Motor control primitives arising from a learned dynamical systems model of speech articulation

Vikram Ramanarayanan¹, Louis Goldstein² and Shrikanth Narayanan¹,²

¹Department of Electrical Engineering, University of Southern California, Los Angeles, CA
²Department of Linguistics, University of Southern California, Los Angeles, CA

Abstract

We present a method to derive a small number of speech motor control “primitives” that can produce linguistically-interpretable articulatory movements. We envision that such a dictionary of primitives can be useful for speech motor control, particularly in finding a low-dimensional subspace for such control. First, we use the iterative Linear Quadratic Gaussian with Learned Dynamics (iLQG-LD) algorithm to derive (for a set of utterances) a set of stochastically optimal control inputs to a learned dynamical systems model of the vocal tract that produces desired movement sequences. Second, we use a convolutive Nonnegative Matrix Factorization with sparseness constraints (cNMFsc) algorithm to find a small dictionary of control input primitives that can be used to reproduce the aforementioned optimal control inputs that produce the observed articulatory movements. The method performs favorably on both qualitative and quantitative evaluations conducted on synthetic data produced by an articulatory synthesizer. Such a primitives-based framework could help inform theories of speech motor control and coordination.

Index Terms: speech motor control, motor primitives, synergies, dynamical systems, iLQG, NMF.

1. Introduction

Mussa-Ivaldi and Solla (2004) [1] argue that in order to generate and control complex behaviors, the brain does not need to solve systems of coupled equations. Instead a more plausible mechanism is the construction of a vocabulary of fundamental patterns, or primitives, that are combined sequentially and in parallel for producing a broad repertoire of coordinated actions. An example of how these could be neurophysiologically implemented in the human body could be as functional units in the spinal cord that each generate a specific motor output by imposing a specific pattern of muscle activation [2]. Although this topic remains relatively unexplored in the speech domain, there has been significant work on uncovering motor primitives in the general motor control community. For instance, [3, 2] proposed a variant on a nonnegative matrix factorization algorithm to extract muscle synergies from frogs that performed various movements. More recently, [4] extended these ideas to the control domain, and showed that the various movements of a two-joint robot arm could be effected by a small number of control primitives.

The working hypothesis of this paper is that a small set of control primitives can be used to generate the complex vocal tract actions of speech. In previous work [5, 6], we proposed a method to extract interpretable articulatory movement primitives from raw speech production data. Articulatory movement primitives may be defined as a dictionary or template set of articulatory movement patterns in space and time, weighted combinations of the elements of which can be used to represent the complete set of coordinated spatio-temporal movements of vocal tract articulators required for speech production. In this work, we propose an extension of these ideas to a control systems framework. In other words, we want to find a dictionary of control signal inputs to the vocal tract dynamical system, which can then be used to control the system to produce any desired sequence of movements.

2. Data

We analyzed synthetic VCV (vowel-consonant-vowel) data generated by the Task Dynamics Application (or TaDA) software [7, 8] – which implements the Task Dynamic model of inter-articulator coordination in speech within the framework of Articulatory Phonology [9]. We chose to analyze synthetic data since (i) articulatory data is generated by a known compositional model of speech production, and (ii) we can generate a balanced dataset of VCV observations. TaDA also incorporates a coupled-oscillator model of inter-gestural planning, a gestural-coupling model, and a configurable articulatory speech synthesizer [10, 11] (see Figure 1). TaDA generates articulatory and acoustic outputs from orthographical (ARPABET) input. The ARPABET input is syllabified, parsed into gestural regimes and inter-gestural coupling relations using hand-tuned dictionaries and then converted into a gestural score. The obtained gestural score is an ensemble of constriction tasks, or gestures, for the utterance, specifying the intervals of time during which particular constriction tasks are active. This is finally used by the Task Dynamic model implementation in TaDA to calculate the time functions of the articulators whose motions achieve the constriction tasks (sampled at 200 Hz).

We generated 972 VCVs corresponding to all combinations of 9 English monophthongs and 12 consonants (including stops, fricatives, nasals and approximants). Each VCV can be represented as a sequence of articulatory states. In our case, the articulatory state at each sampling instant is a ten-dimensional vector comprising the eight articulatory parameters plotted in Figure 1 and two additional parameters to capture the nasal aperture and glottal width. We then downsampled the articulatory state trajectories to 100 Hz. We further normalized data in each channel (by its range) such that all data values lie between 0 and 1.

We acknowledge the support of NIH Grant DC007124.
3. Computing control synergies

In order to find primitive control signals, we first need to use optimal control techniques to compute appropriate control inputs that can drive the dynamical system given in Equation 1 to produce the set of articulatory data trajectories corresponding to each of our synthesized VCVs. Once we estimate the control inputs, we can use these as inputs to algorithms that learn spatiotemporal dictionaries such as the cNMFsc algorithm to obtain control primitives.

3.1. Computing optimal control signals

To find the optimal control signal for a given task, a suitable cost function must be minimized. Unfortunately, when using nonlinear systems such as the vocal tract system described above, this minimization is computationally intractable. Researchers typically resort to approximate methods to find locally optimal solutions. One such method, the iterative linear quadratic gaussian (iLQG) method [13, 14, 4], starts with an initial guess of the optimal control signal and iteratively improves it. The method uses iterative linearizations of the nonlinear dynamics around the current trajectory, and improves that trajectory via modified Riccati equations.

However, iLQG in its basic form still requires a model of the system dynamics given by the equation $\dot{x} = f(x, u)$, where $x$ is the articulatory state and $u$ is the control input. In order to eliminate this need and enable the to algorithm adapt to changes in the system dynamics in real time, Mitrovic et al. proposed an extension, called iLQG with Learned Dynamics, or iLQG-LD, wherein we learn the mapping $f$ using a computationally efficient machine learning technique such as Locally Weighted Projection Regression, or LWPR [15].

In our case, we pass as input to this algorithm articulator trajectories (see Section 2), and obtain as output a set of control signals (timeseries) $\tau$ that can effect those sequence of movements (one timeseries per articulator trajectory). In order to initialize the LWPR model of the dynamics, we used a linear, second-order critically-damped model of vocal tract articulator dynamics (after the Task Dynamics model of speech articulation [16]):

$$\ddot{\phi} + M^{-1}B\dot{\phi} + M^{-1}K\phi = \tau$$

where $\phi$ is a vector of articulatory variables. In our experiments, we found that choosing $M = I$, $B = 2\omega I$, and $K = \omega^2$ worked well for LWPR model initialization purposes (where $I$ is the identity matrix and $\omega$ is the critical frequency of the (critically-damped) spring-mass dynamical system, which we set as 0.6$^2$).

3.2. Extraction of control primitives

Modeling data vectors as sparse linear combinations of basis elements is a general computational approach (termed variously as dictionary learning or sparse coding or sparse matrix factorization depending on the exact problem formulation) which we will use to solve our problem [17, 18, 19, 20, 21]. If $\tau_1, \tau_2, \ldots, \tau_N$ are the $N = 972$ control matrices obtained using iLQG for each of the 972 VCVs, then we will first concatenate these matrices together to form a large data matrix $V = [\tau_1 | \tau_2 | \ldots | \tau_N]$. We will then use convolutive non-negative matrix factorization or cNMF [19] to solve our problem.
cNMF aims to find an approximation of the data matrix $V$ using a basis tensor $W$ and an activation matrix $H$ in the mean-squared sense. We further add a sparsity constraint on the rows of the activation matrix to obtain the final formulation of our optimization problem, termed cNMF with sparseness constraints (or cNMFsc) [5, 6]:

$$\min_{W,H} \|V - \sum_{t=0}^{T-1} W(t) \cdot \hat{H}^t\|^2 \text{s.t. \textbf{sparseness}}(h_i) = S_h, \forall i. \quad (2)$$

where each column of $W(t) \in \mathbb{R}^{K \times T}$ is a time-varying basis vector sequence, each row of $H \in \mathbb{R}^{N \times K \times T}$ is its corresponding activation vector ($h_i$ is the $i^{th}$ row of $H$), $T$ is the temporal length of each basis (number of image frames) and the $j \rightarrow i$ operator is a shift operator that moves the columns of its argument by $j$ spots to the right, as detailed in [19]. Note that the level of sparseness ($0 \leq S_h \leq 1$) is user-defined. See Ramarayanan et al. [5, 6] for the details of an algorithm that can be used to solve this problem.

4. Experiments and Results

The three-dimensional $W$ matrix and the two-dimensional $H$ matrix described above allows us to form an approximate reconstruction, $V_{\text{recon}}$, of the original control matrix $V$. This matrix $V_{\text{recon}}$ can be used to reconstruct the original articulatory trajectories for each VCV by simulating the dynamical system in Equation 1. Figures 3a and 3b show the performance of the algorithm in recovering the original control signals and movement trajectories in such a manner, respectively. We observed that the model accounts for a large amount of variance in the original data and the root mean squared errors of the original movements and controls were 0.16 and 0.29, respectively, on average. The cNMFsc algorithm parameters used were $S_h = 0.65$, $T = 10$. The sparseness parameter was chosen empirically to reflect the percentage of gestures that were active at any given sampling instant ($\sim 35\%$), while the number of bases were selected based on the Akaike Information Criterion or AIC [22], which in this case tends to prefer more parsimonious models. The temporal extent of each basis was chosen to capture effects of the order of 100ms. See [6] for a more complete discussion on parameter selection.

Note that each control primitive could effect different movements of vocal tract articulators depending on their initial position/configuration. For example, Figure 4 shows 8 movement sequences effected by 8 control primitives for one particular choice of a starting position. Each row of plots were generated by taking one control primitive sequence, using it to simulate the dynamical system learned using the iLQG-LD algorithm, and visualizing the resulting movement sequence. Figure 5 shows the median activations of each of the eight bases in Figure 4 for selected phones of interest. We see that the primitives produce movements that are interpretable: for instance, the bases that are activated the most for $P$, $T$, and $K$ are those involved in lip, tongue tip, and tongue dorsum constrictions respectively. For vowels, we also observe linguistically-meaningful patterning: $IY$, $AA$, and $UW$ involve high activations of controls that produce palatal, pharyngeal and velar/uvular constrictions, respectively.

5. Conclusions and Outlook

We have described a technique to extract synergies of control signal inputs that actuate a learned dynamical systems model of the vocal tract. We further observe, using data generated by the TaDA configurable articulatory synthesizer that this method allows us to extract control primitives that effect linguistically-meaningful vocal tract movements.

Work described in this paper can help in formulating speech motor control theories that are control synergy- or primitives-based. The idea of motor primitives allows us to explore many longstanding questions in speech motor control in a new light. For instance, consider the case of coarticulation in speech, where the position of an articulator/element may be affected by the previous and following target [23]. In other words, different movement sequences could result from changes in the timing and ordering of the same set of control primitives. Constructing internal control representations from a linear combination of a reduced set of modifiable basis functions tremendously simplifies the task of learning new skills, generalizing to novel tasks or adapting to new environments [24].

6. References


[2] The extreme overshoot/undershoot in some cases could be an artifact of normalization. Having said that, it is important to remember that the original data will be reconstructed by a scaled-down version of these primitives (weighted down by their corresponding activations).
Figure 4: Spatio-temporal movements of the articulator dynamical system effected by 8 different control primitives for a given choice of initial position. Each row represents a sequence of vocal tract postures plotted at 20 ms time intervals, corresponding to one control primitive sequence. The initial position in each case is represented by the first image in each row. The cNMFsc algorithm parameters used were $S_h = 0.65$, $K = 8$ and $T = 10$ (similar to [6]). The front of the mouth is located toward the right hand side of each image (and the back of the mouth on the left).


