Interval-Based Linear Hybrid Dynamical System for Modeling Cross-Media Timing Structures in Multimedia Signals

Hiroaki Kawashima and Takashi Matsuyama
Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo, Kyoto 606-8501, Japan
{kawashima,tm}@i.kyoto-u.ac.jp

Abstract

In this paper, we propose a computational scheme named an interval-based linear hybrid dynamical system (ILHDS) to represent complex dynamic events based on temporal intervals, each of which is characterized by linear dynamics and its duration. We then propose a cross-media timing-structure model to represent dynamic structures among multiple media signals based on the relation of temporal intervals described by multiple ILHDSs. To evaluate the proposed scheme, we conducted experiments on media conversion that generates lip video from an input audio signal.

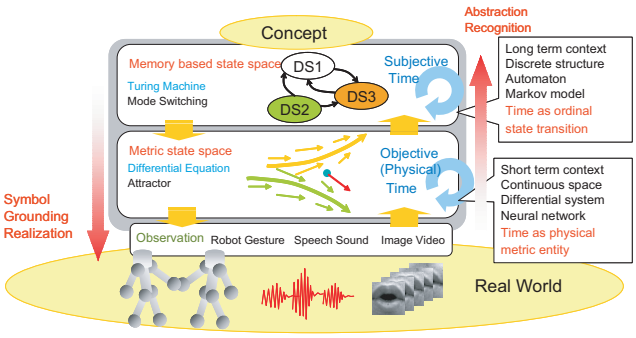

Figure 1. Architecture of hybrid dynamical systems.

1. Introduction

Understanding the meaning of user commands and presenting appropriate information to a user is one of the primary objectives of human-machine interaction systems. Most of the existing approaches, therefore, set the goal to realize interaction systems that understand semantic information specified by a user and generate attractive presentation to a user using multimedia data such as text, graphs, pictures, video, sound, and so on.

While such multimedia interaction systems are important, users sometimes feel frustration when the systems get out of human interaction protocols. That is, the systems often ignore dynamic features such as acceleration patterns, pause lengths, tempo speed, and rhythms, which convey rich nonverbal and non-semantic information in human communication.

In this paper, we attempt to model such dynamic features or temporal structures in verbal and nonverbal communication based on a novel computational model, named an interval-based linear hybrid dynamical system (ILHDS). A hybrid dynamical system is the integration of two types of dynamical systems: one described by differential equations, which is suitable for describing physical phenomena (consider time as physical metric entity), and a discrete-event system, which is suitable for describing human subjective or intellectual activities (consider time as ordinal state transition) (Figure 1).

We developed ILHDS based on the following rationale. Firstly, we assume that a complex human behavior consists of dynamic primitives, which are often referred to as motion elements, movemes, visemes, and so on. For example, a cyclic lip motion can be described by a cyclic sequence of simple lip motions such as “open”, “close”, and “remain closed”. Once the set of dynamic primitives is determined, a complex behavior can be partitioned into “temporal intervals”, each of which is characterized by a dynamic primitive and its temporal duration.

Secondly, we assume that not only temporal orders of motion elements but also their duration lengths or temporal differences among beginning and ending timing of the temporal intervals convey rich information in human communication.

Based on the assumptions above, we proposed ILHDS for modeling dynamic events in terms of temporal intervals. The system has a two-layer architecture consisting of a finite state automaton and a set of linear dynamical systems. In this architecture, each linear dynamical system represents

the dynamics of a motion primitive and corresponds one to one to a discrete state of the automaton. In other words, the automaton controls the activation order and timing of the linear dynamical systems. Thus, ILHDS can model and generate multimedia signals that represent complex human behaviors.

In spite of the flexibility of the systems, the learning process has a difficulty due to its paradoxical nature; that is, it requires us to solve temporal segmentation and system identification problems simultaneously. We therefore propose a two-step learning method as we describe in Section 4.

Applying multiple ILHDSs to human communication, we can successfully extract dynamic features of the behaviors based on relations of temporal intervals. In this paper, we focus on modeling cross-media timing structures and analyzed synchronization/delay mechanisms between mouth motion and speech utterance. Thanks to the model, we successfully generate lip video from an input audio signal (see Section 5 for details).

2. Related Work

Segment models [6] have been proposed in speech recognition fields as the unified model of segmental HMMs [4]. This model uses the *segment* as a descriptor, which has a duration distribution. The segment represents a temporal region in which one of the states is activated, and the total system represents phonemes and subwords as a sequence of the segments.

While ILHDS also explicitly represents duration, the concept of the system is different from the segment models because we concentrate on modeling temporal structure among multiple events rather than only the duration lengths of events. We use the term *intervals* instead of segments because our motivation is bringing Allen's temporal interval logic [1], which exploits 13 topological relations between two intervals (e.g., meets, during, starts with, etc.), into the class of hybrid systems.

Once the intervals are explicitly defined, we can fabricate more flexible models to represent cross-modal temporal relations such as timing among concurrent dynamics appeared in man-machine interaction.

3. Interval-Based Linear Hybrid Dynamical System

3.1. System Architecture

ILHDS has a two-layer architecture (Figure 2). The first layer (the top of Figure 2) records a finite state automaton as a discrete-event system that models stochastic transitions between discrete events. The second layer (the second top

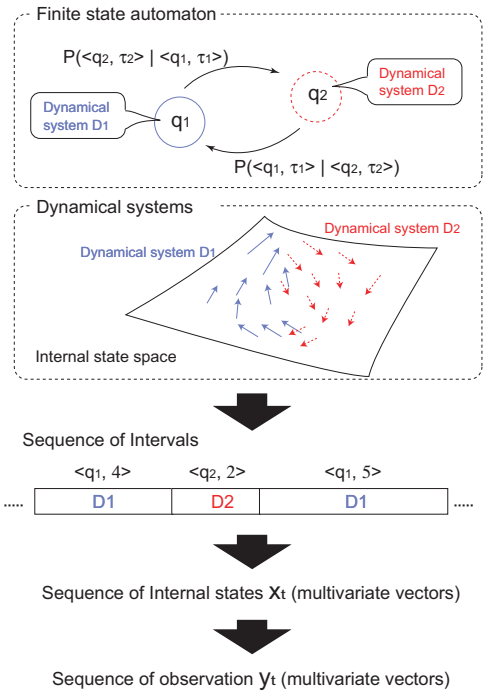

Figure 2. Interval-based linear hybrid dynamical system for modeling a single signal

of Figure 2) consists of a set of linear dynamical systems $\mathcal{D} = \{D_1, ..., D_N\}$. To integrate these two layers, we introduce *intervals* (the bottom of Figure 2): each interval is described by $< q_i, \tau >$, where q_i denotes a state in the automaton and τ the physical temporal duration of the interval. Each state in the automaton corresponds to a unique linear dynamical system recorded at the second layer: q_i denotes the label of the corresponding linear dynamical system as well as a state in the automaton. As a result, the automaton has a discrete state set $\mathcal{Q} = \{q_1, ..., q_N\}$. Note that multiple different intervals can correspond to the same state in the automaton; that is, their dynamics are described/controlled by the same linear dynamical system.

When a temporal sequence of observed signal data $y_t \in \mathbf{R}^m$, is given, it is first transformed into a sequence of internal states $x_t \in \mathbf{R}^n$. Then, that sequence is partitioned into a sequence of intervals. That is, the internal state sequence is partitioned into a group of sub-sequences so that the dynamic state variation in each sub-sequence can be described by a linear dynamical system, which is denoted by q_i recorded in the interval covering that sub-sequence.

Once ILHDS has been constructed by learning as will be described in Section 4, it can generate a multivariate signal sequence by activating the automaton: the activated automaton first generates a sequence of intervals, each of which then generates a signal sequence based on its corresponding linear dynamical system (the bottom of Fig-

ure 2). Note that the activation timing and period of the linear dynamical system are controlled by the duration length recorded in the interval.

3.2. Linear Dynamical Systems

The state transition of dynamical system D_i in the internal state space, and the mapping from the internal state space to the observation space is modeled by the following linear equations:

$$\begin{aligned} x_t &= F^{(i)}x_{t-1} + g^{(i)} + \omega_t^{(i)} \\ y_t &= Hx_t + \upsilon_t, \end{aligned} \quad (1)$$

where $F^{(i)}$ is a transition matrix and $g^{(i)}$ is a bias vector. H is an observation matrix that defines linear projection from the internal state space to the observation space. $\omega^{(i)}$ and υ is the process noise and the observation noise, which are modeled by Gaussian distributions respectively. Note that each dynamical system is defined by $F^{(i)}$, $g^{(i)}$, and $\omega^{(i)}$.

3.3. Interval-Based State Transition

In this section, we define the transition of discrete states in the automaton that generate interval sequences. Here, we assume first-order Markov property for the generated intervals. A major difference from conventional state transition models, such as hidden Markov models, is that the automaton models the correlation between duration lengths of adjacent intervals as well as the transition of discrete states.

Let $\mathcal{I} = I_1, ..., I_K$ be an interval sequence generated by the automaton. To simplify the model, we assume that adjacent intervals have no temporal gaps or overlaps. Here, the interval I_k depends only on the previous interval I_{k-1} because of the Markov property assumption. Then, the Markov process of intervals can be modeled by the following conditional probability:

$$P(I_k =< q_j, \tau > | I_{k-1} =< q_i, \tau_p >),$$

which denotes the probability that interval $< q_j, \tau >$ occurs after interval $< q_i, \tau_p >$.

The computation of probability $P(I_k =< q_j, \tau > | I_{k-1} =< q_i, \tau_p >)$ requires a large parameter set, which does not only increase computational cost but also incur the problem of over-fitting during a training phase. We therefore use a parametric model for the duration length distribution. That is, for each state transition in the automaton, we record $P(q_j|q_i)$ together with a parametric distribution for $P(\tau|\tau_p, q_i, q_j)$.

4. Learning Process for ILHDS

Let us assume that only a group of multivariate signal sequences is given as training data. Then, in most of hybrid dynamical systems, the system identification process that estimates system parameters becomes difficult because of its paradoxical nature. That is, the system consists of a set of subsystems (in our case, linear dynamical systems) and the parameter estimation of each subsystem requires partitioned training data to be modeled by that subsystem, while the segmentation process of training data requires a set of identified subsystems. Moreover, the number of subsystems is also unknown in general.

The expectation-maximization (EM) algorithm is one of the most common approaches to solve this kind of paradoxical problems. The algorithm estimates parameters based on the iterative calculation. In each step, the algorithm conducts model fitting to training data using the model parameters that were updated in the previous step. Then, the parameters are updated based on the result of the current model fitting process.

However, the EM algorithm-based parameter estimation method involves two problems: (1) initialization of the EM algorithm, and (2) estimation of the number of subsystems.

To solve the difficulties in the learning process, we divide the estimation process into two steps: clustering of dynamical systems to estimate a set of required dynamical systems and parameter refinement of the overall system.

We here assume that internal-state sequences have been estimated from observation sequences; that is, an observation matrix H and distribution parameters of observation noise υ have been estimated based on prior knowledge or system-identification techniques [7].

[Step 1] Clustering of Dynamical Systems. The first step is a clustering process that finds a set of dynamical systems required to describe training data: the number of the systems and their parameters. This step employs a typical data sequence as training data. Then, an agglomerative hierarchical clustering is applied to the training data to estimate a set of dynamical systems required to model the data (Figure 3):

1. Partition the training sequence into a group of very short sub-sequences and estimate a dynamical system that can model each sub-sequence respectively.

2. Compute the distance between each pair of estimated dynamical systems.

3. Integrate the closest pair of dynamical systems: compute parameters of the integrated dynamical system based on such sub-sequences that were modeled by the pair of dynamical systems to be integrated.

4. Iterate the above integration process until the closest distance between a pair of dynamical systems becomes greater than a pre-specified value.

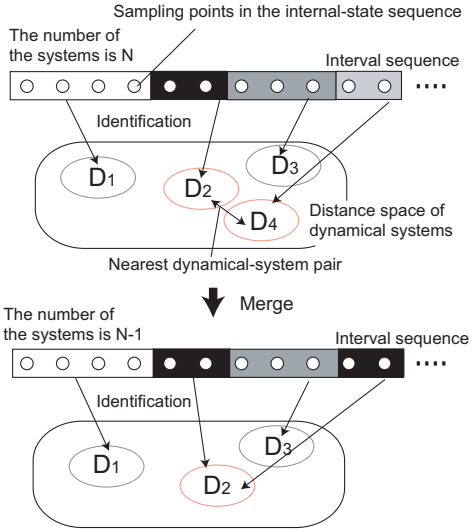

Figure 3. Clustering of dynamical systems.

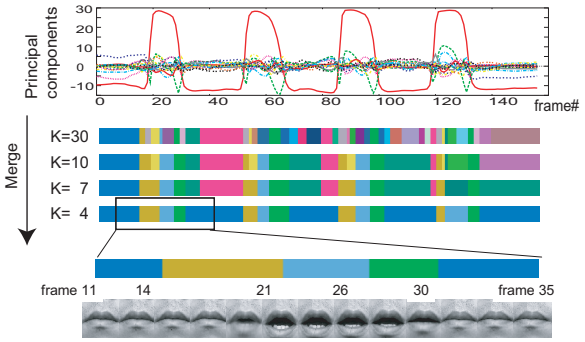

Figure 4. The result of lip motion segmentation using clustering of dynamical systems.

After this process, we get the number of required dynamical systems N and approximate parameters of the dynamical systems. Since this approach agglomerates not the input data but dynamical models, the method can be considered as one of model-based hierarchical clustering techniques [8].

Figure 4 shows the result of lip motion segmentation using the proposed clustering method. We see that the lip sequence is divided by four dynamics: "remain closed", "open", "remain open", and "close".

[Step 2] Refinement of the Parameters. The second step is a refinement process of the system parameters based on the EM algorithm. The process is applied to all training data, whereas the clustering process is applied to a selected typical training sequence. While the EM algorithm strongly depends on its initial parameters, the clustering step provides an initial parameter set that is relatively close to the optimum.

Once the system parameters have been identified, each sequence in the training data set can be described by a sequence of intervals respectively, which then is used to estimate parameters of the automaton. Firstly note that a set of discrete states have been determined uniquely from the set of dynamical systems obtained by the clustering process. Then, for each pair of discrete states, the transition probability and the duration length distribution associated with the state transition are computed. Thus, ILHDS is identified.

5. Modeling Cross-Media Timing Structures

5.1. Timing Structures in Multimedia Data

Measuring dynamic human actions such as speech and music performance with multiple sensors, we can obtain multimedia signal data. We human usually sense/feel cross-modal dynamic structures fabricated by multimedia signals such as synchronization and delay. For example, it is well-known that the simultaneity between auditory and visual patterns influences human perception.

The cross-modal timing structure is also important to realize multimedia systems such as human computer interfaces (e.g., audio-visual speech recognition systems [5]) and computer graphics techniques that generate some media signal from another (e.g., lip sync to input speech [2]).

Dynamic Bayesian networks, such as coupled hidden Markov models [5], are often used as media integration methods. These methods describe co-occurrence or temporal adjacency of states in different media data. While such methods enable us to represent short-term cross-media relations, they are not well suited to describe systematic and long-term cross-media relations. For example, an opening lip motion is strongly synchronized with an explosive sound /p/, while the lip motion is loosely synchronized with a vowel sound /e/.

To represent such systematic and long-term synchronization/delay and mutual dependency among multimedia signals, here we propose a novel model based on ILHDS. For each media signal sequence in multimedia data, we first apply ILHDS to obtain the interval sequence respectively. Then, by comparing intervals of different media signals, we construct a *cross-media timing-structure model*, which is a stochastic model to describe temporal structures across multimedia signals.

5.2 Modeling Cross-Media Timing Structures

Applying ILHDS to each media signal sequence in multimedia data, we obtain a group of interval sequences (the top in Figure 5). Let I_k be an interval of mode M_i in one of

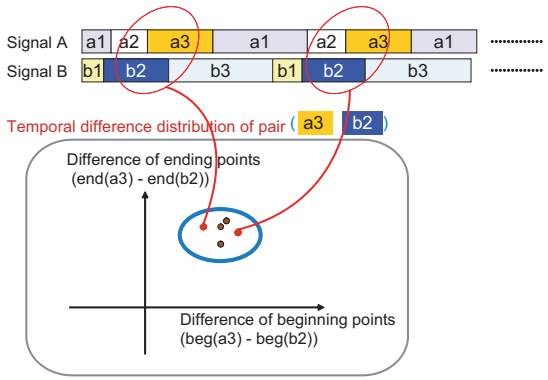

Figure 5. Learning timing-structure model.

the obtained interval sequences and $I'_{k'}$ an interval of mode M'_p in another interval sequence overlapping with I_k. Note that modes M_i and M'_p specify the linear dynamical systems that describe dynamics in intervals I_k and $I'_{k'}$ respectively. Let $b_k(e_k)$ and $b'_{k'}(e'_{k'})$ denote the beginning (ending) points of intervals I_k and $I'_{k'}$, respectively.

To model the cross-media relation between modes M_i and M'_p, we collect all pairs of overlapping intervals that satisfy the same temporal relation as that between I_k and $I'_{k'}$ and compute

$$P(b_k - b'_{k'}, e_k - e'_{k'} | m_k = M_i, m'_{k'} = M'_p), \quad (2)$$

where m_k and $m'_{k'}$ are the modes of interval I_k and $I'_{k'}$ (the bottom in Figure 5). We refer to this distribution as a *temporal difference distribution*. This distribution represents rich cross-media synchronization structures between a pair of different media signals. For example, if the peak of the distribution comes to the origin, the two modes tend to be synchronized each other at both beginning and ending points, while if $b_k - b_{k'}$ has large variance, the two modes loosely synchronized at their onset timing.

Note that we compute temporal difference distributions for all possible mode pairs and record them as fundamental characteristics of the cross-media timing structure of a given multimedia signal data. In addition to a set of such temporal difference distributions, we also model which mode pair tends to overlap with each other across different media (co-occurrence probabilities of modes), and which mode pair tends to appear in neighboring intervals in each media signal data (mode-transition probabilities). The cross-media timing structure is defined by these mutual dependency relations between modes.

5.3. Timing-Based Media Conversion

Once the cross-media timing-structure model is learned from simultaneously captured multimedia signal data, we can exploit the model for generating one media signal from another related media signal. The overall flow of the media conversion from signal S' to S is as follows:

1. A reference (input) signal S' is partitioned into an interval sequence $\mathcal{I}' = \{I'_1, ..., I'_{K'}\}$.
2. An interval sequence $\mathcal{I} = \{I_1, ..., I_K\}$ is generated from $\mathcal{I}'$ based on the cross-media timing-structure model. (K and K' is the number of intervals in $\mathcal{I}$ and $\mathcal{I}'$, and note that $K \neq K'$ in general.)
3. Signal S is generated from $\mathcal{I}$.

The key process of this media conversion lies in step 2. Let Φ be the cross-media timing-structure model that is learned in advance. Then, the problem of generating an interval sequence $\mathcal{I}$ from $\mathcal{I}'$ can be formulated by the following optimization:

$$\hat{\mathcal{I}} = \arg\max_{\mathcal{I}} P(\mathcal{I}|\mathcal{I}', \Phi). \quad (3)$$

In the equation above, we have to determine the number of intervals K and their properties, which can be described by triples $< b_k, e_k, m_k > (k = 1, ..., K)$, where $b_k, e_k \leq T$ and $m_k \in \mathcal{M}$. Here, T is the length of signal S', and $\mathcal{M}$ is the set of modes of intervals (i.e., set of linear dynamical systems fixed at the learning process). If we searched for all possible interval sequences $\{\mathcal{I}\}$, the computational cost would increase exponentially as T becomes longer. We therefore use a dynamic programming method to solve Equation (3), where we assume that generated intervals have no gaps or overlaps; thus, pairs $< e_k, m_k > (k = 1, ..., K)$ are required to be estimated under this assumption.

5.4. Experiments

To evaluate the descriptive power of the proposed cross-media timing structure model and the performance of the media conversion method, we conducted experiments on the lip video generation from an input audio signal.

Feature extraction. A continuous utterance of five vowels /a/,/i/,/u/,/e/,/o/ (in this order) was captured using mutually synchronized camera and microphone. The utterance was repeated nine times (18 sec.). A lip region in each video image was extracted by the active appearance model (AAM) [3]. Filter bank analysis was used for the audio feature extraction and the principal component analysis (PCA) was used for visual feature extraction of the lip motion. These features were used as observed data to train ILHDS.

Learning the cross-media timing-structure model. Using the extracted audio and visual feature vector sequences as signal S' and S, we estimated the number of modes and parameters of each mode, partitioned each signal into an interval sequence, and then computed the cross-media timing structure according to the method described

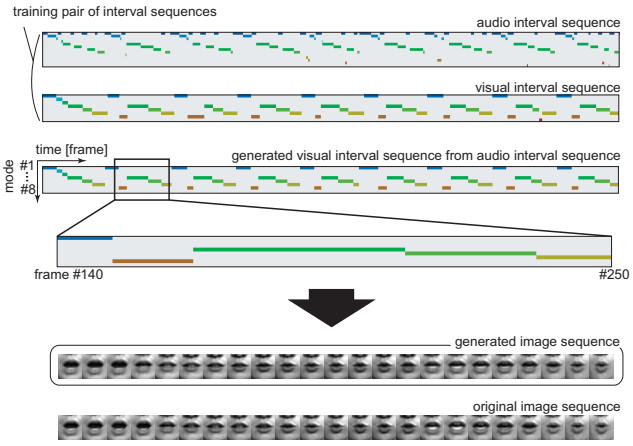

Figure 6. Media conversion.

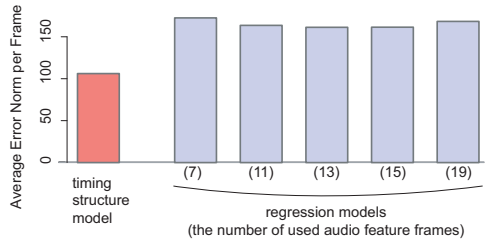

Figure 7. Average error norm per frame between generated and original sequences.

in Section 5.2. The estimated number of modes was 13 and 8 for audio and visual modes, respectively. The segmentation results are shown in Figure 6 (the first and second rows). Because of the noise, some vowels were divided into several different audio modes.

Evaluation of timing generation. Based on the estimated cross-media timing-structure, we applied the media conversion method in Subsection 5.3: we used an audio signal interval sequence included in the training data of ILHDS as an input (source) media (top row in Figure 6) and converted it into a video signal interval sequence (third row in Figure 6).

Then, to verify the performance of the media conversion method, we first compared the converted interval sequence with the original one which was generated from the video data measured simultaneously with the input audio data (second row in Figure 6). Moreover, we also compared the pair of video data: one generated from the converted interval sequence (second bottom row in Figure 6) and the originally captured one (bottom row in Figure 6). From these data, the media conversion method seemed to work very well.

To quantitatively compare our method with others, we generated feature vector sequences based on several regression models. Seven regression models were constructed, each of which estimated visual feature vector y_t from $2a+1$ frames of audio feature vectors $y_{t-a}, y_{t-a+1}, ..., y_t, ..., y_{t+a}$, where $a = 3, 4, ..., 9$. Figure 7 shows the average error norm per frame of each model when $a = 3, 5, 6, 7, 9$. We see that our method provide the smallest error compared to the regression models [1].

6. Conclusion

We proposed ILHDS as a novel computational model to represent dynamic events and structures. Applying ILHDS to human behavior analysis, we can successfully extract dynamic features based on the relation of temporal intervals, and analyze the synchronization/delay mechanism between mouth motion and speech utterance.

In this paper, we concentrated on modeling a single human behavior rather than multiparty interaction, because our first concern is to see the effectiveness of ILHDS for modeling and learning dynamic events and structures from multimedia signals. Currently we are extending the proposed scheme to model multiparty interaction by describing timing structures among dynamic primitives (e.g., pitch and intensity patterns in utterances) appeared in each of individuals, and to realize natural human-machine interaction.

Acknowledgment: This study is supported by Grant-in-Aid for Scientific Research No.18049046 of the Ministry of Education, Culture, Sports, Science and Technology.

References

[1] J. F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.

[2] M. Brand. Voice puppetry. *Proc. SIGGRAPH*, pages 21–28, 1999.

[3] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance model. *Proc. European Conference on Computer Vision*, 2:484–498, 1998.

[4] S. E. Levinson. Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language*, 1:29–45, 1986.

[5] A. V. Nefian, L. Liang, X. Pi, X. Liu, and K. Murphy. Dynamic Bayesian networks for audio-visual speech recognition. *EURASIP Journal on Applied Signal Processing*, 2002(11):1–15, 2002.

[6] M. Ostendorf, V. Digalakis, and O. A. Kimball. From HMMs to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Trans. Speech and Audio Process*, 4(5):360–378, 1996.

[7] P. V. Overschee and B. D. Moor. A unifying theorem for three subspace system identification algorithms. *Automata*, 31(12):1853–1864, 1995.

[8] S. Zhong and J. Ghosh. A unified framework for model-based clustering. *Journal of Machine Learning Research*, 4(11):1001–1037, 2003.

[1] Correction has been made for bug fix in this paragraph and Fig.7. Originally the regression model takes minimum at $a = 3$, however, $a = 6$ is correct.