampling [28] and Data-Driven MCMC [20].

2 Contributions and Related Work

In this paper, we address an important limitation of the standard SLDS model : limitations in duration modeling. We propose a novel solution to address this problem : a segmental SLDS model which improves the limited duration modeling power of standard SLDSs. In the sections below we discuss our approach along with the related work that provided the inspiration for them.

2.1 Improved Duration modeling

The duration modeling capabilities of a standard SLDS are limited by the Markov assumption which is imposed upon the transitions at the discrete switching states. As a consequence of Markov assumption, the probability of remaining in a given switching state follows a geometric distribution :

$$P(d) \quad = \quad a^{d-1}(1-a) \tag{5}$$

Above, d denotes the duration of a given switching state and a denotes Markov transition probability to make a self-transition which has a value between zero and one. As a consequence, a duration of one time-step come to possess the largest probability mass.

In contrast, many natural temporal phenomena exhibit patterns of regularity in the duration for which a given model or regime is active. In such cases the standard SLDS model would be inappropriate to effectively encode the regularity of durations in data. A honey bee dance is an example: a dancer bee will attempt to stay in the waggle regime for a certain duration to effectively communicate a message. In such cases, it is clear that the actual duration diverges from a geometric distribution.

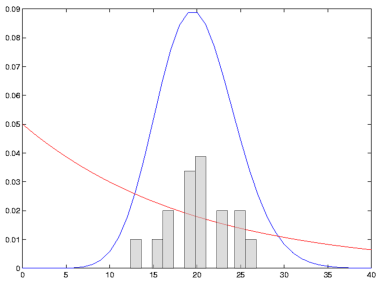


Figure 5: A realistic Gaussian (blue) and a limited geometric duration model (red). Models are learned from data.

For example, we learned a duration model for the waggle phase using a realistic Gaussian density and a conventional geometric distribution from one of the manually labeled dance sequences depicted in Figure 10. Figure 5 shows the learned geometric (red) and Gaussian (blue) distributions for comparison. It can be observed that the learned geometric duration model does not exhibit any pattern of regularity in durations. Hence, standard SLDS models are inappropriate for data which exhibits temporal patterns that deviate from geometric distributions.

2.2 Related Work

The limitation of a geometric distribution was also previously addressed by the HMM communities, and HMM models with enhanced duration capabilities were introduced [10, 17, 23]. HMMs has been widely studied by the speech recognition and the machine learning communities to enhance its duration modeling capabilities. The variable duration HMM (VD-HMM) was introduced in [10]: state durations are modeled explicitly in a

variety of PDF forms. Later, a different parameterization of the state durations was introduced where the state transition probabilities are modeled as functions of time, which are referred to as non-stationary HMMs (NS-HMM) [17]. It has since been shown that the VD-HMM and the NS-HMM are duals [7]. Ostendorf et.al. provides an excellent discussion on segmental HMMs [23].

We adopt similar ideas to arrive at SLDS models with enhanced duration modeling. The resulting segmental SLDS model is described in Section. 3.

3 Segmental SLDS

We introduce the segmental SLDS (S-SLDS) model, which improves on the standard SLDS model by relaxing the Markov assumption at a time-step level to a coarser *segment level*. The development of S-SLDS model is motivated by the regularity in durations being exhibited by the honey bee dances. As discussed in Section 2.1, a dancer bee will attempt to stay in the waggle regime for a certain duration to effectively communicate a message. In such a case, the geometric distribution induced in standard SLDSs is not an appropriate choice to model the duration patterns. Fig. 5 shows that a geometric distribution accords the highest probability on the duration of only one time step. As a result, the inference in standard SLDSs is susceptible to over-segmentation due to the noise in data.

In an S-SLDS, the durations are first modeled explicitly and then non-stationary duration functions are derived from them. Both of them are learned from data. As a consequence, the S-SLDS model has more descriptive power in modeling duration, and more robust inference capabilities than the standard SLDS. Nonetheless, we show that one can always convert a learned S-SLDS model into an equivalent standard SLDS, operating in a different label space. Hence, as a significant advantage we are able to reuse the large array of approximate inference and learning techniques developed for SLDSs.

3.1 Conceptual view on the generative process of S-SLDS

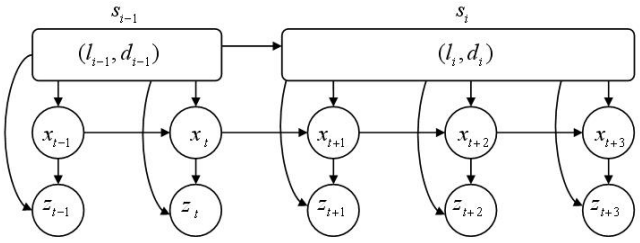


Figure 6: A schematic sketch of an S-SLDS with explicit duration models.

Conceptually, in an S-SLDS, we deal with segments of finite duration, i.e. each segment $s_i \triangleq (l_i, d_i)$ is described by a tuple of label l_i and duration d_i. Within each segment a fixed LDS model M_l is used to generate the continuous state sequence for the duration d_i. Similar to SLDSs, we take an S-SLDS to have an initial distribution $\pi(l_1)$ over the initial label l_1 of the first segment s_1, and an $n \times n$ semi Markov label transition matrix $\tilde{B}$ that defines the switching behavior between the segment labels. The tilde denotes that the matrix is a semi-Markov transition matrix. Additionally, however, we associate each label l with a fixed *duration model* D_l, represented as a multinomial. We denote the set of n duration models as $D \triangleq \{D_l(d) | 1 \leq l \leq n\}$, and refer to them in what follows as *explicit duration models*. In summary, an S-SLDS is defined by a tuple $\Theta \triangleq \left\{\pi, \tilde{B}, D \triangleq \{D_l | l = 1..n\}, M \triangleq \{M_l | l = 1..n\}\right\}$.

A schematic depiction of an S-SLDS is illustrated in Fig.6. The top chain in the figure is a series of segments where each segment is depicted as a rounded box. In the model, the current segment $s_i \triangleq (l_i, d_i)$ generates a next segment s_{i+1} in the following manner: first, the current label l_i generates the next label l_{i+1} based on the label transition matrix $\tilde{B}$; then, the next duration d_{i+1} is generated from the duration model for the label l_{i+1}, i.e. $d_{i+1} \sim D_{l_{i+1}}(d)$. The dynamics for the continuous hidden states and observations are identical to a standard SLDS : a segment s_i evolves the continuous hidden states X with a corresponding LDS model M_{l_i} for the duration d_i, then the observations Z are generated given the labels L and the continuous states X.

3.2 Graphical Representation of S-SLDS

In this section we present a graphical representation of S-SLDSs, transforming the conceptual generative model described in Section 3.1 into a concrete model that uses conventional model switching at every time-step. To maintain the same duration semantics, we introduce *counter variables* $C \triangleq \{c_t | 1 \leq t \leq T\}$. The resulting graphical model of S-SLDS is illustrated in Fig.7, and is identical to the graphical model of an SLDS in Fig.3(b), but with additional top-chain representing a series of counter variables C.

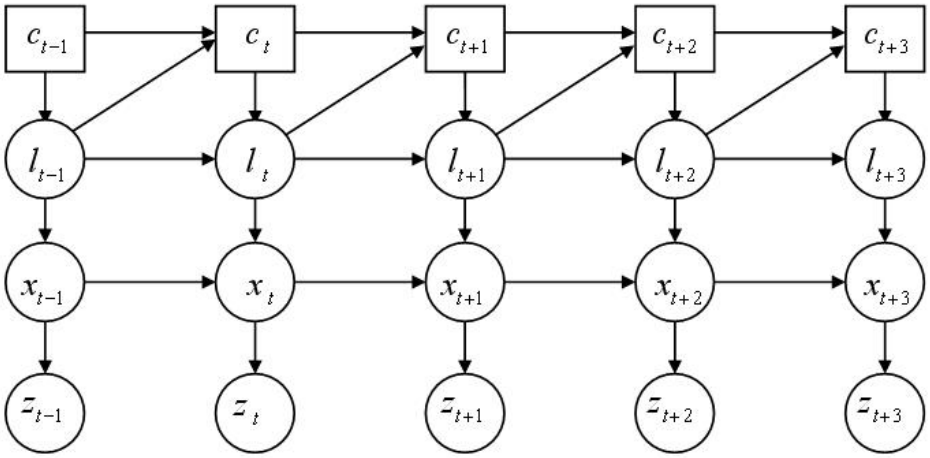


Figure 7: Graphical representation of an S-SLDS

The counter chain C maintains an incremental counter which evolves based on a set of *non-stationary transition functions* (NSTFs) $U \triangleq \{U_l(c) | 1 \leq l \leq n\}$. An NSTF U_l for the current label l_t defines the conditional dependency of the next counter variable c_{t+1} given the current counter variable c_t and the label l_t :

$$U_l(c_t) \quad = \quad P(c_{t+1} | c_t, l)$$

The system can either increment the counter, i.e. $c_{t+1} \leftarrow c_t + 1$, or reset it to one, i.e. $c_{t+1} \leftarrow 1$. If the counter variable c_{t+1} is reset, then a label transition occurs, i.e. a new segment is initialized. A new label l_{t+1} is chosen based on the label transition matrix B. If the counter simply increments, then the new label is set to be the current label l_t, i.e. $l_{t+1} \leftarrow l_t$.

While the explicit duration models D introduced in Section 3.1 are more understandable and readily obtained from the labeled data, it is necessary to transform the explicit duration models D into an equivalent

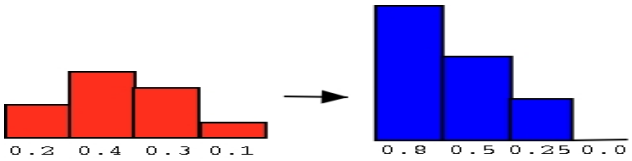


Figure 8: Evaluating an NSTF (blue) from an explicit duration model (red).

NSTFs U to incorporate the knowledge in durations into a framework based on graphical models. To do this, we can observe that the explicit duration models D and the NSTFs U are analogous to the duration models of VD-HMMs [10] and NS-HMMs [17] respectively. Hence, we can exploit the duality between the VD-HMMs and NS-HMMs, which appeared in [7]. The equivalent NSTFs U are exactly evaluated from the explicit duration models D as follows :

$$U_l(c_t) \quad = \quad 1 - \left(D_l(c_t) / \sum_{d=c_t}^{D_l^{max}} D_l(c_t) \right) \tag{6}$$

Above, D_l^{max} denotes the maximum duration allowed for the l th model. Intuitively, the latter composite term on the r.h.s. denotes the probability to reset the counter variable c_{t+1}. It represents the ratio of the probability of current duration c_t over the sum of durations equal or greater than c_t in the corresponding duration model D_l.

In summary, an S-SLDS model is completely defined by a tuple $\Theta \triangleq \left\{ \pi, \tilde{B}, U \triangleq \{U_l | 1 \leq l \leq n\}, M \triangleq \{M_l | 1 \leq l \leq n\} \right\}$ where the NSTFs U are obtained from the explicit duration models D.

3.3 Learning in Segmental SLDS

Learning in S-SLDS is analogous to learning in SLDS, using EM. The initial distribution π, and LDS model parameters M are learned in exactly the same manner as in SLDS. However, it is necessary to learn the additional duration models D and the semi-Markov transition matrix $\tilde{B}$. These two additional model parameters only influence the label sequence L, and hence the ML estimates of these two parameters can be evaluated from a segmental representation of the label sequence L, i.e., $L = \cup_{j=1}^{|s|} s_j$. The specific functional forms of ML estimation depends on the choice of duration models. An example is demonstrated in Section 2.1 where we learn the duration models from the honey bee dance sequences.

3.4 Inference in Segmental SLDSs

Below we demonstrate that an S-SLDS can be always converted to an equivalent SLDS. This is an important advantage as it allows us to readily reuse the large array of approximate inference algorithms discussed in Section 1.3.3. In other words, the inference in S-SLDS is identical to that of the standard SLDS, simply with additional conversion from an S-SLDS to its corresponding SLDS.

The overall idea of inference is depicted in Figure 9. In step 1, we convert an S-SLDS model into an equivalent SLDS model. Then, we perform step 2 (inference) using any of the approximate inference algorithms for the standard SLDSs. Once the parameters of the equivalent standard SLDS are learned via EM, the obtained SLDS model is converted back to S-SLDS model and the inference in S-SLDS concludes.

The model conversion from an S-SLDSs to an equivalent SLDS is possible by applying the standard technique of merging multiple discrete variables into meta variables. Specifically, all possible pairs of a label l_t and a counter value c_t are merged and form a set of "lc" variables where $\mathcal{LC} \triangleq \{(l, c_i) | 1 \leq l \leq n, 1 \leq c_i \leq$

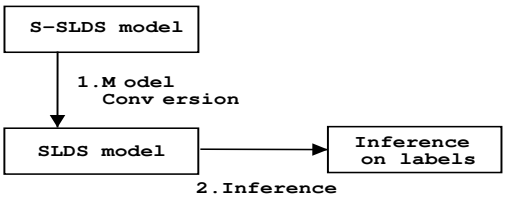


Figure 9: Inference in S-SLDS.

$D_l^{max}\}$. To obtain a complete SLDS model, an equivalent $n' \times n'$ transition matrix B' where $n' \triangleq \sum_{l=1}^{n} D_l^{max}$ is constructed from the semi-Markov transition matrix $\tilde{B}$ and the NSTFs U, as follows :

$$B'_{(l_i,c_i),(l_j,c_j)} = \begin{cases} U_{l_i}(c_i) & \text{increment} \\ \tilde{B}_{l_i,l_j}(1 - U_{l_i}(c_i)) & \text{reset} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

In Eq. 7, the three cases differ as follows : (increment) $l_i = l_j$ and $c_j = c_i + 1$. (reset) $c_j = 1$. (otherwise) all other cases. In addition, the initial label distribution π' for the equivalent SLDS can similarly be constructed from the S-SLDS initial distribution π :

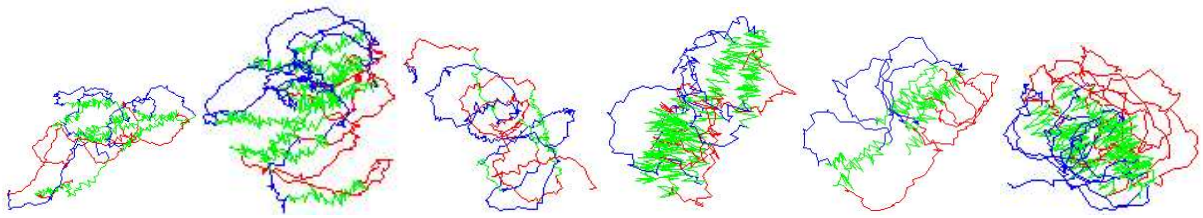
$$\pi'(l_i, c_i) = \begin{cases} \pi(l_i) & \text{if } c_i = 1 \\ 0 & \text{otherwise} \end{cases}$$

Nonetheless, it is important to note that the naive reuse of the learning and inference algorithms for SLDS to S-SLDS may induce substantial increase in computational overhead. The issues regarding computational considerations are presented in the next section.

3.5 Computational Considerations

As mentioned in Section 3.4, an equivalent SLDS can always be constructed from an arbitrary S-SLDS. However, if we reuse the original learning and inference algorithms for SLDSs in a naive manner the cost of inference will be on the order of $O(TD_{max}^2|L|^2)$ for S-SLDSs, while it takes $O(T|L|^2)$ for SLDSs without duration models, where $D_{max} \triangleq \max\{D_l^{max}\}_{l=1}^{n}$, i.e. the number of all meta variables. Thus, there is a considerable computational overhead, by a factor of $O(D_{max}^2)$. This increased asymptotic running time overhead applies to the approximate inference algorithms with HMM-type components in general, e.g. approximate Viterbi [26] and a variational method [26, 11, 22], as they require the computations between all possible state pairs from the previous time-step to the next time-step.

Nonetheless, we can still maintain linear efficiency w.r.t. the maximum duration D_{max} by exploiting the sparseness of the constructed SLDS matrix B'. It can be observed from Eq.7 that the SLDS matrix B' is mostly sparse, i.e. only a few transitions are allowed between the states in $\mathcal{LC}$. In fact, only $|L| + 1$ transitions allowed for every lc state. The allowable transitions include the resets to $|L|$ labels and one increment transition. Hence, we can achieve an overall performance of $O(TD_{max}|L|^2)$ via exploiting this fact, which results in reduced overhead by a factor of $O(D_{max})$. The number is derived from the fact that there are total $O(D_{max}|L|)$ states at time $t-1$, and the number of transitions allowed for each state to time t reduces to $O(|L|)$ from $O(D_{max}|L|)$. This reduction in complexity allows us to incorporate a duration model with a large D_{max} and maintain computational efficiency. As a consequence, we can adopt the more powerful duration modeling capabilities of an S-SLDS at the cost of a modest complexity increase over the standard SLDS model.



(1) (2) (3) (4) (5) (6)

Figure 10: Bee dance sequences used in the experiments. Each dance trajectory is the output of a vision-based tracker.

4 Experimental Results

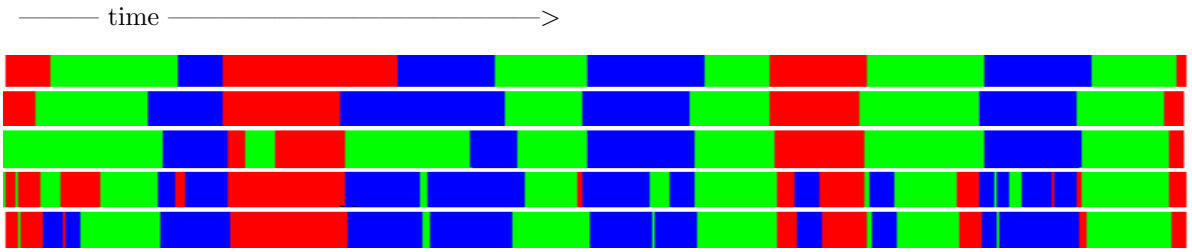
The experimental results below show that S-SLDSs provide superior behavior interpretation capabilities over standard SLDSs.

For the experiments, we used total six real-world dancer honeybee tracks, which are shown in Fig.10. They were obtained using a vision-based tracker [13] from videos of bees dancing inside the beehive. We re-parametrize the output of the tracker as a time-series sequence of vectors $z_t = [x_t, y_t, \cos(\theta_t), \sin(\theta_t)]^T$ where x_t,y_t and θ_t respectively denote the 2D coordinates and the heading angle at time t. The red, green and blue colors in Fig.10 represent manually marked right-turn, waggle and left-turn phases. The lengths of the sequences were 1058, 1125, 1054, 757, 609 and 814 frames, respectively. The ground-truth sequences are labeled manually for purposes of comparison and learning. The dimensionality of the continuous hidden states was set to four.

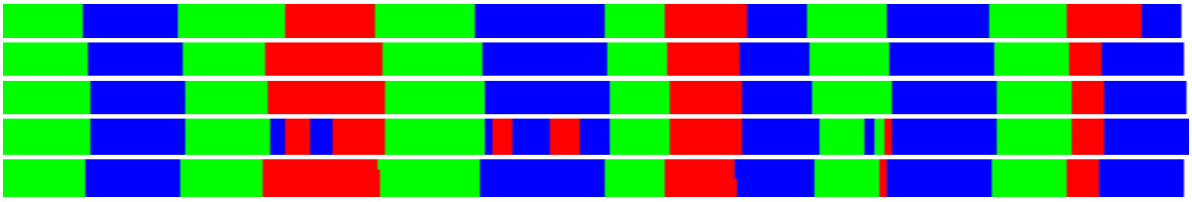
Given the relative difficulty of obtaining this data, which has to be labeled manually to allow for a ground-truth comparison, a leave-one-out strategy is adopted. The parameters are learned from five out of six datasets, and the learned model is applied to the left-out dataset to perform inference on the label sequence. Both standard SLDS parameters and S-SLDS parameters are learned from the training data. For the S-SLDS we use a Poisson distribution model for the duration, and converted that into the NSTFs used in the S-SLDSs implementation.

For inference, an approximate Viterbi method [26] and a structured variational inference method [26, 22] are used for both a standard SLDS and a segmental SLDS. The variational method was initialized using the approximate Viterbi results. The label inference results on sequence 1 and sequence 2 are shown in Fig.11. The five color strips in each figure respectively show the ground-truth, S-SLDS Viterbi, S-SLDS variational, SLDS Viterbi, SLDS variational method labels, from top to bottom. The x-axis represents time flow and the color is the corresponding label at that corresponding video frame. On other four datasets, the S-SLDS results were superior or comparable to SLDS results.

From the experimental results, it can be observed that the inference results of S-SLDSs agree well with the ground truth, and S-SLDSs yield more accurate results than standard SLDS model. In particular, most of the over-segmentations in standard SLDSs (4th and 5th from the top) that occur in noisy segments, disappear in the S-SLDS results (2nd and 3rd from the top). The superior interpretation capabilities of S-SLDSs over SLDSs demonstrate the benefits of incorporating accurate duration models that can be learned from the data.



(a) Sequence 1



(b) Sequence 2

Figure 11: Label inference results on two tracker sequences: Ground-truth, S-SLDS Viterbi, S-SLDS variational, SLDS Viterbi, SLDS variational method results from the top to the bottom.

5 Conclusion

We presented a segmental switching linear dynamic systems (S-SLDS), which incorporates duration models that can be learned from the data. It overcomes the limitations of the simple geometric duration models induced in standard SLDSs. We introduced the graphical representation of S-SLDSs and the associated non-stationary transition functions. The concrete formulation of the S-SLDS is derived by incorporating additional counter variables to the standard SLDS model.

An S-SLDS can be converted to an equivalent SLDS by creating meta states, and consequently a large array of approximate inference and learning algorithms for standard SLDSs can be readily adopted in the new S-SLDS framework. Additionally, computational efficiency is maintained by exploiting the sparse structure of the resulting transition matrix. In summary, the proposed S-SLDS framework provides more powerful duration modeling capabilities over the standard SLDSs at a modest cost, and does not necessitate developing new inference and learning algorithms. Its benefits were experimentally validated by means of the bee behavior recognition task, using both approximate Viterbi and a variational inference methods.

References

[1] T. Balch, F. Dellaert, A. Feldman, A. Guillory, C. Isbell, Z. Khan, A. Stein, and H. Wilde. How A.I. and multi-robot systems research will accelerate our understanding of social animal behavior. *Proceedings of IEEE*, 2005. Accepted for publication.

[2] T. Balch, Z. Khan, and M. Veloso. Automatically tracking and analyzing the behavior of live insect colonies. In *Proc. Autonomous Agents 2001*, Montreal, 2001.

[3] Y. Bar-Shalom and X. Li. *Estimation and Tracking: principles, techniques and software*. Artech House, Boston, London, 1993.

[4] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data-association. *Automatica*, 11:451–460, 1975.

[5] Christoph Bregler. Learning and recognizing human dynamics in video sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1997.

[6] 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*, 39(1):1–38, 1977.

[7] P. M. Djuric and J-H. Chun. An MCMC sampling approach to estimation of nonstationary hidden Markov modles. *IEEE Trans. Signal Processing*, 50(5):1113–1123, 2002.

[8] A. Doucet and C. Andrieu. Iterative algorithms for state estimation of jump markov linear systems. *IEEE Trans. Signal Processing*, 49(6), 2001.

[9] A. Doucet, N. J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Trans. Signal Processing*, 49(3), 2001.

[10] J. Ferguson. Variable duration models for speech. In *Symposium on the Aplication of HMMs to Text and Speech*, 1980.

[11] Z. Ghahramani and G. E. Hinton. Variational learning for switching state-space models. *Neural Computation*, 12(4):963–996, 1998.

[12] A. Howard and T. Jebara. Dynamical systems trees. In *Conf. on Uncertainty in Artificial Intelligence*, pages 260–267, Banff, Canada, 2004.

[13] Z. Khan, T. Balch, and F. Dellaert. A Rao-Blackwellized particle filter for EigenTracking. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[14] C.-J. Kim. Dynamic linear models with Markov-switching. *Journal of Econometrics*, 60, 1994.

[15] U. Lerner and R. Parr. Inference in hybrid networks: Theoretical limits and practical algorithms. In *Proc. 17th Annual Conference on Uncertainty in Artificial Intelligence (UAI-01)*, pages 310–318, Seattle, WA, 2001.

[16] U. Lerner, R. Parr, D. Koller, and G. Biswas. Bayesian fault detection and diagnosis in dynamic systems. In *Proc. AAAI*, Austin, TX, 2000.

[17] S. E. Levinson. Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language*, 1(1):29–45, 1990.

[18] P. Maybeck. *Stochastic Models, Estimation and Control*, volume 1. Academic Press, New York, 1979.

[19] B. North, A. Blake, M. Isard, and J. Rottscher. Learning and classification of complex dynamics. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(9):1016–1034, 2000.

[20] S. M. Oh, J. M. Rehg, T. Balch, and F. Dellaert. Data-driven MCMC for learning and inference in switching linear dynamic systems. In *AAAI Nat. Conf. on Artificial Intelligence*, 2005.

[21] S. M. Oh, J. M. Rehg, T. Balch, and F. Dellaert. Learning and Inference in Parametric Switching Linear Dynamic Systems. In *Intl. Conf. on Computer Vision (ICCV)*, 2005.

[22] S.M. Oh, A. Ranganathan, J.M. Rehg, and F. Dellaert. A variational inference method for switching linear dynamic systems. Technical Report GIT-GVU-05-16, GVU, College of Computing, 2005.

[23] M. Ostendorf, V. V. Digalakis, and O. A. Kimball. From hmm's to segment models : A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):360–378, 1996.

[24] V. Pavlović and J.M. Rehg. Impact of dynamic model learning on classification of human motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2000.

[25] V. Pavlović, J.M. Rehg, T.-J. Cham, and K. Murphy. A dynamic Bayesian network approach to figure tracking using learned dynamic models. In *Intl. Conf. on Computer Vision (ICCV)*, 1999.

[26] V. Pavlović, J.M. Rehg, and J. MacCormick. Learning switching linear models of human motion. In *Advances in Neural Information Processing Systems (NIPS)*, 2000.

[27] L. Ren, A. Patrick, A. Efros, J. Hodgins, and J. M. Rehg. A data-driven approach to quantifying natural human motion. *ACM Trans. on Graphics, Special Issue: Proc. of 2005 SIGGRAPH Conf.*, 2005.

[28] A-V.I. Rosti and M.J.F. Gales. Rao-blackwellised Gibbs sampling for switching linear dynamical systems. In *Intl. Conf. Acoust., Speech, and Signal Proc. (ICASSP)*, volume 1, pages 809–812, 2004.

[29] S. Roweis and Z. Ghahramani. A unifying review of Linear Gaussian Models. *Neural Computation*, 11(2):305–345, 1999.

[30] R.H. Shumway and D.S. Stoffer. Dynamic linear models with switching. *Journal of the American Statistical Association*, 86:763–769, 1992.

[31] S. Soatto, G. Doretto, and Y.N. Wu. Dynamic Textures. In *Intl. Conf. on Computer Vision (ICCV)*, pages 439–446, 2001.

[32] R. Vidal, A. Chiuso, and S. Soatto. Observability and identifiability of jump linear systems. In *Proceedings of IEEE Conference on Decision and Control*, 2002.

[33] Y.Li, T.Wang, and H-Y. Shum. Motion texture : A two-level statistical model for character motion synthesis. In *SIGGRAPH*, 2002.

[34] O. Zoeter and T. Heskes. Hierarchical visualization of time-series data using switching linear dynamical systems. *IEEE Trans. Pattern Anal. Machine Intell.*, 25(10):1202–1215, October 2003.