© 2017 William Jacob Wagner

# UNSUPERVISED LEARNING OF VOCAL TRACT SENSORY–MOTOR SYNERGIES

BY

WILLIAM JACOB WAGNER

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Mechanical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Adviser:

Professor Stephen Levinson

# ABSTRACT

The degrees of freedom problem is ubiquitous within motor control arising out of the redundancy inherent in motor systems and raises the question of how control actions are determined when there exist infinitely many ways to perform a task. Speech production is a complex motor control task and suffers from this problem, but it has not drawn the research attention that reaching movements or walking gaits have. Motivated by the use of dimensionality reduction algorithms in learning muscle synergies and perceptual primitives that reflect the structure in biological systems, an approach to learning sensory-motor synergies via dynamic factor analysis for control of a simulated vocal tract is presented here. This framework is shown to mirror the articulatory phonology model of speech production and evidence is provided that articulatory gestures arise from learning an optimal encoding of vocal tract dynamics. Broad phonetic categories are discovered within the low-dimensional factor space indicating that sensory-motor synergies will enable application of reinforcement learning to the problem of speech acquisition.

# ACKNOWLEDGMENTS

In my first year at the University of Illinois I was fortunate enough to take the Machine Learning for Signal Processing course taught by Paris Smaragdis where I was exposed to the concepts of unsupervised learning and dimensionality reduction. I was intrigued by the idea that biological perceptual processing systems could be understood from the aspect of optimal encoding and was interested if there were analogous approaches in motor control beginning my interest in the topic of this thesis. I want to think Dr. Stephen Levinson for his guidance and support and in general for fostering an atmosphere for creative thinking and stimulating discussion on the topic of intelligence within the Language Acquisition and Robotics (LAR) Laboratory. I also want to thank my fellow LAR lab mates Jacob Bryan, Luke Wendt, and Yuchen He for helping formulate my research questions, navigate the coursework, and putting up with my text message ringer. Jacob Bryan's help with the Praat software modifications is very much appreciated as well.

I want to thank the faculty, students, and administration that made the NSF IGERT Neuroengineering program possible. This traineeship enabled me to study the topics I found interesting and led me to join the LAR lab. I also want to thank my colleagues at the Construction Engineering Research Laboratory whom have ensured that I keep my education a priority and exposed me to many new problems that I hope to explore in the future. I also want to acknowledge that this research started as a part of a course project in Dr. Levinson's Mathematical Models of Language. This would not have been possible without the constant support and affirmation I have received from my parents throughout my life. Thank you Mom and Dad. Finally I want to thank my wife for being there for me every day through this roller coaster that is graduate school. Thank you Elle for listening to my daily concerns, giving me perspective, and motivating me.

# TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION . . . . . 1
1.1 Background . . . . . 1
1.2 Description of Research . . . . . 4

CHAPTER 2 LITERATURE REVIEW . . . . . 5
2.1 Vocal Tract Modeling . . . . . 5
2.2 Redundancy and Redundancy Resolution . . . . . 13
2.3 Learning Modular Representations for Speech Production . . . 50

CHAPTER 3 VOCAL TRACT SENSORY-MOTOR SYNERGIES . . 56
3.1 Vocal Tract Model . . . . . 56
3.2 Sensory-Motor Synergy Model . . . . . 62
3.3 Experiments . . . . . 71

CHAPTER 4 CONCLUSION . . . . . 113
4.1 Future Work . . . . . 116

REFERENCES . . . . . 118

# CHAPTER 1

# INTRODUCTION

## 1.1 Background

The central question motivating this work is how can we design intelligent systems? In the seminal paper [1], Turing proposes that creating a machine that can think may be accomplished by developing a logical inference system in which definitions and propositions are programmed into the machine that it can use to evaluate statements about the world. The first wave of artificial intelligence was motivated by this line of thinking and the physical symbol system hypothesis [2] which fleshes out Turing's idea of intelligence as the manipulation of symbols and emphasizes the importance of the connection of symbols to physical systems. This approach, sometimes referred to as good old fashioned artificial intelligence or GOFAI, enabled the creation of so-called expert systems which were developed by encoding the knowledge of human experts into logical processing systems. However, it has fallen out of favor due partially to the heavy reliance on expert knowledge and the amount of time required to construct a system.

In its place, statistical learning theory and machine learning have flourished. Instead of relying on human experts to develop logical rules, these systems are trained and learn from large amounts of data. In fact, this approach was also encouraged by Turing in his 1950 paper [1], where he discusses the potential for constructing a machine that can be taught via reinforcement and punishment. In the final paragraph he offers this piece of advice:

> "It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to learn and understand English. This process could follow the normal teaching of a child. Things would be pointed out and

named, etc."

This advice to incorporate sense organs into the learning of symbols has been, if not overlooked, misinterpreted. Research in artificial intelligence has become segmented into subdisciplines separating perception from cognition from action. The field of computer vision has made great advances in object recognition, but struggles in scene understanding. Natural language processing on the other hand has advanced voice recognition to the point where people can regularly use it for transcription, but struggles with answering simple questions or responding to basic commands because the systems lack understanding. Control theory has enabled the use of robotic systems for manufacturing operations, but they have been primarily confined to performing repetitive tasks in highly controlled environments. It is possible that the deficiencies in each subfield will be resolved by better technology and more advanced learning algorithms. But according to the philosophy of embodied cognition, this approach is not sufficient. Alternatively, it suggests that intelligent systems must have a means of interacting with and learning from the world. Therefore, the artificial systems must be capable of influencing the external world through motor function and be able to access information about the world through sensory systems.

So, following Turing's advice and ascribing to the theory of embodied cognition, I chose to pursue development of a system that can learn to produce speech. Speech is produced via coordination of articulatory muscles which vary the shape of the vocal tract and the flow of air through it. This is a very complex motor control task due to the non-linear relationship between muscle activations, vocal tract shape, and acoustic output. Making this task even more difficult is the high number of degrees of freedom required to adequately characterize the process. In the Praat articulatory synthesis model, a somewhat sophisticated simulator of the vocal tract, 29 different articulatory muscles control the shape of the vocal tract represented by 89 different acoustic tube sections [3]. A one dimensional acoustic signal is generated by simulating airflow through this model, but unfortunately, it is very difficult to obtain meaningful information from this raw signal. Instead, it is common to transform the signal into a time-frequency representation. This new representation is often represented by many more degrees of freedom. The point of this example is to show that speech production requires dealing with

very high dimensional signals.

Reinforcement learning (RL) is typically used to approach problems where developing controllers for complex dynamical systems is desired, but the computational demands of this approach increase exponentially with the number of degrees of freedom of the system. This phenomenon is known as the curse of dimensionality. To deal with this curse we to look to biology for inspiration. Bernstein, one of the first scientists to study human motion and coordiantion, posited that complex motor control is aided by the use of so called muscle-synergies. Essentially, synergies are coordinated responses of muscles that can be superimposed on one another and concatenated together to produce more complex motions. They can be thought of as the fundamental building blocks of motion. But if such synergies do exist, where do they come from?

Interestingly, similar questions arise in the study of perceptual processing systems where the responses of individual neurons to stimuli reveal characteristic patterns of activation. The range of stimuli that elicits a response from an individual neuron is referred to as a receptive field. Barlow originally hypothesized that perceptual processing systems evolve based on a principle of optimal encoding. More recently, many studies have shown that various unsupervised dimensionality reduction (DR) methods, which attempt to learn optimal encoding schemes, yield filters similar to the receptive fields in human visual and auditory processing systems. Other researchers have used these same methods to look for evidence of muscle synergies with mixed results. One problem with using DR methods to look for muscle synergies is that there is no real ground truth to reference as in the case of perceptual systems and receptive fields. This is problematic because the argument can be made that although one can apply DR methods to motion recordings or electromyograph (EMG) signals and find low dimensional representations of the signals, the resulting synergies may be more reflective of the task being performed than of the underlying functional units of control.

To address this problem and separate out these different effects it has been suggested that synergies composed of both observations of a system and the inputs that control the system be learned instead. These new sensory-motor synergies should efficiently encode the dynamics of the system and provide a means for efficiently exploring the control space. This may be a way of lifting the curse of dimensionality and enabling learning of complex motor

control tasks such as speech production.

This is a compelling idea, but how does this relate to our current understanding of speech production? First of all, speech is inherently symbolic. Sentences are composed of words which are composed of syllables which are composed of phonemes. According to the theory of articulatory phonology, all of these symbols are constructed from lower level symbols called gestures. Gestures are described as the coordinated movements of articulators accompanied by the activations of articulatory muscles recruited to produce those movements. Gestures, like synergies, are weighted by activation levels and combined with one another in what are referred to as gesture scores to produce the higher level speech symbols. So, within this framework, gestures are analogous to sensory-motor synergies and gestures scores analogous to the synergy activations over time. This reasoning led me to believe that developing a system that learns to control a vocal tract by using sensory-motor synergies would be fruitful.

## 1.2 Description of Research

In this thesis I describe one approach to learning vocal tract sensory-motor synergies and develop methods for evaluating the resulting model. I use the Praat articulatory synthesizer as a basis for this work because it models the human vocal tract in a biologically plausible way, incorporates dynamic movement of the articulators, and is open source [3]. I then modify this software to enable complete control of the simulator and recording of all relevant states of the model. By randomly articulating the model, I then generate a database of articulatory muscle activations, vocal tract area functions, and acoustic signals that is used to learn synergies. Motivated by the success of Todorov and Ghahramaniin in learning of sensory-motor primtives for control of a simulated arm, I chose to use a similar dimensionality reduction algorithm called dynamic factor analysis (DFA) to learn the vocal tract sensory-motor synergies. I evaluate the usefulness of the learned synergies by analyzing the learned patterns of coordination and by analyzing the factor trajectories in the lower-dimensional factor space with respect to separation between phonemic classes.

# CHAPTER 2

# LITERATURE REVIEW

In this chapter I review previous research in the fields of vocal tract modeling, redundancy resolution, and speech learning as it pertains to the problem of learning to control a realistic vocal-tract simulator to produce speech. I also provide motivation for approaching this problem as learning of vocal tract sensory-motor synergies.

## 2.1 Vocal Tract Modeling

As the goal is to develop low-level sensory-motor control primitives for speech production, it is vital to understand and adequately characterize the system that we aim to control. If we were to stop an individual on the street and ask them the question "How is speech produced?" most people would find it difficult to come up with an answer. That is because speech comes so naturally to humans that most of us don't bother to give it much thought. However, speech production is far from simple and, in fact, has a rich academic history.

### 2.1.1 The Speech Signal

The human vocal tract is a hollow flexible passage through which air flows to produce speech. The lungs connect to the trachea which is a cartilaginous tube and is sometimes referred to as the windpipe. The trachea connects in-turn to the larynx, colloquially referred to as the voice box, which contains two mucous membranes called the vocal folds or vocal cords. This area of the vocal tract is called the glottis. The glottis also describes the opening between the vocal folds. These membranes can be tensioned by muscles in the glottis to enable vibration of the vocal cords, or relaxed to allow air to pass unrestricted through the glottis. Located at the end of larynx is the epiglottis

which is a flap of elastic cartilage covered in a mucous membrane that closes off the lower vocal tract from the upper vocal tract and acts as a valve diverting liquids to esophagus to prevent aspiration. The pharynx lies between the epiglottis and the velum. The velum, or soft palate, is a muscular structure at the back of the mouth that can close off air from flowing through the nasal cavity when raised. Other than the velum, the nasal cavity is an unarticulated structure that terminates at the nostrils. The shape of the oral cavity however, is determined by the position of the tongue, jaw, and lips. The walls of the vocal tract are made up of cartilage, bone, mucous membranes, and muscles which have differing stiffness characteristics and may deflect as air passes through the tract. Fant [4] and Rabiner and Schafer [5]provide a more thorough overview of the physical elements of speech production than is presented here.

Speech sounds are produced when air from the lungs is forced through the vocal tract resulting in changes in the air pressure at the lips and nose. This results in an acoustic wave being radiated into the environment. Various speech sounds are produced by altering the shape of the vocal tract in a process referred to as articulation. The articulators are elements of the vocal tract that can be moved to change the shape of the tract and include the tongue, lips, jaw, velum. A variety of articulatory models relating positions or activations of the articulators to the shape of the vocal tract have been proposed. One of the earliest models, proposed by Coker uses five variables to parameterize articulation, namely tongue body height, anterior-posterior position of the tongue body, tongue tip height, mouth opening, and pharyngeal opening. A sixth parameter is also used to alter the static nominal tract length of 17 cm [6]. Modulation of the air flow through the tract by the lungs and diaphragm also plays vital role, and is sometimes considered in articulatory models.

The speech signal is composed of sequences of speech sounds, also known as phones. A phoneme is a useful concept that represents a category of similar phones, allowing for classification of speech sounds into discrete categories [5]. This enables transcription of the speech signal into a sequence of phonemes represented as symbols. They are often called the building blocks of spoken language. Now it is important to point out that phonemes are a theoretical construct used to aid in the analysis and understanding of the speech signal and not an exact speech sound. In other words there exists a certain amount

of ambiguity or underspecification in the definition of a phoneme whereas a phone can be considered one realization of a specific phoneme out of infinitely many possible realizations of that same phoneme.

The field of linguistics is devoted to the study of language and the science's practitioners, linguists, have traditionally studied the connection between speech sounds and meaning. The two fields of linguistics that are most relevant to this research are phonetics and phonology. Phonetics is defined as the study of the physical properties of speech sound production and perception and phonology is defined as the study of sounds as abstract elements in the speaker's mind that distinguish meaning. In other words phonetics is concerned with the production of phones and phonology is concerned with the categorization and organization of phonemes.

Speech sounds can be classified into three broad classes according to the mode of excitation: voiced sounds, unvoiced sounds or fricatives, and plosives [5, 7]. Voiced sounds are produced when air is forced past the tensioned vocal folds producing a periodic excitation of the vocal tract. Fricatives or unvoiced sounds are produced by constricting air flow at some point along the vocal tract causing turbulent air flow. This produces broad-spectrum noise that excites the vocal tract. Plosives are produced when airflow is stopped, by making a complete closure of the vocal tract, and then released. Pressure is allowed to build up during this closure and when release creates a burst of turbulent air flow.

As mentioned above, articulation changes the shape of the vocal tract and subsequently the sound produced. The raw acoustic waveform can provide some indication of a change in sound, but only enables general observations on the periodicity and amplitude of the speech signal. The spectral content of the speech signal has proven to be much more useful. Articulation changes the resonant frequencies of the vocal tract, which can easily be seen by performing a frequency decomposition of the speech signal using the Fourier transform. The change in resonance of the vocal tract is analogous to the change in pitch produced by musical instruments such pipe organs or any woodwinds when a different note is played. In phonetics, these resonances are referred to as the formants and are represented by the symbol $\mathrm{f}_i$ where $i$ refers to the $i^{\text{th}}$ formant ordered by increasing frequency. However, since the shape of the vocal tract is changing over time to produce sequences of phones, a time varying representation of the spectrum is needed. The most common

approach is to employ the use of the spectrogram, which is mathematically described by the amplitude spectrum given by the short time Fourier transform (STFT) as shown in equation 2.41. More details on this technique can be found later in this chapter.

Using these different means of characterization, linguists have identified many different categories of phonemes and have developed a written representation of these sounds called the International Phonentic Alphabet (IPA) which contains over 500 distinct phonemes [8]. However, individual languages only utilize a subsets of these phonemes to represent meaning. For example, American English is comprised of 42 different phonemes [5]. These phonemes are broken down into three broad phonetic categories: vowels, consonants, dipthongs, and semivowels. Vowels are produced by voiced excitation of a fixed shape vocal tract. Different vowels are produced primarily by differing positions of the tongue, but position of the jaw, lips, and velum have a small effect as well. The difference between the vowels is most easily seen by viewing vowel pronunciations in the $(f_1, f_2)$ plane. This reveals the existence of the so-called"vowel triangle", which shows the relationship between jaw opening and tongue position to $f_1$ and $f_2$ respectively [5]. A dipthong is defined as a gliding monosyllabic speech item that starts at or near the articulatory position for one vowel and moves to or towards a position for another. Semivowels are difficult to characterize, but can best be defined as transitional, vowel-like sounds that are highly context dependent.

Consonants are a broad category of sounds that are produced by partial or full closure of the vocal tract and can be broken down into 4 categories in English. Nasal are voiced consonants produced by closing off the vocal tract completely at some point in the oral cavity while lowering the velum and allowing air to pass through the nasal cavity. The location where the oral cavity is closed affects the resonant properties of the vocal tract producing different nasals. Fricatives are produced through constriction of the vocal tract as described above and can be either voiced or unvoiced. Each voiced fricative has an unvoiced counterpart that only varies in voicing not in articulation. Stops or plosives are produced by closing off and subsequently opening the vocal tract, releasing a transient burst of turbulent air. Stops can be voiced or unvoiced differing only in the presence or absence of vocal cord vibration. And finally, affricates are a concatenation of a stop and a fricative.

### 2.1.2 The Physics of Speech Production

Although speech production may seem straight forward, the vocal tract is an extremely complex non-linear time varying aero-dynamical system that is very difficult to model accurately. However, there are a few assumptions that can be made to greatly simplify the model. The standard approach is to model the vocal tract as a lossless acoustic tube of nonuniform slowly time-varying cross-sectional area $A(x,t)$ where air flowing through the tract is assumed to travel as a plane-wave along a single dimension $x$. Portnoff and Sondhi [9, 10] have shown that under these assumptions that applying Newton's second law and the principle of the conservation of mass to this model yields the following two equations

$$-\frac{\partial p}{\partial x} = \frac{\rho}{A(x,t)}\frac{\partial u}{\partial t} \tag{2.1}$$

$$-\frac{\partial u}{\partial x} = \frac{A(x,t)}{\rho c^2}\frac{\partial p}{\partial t} \tag{2.2}$$

where $p(x,t)$ is the pressure, $u(x,t)$ is the volume velocity, $\rho$ is the air density, and $c$ is the speed of sound for the given air density. Differentiating 2.1 and 2.2 with respect to space and time respectively and eliminating $\frac{\partial^2 u}{\partial x \partial t}$ and $\frac{\partial u}{\partial t}$ from the system of equations results in the Webster equation for pressure

$$\frac{\partial^2 p}{\partial x^2} + \frac{1}{A(x,t)}\frac{\partial p}{\partial x}\frac{\partial A(x,t)}{\partial x} = \frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} \tag{2.3}$$

which is a differential equation that describes the relationship between the vocal tract area function and pressure along the tract over time. Closed form solutions are generally not possible and only exist in trivial cases, but numerical solutions can be computed given appropriate boundary conditions. One approach is to discretize the Webster equation by breaking up the vocal tract into $n$ concatenated tubes each with a constant area function. The first and second order derivatives can then be approximated as first backward differences and second central differences respectively yielding a finite difference equation.

However, this does not account for losses due to wall displacement, viscous air flow at the walls, and heat conduction in the walls. To incorporate the effects into the model, a frequency domain representation is obtained by

assuming a time invariant tube and modeling the glottal boundary condition as a complex volume velocity source given by

$$u(0,t) = U_G(w)e^{jwt} \tag{2.4}$$

Additionally assuming that the equations governing the losses are linear time invariant yields

$$p(x,t) = P(x,w)e^{jwt} \tag{2.5}$$

$$u(x,t) = U(x,w)e^{jwt} \tag{2.6}$$

and Equations 2.1 and 2.2 can be rewritten as

$$-\frac{dP}{dx} = Z(x,w)U(x,w) \tag{2.7}$$

$$-\frac{dU}{dx} = Y(x,w)P(x,w) \tag{2.8}$$

where $Z(x,w)$ and $Y(x,w)$ are defined as the acoustic impedance and admittance per unit length respectively. The Webster equation can then be rewritten as a function of frequency and reformulated in terms of volume velocity.

$$\frac{d^2U}{dx^2} = \frac{1}{Y(x,w)}\frac{dU}{dx}\frac{dY}{dx} - Y(x,w)Z(x,w)U(x,w) \tag{2.9}$$

Porntoff assumes uniform displacement of the vocal tract wall at a given position and models the relationship between displacement $\xi(x,t)$ and pressure as a simple mass-spring-damper system

$$p(x,t) = M\frac{\partial^2\xi(x,t)}{\partial t^2} + b\frac{\partial\xi(x,t)}{\partial t} + k(x)\xi(x,t) \tag{2.10}$$

where $M$ is the unit length wall mass, $b$ is the damping coefficient, and $k$ is the spring constant [9]. The impedance and admittance contributions of these three losses is reviewed in Rabiner and Schafer [5] and Levinson [7], but we will not review it here. The resulting lossy version of Webster's equation can then be computed. However this method is somewhat computationally expensive and requires measurement of physical constants of vocal tract tissue. In [10], Sondhi proposes an alternate formulation with impedance and admittance equations that avoids these difficulties and approximates Portnoff's formulation with reasonable accuracy.

Solving either Webster's equation or the lossy Webster equation then comes down to solving a boundary value problem. Dunn et al. [11] has shown that the glottal boundary condition can be approximated as a constant volume source with an asymmetric triangular waveform with amplitude $V$.

$$U_g(w) = \frac{V}{w^2} \tag{2.11}$$

Assuming the relationship between sinusoidal steady-state pressure and volume velocity given by

$$P(L, w) = Z_r(w)U(L, w) \tag{2.12}$$

whre $Z_r$ is the radiation impedance at the lips approximated by a piston in an infinite plane baffle give as

$$Z_r(w) = jwL_r/(1 + jwL_r/R) \tag{2.13}$$

where $L_r$ and $R$ are constants [5, 7].

This methodology is useful and was very important in the early days of speech research, but many of the assumptions that were made limit this approach. One of the problems is that the cross-sectional area function $A(x, t)$ is only quasi-stationary, not stationary [7]. This is meant to indicate that although the area function changes with time, for many speech sounds it changes slowly in comparison to the pressure change over time or

$$\left|\frac{\partial A}{\partial t}\right| \ll \left|\frac{\partial p}{\partial t}\right| \tag{2.14}$$

This model turns out to approximate vowel production well, but has difficulty approximating other speech sounds [5]. Another assumption made by this model, that the air flow is described by plane waves, may be overly simplistic. If 2 dimensional wave propagation is considered instead and viscous and convective effects are considered, the 2 dimensional Reynolds averaged, Navier-Stokes equations for slightly compressible flow is arrived at instead of the Webster equation. These equations can also be solved numerically and may represent the physical system more accurately [7]. Other improvements to the vocal tract model include consideration of nasal coupling, dynamic modeling of the vocal folds, and variable length tracts.

### 2.1.3 The Source Filter Model and Linear Prediction

The source filter model of speech production is an electrical analogue to the acoustic model reviewed in the previous section. OShaughnessy [12] provides a review of this topic which is covered briefly here. The primary assumption of this model is that the vocal tract can be decomposed into 3 main components, each one modeled independently: a glottal source, a vocal tract filter, and an acoustic impedance at the lips. The glottal source is typically modeled as a periodic pulse train for voiced sounds and a white noise source for frication. The vocal tract filter is modeled as a time varying digital filter with a transfer function of the form

$$H(z) = G\frac{\sum_{l=0}^{N} b_l z^{-l}}{1 - \sum_{k=1}^{M} a_k z^{-k}} \tag{2.15}$$

where $G$ is a gain term, $N$ is the number of zeros, $M$ is the number of poles, and the $b_l$'s and $a_k$'s are scalar coefficients that vary with time. If we remove all zeros from this mode, as is commonly done, by letting $b_1 = 1$ and $b_l = 0 \; \forall \; l \neq 1$ then 2.15 becomes

$$H(z) = \frac{G}{1 - \sum_{k=1}^{M} a_k z^{-k}} \tag{2.16}$$

The source filter model has a strong connection to linear prediction based models of the form

$$s(n) = \sum_{k=1}^{M} a_k s(n-k) + G\sum_{l=1}^{N} b_l u(n-l) \tag{2.17}$$

which has an equivalent transfer function to the general source-filter model of speech 2.15. In words, the general linear prediction model uses a linear combination of past values of the signal $s(n-k)$ and a linear combination of past values of the input $u(n-l)$ to predict future values of the signal $s(n)$. This is sometimes referred to as the autoregressive moving average model or ARMA. In practice, it is difficult to estimate the zeros so the moving average portion of the equation is removed leaving an all pole model

$$s(n) = \sum_{k=1}^{M} a_k s(n-k) + e(n) \tag{2.18}$$

where the error term $e(n)$ is assumed to be $Gu(n)$, giving the same transfer function as Equation 2.16. This model is therefore referred to simply as autoregressive or AR and the coefficients $a_k$ of the all pole model are referred to as linear predictor coefficients or LPCs.

Many methods exist to solve this model, including the autocorrelation, covariance, and lattice methods which all involve the use of what are called the partial correlation coefficients (PARCORs) given by

$$k_i = -\frac{A_{i+1} - A_i}{A_{i+1} + A_i} \tag{2.19}$$

where $A_i$ is the area of the $i^{\text{th}}$ tube in the discretized vocal tract [5]. This indicates that there is a relationship between the LPC model and the vocal tract model. In fact, it has been shown that the LPC model is equivalent to the discretized concatenated tube models derived from the lossless Webster equation, Equation 2.3 [13]. Additionally, the resonances or formants of the vocal tract are simply the the poles of 2.16.

## 2.2 Redundancy and Redundancy Resolution

At the core of problems within perception and motor control is the need to reduce the redundancy of the sensory and control signals. Dimensionality reduction methods are mathematical techniques designed to perform this task. These methods are used heavily in the study of perceptual processing where neural filters called primitives have been shown to respond to distinct areas of the perceptual space called receptive fields. These primitives are thought to have developed to reduce the dimensionality of perceptual signals and optimally encode the information contained within the signals. The motor equivalence problem in motor control is the dual to the perceptual processing problem. It points out the difficulty in selecting the appropriate control actions to perform a motor task when there are many, often infinitely many, ways of performing the task. A relatively unexplored approach to dealing with the motor equivalence problem is to combine perceptual and motor features to develop hybrid synergies. I review all of these concepts here to motivate my approach to developing vocal tract sensory-motor synergies and to place it in the context of previous work.

### 2.2.1 Dimensionality Reduction Methods

Dimensionality reduction (DR) methods are used in a variety of disciplines when it is desired to more compactly represent data than in its raw high-dimensional form. Often this is desired to reduce computational demands and memory requirements of analyzing the data or using it in various algorithms. Some common application areas include image processing, economics research, and speech recognition, but it is generally useful in signal processing and statistical analysis. DR methods all rely on the assumption that high-dimensional data has a lower intrinsic dimensionality, meaning that if redundancy exists in the data it can therefore be represented using fewer parameters. The number of parameters required to completely represent the data is called the intrinsic dimensionality [14]. DR can be applied in cases where the intrinsic dimensionality is still high if the contribution of some dimensions to the data is relatively small, meaning that the reconstruction error is also very small. These techniques are mostly performed using different matrix factorization algorithms and therefore the data must be represented in matrix form. Let $\mathbf{Y}$ be a $D \times n$ matrix formed by $n$ row datavectors $\mathbf{y}_j (j \in 1, 2, \ldots n)$ of dimensionality $D$ [15]. The intrinsic dimensionality of the dataset is $d$ and is assumed to be $d < D$ or sometimes $d << D$. Dimensionality reduction techniques take advantage of this assumption and represent $\mathbf{Y}$ instead as a matrix decomposition or matrix multiplication.

#### 2.2.1.1 Principle Component Analysis

The most commonly used DR method is Principle Component Analysis (PCA) which is also known as the Karhunen-Loéve transform in the signal processing domain. It was first discovered by Karl Pearson in 1901 [16]. PCA is a mathematical transformation that attempts to linearly project a dataset onto an orthogonal coordinate space where the variance of the projected data along each successive principle axis is decreasing. The PCA model can be represented as a matrix multiplication

$$\mathbf{Z} = \mathbf{W}^\top \mathbf{Y} \tag{2.20}$$

where $\mathbf{Z}$ is a $d \times n$ dimensional matrix with row vectors $\mathbf{z}_j (j \in 1, 2, \ldots n)$ each of dimensionality $d$, $\mathbf{Y}$ is defined as above, and $\mathbf{W}$ is the $D \times d$ feature

matrix [17]. The matrix $\mathbf{Z}$ exists on a $d$ dimensional manifold embedded in a $D$ dimensional space and is sometimes referred to as the weight or score matrix.

PCA requires that the data $\mathbf{Y}$ first be centered; i.e., have a zero column wise mean. In addition, this method is not scale invariant, and therefore the rows of $\mathbf{Y}$ are often scaled by the square root of the variance, especially in cases where the features have different units. The first principle component is then found by minimizing the Rayleigh quotient $\mathbf{w}_k = \arg\max \frac{(\mathbf{w}_k^\top \mathbf{Y})(\mathbf{w}_k^\top \mathbf{Y})^\top}{\mathbf{w}_k^\top \mathbf{w}_k}$ where $k \in 1, 2, \ldots d$ [15]. Successive principle components are found in the same manner, with $\mathbf{Y}$ being replaced with $\mathbf{Y}_k = \mathbf{Y} - \sum_{s=1}^{k-1} \mathbf{Y}\mathbf{w}_k\mathbf{w}_k^\top$ for the $k^{\text{th}}$ component. This can also be thought of as attempting to decorrelate the columns of the weight matrix $\mathbf{Z}$.

The solution to this minimization is found by setting the columns of $\mathbf{W} = \mathbf{N}_d$, which is the matrix containing the first $d$ eigenvectors of the autocovariance matrix $\mathbf{\Sigma} = \mathbf{Y}\mathbf{Y}^\top$ corresponding to the $d$ largest eigenvalues $\mathbf{\Lambda}_d$. The eigendecomposition is shown below for clarity.

$$\mathbf{\Sigma}\mathbf{N}_d = \mathbf{\Lambda}_d\mathbf{N}_d \tag{2.21}$$

where $\mathbf{\Lambda}_d$ and $\mathbf{N}_d$ are the diagonal matrix containing the $d$ largest eigenvalues, and the $D \times d$ matrix with columns containing the corresponding $d$ eigenvectors respectively. Alternatively, the feature matrix $\mathbf{W}$ can also be found by equating it to $d$ right singular vectors from the singular value decomposition (SVD) of $\mathbf{Y}$ [17]. This SVD method is often preferred as algorithms exist to more efficiently perform SVD. The dimensionality of the data $\mathbf{Y}$ is reduced by allowing $d < D$ when performing PCA. By keeping the principle components corresponding to the largest $d$ eigenvalues or singular values, this method will minimize the mean square error of reconstructing $\mathbf{Y}$ from the score and feature matrices, $\mathbf{Z}$ and $\mathbf{W}$ [18].

#### 2.2.1.2 Factor Analysis

Factor analysis (FA) is another dimensionality reduction technique that is more commonly used within the social sciences. It was originally developed by Charles Spearman during his study of human intelligence [19]. FA is a statistical method that attempts to find a small number of uncorrelated

unobserved variables that explain the correlations between a larger number of correlated observed variables [20]. Mathematically, the concept behind FA is that a $D$ dimensional random variable can be represented by a linear combination of $d < D$ dimensional hidden or latent random variables called common factors in addition to $D$ error terms known as unique or special factors. More explicitly

$$y_i = a_{i1}x_1 + a_{i2}x_2 + \cdots a_{ik}x_k + \gamma_i \tag{2.22}$$

for each $i \in \ 1, 2, \ldots D$. In vector form this becomes

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \gamma \tag{2.23}$$

where $\mathbf{x}$ is a $d$ dimensional random variable represented as a column vector containing the common factors, $\mathbf{y}$ is a $D$ dimensional random variable represented as a column vector of observed variables, each row of $\mathbf{A}$ contains the factor loadings for each observed random variable in $\mathbf{y}$, and $\boldsymbol{\gamma}$ is a $D$ dimensional random variable represented as a column vector and containing the unique factors [17]. Intuitively, one can think of the common factors as the underlying hidden states that produce the observations $\mathbf{y}$ which is corrupted by noise $\boldsymbol{\gamma}$.

Factor analysis can be used as a means of testing a hypothesis regarding the relationship of hidden variables to the observed variables as is done in confirmatory factor analysis (CFA), or it can be used to discover the underlying latent structure of a set of random variables, as in exploratory factor analysis (EFA). In EFA, the linear weightings of the common factors, or factor loadings, are unknown, while in CFA the factor loadings are assumed to be known. In the context of unsupervised dimensionality reduction, EFA is the technique used because the goal is to discover structure in the system not to test for a hypothesized structure.

In order to fit this factor model, samples of the system are required. It is often convenient to the represent samples of the model in Equation 2.23 in matrix form

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \boldsymbol{\Gamma} \tag{2.24}$$

where each random variable in 2.23 has been replaced by a matrix with each column consisting of a sample of that random variable. For $n$ samples of the

system, $\mathbf{Y}$ and $\mathbf{\Gamma}$ are $D \times n$ matricies, $\mathbf{X}$ is a $d \times n$ matrix, and $\mathbf{A}$ is , as defined as before, a $D \times d$ matrix [18].

There are a number of assumptions that have to be made about the structure of our model in order to find a solution to the EFA problem outlined by many [17, 18, 20]. As with PCA, we assume that the data is zero mean or has been mean centered; i.e., the row mean $\mathbb{E}[\mathbf{y}] = 0$. We also assumed that $E[\mathbf{x}] = 0$ and $E[\boldsymbol{\gamma}] = 0$. In addition, we assume that the unique factors are uncorrelated $\mathbb{E}[\boldsymbol{\gamma}\boldsymbol{\gamma}^\top] = \Psi$ where $\Psi$ is diagonal. This really just reiterates the basic idea of FA, that the common factors and factor loadings capture all of the covariances of the variables in $\mathbf{y}$. Another important assumption is that the common factors are uncorrelated with the unique factors $\mathbb{E}[\mathbf{x}\boldsymbol{\gamma}^\top] = \mathbf{0}$. Finally, it is often assumed that the common factors are uncorrelated and have unit variance $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \mathbf{I}_d$, but this constraint can be relaxed to allow correlations between common factors.

Even with all of these assumptions, the FA model is under-constrained. To make this indeterminacy clear, we first compute the covariance matrix from both sides of Equation 2.23

$$\mathbf{\Sigma} = \mathbb{E}[\mathbf{y}\mathbf{y}^\top] = \mathbf{Y}\mathbf{Y}^\top = (\mathbf{A}\mathbf{f} + \boldsymbol{\gamma})(\mathbf{A}\mathbf{f} + \boldsymbol{\gamma})^\top = \mathbf{A}\mathbf{A}^\top + \mathbf{\Psi} \tag{2.25}$$

where the earlier assumptions regarding the unique and common factors enable the simplification [17]. Note that often times the sample covariance matrix $\mathbf{S}$ is used in place of the covariance matrix $\mathbf{\Sigma}$. If a solution to 2.25 is found to be $\mathbf{A}$ and $\mathbf{\Psi}$ then $\mathbf{A}^* = \mathbf{A}\mathbf{T}$ and $\mathbf{\Psi}$ is also a solution if $\mathbf{T}$ is orthogonal because [17, 20]

$$\begin{aligned}
\mathbf{A}^*\mathbf{A}^{*\top} &= (\mathbf{A}\mathbf{T})(\mathbf{A}\mathbf{T})^\top \\
&= \mathbf{A}\mathbf{T}\mathbf{T}^\top\mathbf{A}^\top \\
&= \mathbf{A}\mathbf{A}^\top
\end{aligned}$$

Therefore, solutions to the EFA problem are not unique. In order to find a unique solution, most solution methods add further restrictions on $\mathbf{A}$ [17, 20]. After finding this initial solution, a "rotation method" is then applied, whose goal is to find the *best* rotation matrix $\mathbf{T}$. These different methods quantify *best* by minimizing different cost functions which may be appropriate for different applications. Some of the most common methods include varimax,

quartimax, and promax, but there are many, some of which even relax the orthogonality constraint on $\mathbf{T}$, e.g. oblimax [17, 21, 22].

Often, it is desirable to have some measure that describes how well the FA model describes the data. One way of quantifying this goodness of fit is by looking at the variance of each variable $y_i$ that is described by the common factors $\mathbf{x}$. This is referred to as the communality $h_i^2$ and can be found by looking at the diagonal elements of Equation 2.25.

$$Var(y_i) = \sigma_{ii} = \sum_{k=1}^{d} a_{ik}^2 + \psi_{ii} = h_i^2 + \psi_{ii}$$

where $\psi_{ii}$ is the variance of the $i^{\text{th}}$ unique factor $\gamma_i$ known as the unique variance [18, 20]. It contains the variance not accounted for by the common factors in $y_i$.

The FA model has the interesting property of scale equivariance unlike PCA. If we have a factor model $\mathbf{y} = \mathbf{A}_y\mathbf{x} + \boldsymbol{\gamma}_y$ then then model for $\mathbf{z} = \mathbf{C}\mathbf{y}$ where $\mathbf{C}$ is a diagonal scaling matrix is

$$\mathbf{z} = \mathbf{C}\mathbf{y} = \mathbf{C}(\mathbf{A}_y\mathbf{x} + \boldsymbol{\gamma}_y) = \mathbf{C}\mathbf{A}_y\mathbf{x} + \mathbf{C}\boldsymbol{\gamma}_y = \mathbf{A}_z\mathbf{x} + \mathbf{C}\boldsymbol{\gamma}_y$$

meaning that the same common factors describe the scaled $\mathbf{z}$ [17, 20, 18]. The new factor loadings $\mathbf{A}_z = \mathbf{C}\mathbf{A}_y$ are merely a scaled version of the factor loadings for $\mathbf{y}$. The unique variances $\boldsymbol{\Psi}_z = \mathbb{E}[\mathbf{C}\boldsymbol{\gamma}_y\boldsymbol{\gamma}_y^\top\mathbf{C}^\top] = \mathbf{C}\boldsymbol{\Psi}_y\mathbf{C}^\top$ are also just re-scaled. This means that the choice between using either the covariance or the correlation is less important than in PCA because one can always obtain the alternate formulation by simple scaling.

In order to obtain an initial solution, a variety of methods can be used. One of the early approaches, called the centroid method, takes the ratio of the sum of each column of the the correlation matrix to the sum of all of the elements in the correlation matrix to estimate each factor loading [23, 22]. This technique is crude and lacks a strong statistical foundation, but was devised to be computable by hand before computers were widely used and often gives feasible results. Unfortunately, this method and similar methods gave factor analysis a bad name and led to it be ignored by many mathematicians and statisticians as a valid tool for latent variable discovery. Other solution methods, however, rest on solid theoretical ground including Bayesian

approaches [24], cannonical correlation analysis, and maximum likelihood estimation (MLE) of $\mathbf{A}$ and $\mathbf{\Psi}$ assuming multivariate normality of $\mathbf{x}$ and $\boldsymbol{\gamma}$ [25]. The MLE method is solved using an iterative Expectation-Maximization procedure.

Another older method that is still commonly used today is called the principle factors method, not to be confused with the principle component method. The only input for this method is either the sample covariance matrix $\mathbf{S}$ or sample correlation matrix $\mathbf{R}$.

First, an estimate of each communality $\hat{h}_i^2$ is obtained. Each estimate is found by first performing a multiple regression for each variable $y_i$ regressed on the other $D-1$ variables $y_j (j \neq i)$. The coefficient of determination $R_i^2$, also known as the squared multiple correlation coefficient, is obtained for each regression. The communality is then estimated to be either $s_{ii}R_i^2$ or $s_{ii} \max_{j \in (1 \ldots D)} |r_{ij}|$ if using the covariance matrix or either $R_i^2$ or $\max_{j \in (1 \ldots D)} |r_{ij}|$ if using the correlation matrix. If $\mathbf{R}$ is invertible it can be shown that $R_i^2$ can be found from $\mathbf{R}$ meaning that only $\mathbf{Y}$ is required to find this initial estimate of the communalities [18, 20].

Next we define the reduced covariance matrix as $\mathbf{S} - \hat{\mathbf{\Psi}}$ which is just $\mathbf{S}$ with the diagonal elements replaced by $\hat{h}_i^2$. A $d$ rank eigendecomposition of the reduced covariance is then

$$\mathbf{S} - \hat{\mathbf{\Psi}} = \mathbf{N}_d \mathbf{\Lambda}_d \mathbf{N}_d^\top \tag{2.26}$$

where $\mathbf{\Lambda}_d$ is a diagonal matrix containing the first $d$ largest eigenvalues in decreasing order and $\mathbf{N}_d$ contains the corresponding eigenvectors in each column. Equation 2.26 can be rewritten as

$$\mathbf{S} - \hat{\mathbf{\Psi}} = \mathbf{N}_d \mathbf{\Lambda}_d^{1/2} \mathbf{\Lambda}_d^{1/2} \mathbf{N}_d^\top = \hat{\mathbf{A}} \hat{\mathbf{A}}^\top$$

with $\hat{\mathbf{A}} = \mathbf{N}_d \mathbf{\Lambda}_d^{1/2}$. The communalities can then be re-estimated to be the diagonals of the rank $d$ decomposition of the reduced covariance matrix and the process can be repeated until some convergence criteria is met [18, 20]. Essentially, the principle factor method estimates the factor model by performing PCA on the reduced covariance matrix $\mathbf{S} - \hat{\mathbf{\Psi}}$ and iterating [17]. This method has little more theoretical justification than the centroid method, but it is one of the more common approaches. Note that as this method relies on

spectral decomposition and therefore scale equivariance does not hold and will give different solutions if the covariance matrix is chosen instead of the correlation matrix [20].

With a slight tweak, the principal factor method becomes the principle component method of factor analysis. By replacing the estimate of the communalities with zeros, the reduced covariance matrix becomes the covariance matrix $\mathbf{S} - \hat{\mathbf{\Psi}} = \mathbf{S}$ and the factor analysis solution reduces to the principle component solution. However, this solution is problematic as the assumption that $\mathbf{\Psi} = 0$ is in general not true and leads to some of the FA model assumptions not being satisfied [20, 17].

The PCA projection, Equation 2.20, repeated here $\mathbf{Z} = \mathbf{W}^\top \mathbf{Y}$, can be rearranged to better compare the two techniques. First, since $\mathbf{W}$ is orthogonal, $\mathbf{Y} = \mathbf{W}\mathbf{Z}$. $\mathbf{W}$ can then be decomposed into two submatricies $\mathbf{W} = (\mathbf{W}_d | \mathbf{W}_{D-d})$ containing the eigenvectors corresponding to the $d$ largest eigenvalues and the remaining $D-d$ eigenvalues respectively of the covariance matrix $\mathbf{\Sigma}$. Decomposing $\mathbf{Y}$ in a similar fashion as $\mathbf{W}$ gives $\mathbf{Z} = \begin{pmatrix} \mathbf{Z}_d \\ \hline \mathbf{Z}_{D-d} \end{pmatrix}$ resulting in

$$\mathbf{Y} = (\mathbf{W}_d | \mathbf{W}_{D-d}) \begin{pmatrix} \mathbf{Z}_d \\ \hline \mathbf{Z}_{D-d} \end{pmatrix} = \mathbf{W}_d \mathbf{Z}_d + \mathbf{W}_{D-d} \mathbf{Z}_{D-d}$$

which can be transformed into the factor model

$$\mathbf{Y} = \mathbf{W}_d \mathbf{\Lambda}_d^{1/2} \mathbf{\Lambda}_d^{-1/2} \mathbf{Z}_d + \mathbf{W}_{D-d} \mathbf{Z}_{D-d} = \mathbf{A}\mathbf{X} + \mathbf{\Gamma}$$

where $\mathbf{A} = \mathbf{W}_d \mathbf{\Lambda}_d^{1/2}$, $\mathbf{X} = \mathbf{\Lambda}_d^{-1/2} \mathbf{Z}_d$, and $\mathbf{\Gamma} = \mathbf{W}_{D-d} \mathbf{Z}_{D-d}$. Now we can check if the FA model assumptions are fulfilled by this solution.

$$\begin{aligned} \mathbb{E}[\mathbf{x}\mathbf{x}^\top] &= \mathbf{X}\mathbf{X}^\top \\ &= \mathbf{\Lambda}_d^{-1/2} \mathbf{Z}_d \mathbf{Z}_d^\top \mathbf{\Lambda}_d^{-1/2} \\ &= \mathbf{\Lambda}_d^{-1/2} \mathbf{\Lambda}_d \mathbf{\Lambda}_d^{-1/2} \\ &= \mathbf{I} \end{aligned}$$

where the key is that since $\mathbf{W}$ contains the eigenvectors of $\mathbf{Y}\mathbf{Y}^\top$ in its

columns the equation below is just an eigendecomposition

$$\mathbf{Y}\mathbf{Y}^\top = \mathbf{W}\mathbf{Z}\mathbf{Z}^\top\mathbf{W}^\top$$

meaning that $\mathbf{Z}_d\mathbf{Z}_d^\top = \mathbf{\Lambda}_d$. The second assumption is also fulfilled

$$\mathbb{E}[\mathbf{x}\boldsymbol{\gamma}^\top] = \mathbf{X}\mathbf{\Gamma} = \mathbf{\Lambda}_d^{-1/2}\mathbf{Z}_d\mathbf{Z}_{D-d}^\top\mathbf{W}_{D-d}^\top = \mathbf{0} \tag{2.27}$$

However, the final assumption

$$\mathbb{E}[\boldsymbol{\gamma}\boldsymbol{\gamma}^\top] = \mathbf{\Gamma}\mathbf{\Gamma}^\top = \mathbf{W}_{D-d}\mathbf{Z}_{D-d}\mathbf{Z}_{D-d}^\top\mathbf{W}_{D-d}^\top = \mathbf{W}_{D-d}\mathbf{\Lambda}_{D-d}\mathbf{W}_{D-d}^\top \neq \mathbf{\Psi}$$

because although $\mathbf{\Lambda}_{D-d}$ is diagonal it has, in general, non-uniform values along the diagonal meaning that $\mathbf{\Psi}$ will not be diagonal indicating correlation between unique factors [20, 17].

#### 2.2.1.3 Independent Component Analysis

Independent Component Analysis (ICA) refers to a collection of techniques for finding a linear transformation of multivariate data into new features that, as the name implies, are statistically independent from each other. The first ICA technique was developed by Jutten as part of his PhD thesis and by Jutten and Hrault in the late 1980's and early 1990's [26, 27]. The following discussion of ICA was condensed from the review paper by Hyvrinen [28]. There are two basic formulations that are referred to as the noiseless and noisy cases respectively

$$\mathbf{Y} = \mathbf{A}\mathbf{X} \tag{2.28}$$

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{\Gamma} \tag{2.29}$$

where $\mathbf{Y}$ is the $D \times n$ data matrix as defined throughout this section, $\mathbf{A}$ is the $D \times d$ mixing matrix, $\mathbf{X}$ is a $d \times n$ matrix of representing $n$ samples of $d$ latent variables and is sometimes called the source matrix, and $\mathbf{\Gamma}$ is a $d \times n$ matrix representing $n$ samples of a $d$ dimensional noise vector. Looking back at Equations 2.20 and 2.24, it is easy to see the resemblance of ICA to both PCA and FA.

To make the distinction clear between ICA and these other classical meth-

ods it is important to remember the difference between independence and correlation. Two variables are said to be uncorrelated when

$$\mathbb{E}[xy] = \mathbb{E}[x]\,\mathbb{E}[y]$$

The condition for independence is stronger requiring the two variables' joint probability density to factor into the product of their marginal densities

$$\boldsymbol{f}_{xy}(i,j) = \boldsymbol{f}_{x}(i)\boldsymbol{f}_{y}(j)$$

PCA and FA are both second order techniques, meaning that they rely solely on second order statistics, such as the correlation or covariance (assuming the data has been centered). The major difference between these methods and ICA is that it relies on higher order statistics, e.g. the fourth moment, kurtosis, as a measure of independence. Although the PCA and FA methods do not technically make any distributional assumptions, since they are minimizing correlations, they are only finding independent components if all of the latent variables are Gaussian. Therefore, if we are truly interested in finding independent latent variables then PCA and FA may not be adequate. In fact, if latent variables are mixed non-linearly $\mathbf{Y} = \mathrm{f}(\mathbf{X})$, linear ICA will only be capable of providing an approximate fit, but that will not be covered in this review.

Often some form of data preprocessing is required before performing ICA. The most common requirement among the different ICA techniques is that the data must be whitened

$$\mathbf{Y}_w = \mathbf{Q}\mathbf{Y} \tag{2.30}$$

where $\mathbf{Y}_w$ is the whitened data and $\mathbf{Q}$ is square $D$ dimensional whitening matrix that results in the covariance matrix of the whitened random variable being an identity matrix $\mathbb{E}[\mathbf{y}_w\mathbf{y}_w^\top] = \mathbf{I}$. This is commonly accomplished using PCA or eigendecompostition

$$\mathbf{Q}_{pca} = \mathbf{\Lambda}^{-1/2}\mathbf{N}^\top \tag{2.31}$$

where $\mathbf{\Lambda}$ and $\mathbf{N}$ are the diagonal matrix of decreasing eigenvalues and the corresponding eigenvectors or the covariance matrix $\mathbf{\Sigma} = \mathbf{Y}\mathbf{Y}^\top$ respectively as defined in Equation 2.21. Combining Equation 2.30 with Equation 2.29

then gives

$$\mathbf{Y}_w = \mathbf{G}\mathbf{X} \tag{2.32}$$

where $\mathbf{G} = \mathbf{Q}\mathbf{A}$ and $\mathbf{G}$ is an orthogonal matrix because the covariance of the whitened data can be written as

$$\mathbf{\Sigma} = \mathbb{E}[\mathbf{y}_w \mathbf{y}_w^\top] = \mathbf{G}\,\mathbb{E}[\mathbf{y}\mathbf{y}^\top]\mathbf{G}^\top = \mathbf{G}\mathbf{G}^\top = \mathbf{I} \tag{2.33}$$

as was the desired effect of whitening. This whitening actually makes the problem of finding a solution to the ICA problem easier, because obtaining an arbitrary matrix $\mathbf{A}$ to satisfy 2.29 now becomes the easier problem of finding an orthogonal matrix $\mathbf{G}$.

Now, up to this point we have just considered ICA a matrix decomposition method that produces optimally independent components. To reduce the dimensionality of a matrix as well, the data preprocessing step normally involves excluding a number of principle components as is done in PCA.

$$\mathbf{Q}_{pca} = \mathbf{\Lambda}_d^{-1/2} \mathbf{N}_d^\top$$

This is somewhat justified if it is assumed that the noise is low and therefore energy of $Y$ is concentrated in the subspace spanned by the first $d$ principle components.

As mentioned earlier, ICA refers to a collection of techniques where the techniques differ based on the specific objective function they choose optimize and the optimization method used to arrive at a solution. Options for objective function include log likelihood, network entropy or infomax, mutual information or KL divergence, negentropy, general contrast functions, kurtosis and other measures of non-Guassianity, and many more. The solution methods are all incremental and typically are presented as an update function of the inverse of the mixing matrix sometimes referred to as the unmixing matrix $\mathbf{W} = \mathbf{A}^{-1}$. One of the earlier algorithms performs a gradient ascent on an infomax objective function leading to the update rule

$$\mathbf{\Delta W} \propto [\mathbf{W}^\top]^{-1} - 2\tanh(\mathbf{W}\mathbf{y})\mathbf{y}^\top \tag{2.34}$$

A very commonly used ICA method is called FastICA and is based on a batch style fixed point iteration optimization method for general contrast functions

giving the following update equation

$$\mathbf{w}(k) = \mathbb{E}[\mathbf{x}\mathrm{g}(\mathbf{w}(k-1)^\top \mathbf{x})] - \mathbb{E}[\mathrm{g}^{'}(\mathbf{w}(k-1)^\top \mathbf{x})]\mathbf{w}(k-1) \tag{2.35}$$

where g() is the derivative of a nonquadradic nonlinearity function used in the contrast function, $\mathrm{g}^{'}()$ is its second derivative, and $\mathbf{w}$ is normalized after each update.

Note that this section was meant to provide a cursory look at ICA as it relates to the problem of dimensionality reduction. For a more comprehensive look at this collection of techniques, see the review by Hyvrinen [28].

#### 2.2.1.4 Non-Negative Matrix Factorization

Non-Negative Matrix Factorization (NMF) is a newer dimensionality reduction technique, compared to PCA and FA, developed in 1999 by Lee and Seung [29]. As its name hints, NMF is a matrix factorization method that can be applied to exclusively non-negative data represented as a $D \times n$ matrix $\mathbf{Y}$. Of course, PCA and FA can be applied to non-negative data, but will give negative values in the feature/loading and score/factor matrices which can be difficult to interpret. NMF is unique because it imposes positivity constraints on every element of the decomposing matrices. This constraint has been shown to produce features that can have semantic meaning [29]. The current reasoning for why NMF finds semantically meaningful features is that the model's positivity constraint mirrors the human interpretation that objects are composed of separate parts.

$$\mathbf{Y} = \mathbf{WH} \tag{2.36}$$

where for $d$ features $\mathbf{W}$ is a $D \times d$ matrix and $\mathbf{H}$ is a $d \times n$ matrix.

Depending on what is cost function or distance metric is chosen, different solutions arise that are detailed in the 2001 paper [30] and discussed below. The most common NMF solution methods are iterative and utilize multiplicative update rules, meaning that the current estimate is multiplied by some factor to generate the new estimate. If we choose to minimize the squared Euclidian distance metric $\|\mathbf{V} - \mathbf{WH}\|^2$, the least squares error, subject to the non-negativity constraints $W_{ik}, H_{kj} > 0 \; \forall \; i, j, k$, then the update

equations become

$$H'_{kj} \leftarrow H_{kj} \frac{(\mathbf{W}^\top \mathbf{Y})_{kj}}{(\mathbf{W}^\top \mathbf{W}\mathbf{H})_{kj}} \qquad W'_{ik} \leftarrow W_{ik} \frac{(\mathbf{Y}\mathbf{H}^\top)_{ik}}{(\mathbf{W}\mathbf{H}\mathbf{H}^\top)_{ik}} \tag{2.37}$$

for $i \in (1 \dots D), j \in (1 \dots d), k \in (1 \dots n)$ where the $kj$ subscript of $H_{kj}$ indicates the element of $\mathbf{H}$ in the $k^{th}$ row and $j^{th}$ column, $W_{ik}$ is defined similarly, and the prime $'$ indicates the updated estimate. This Euclidian distance is non-increasing under these rules and is invariant *if and only if* $\mathbf{W}$ and $\mathbf{H}$ are at a stationary point of the distance, which occurs when $\mathbf{Y} = \mathbf{W}\mathbf{H}$.

If instead we choose to minimize a slightly modified form of the Kullback-Leibler (KL) divergence that we will refer to simply as the Divergence

$$D(\mathbf{X} \| \mathbf{W}\mathbf{H}) = \sum_{ij} \left( X_{ij} \log \frac{X_{ij}}{(\mathbf{W}\mathbf{H})_{ij}} - X_{ij} + (\mathbf{W}\mathbf{H})_{ij} \right) \tag{2.38}$$

subject to the non-negativity constraints $W_{ik}, H_{kj} > 0 \ \forall \ i, j, k$, we end up with a different set up of update equations

$$H'_{kj} \leftarrow H_{kj} \frac{\sum_a W_{ak} X_{aj} / (\mathbf{W}\mathbf{H})_{aj}}{\sum_a W_{ak}} \quad W'_{ik} \leftarrow W_{ik} \frac{\sum_b H_{kb} X_{ib} / (\mathbf{W}\mathbf{H})_{ib}}{\sum_b H_{kb}} \tag{2.39}$$

where the matricies are defined as above. This Divergence is non-increasing under these rules and is invariant *if and only if* $\mathbf{W}$ and $\mathbf{H}$ are at a stationary point of the divergence, which occurs when $\mathbf{Y} = \mathbf{W}\mathbf{H}$.

### 2.2.2 Perceptual Primitives and Receptive Fields

The previous discussion of dimensionality reduction was motivated by an idea expressed by the neuroscientist Horace Barlow in 1961 [31]. Barlow studied visual processing and his observations on the structure and organization of sensory neurons led him to ask if there is an underlying principle that explains why different sensory processing neurons respond to different ranges of stimuli. At this time, the idea that reduction in dimensionality is an important characteristic of sensory processing was gaining popularity. Barlow provided the interesting hypothesis that receptive fields were optimized to reduce the dimensionality of the incoming sensory information by extracting signals of high relative entropy. The term receptive field, originally coined

by Sherrington, is used here to describes a region of the sensory space that elicits a response by a specific neuron.

But, reduced dimensionality is not an intrinsically desirable characteristic. It is possible that a high dimensionality is required to represent the range of stimuli adequately and reducing the dimensionality of that signal would discard useful information. Dimensionality reduction is really only useful when the intrinsic dimensionality of the signal is lower than that of the signal itself. Therefore, what Barlow's hypothesis indirectly implies is that there is structure in our sensory observations, and, consequently, structure in the world. It is this structure in the world that our sensory processing systems have adapted to reflect in order to efficiently process information. By taking advantage of the statistical regularities in the sensory signals, the dimensionality of the observations are reduced, simplifying computational demands in the process. There is a growing body of evidence that suggests Barlow's intuition was correct and bolsters this structured world interpretation. We will review some of that evidence here, focusing on results from the visual and auditory sensory domains.

#### 2.2.2.1 Visual Primitives

Neuroscientists Hubel and Wiesel investigated the receptive fields of neurons in the visual processing stream of cats. In a string of experiments, they recorded electrical signals from neurons in the primary visual cortex (V1) of cats as they projected different simple black and white patterns on the cats' retinas [32, 33]. These experiments revealed that certain cells were sensitive to specific orientations of lines and gratings within a specific region of the visual space. They deemed neurons with this behavior simple cells. They also discovered neurons that responded to oriented lines and gratings similarly as simple cells, but exhibited some invariance as to the location of the stimulus on the retina or visual space. In other words, these other neurons were sensitive to oriented patterns not just in one location of the visual space, but within a range of locations. They deemed these cells with spatial invariant properties complex cells. Later research helped to show that complex cells are aggregators of responses from simple cells responding to the same orientated lines at different locations in the visual space, enabling their spatial invariance properties [34].

Simple and complex cells have also been shown to be sensitive to specific spatial frequencies [35]. Simple cells however, still exhibit the property of localization, meaning that they can not be simply considered spatial frequency detectors and represented via a 2-D Fourier transform. However, the Gabor filter has been shown to be a good approximation of simple cell receptive fields [35]. A 2D Gabor filter is simply the multiplication of a Gaussian kernel with sinusoidal plane wave. One can consider simple cells a type of visual primitive. Each simple cell's receptive field can be considered a feature or basis, and in the linear case, a column of the feature matrix as in Equation 2.20.

In fact, many computer vision systems use Gabor filters and similar edge filters to generate features that are used in object detection, character recognition, and movement tracking algorithms. But what is truly remarkable is that experiments that apply dimensionality reduction techniques to images of natural environments produce features that resemble the receptive fields of simple cells. To apply dimensionality reduction to $h \times w$ dimensional images, one can apply a vectorization operation $vec()$ to each image $I$ which successively stacks the columns of the image into a $h \cdot w \times 1$ vector. The inverse of the the vectorization operaiton $vec^{-1}()$ can later be used to reconstruct a vectorized matrix.

Bell and Sejnowski utilized a natural gradient method of infomax ICA to develop a set of visual features from image patches from pictures of natural scenes of trees and leaves [36]. These image patches were first vectorized and then placed in the columns of $\mathbf{Y}$ as in Equation 2.29. Olshausen and Field took a slightly different approach and developed a linear decomposition to maximize sparsity of the basis [37]. Interestingly, both of these approaches result in filters that resemble the receptive fields of cells in the visual cortex. One reason for that may be that the sparsity constraint is very closely related to the independence constraint [38].

For some computer vision applications it sometimes makes sense to develop larger scale features from an entire image instead of image patches. For example, for the task of face recognition, computer scientists have developed "face" features by performing dimensionality reduction on entire images of faces, where each face image is vectorized and becomes a column in $\mathbf{Y}$. Turk and Pentland first proposed this concept and developed what they called the eigenfaces [39]. They framed the problem in the context of information

theory, where the goal was to find an efficient code to extract the information from a facial image that would enable facial recognition. However, at the time, ICA was a very new technique and the infomax method had not yet been created. They instead used PCA to construct their eigenfaces, which generates uncorrelated components that approximate independence using second order statistics. The name eigenfaces is apt as they are comosed of the columns of the feature matrix in Equation 2.20 which contain the eigenvectors of the correlation matrix. Turk and Pentland were able to use these eigenfaces to perform face classification and face detection in a constrained setting. The eigenfaces can be visualized by performing an inverse vectorization operation on each column of the feature matrix in Equation 2.20.

$$\mathrm{k}^{th} \text{ eigenface} = vec^{-1}(\mathbf{w}_k)$$

The eigenfaces are somewhat face-like in appearance, but have both positive and negative components which make them difficult to interpret.

The NMF algorithm was actually created to address this problem of component interpretation [29]. Lee and Seung cite psychological and physiological evidence pointing to a parts based representation in the brain as motivation for developing features that are more efficient and more easily interpreted than the holistic "face" features of eigenface and similar approaches [40]. They hypothesized that in order to produce more interpretable components for data that is inherently non-negative, the basis must be entirely additive and not rely on inter-feature cancellation, leading to the non-negativity constraints of NMF [29]. The NMF algorithm does produce more component based face features. In fact, it picks up on facial features that we have names for, such as noses, eyes, lips, beards, etc. and represents these each as individual components. There is some evidence to indicate that cells in the visual cortex may have receptive fields that are somewhat component based [40]. It is possible that these cells exist at a higher level of the visual processing hierarchy and are aggregating inputs from other cells earlier in the chain of processing such as simple cells or complex cells.

These higher level "face" features are interesting, but have limited use because they are not invariant to translation, scale, or angle of the face to the camera. One way of adding invariance is through use of a hierarchy, which is the approach taken by Coates et al [41]. They used K-means clustering

to construct what they call simple cells. It is important to point out that K-means is a technique that gives very similar results to that of PCA [42]. In fact, K-means can be thought of as a sparse PCA that maximizes the same least squares objective function but with the addition of a categorical constraint. So, although we have only covered PCA, the simple cells produced using k-means can be thought of as behaving similarly. The output of the simple cells is fed into an agglomerative clustering algorithm which acts as a means for performing max-pooling. Pooling refers to a general technique of condensing multiple signals or responses into a single signal. Max-pooling specifically assigns the output of a polling unit to be the maximum value of all the input signals. Coates et al. call these max-pooling units complex-cells, because the pooling groups are chosen in such a way to enable the outputs to be invariant to translations of features [41]. To build a deeper hierarchy, Coates et al. alternatingly stacks layers of complex cells and simple cells on top of each other.

They applied this approach to the problem of learning a hierarchy of image features. First, they collected their training data: a set of 52 million $32 \times 32$ pixel image patches by randomly sampling YouTube videos. They then used these images to train a 4 layer feature hierarchy of alternating simple and complex cells [41]. As the training data was sampled randomly from YouTube and no supervision was used, the images are of partial views of objects and clutter, unlike the previously discussed NMF and eigenface experiments which rely on images cropped to have centered faces [39, 29]. However, Coates et al. estimate the the most commonly occurring object category is a human face, but that it well-framed images of faces account for less than 0.1% of the image patches. So, it is somewhat surprising that a number of the second tier simple cell receptive fields resemble partial views of faces. In addition, there are a number of second tier complex cells that also resemble faces and are maximally activated when the network is passed images of faces at various angles and scales.

The lower level features are also interesting. The simple cells resemble edge and spot detectors much like biological simple cells. And some of the lower level complex cell pooling units are comprised of simple cells with very similar structure, often containing edge-like filters at various translations. What these results demonstrate is that through hierarchical unsupervised learning, it is possible to develop features that exhibit some properties of

invariance similar to those of neurons found in the visual processing stream.

#### 2.2.2.2 Auditory Primitives

The field of auditory and, more specifically, speech processing was primarily advanced by the telecommunications industry during the creation of the telephone network. In fact, the study of auditory signals led to the development of many of the tool used in signal processing today [7]. Consequently, development of auditory primitives precedes that of visual features. There are some significant differences between auditory signals and visual signals. Unlike visual signals, which are very high dimensional vectors or matrices, raw audio signals are normally 1 or 2 dimensional time indexed signals. In addition, while time can be somewhat ignored when developing visual primitives, it is an absolutely essential component in characterizing auditory signals.

Audio signals can be thought of as being composed of sums of pure sin waves of different amplitudes through the use of a Fourier decomposition. One of the most common approaches is to decompose the audio signal into a time-frequency representation via a short time Fourier transform (STFT). The STFT was first introduced by Gabor in 1946 [43] and consists of applying the DFT successively to a windowed version of the time domain signal as below

$$\hat{\mathbf{x}}_\tau = \mathcal{F}^N(\mathbf{x}_\tau \otimes \mathbf{w}) \tag{2.40}$$

where $\hat{\mathbf{x}}_\tau = [f(\tau - (N-1)/2) \cdots f(\tau + (N-1)/2)]^\top$ is the length $N$ segment of the signal centered at sample $\tau$, $\mathcal{F}^N_{(k,n)} = \exp 2\pi i \frac{kn}{N}$ is the $k^{th}, n^{th}$ element of the $N \times N$ discrete Fourier transform (DFT) matrix corresponding to the $k^{th}$ frequency , $\hat{\mathbf{x}}_\tau = [\hat{x}(\tau, 1) \cdots \hat{x}(\tau + N, 1)]^\top$ is the DFT of the $N$ sample signal segment centered at sample $\tau$ at discrete frequencies $k/N$, and $w$ is a window function that is used to reduce artifacts induced by truncating the signal. The STFT is finally given by combining the column vectors $\hat{\mathbf{x}}_\tau$ into a matrix

$$\hat{\mathbf{X}} = [\hat{\mathbf{x}}_{\tau_1}, \hat{\mathbf{x}}_{\tau_2}, \cdots] \tag{2.41}$$

where the overlap $l = \tau_j - \tau_i$ for $j = i + 1$. determines the spacing between window centers. See [5, 44] for a more thorough overview.

The transform can be thought of as applying a Fourier transform to successive overlapping segments of a signal. This is often referred to as a sliding

window. This approach enables building a time indexed frequency-amplitude decomposition and allows us to see how the spectrum of a signal varies over time. The STFT, and similar transforms, have proven to be extremely valuable tools in understanding audio signals including speech and music.

The spectrogram, which is used heavily in phonetics, is either the magnitude or power of the STFT and is typically plotted on a log scaled frequency axis. One interesting limitation of the STFT is that there is a trade-off between time resolution and frequency resolution of the transform in accordance with the Heisenberg uncertainty principle, meaning that the we cannot precisely identify the frequency of a signal at a specific time [43]. This is related to the window width $N$; as we decrease $N$ the STFT time resolution improves, but the STFT frequency resolution decreases. A similar time-frequency representation called the wavelet transform was designed to overcome the uncertainty problem by using narrower windows for transforming higher frequencies and wider windows for lower frequencies.

One of the reasons for for viewing audio signals in the time-frequency plane is that the human ear has been shown to perform a similar harmonic decomposition. This decomposition is performed by the cochlea, which is a hollow fluid filled spiraled shell containing something called the basilar membrane which resonates with incoming sounds. The basilar membrane is lined by the Organ of Corti, which is peppered with groups of hair cells called sterocilia that convert mechanical movement of the fluid into electrical signals. In other words the stereocilia are transducers. When a sound wave hits the ear drum it cause the membrane to oscillate. That oscillation is relayed to the cochlea through a linkage of 3 bones causing displacement of the fluid within the cochlea. Due to the mechanical characteristics of the cochlea, the basilar membrane resonates at different locations along it corresponding to different frequencies, thereby performing a harmonic analysis. The changing resonant frequency along membrane can be described using a place-frequency mapping.The inner ear is very complicated and has been studied extensively. In fact, the cochleogram is a time-frequency transform that utilizes a linear model of the basilar membrane and a leaky integrator model of stereocilia activations to produce a power spectrum representation similar to that of a spectrogram [45]. This representation is not as commonly employed due to its increased computation requirements, but advocates of this technique argue that it is more physiologically faithful and doesn't introduce disconti-

nuity artifacts that windowing approaches suffer from. Other features that emulate the response of the cochlea include gammatone filterbanks, correlograms, and the weft. A variety of additional transform methods and audio features have been engineered over the years including the discrete cosine transform (DCT), which is a variation of the Fourier transform that has only real values, and the Mel-frequency cepstral coefficients (MFCCs), which is the DCT of the log power spectrum of the signal.

However, another approach to developing features is to use the data to create a new basis. This approach is motivated by Barlow's hypothesis that sensory processing systems have evolved to represent precepts optimally with respect to the statistics of the environment [31]. This can be done through LPC [5], as discussed earlier in Section 2.1.3, or by applying dimensionality reduction techniques. Although much of the work in learning perceptual primitives has focused on the visual domain, similar approaches have been taken in the auditory domain. To apply a dimensionality reduction method, the audio signal must first be placed into a matrix. This is accomplished using the sliding window method as we did for the STFT. For a single channel system with signal $f(t)$ a window of width $N$ can be taken specifying the length of the primitive. The data matrix $\mathbf{Y}$ is then constructed by setting successive length $N$ signal samples $\mathbf{f}_\tau = [f(\tau+1) \cdots f(\tau+N)]^\top$ as the columns [44]. Then, techniques such as PCA, ICA, and NMF can be applied.

Initial data based approaches focused on learning features using artificial neural networks with Hebbian based learning rules. These methods have been described as very similar to PCA [46]. When PCA is applied to speech sounds, it produces a collection of approximate sinusoids of varying frequency. Interestingly, this PCA basis derived from speech sounds very closely resembles the DCT components [44], which are just frequency localized sinusoids [47]. And, it has been shown that the DCT is asymptotically equivalent to PCA applied to time coherent data [48].

But, the PCA approach suffers from the inability to localize events in time similar to that of the DFT. One way of dealing with this is to develop a time localized basis functions using ICA which relies on higher order statistics to ensure independence between the features. In fact, when applied to speech sounds ICA produces frequency and phase localized sinusoids [44, 49, 46]. In addition the ICA features enable a much more sparse encoding of sounds compared to PCA [44]. In other words, a fewer number of ICA bases is

required to achieve the same reproduction accuracy as is with PCA. This evidence supports Barlow's initial hypothesis that sensory processing systems have evolved to encode common precepts optimally in a statistical sense.

An even stronger case for this hypothesis was made by Lewicki who applied ICA to different collections of natural sounds [49]. He found that ICA applied to sounds of animal vocalizations resulted in features similar to those of a Fourier transform. When applied to non-biological environmental sounds, the resulting features resembled a wavelet transformation. But, he found that when applied to an ensemble of both sets of sounds, the algorithm produced features that even more closely match biological data. Specifically the features from the ensemble of sounds exhibit a sublinear power law relationship of filter sharpness versus center frequency that resemble the distribution of tuning frequencies along the length of the cochlea. In addition, Lewicki points out that ICA applied to human speech produces results very similar to those of the ensemble. This may imply that speech evolved to encode communications optimally with respect to existing perceptual processing ability.

These dimensionality reduction approaches produce interesting and efficient primitives for low-level encoding of auditory signals, but it is often useful to obtain primitives on a longer time scale with more invariant properties. In order to develop higher level features, a hierarchy can be constructed. Lee et al. took an unsupervised deep learning approach to creating higher level features [50]. They trained a convolutional deep belief net (CDBN) on whitened spectrograms of speech samples from the TIMIT database. CDBNs are formed by successively training restricted Boltzmann machines (RBMs) and stacking them on top of each other. Some of the low level features learned by the CDBN were shown to correspond to individual phones. Interestingly, when the first layer features were used to perform a phone recognition task they underperformed the use of standard MFCCs significantly. However, when the first layer features were combined with MFCCs an improvement of 0.7% was achieved. The accuracy of phone classification using the higher level features however, was not reported. Lee et al. also used the features to perform gender classification. Interestingly, the training of a classifier using the high level features proved to be the most accurate method compared to use of MFCCs, low level features, and combined low and high level features. This suggests that the high level features learned invariance that preserves gender information.

### 2.2.3 The Motor Equivalence Problem

Consider the seemingly simple action of striking a chisel with a hammer. At first glance there is nothing particularly outstanding about this movement; it is a task that most humans are capable of. However, when one takes into account the complex and highly redundant structure of the human motor system this gesture can be seen as truly spectacular. Nikolai Bernstein was one of the first people to study coordinaiton of motion. In fact, many of the first movements that he studied were of industrial workers performing their jobs, including the action of hammering [51]. He observed that there is a great deal of redundancy in the motor system, meaning that humans have many more degrees of freedom (DOFs) than are required to perform a motor task.

This redundancy arises at several levels. A motor unit consists of an alpha motor neuron which innervates a number of muscle fibers within an individual muscle [52]. Motor neuron cell bodies are clustered in columns within the spinal cord making up motor pools. Each muscle pool exclusively contains all of the motor neurons that innervate a single muscle. The force that a muscle exerts is affected by both the rate at which individual motor neurons fire , rate coding, and the number of motor units that are being recruited. So, there is a redundancy at the motor level as there are many more motor units than necessary to exert a specific force from a muscle. One way that we know of that the nervous system uses to reduce this redundancy is via the motor neuron size principle. The size principle states that motor units are recruited in order of smaller motor units to larger motor units as the strength of the input to the motor pools increase. However, additional redundancy exists at the joint level where multiple muscles affect the torque about different limb joints. Redundancy also exists at the limb level. For example the positioning of the hand in 3-D space requires 6 DOFs, but the human arm has 7 DOF. Another level of redundancy exists at the level of the limb trajectory. In the task of hammering it can be seen that there are many different joint trajectories that result in the nail being struck.

The problem of how to coordinate muscle activations for performing specific task given that they system is highly redundant is known as the motor equivalence problem or the degrees of freedom problem. Stated more succinctly the motor equivalence problem is the idea that there are many motor

actions available that will achieve a single motor goal. In many cases there is actually an infinite number of solutions to a given task. From a control theory perspective this is an extremely challenging problem. Yet, humans can not only hammer chisels, but they can build houses, perform gymnastics, and produce speech. Many different theories have been put forth to address this problem and include ideas from neuroscience and the study of motor control as well as models adopted from the domains of control theory and robotics.

#### 2.2.3.1 Motor Primitives

The term motor primitives refers to the general idea that motor actions and motions are composed of fundamental building blocks. This hypothesizes that the entire repertoire of motor actions is spanned by these motor primitives and specific transformations of them [53]. These primitives take many forms, may be part of a complex motor control hierarchy, and may capture coordination at the neural, joint, kinematic, and/or dynamic levels of movement.

Bernstein pioneered the concept of motor or muscle synergies as one potential solution to the motor equivalence problem [51]. A synergy is generally used to describe a weighted co-activation of muscles [54, 55]. For a system with $D$ muscles and $N$ primitives this can be represented as a vector matrix product

$$\mathbf{m} = \mathbf{cw} \tag{2.42}$$

where $\mathbf{m}$ is a $D$-dimensional vector representing the muscle activations, $\mathbf{c}$ is an $N$-dimensional vector of synergy weightings, and $\mathbf{w}$ is a $D$-dimensional synergy or co-activation of muscles [55]. However, the term synergy has taken on a number of meanings within the motor control domain. Most uses of muscle synergy refer to either time-invariant also known as spatial synergies

$$\mathbf{m}(t) = \mathbf{c(t)w} \tag{2.43}$$

which allow the weightings to vary over time but describe a static co-activation

of muscles, or time-varying synergies also known as spatiotemporal synergies

$$\mathbf{m}(t) = \sum_{i=1}^{N} c_i \mathbf{w}(t - t_{oi})\delta(t - t_{oi}) \tag{2.44}$$

which capture temporal regularities in muscle co-activations as well.The function $\delta(t - t_{io})$ in Equation 2.44 is the dirac delta function which indicates that spatiotemporal synergies may be combined asynchronously in time.

The first indication that synergies or motor primitives could describe the method by which the CNS overcomes the motor equivelence problem was provided by Bernstein [51]. In his pioneering work in the study of human movements, he recognized that rhythmical motions in particular, such as hammering or walking, could be represented to within 1-3mm or 1% of the total movement as a Fourier series of only 3 to 4 terms. Each term of a Fourier series has only two variables, an amplitude and a phase shift meaning that it is possible to represent these complex coordinated movements that Bernstein studied with only 6 to 8 parameters. This is, in effect, a massive reduction in dimensionality. To Bernstein, the fact that there exists such a high degree of structure in these motions points to the existence of a highly structured internal physiology that produces these motions.

A number of studies have provided evidence for Bernstein's intuition by showing that microstimulation of spinal cord premotor neural circuits generate balanced contractions in muscle groups[56, 57, 58] indicating that low dimensional representations may help to generate complex movements. One commonly cited study measured position dependent force fields generated by stimulation of individual spinal circuits in frog hind limbs [59]. They showed that the force fields generated by stimulating individual spinal circuits sum vectorially when stimulating them simultaneously. This was expected for non-redundant configurations of the leg, but it was surprising that this held for redundant configurations where many possible force field combinations could have produce the summed force. These results suggest that spinal synergies or motor primitives could be responsible for part of this coordination that Bernstein first observed.

The concept of neural timing circuits in the spinal cord called central pattern generators (CPGs) has similarities to spatiotemporal synergies and Bernstein's early work in that CPGs produce rhythmic coordinated muscle

activations [60, 61]. CPGs are hypothesized to be capable of oscillating independently of afferent sensory inputs or efferent rhythmic stimulation. They do require some sort of efferent activation and allow for top-down modulation. These neural circuits may aid in a variety of rhythmic motor control processes such as respiration, walking, swimming, flying. Most evidence supporting the idea of CPGs comes from either studies of fictive motor patterns in which neural circuits in extracted spinal tissue are stimulated and their firing patterns observed, or from studies using deafferented animals [62, 63, 64].

Some other evidence for the existence of synergies comes from studies in electrophysiology where electromyographical (EMG) signals are decomposed into lower dimensional spaces using dimensionality reduction techniques such as Principle Component Analysis (PCA) [65], Independent Component Analysis (ICA)[66], or non-Negative Matrix Factorization (NMF) [67]. Other approaches have applied dimensionality reduction techniques to discovery of kinematic primitives in walking, reaching, grasping, tounge motion, etc. [68, 69, 70]. These methods do not consider how redundancy at the neural level is resolved as they rely on analysis of the motion of the actuated system.

The use of some type of motor synergy by the CNS (Central Nervous System) to perform coordinated movements could greatly reduce the computation for the planning and or execution of motion [71]. If it is possible to perform all desired motions with a number of synergies significantly lower than the number of muscles, then a great deal of redundancy in performing a task has been reduced making it less difficult to find a weighting of the synergies. It is important to note that although spatiotemporal synergies require the CNS to select a weighting as well as individual activation times $t_{io}$ they have the potential to reduce redundancy at the limb and muscular level, as with spatial synergies, and at the trajectory level.

Another formulation of motor primitives coming from the robotics domain is Dynamic Movement Primitives [72, 73] which model coordination as coupled multi-dimensional non-linear dynamical equations with point or limit-cycle attractors and adjustable attractor landscape.

$$\tau \dot{v} = K(g - x) - Dv - K(g - x_0)s + Kf(s) \tag{2.45}$$

$$\tau \dot{x} = v \tag{2.46}$$

$$\tau \dot{s} = -\alpha s \tag{2.47}$$

where $x$ and $v$ are the position and velocity of the system, $x_0$ and $g$ refer to the starting and goal states, $K$ and $D$ are analogous to a spring constant and a damping coefficient, $\tau$ is a time scaling factor, $\alpha$ is a predefined time constant, and $f(s)$ is a non-linear forcing function. This formulation has a number of interesting properties including the ability to modulate primitives to change the scale, speed, phase, and initial and final state of the movement. These modulations endow the primitives with the ability to generalize to new tasks or variations within a given task. For example, a set of DMPs, each one representing a joint of the robot arm, can represent the collective movement of all joints in a robotic arm. This motion can be modified by changing the goal parameter $g$ and the time scaling factor $\tau$ of the DMP to produce a similar movement in a different location within the robots workspace at a different speed. DMPs can be learned from demonstration; i.e., a human can demonstrate a motion to the robot by moving the robot's arm. From this motion a set of DMP parameters, all $\theta_i$'s in Equation 2.2.3.1 can be found using simple regression methods such as [74]

$$f(s) = \frac{\sum_i \phi_i(s)\theta_i s}{\sum_i \phi_i(s)} \tag{2.48}$$

$$\phi_i(s) = \exp(-h_i(s - c_i)^2) \tag{2.49}$$

or via locally weighted regression [73]

$$f_{target}(s) = \frac{\tau \dot{v} - K(g - x) + Dv + K(g - x_0)s}{K} \tag{2.50}$$

$$\theta = \arg\min_{\theta} \sum_s (f_{target}(s) - f(s))^2 \tag{2.51}$$

DMP's are well known within the robotics community and have, for example, enabled robotic systems to play table tennis [75] and perform a tennis backhand and forehand swing [73]. However, in the original formulation [72]

there is not a clear way of combining and superimposing primitives in an analogous fashion to 2.44. In fact, most uses of the DMP framework learn either a single primitive or a collection of primitives that may be concatenated to perform longer actions. A recent probabilistic reformulation of DMPs is more amenable to performing superposition in time of primitives via joint primitive distributions [76].

#### 2.2.3.2 Equillibrium Point Hypothesis

The equillibrium point hypothesis (EPH) describes another means by which redundancy in the human motor system may be reduced. Originally developed by Feldman, it is a physiologically motivated model of muscle control proposing that through the combined effects of muscle dynamics and spinal reflex arcs, the CNS is able to control and coordinate movement simply by setting muscle threshold lengths [77]. A threshold length is considered to be the length beyond which a muscle begins exerting an opposing force [78]. The equillibrium point is the state acheived where the forces exerted by the individual muscles and the inertial and external forces are balanced. In addition, the system is required to have a zero state velocity to be considered an equillibrium point. Essentially, this all means that the local feedback is performing a type of inverse dynamics, allowing the CNS to simply command muscle lengths.

Shadmehr provides a concise yet rigorous introduction to the EPH in [78]. The basic equation of motion for a general mechanical system can be written as

$$\ddot{\boldsymbol{\theta}} = \mathbf{I}(\boldsymbol{\theta})^{-1}(\mathbf{f_c}(\dot{\boldsymbol{\theta}}, \boldsymbol{\theta}, \mathbf{u}(t)) - \mathbf{f_m}(\dot{\boldsymbol{\theta}}, \boldsymbol{\theta} - \mathbf{f_g}(\boldsymbol{\theta})) \tag{2.52}$$

where $\mathbf{I}$ is the system's inertia matrix, $\boldsymbol{\theta}$ is the state, $\mathbf{f_c}$ is the force exerted by the muscles on the system, $\mathbf{f_m}$ contains the centripetal and Corriolis forces, and $\mathbf{f_g}$ is the gravitational force [78, 79]. The equillibrium point of this system is defined as the the point where the state $\boldsymbol{\theta}$ and state velocity $\dot{\boldsymbol{\theta}}$ are zero, which only occurs if $\mathbf{f_c} - \mathbf{f_m} - \mathbf{f_g}$ is zero.

The inverse dynamics solution to tracking a desired trajectory $\boldsymbol{\theta}_d(t)$ is to simply solve Equation 2.52 for the muscle force $\mathbf{f_c} = \hat{\mathbf{f}}_\mathbf{m} + \hat{\mathbf{f}}_\mathbf{g} + \hat{\mathbf{I}}\ddot{\boldsymbol{\theta}}_d$, where the hat symbol $\hat{x}$ refers to the system's estimate of $x$. However, due to model and environmental uncertainties as well as un-modeled external forces, this

force may not produce the desired motion exactly. To compensate for this uncertainty and to track the desired state more accurately we can incorporate a feedback component into this inverse dynamics model

$$\mathbf{f_c} = \hat{\mathbf{f}}_\mathbf{m} + \hat{\mathbf{f}}_\mathbf{g} + \hat{\mathbf{I}}\ddot{\boldsymbol{\theta}}_d - \mathbf{B}(\dot{\boldsymbol{\theta}} - \dot{\boldsymbol{\theta}}_d) - \mathbf{K}(\ddot{\boldsymbol{\theta}} - \ddot{\boldsymbol{\theta}}_d) \tag{2.53}$$

where $\mathbf{B}$ and $\mathbf{K}$ are positive definite gain matricies.

Equation 2.53 implies that the CNS has an internal dynamics model that it uses to track desired trajectories. What the equillibrium point hypothesis posits is that the feedback portion of 2.53 is designed in such a way so that the internal inverse dynamics model is not necessary. The dynamics of the muscles and the low-level spinal reflexes are captured by the feedback components in this model. However, there has been much debate about what is the appropriate form of $\mathbf{f_c}$ [77, 80, 81]. One common feature of $\mathbf{f_c}$ assuming a static system is that it is approximately exponential with respect to some linear difference in the desired and observed trajectories. However, Gomi and Kawato demonstrated that a static $\mathbf{f_c}$ is not adequate for describing human reaching movements and argued that this implied the CNS must be doing some form of inverse dynamics [82]. However, Gribble et al. [83] demonstrated that a dynamic model incorporating delayed feedback and velocity dependent muscle force, could potentially address the concerns of Gomi and Kawato, while having similar static behavior as earlier models.

The EPH could be one way that the CNS simplifies motor control tasks by reducing the dimensionality at the joint level, by providing a means by which joint trajectories or synergies can be tracked without the need to perform inverse dynamics. Granted, redundancy still exists at the limb and trajectory levels. However, the EPH formulation does not preclude the use of some higher level redundancy resolution strategy. In fact, Mussa-Ivaldi has proposed a way by which state and time dependent muscle synergies could be integrated into the EPH framework [84].

#### 2.2.3.3 Operational Space Control and the Uncontrolled Manifold Hypothesis

In order to resolve redundancy at the limb and trajectory levels additional constraints are required. Operation space control, also known as task space

control, is a method developed in robotics used to track robotic end effector trajectories or honor end effector constraints while taking into account the robot's dynamics [85]. The key idea in operational space control is to transform movements in the task space into an internal space where stable controllers, such as inverse dynamics controllers, can be used to perform the task [86]. For the task of finding $n$ joint velocities $\dot{\mathbf{q}}$ in a serial robotic manipulator from $m = 6$ end effector velocities $\dot{\mathbf{x}}$ one can utilize the $m \times n$ manipulator Jacobian $\mathbf{J}$ [87] which is defined as

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{2.54}$$

In redundant systems, where the number of task constraints is less than the manipulator degrees of freedom, $m < n$, there is an ambiguity in the transformation $\mathbf{J}$. This ambiguity or unconstrained region of the internal space is referred to as the null space within robotics and engineering and is known as the uncontrolled manifold within the motor control literature [87, 88, 86]. In the velocity operational space problem, this null space is represented by the second term in the below equation

$$\dot{\mathbf{q}} = \mathbf{J}^{+}\dot{\mathbf{x}} + (\mathbf{I} + \mathbf{J}^{+}\mathbf{J})\mathbf{b} \tag{2.55}$$

where $\mathbf{J}^{+} = \mathbf{J}^{T}(\mathbf{J}\mathbf{J}^{T})$ is the right pseudoinverse of $\mathbf{J}$, $\mathbf{I}$ is the $n \times n$ identity matrix, and $\mathbf{b} \in \Re^{n}$ is an arbitrary vector [87]. Movement of the system in the null space does not affect the performance of the task. For example, consider the task of positioning the end effector of a 3 DOF planar revolute manipulator at a position in the plane. Because there are only two constraints imposed by the task, positioning in 2-D, and the manipulator has 3 DOF, there are an infinite number of positions by which the manipulator can achieve the goal. In other words this system is redundant. The uncontrolled manifold, in this case, is the space of joint movements that do not affect the position of the end effector.

There is an increasing consensus within the the field of motor control that humans plan their motions in task space [89]. Some evidence for this comes from studies of response times, which have been shown to be a function of spatial task parameters [90]. On the neurophysiological level, cells within the motor and premotor cortex have been shown to exhibit farily uniform tuning

to spatial parameters such as end effector motions or forces applied to the end effector [91, 92, 93].

Choices for how to resolve this ambiguity include not performing any control in the nullspace by forcing $\mathbf{b} = \mathbf{0}$, minimal intervention, energy minimization, avoidance of joint limits, maximizing comfort, task specific optimization. The idea of minimal intervention arises from the observations of human movements in reaching and force control tasks [94, 95]. These studies have revealed that variation in the uncontrolled manifold is much higher than variation in task relavent portion of the internal space [96, 97]. Latash et al. provides a fairly comprehensive review of the applicaiton of the the uncontrolled manifold concept to analysis of human movements in [89]. In other words, the CNS allows inconsistency in arm movements from trial to trial that do not affect the performance of the task. This implies that the CNS is prioritizing stabilizing with respect to a task and allowing the orthogonal space, the uncontrolled manifold, to vary, potentially absorbing variation from disturbances and errors in the task space.

Another question that arises when developing an operational space controller is, which variables should be transformed from task-space into joint-space? Roboticists have developed a variety of operational space controllers for task space variables including end effector positions, velocities, accelerations, and forces [98, 85].

#### 2.2.3.4 Optimal Control

The framework of optimal control provides a systematic means for coordinating movements and reducing DOFs by allowing one to optimize performance of a task with respect to a cost function. Research has shown that humans behave near optimally in a number of reaching and force balancing tasks if the correct cost is chosen [99, 100, 101]. A cost is typically defined as an integral of an instantaneous cost over a period of time and is typically a function of end effector states, internal state variables, model dynamics, and motor torques or muscle activations [102].The optimal action and movement is then found by minimizing the total cost. This process can be thought of as adding additional constraints to the task, thereby reducing the redundancy in the system in a principled manner.

Many different costs have been proposed for describing human movements

which rely on different model assumptions. Kinematic costs are functions of end effector and joint positions, velocities, accellerations, and their higher order derivatives whereas dynamic costs consider the link masses and inertias, end effector forces, and joint torques required to perform specified movements [103, 102]. A commonly chosen kinematic cost is that of minimum jerk, or minimizaiton of the first derivative of task space acceleration, which is motivated by the the observation that end effector veocities vary smoothly in most reaching movements [100, 101]. Other common choices for costs include minimum torque change, energy, time, and variance [104].

One major difference between kinematic and dynamic costs is in how they separate the movement planning and movement execution phases [103, 102]. Kinematic based costs give solutions that are trajectories in either joint or task space, and rely on some base controller, such as an inverse dynamics controller, to track the desired trajectory. This method completely separates the planning and execution stages and the can be thought of as a hierarchical approach to motor control. Alternatively, solutions to dynamic cost optimal control problems take the form of trajectories within the joint torque or muscle force space and therefore no distinction can be made between the planning and execution of a movement.

In general, there is no agreement on what type of cost the human motor system is optimizing. But, some experiments attempt to draw conclusion about the correct cost by using the fact that these two categories of costs predict differing behaviors under visual and force field perturbations during movement [103]. When the visual feedback is altered artificially, and the perturbation decays to zero at the start of and by the end of the movement, the dynamic cost is unvaried and predicts that the task can be executed with the same set of motor commands as generated without the altered feedback. Whereas in the kinematic cost case, the cost increases and the generated trajectory is altered to bring the desired and observed visual position into consensus. If an artificial force field is imposed on the system the kinematic cost solution predicts an increased cost but maintains the original optimal kinematic trajectory, whereas in the dynamic cost case a new path is found that will minimize the cost in the presence of this force field perturbation.The primary reason for these predicted differences is the separation of planning from execution in kinematic solutions versus the integrated solution in dynamic solutions.

Researchers have taken advantage of these differing predictions and have found evidence for the hierarchical plan and execute strategy implied by the kinematic cost model in both the case of visual-feedback alteration [103, 105] and force field perturbation [106, 107, 108]. This would seem to resolve the question between kinematic and dynamic costs, but studies of more complex movements indicate that knowledge of arm dynamics is vital to planning. One interesting observation is that when executing reaching movements in which an obstacle is present, humans tend to configure their arms to be optimally inertially stable when passing near the object [109].

Todorov has proposed a alternate categorization of optimal control approaches to motor control by instead differentiating between open-loop and closed-loop control laws [104]. Optimization models that rely on open-loop controllers plan the optimal muscle activation, joint torques, or limb trajectories whereas models with closed-loop controllers additionally take into account motor noise, sensory uncertainty, and delayed feedback. The major advantage of the closed-loop optimization approach is that instead of yielding a desired movement, as with open-loop optimization where the movement is executed using some base-level pre-determined feedback controller such as an inverse dynamics controller, the system finds a feedback controller that is optimal in the presence of delayed feedback and noise. This is a somewhat subtle distinction, but the difference is important.

Todorov has shown that optimal feedback control is an incredibly powerful tool and that it has the potential to subsume and unite many of the conflicting theories of optimal motor control [94]. Through the lens of optimal feedback control phenomenon such as Fitts' law, the scaling of movement duration with amplitude and desired accuracy, the uncontrolled manifold and the minimal intervention principle, and smooth end effector trajectories can all be explained [104, 94]. See Todorov [104] for a fairly comprehensive review of optimal control applied to modeling of human motor control.

However appropriate optimal control may be for describing human movements it does not offer an adequate explanation of how these skills may be acquired. Instead, it presupposes that a model of the musculo-skeletal system is available, a family of control laws is specified, and requires a description of the task in the form of a cost function [104]. It is unlikely that this is how the CNS solves the motor skill learning and redundancy resolution problem. There exists, however, an approximate form of optimal control called rein-

forcement learning (RL) or approximate dynamic programming that assumes no knowledge of a system model and relies on repeated interaction with the environment in order to find an optimal movement. Although, RL fails to provide a bulletproof solution for learning optimal movements. One major issue with RL is problematic enough that it has been give its own name, the curse of dimensionality, which states that the amount of time required to find an optimal solution is exponentially related to the number of states and actions [110]. In the case of human motor control the state space consists of the estimated positions, velocities, accelerations, and jerks of each joint and the end effector, the individual muscle activations and joint torques, and complex perceptual signals such as vision and audition. In addition to the massive dimensionality of the state, the fact that this is a continuous system and the length of a desired action is variable and not available a priori further complicates the problem. Another problem with the optimal control or RL formulation is that each new task, or new cost function, requires solution of another optimization problem. This is impractical as it does not allow for transfer of knowledge about one task to performing of a new yet similar task sometimes referred to as multi-task learning. It does not have any means of performing generalization.

### 2.2.4 Hybrid Perceptual-Motor Primitives

One critique of motor primitive and muscle synergy based models is that although primitives can be found from sets of muscle activations or movement recordings and used to explain a large portion of the variance of recordings for a given task, that is not proof that these primitives are components of an underlying control architecture, and not merely an artifact of the structure imposed by the task itself. In other words, many researchers infer existence of a control system that recruits motor primitives to perform a variety of tasks from the observation that the output of the system during performance of a single task has statistical regularities. This is flawed logic. If movement is indeed controlled by combining motor synergy activations, the synergies themselves are being recruited in a coordinated manner, implying that the structure observed in the movement itself is due to the structure of both the synergies and the controller for the given task. Therefore it is not possible

to separate out the contributions of the controller and the underlying synergies from the learned motor primitives. It is also possible that there are no underlying synergies and these experiments are merely picking up on regularities in the controller for a specific task. This is a fundamental problem for neuroscientists and physiologists hoping to draw conclusions about motor control in biological systems. In addition, the criteria of reproducing a large portion of the variance does not necessarily lead to discovery of synergies that if recruited would produce the desired action, because there is no link between actions and observations. Combining perceptual and motor features to produce some form of hybrid synergies could make this link and provide more robust evidence for the existence of synergies.

#### 2.2.4.1 Functional Synergies

One way of dealing with this issue is to create what are called functional synergies, which are synergies that include both EMG signals and task related variables. That is, dimensionality reduction is applied to the combined set of EMG signals and task variables. Chvatal et al. used the functional synergy approach to discover synergies for a human postural balancing task [111]. In the experiment, participants stood on an platform that was then tilted randomly causing the participants to react in order to maintain balance. EMG signals of various postural muscles were measured along with the contact forces between the subject's feet and the platform. Functional synergies were computed for each participant from the collected EMG and force data. These synergies were then used to reconstruct the measurements from a second postural experiment where subjects were taking a step at the time of perturbation. Interestingly, the functional synergies explained the second experiments quite well, although one extra synergy was added in order to obtain better reconstruction of the EMG signal. This line of investigation is promising as it affords a means to test the motor primitive control hypothesis that other approaches do not. Alessandro et al. provides a fairly comprehensive review of different synergy based motor control approaches in neuroscience as well as robotics and advocates for including both input and output variables in primitive formulations [112].

#### 2.2.4.2 Sensory-Motor Primitives

A similar argument is made by Todorov and Ghahramani in [113] and is reviewed here. They pose the question of "How can *good* motor primitives be constructed from first principles?" Whereby *good* they mean primitives that reduce the dimensionality and complexity of the state space, but still ensure that the a solution is achievable. It is possible that by reducing the dimensionality of the system that the desired task becomes unachievable or the system becomes uncontrollable. Their main insight is that the only safe assumption to make is that all tasks will be performed with the same physical system. Therefore, primitives should be constructed from the input and output of the system so as to reflect the structure of that system. They refer to primitives constructed in this way as sensory-motor primitives.

Todorov and Ghahramani approach the formation of sensory-motor primitives from a control and estimation perspective. Let us define a discrete time system with two column vectors $\mathbf{u}_t$ and $\mathbf{y}_t$ as the $l$ and $m$ dimensional column vectors representing the input and output of the system at time $t$ respectively. We also assume a general model of the underlying system dynamics given by $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ where sensory observation are given by $\mathbf{y} = s(\mathbf{x}, \mathbf{u})$. Then given a history of input and output behavior from this system we can form the vectors

$$\mathbf{p}_t = [\mathbf{u}_{t-p}^\top, \mathbf{y}_{t-p}^\top, \cdots, \mathbf{u}_{t-1}^\top, \mathbf{y}_{t-1}^\top]^\top \tag{2.56}$$

$$\mathbf{f}_t = [\mathbf{y}_t^\top, \mathbf{u}_{t+1}^\top, \mathbf{y}_{t+1}^\top, \cdots, \mathbf{u}_{t+f}^\top, \mathbf{y}_{t+f}^\top]^\top \tag{2.57}$$

where $p$ and $f$ are the past and future horizons in discrete time steps. Todorov and Ghahramani propose using the sample data to fit a probabilistic latent variable model of the form

$$P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t) = \int P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{h}) P(\mathbf{h}|\mathbf{p}_t) d\mathbf{h} \tag{2.58}$$

where $\mathbf{h}$ is the $n$ dimensional latent variable that is discovered by the unsupervised primitive learning algorithm and the integral is over all possible values of the vector. Selection of the control input $\mathbf{u}_t$ proceeds as follows. At

each time step the low-level controller computes

$$\mathbf{h}_{\text{past}} = \mathbb{E}[\mathbf{h}|\mathbf{p}_t] = \int \mathbf{h} P(\mathbf{h}|\mathbf{p}_t) d\mathbf{h}$$

which is then sent to the high-level task-specific controller

$$\mathbf{g} = G(\mathbf{h}_{\text{past}})$$

which returns the control increment on the hidden state

$$\mathbf{h}_{\text{desired}} = \mathbf{h}_{\text{past}} + \mathbf{g} \tag{2.59}$$

from which the control action is computed by

$$\begin{aligned} \mathbf{u}_t &= \mathbb{E}[\mathbf{u}|\mathbf{h}_{\text{desired}}] \\ &= \int \mathbf{u} P(\mathbf{u}|\mathbf{h} = \mathbf{h}_{\text{desired}}) d\mathbf{u} \\ &= \int \mathbf{u} \int_{\mathbf{f}} P(\mathbf{f}, \mathbf{u}|\mathbf{h} = \mathbf{h}_{\text{desired}}) d\mathbf{f} d\mathbf{u} \end{aligned} \tag{2.60}$$

where the step from line 2 to line 3 is accomplished by marginalizing over $\mathbf{f}$. Note that the probabilities used to arrive at values for $\mathbf{h}_{\text{past}}$ and $\mathbf{u}_t$ are from the probabilistic model in Equation 2.58 composed of a high-level task-specific controller and a low-level task common controller.

Todorov and Ghahramani claim that the model will have some desirable properties. Namely, the hidden state $\mathbf{h}$ will capture the information about past inputs and observations that is most statistically useful in predicting the future. This enables learning of the high-level controller $G$ using RL on a lower-dimensional space. Also, the transformation 2.60 can be thought of as a set of control synergies because it relates the internal desired state $\mathbf{h}_{\text{desired}}$ to the control input $\mathbf{u}_t$. They say that these synergies form a compact representation of the statistics of the sensory-motor dynamics and capture the so called *modes* of the system. They also point out that the data generated prior to fitting the model can generated by applying either a random control or a controller and that the model will incorporate the behavior of the controller into itself.

To fit this probabilistic model, they suggest using what they call, a factor analysis generalized to include both inputs $\mathbf{i}_t = \mathbf{p}_t$ and outputs $\mathbf{o}_t = [\mathbf{u}_t, \mathbf{f}_t]$.

The model is given by

$$\mathbf{h}_t = \mathbf{B}\mathbf{i}_t + \mathbf{w} \tag{2.61}$$

$$\mathbf{o}_t = \mathbf{C}\mathbf{h}_t + \mathbf{v} \tag{2.62}$$

where $\mathbf{w}$ and $\mathbf{v}$ are zero mean Gaussian noise random vectors. They then go on to present an expectation maximization solution to fitting the modified factor analysis model that relies on the use of a Kalman filter in the E-step.

It is not immediately apparent how this model is related to factor analysis. Equation 2.62 resembles the standard factor analysis form as shown in Equation 2.23 with $\mathbf{C}$ as the factor loading matrix, $\mathbf{h}_t$ as the common factors, and $\mathbf{v}$ as the unique factors. In Section 3.2 I will discuss the relationship of this model to a technique called dynamic factor analysis in more detail.

As a demonstration of this approach, Todorov and Ghahramani apply this method to data obtained from a simulation of a biologically inspired 2 joint arm. The primitives that they obtain seem to discover the concept of a joint as demonstrated by the inverse activations of primitives for sets of opposing agonist-antagonist muscles. Intuitively, the model discovers structure in the dynamics of the system. This structure inherently reduces the dimensionality of the space that the system can explore. Therefore, a reduced dimensionality representation of this system could conceivably still allow control over the entire space of accessible states. This is an exciting result that leads me to believe that this method has merit. In addition, Todorov and Ghahramani provides some evidence that these primitives can be utilized in speeding up RL for control of a system. The lower dimensional representation helps overcome the curse of dimensionality that is encountered during RL.

#### 2.2.4.3 Dynamic Movement Primitives and Perceptual Coupling

The earlier discussed method of Dynamic Movement Primitives can be tweaked to allow modification of a movement based on feedback. The idea is that individual executions of a DMP should generate similar sensory traces. If one assumes that a given motion or range of similar motions has a characteristic set of sensory traces, $F_{des}$, associated with it, then one can devise a way to use this information as either feedback or a predictor of task failure. Pastor refers to the combination of a DMP and a set of characteristic sensory traces

as and associated skill memory or ASM [114].

The characteristic sensory traces, $F_{des}$ are found by recording the sensory observations over the course of execution of a DMP and performing some sort of averaging. $F_{des}$ can then used to modifiy the original DMP dynamics shown in Equation 2.2.3.1 via a perceptual coupling term allowing the system to compensate for deviations from the characteristic sensory traces.

$$\tau \dot{v} = K(g - x) - Dv - K(g - x_0)s + Kf(s) + \zeta \tag{2.63}$$

$$\zeta = K_{sensor}(F - F_{des}) \tag{2.64}$$

One problem with this approach is that the mapping from sensory errors to desired changes in the motion $K_{sensor}$ must be specified. However, it is possible to learn this mapping via RL [115]. This approach enabled Kober et al. to teach a robot to perform the ball-in-a cup task with variable initial conditions of the ball position. In general, this the approach of including sensory features in the representation of motor primitives is promising.

## 2.3 Learning Modular Representations for Speech Production

In [116] Gick and Stavness make a call to action, urging researchers to tackle the problem of *modularizing* speech. They believe that the concepts of muscle synergies can be extended beyond the area of limb control to the domain of speech and articulatory control. They describe a lack of fundamental research in this domain and cite the relative complexity of vocal tract articulation as a primary reason. However, there is some work that is worth reviewing. The theory of articulatory phonolgy falls in line well with this modularity hypothesis and introduces the concept of articulatory gestures which can be thought of as sensory-motor synergies. Also, some recent work discussed how articulatory primitives were learned from electromagnetic articulographic recordings providing evidence for the existence of articulatory gestures. A number of neural network based models for control of simulated vocal tracts have been developed that assume modular neural architectures. These systems are also trained via interaction with the simulated vocal tract

and are reviewed as well.

#### 2.3.0.1 Articulatory Gestures

The concept of articulatory gestures comes from the articulatory phonology framework of Browman and Goldstein [117]. In this framework gestures are posited to be the underlying atomic units of speech production and can be combined and concatenated to produce speech segments and syllables. A gesture essentially specifies the location of a constriction within the vocal tract. These gestures are thought to be dynamical systems that can be combined and concatenated to produced complex coordinated motions. The task dynamic model of interarticulator speech coordination rests on the framework of articulatory phonology and models gestures as point attractor dynamical systems [118]. In is worth noting that this is very similar to the dynamic movement primitive approach used in robotics [72, 73]. The articulatory synthesizer TADA implements the task dynamic model with a physical vocal tract model [118]. In the TADA environment, a gesture connects tract variables to model articulator variables. In other words a gesture is composed of a desired constriction and the articulator activations used to create that constriction. So essentially, it is the motion plan for how to produce a desired constriction.

Accompanying each gesture is an activation level that specifies the strength of the gesture. A second layer of coordination called a gestural score is used to describe how gestures can be combined to produce segments and words. The articulatory phonology framework and task dynamics model aligns with the hypothesis that synergies are the fundamental units of motor control and exist as a means for simplifying control of complex dynamical systems. However, these approaches give little thought to the origin of these gestures. For example, in the TADA simulator all gestures are hand specified and require an intimate understanding of phonetics to construct.

#### 2.3.0.2 Articulatory Primitives

The work of Ramanarayanan et al. begins to address these concerns in which they use a variant of NMF called convolutive NMF with sparseness constraints (cNMFsc) to discover articulatory movement primtives from electro-

magnetic articulographic (EMA) recordings [119]. They point out one of the common issues with learning of synergies and movement primitives is that it is difficult to quantitatively evaluate the result. First they seek to answer the question "If we are presented with a set of waveforms or movement trajectories that have been generated by a compositional structure, then can we design and validate algorithms that can recover this compositional structure?" This is a fundamental question that underlies all of the synergy and primitive research.

They offer a very unique approach to answering this question by applying their algorithm to movements of the TADA vocal tract simulator and comparing the learned primitive activations to the ground truth gesture scores of the TADA model. As one might imagine, the activations of the primitives can not be directly compared to the gesture scores. Instead, they first fit a LPC model to both the activation trajectories and the ground truth gesture scores. They then compare the information content, of the learned primitives activations to the gestures scores of the task dynamics model using a canonical correlation analysis on the LPC weights of each. This reveals a strong similarity between some of the learned primitives and the TADA gestures. Although not conclusive, this approach is principled and adds rigor to their analysis.

They also perform the same analysis with the primitives learned from the EMA data with similar results. In addition, they generate phonetic labels to accompany the EMA data and investigate the activation of various primitives with respect to the phoneme being produced. This analysis reveals some selectivity of bases based on vowels and consonants, but the results are fairly ambiguous [119]. It is also important to point out that this EMA data was obtained from individuals that have already learned to speak. What this means is that the primitives learned from this data are going to be reflective of the language of the speakers. If we assume that these primitives are passed down via genetics then this approach is likely fine, but if we instead assume that primitives are learned by each individual throughout their childhood then this does not address that problem. Overall, this study outlined a more rigorous approach to evaluation of primitives learned from biological data and added to the evidence for the existence of articulatory synergies.

#### 2.3.0.3 The DIVA Model

DIVA is a neurobiologically inspired vocal tract articulatory control system [120, 121]. It was developed with the motivating question of "How do infants learn the motor skills necessary to produce speech?" DIVA is composed of of blocks artificial neural networks, referred to as maps, connected to each other in complex architecture that enables learning to produce speech via an artiulatory vocal tract simulator. These maps correspond to neural structures theorized to exist in the brain from fMRI based studies.

The architecture of the DIVA model is fixed, but the synaptic weights of each map and between maps is learned via interaction with the simulator during two stages. During the babbling stage the sensory-to-motor inverse models of desired auditory and proprioceptive signals to articulatory commands are learned. In the imitation stage, the model is presented with example phonemes, syllables, and words from the target language and learns a mapping from perception to auditory targets (formants). It then learns feedback and feedforward control of the model for tracking of these auditory targets.

The name DIVA is an acronym for directions into velocities of articulators, which describes the primary map of direction of vocal tract proprioceptive signals to velocities of the model articulators. The vocal tract simulator that is used is an extension of Maeda's model modified to include 10 articulators instead of the original eight [122]. DIVA is able to learn to a control a vocal tract to produce synthetic speech and has been used to study may different speech production phenomena. Overall, the results of this system are encouraging, but from the perspective of artificial intelligence it is not terribly satisfying due to the amount of prior knowledge that is required to construct the system. Ideally one would like to be able to outline an architecture and learning process that can be applied to a variety of motor control tasks.

#### 2.3.0.4 A Connectionist Model

Plaut and Kello present connectionist model for learning of speech production [123]. Their model has similarities with the DIVA model, but differs in a number of dimensions. First of all, the model was not designed to corre-

late as closely with brain structures although the network architecture was motivated by research in speech learning. This model is also based on something called a simplified recurrent neural network, endowing the system with a *memory* of sorts. The vocal tract model that is used is not a physics based simulation involving an acoustic tube and articulators. Instead it is a set of equations that relate articulatory variables such as constriction locations and voicing to acoustic variables such as formants and frication. This simplification was made to reduce the computational load needed to train the underlying networks. Similar to the DIVA model, this model relies on interaction with the *simulator* to learn the network mappings and involves both babbling and imitation stages. The model is trained with 3.5 million different babbling and imitation presentations. The resulting model is able to reproduce perceived utterances very accurately with a 3.5% error rate. Overall this approach appears to be somewhat more general than the DIVA model in that the architecture is more flexible. However, this more general architecture requires a large amount of training. If a true articulatory synthesizer had been utilized instead the computational demands would increase drastically. However, neural network models are capable of being trained much more efficiently than they were in 1999 using GPU hardware and therefore may make extending of this approach more feasible.

#### 2.3.0.5 Reinforcement-Gated Self-Organizing Maps

Warlaumont et al. is motivated by by better understanding how infants learn to phonate, or produce speech sounds [124]. Instead of viewing the problem as one of imitation of human speech sounds they see RL as having an integral role in learning to phonate. Then the focus becomes on determining what behaviors should be reinforced. The vocal tract model used is the articulatory syntheis module from the Praat software suite [125, 3], see Section 3.1.1.1 for a description of the model. They perform a number of learning trials with reinforcement being a function of formant frequencies and similarity to various vowel sounds.

Their model consists of a 25 neuron neural network where each neuron is connected to the same 20 muscles of the model [124]. This network is trained via what the authors call a reinforcement-gated self-organizing map. Essentially, the the model is the same as a self-organizing map, but the synaptic

weights are only updated when the sound produced via the network articulating the model is deemed to be *good* by the reinforcing function; i.e., the learning is gated by the reinforcement. This method was developed because there is no statistical regularity in the randomized articulator commands. The model is able to learn to produce phonated sounds very reliable using this method. Note that for the learning, the lungs articulator was biased to produce air flow through the tract greatly increasing the likelihood of phonation. In addition, the model learns to produce different vowel sounds depending on whether the reinforcement function rewards Korean or English vowels.

This approach of using reinforcement to gate learning of a self-organizing map is interesting. However, if the self-organizing map was allowed access to sensory observations as well as articulatory commands may have enabled a similar result.

# CHAPTER 3

# VOCAL TRACT SENSORY-MOTOR SYNERGIES

The main objective of this work is to answer the question "How can we design a system that can autonomously learn to articulate a simulated vocal tract in order to produce speech?" This is a very difficult question; therefore, we restrict ourselves to answering a smaller, more tractable question, namely "How can we learn low-level vocal tract controllers that simplify the problem of learning higher level controllers for speech production?" To investigate this, I first use a vocal tract model to generate a collection of articulation commands, vocal tract area functions, and simulated speech sounds. I then apply a sensory-motor synergies discovery algorithm, similar to that introduced by Todorov and Ghahramani [113], to the VT simulation data and human speech signals. The structure of the resulting resulting synergies is then discussed. I also show that it is possible to use these sensory-motor synergies to drive articulation of the vocal-tract simulator.

## 3.1 Vocal Tract Model

In order to answer the questions laid out in this thesis, a database of vocal tract area functions, articulatory muscle activations, and acoustic features is required. This is difficult information to obtain from human subjects because it requires measuring of the articulator positions and of the muscle activations. Measurement of the articulator positions, area function, can be taken using a technique called electromagnetic articulography in which sensor coils placed inside and along the vocal tract pick up on changes in the electromagnetic field generated by induction coils that are mounted to a persons head. This is an invasive technique and has very limited spatial resolution, but very good temporal resolution around 500 Hz citeWrench2000. An alternative technique is to use real-time magnetic resonance imaging which has

improved spatial resolution, but reduced temporal resolution [126].

In addition to area function or articulator position measurements, we require articulator muscle activation measurements in order to develop and evaluate sensory-motor synergies. Muscle activity can be recorded using electromyography EMG. For example, Schultz and Wand [127] measure six EMG signals from electrodes placed on the face and along the outside of the throat. They were then able to develop a speech recognition system based only on EMG activity that achieved a 10% error rate on a 100 vocabulary system. This bolsters the argument for including articulatory activations in our analysis. However, we are interested in obtaining the activations of *individual* muscles for the ultimate purpose of control the vocal tract. Since there are a large number of muscles that articulate the vocal tract, many of which are very small and difficult to measure externally, it is not feasible to use this method.

In addition to the difficulty in obtaining vocal tract articulatory data from human subjects, we are not interested in observing the coordinated vocal tract movements of people who have already learned to speak. Instead, we are interested in developing a method by which a system can learn to produce speech with little prior knowledge. Therefore, we require a simulation of the human vocal tract.

### 3.1.1 Model Characteristics and Comparison

There are a number of vocal tract simulators from which to choose, but most are not suitable for this research. A vocal tract simulator was chosen based on the following criteria:

- Model
  - Biologically faithful
  - Dynamic
  - Low-level control of articulators
- Software
  - Open source
  - Well supported

These requirements all stem from the dual goal of better understanding speech learning and developing a system that can learn to produce speech with little prior knowledge. The model should be biologically faithful and therefore more complex than most early vocal tract models in order to capture phenomenon beyond the simple production of vowels. And since we are interested in better understanding human speech learning the model should be similar to the human vocal apparatus. The human vocal tract is a dynamical system. Many models, however, avoid the incorporation of dynamics into their system and instead relate articulator activations directly to vocal tract configurations kinematically.

The model also needs to allow low-level control of the articulators such as muscle activation as opposed to some high-level specification of articulation such as the place of constriction and openness. This is necessary because we are interested in learning about how coordination of high DOF systems can be learned. The simulator should be open source to enable random articulation of the vocal tract, recording of simulation trials, and feedback control of the vocal tract which will require modification of the source code. The software should also be supported by an active community of developers and users to allow for reproduction of the results by other researchers and aid in troubleshooting. Overall, the simulator should be suitable for research in articulation and not have been designed just for speech synthesis.

After an initial search for simulators using the above criteria, I narrowed down the potential models to TADA and Praat. TADA is an appealing option due to the use of gestures in controlling the vocal tract. The problem is that it relies on hand specified hard coded gestures and we are interested in developing a system that can learn gestures with little direction. Therefore, TADA is unsuitable for this research.

#### 3.1.1.1 Praat

Praat is a software suite that is used by linguists for analyzing, synthesizing, and manipulating speech [3]. Most importantly for this work, Praat contains an articulatory synthesis module. The Praat model is, at its base, an acoustic tube model [128, 125, 128, 129]. Every part of the vocal tract, including the lungs and glottis, is modeled as a tube with walls that can be thought of as

adjustable mass-spring systems following the template

$$m\frac{d^2y}{dt^2} = tension\ force + collision\ force + coupling\ force + damping\ force + air\ pressure\ force \quad (3.1)$$

where $m$ represents the wall mass and $y$ represents the displacement of the wall from the midline of the tract.

The *tension force* is the force exerted by the spring to reach its equilibrium length and is modeled as a third order non-linear spring

$$tension\ force = k^{(1)}(y_{eq} - y) + k^{(3)}(y_{eq} - y)^3 \quad (3.2)$$

where $k^{(1)}$ and $k^{(3)}$ are the linear and cubic spring constants respectively and $y_{eq}$ is the equillibrium length of the spring. The stiffness, damping, and equillibrium lengths of these springs are adjusted in relation to the the activation of 29 different muscles which take on values in the range of 0-1. Although the assumption is made to model the individual muscles as constant stiffness with variable equillibrium length, some of the tube sections are modeled as variable stiffness mass-spring-damper systems. This is because many of the walls of the vocal tract are actually composed of muscles with fibers running tangential to the tract which behave as variable stiffness non-linear springs along the direction perpendicular to the stretch of the muscle.

The *collision force* approximates the force of two walls coming together. This is also modeled as a non-linear third order spring. However, to ensure that the physical laws of the aerodynamic simulation are obeyed a minimum tube width is introduced to ensure the cross-sectional area is never zero along the tract even at a wall collision. The *coupling force* represents the force that adjacent tubes exert on each other. Adjacent tubes are connected by third order non-linear springs. This force is very important in the modeling of the glottis. The *damping force* captures the effect of internal friction in the tissue of the tube walls. It is proportional to the negative of the wall velocity

$$damping\ force = -(B_{open} + B_{closed})\frac{dy}{dt} \quad (3.3)$$

where $B_{open}$ is the damping of the spring and $B_{closed}$ is the damping of the compressed walls when the walls are in contact. These damping coefficients

are dynamic and depend on the spring constants, wall mass, and tissue properties. The *air pressure force* is the force exerted by the air pressure in the tract that forces the walls apart or together depending on the internal pressure relative to atmospheric pressure.

$$air\ pressure\ force = P\delta x\delta z \tag{3.4}$$

where $P$ is the average pressure in a tube section, $\delta x$ is the length of a wall section, and $\delta z$ is the tube depth, making $\delta x\delta z$ the area of the wall.

The sublaryngeal system is modeled as a sequence of 17 or optionally 29 tubes. Different tube sections are divided into a number of parallel subdivisions in order to better model the viscous resistance of the air particle flow along the tract walls. These tubes are all articulated by the single *lungs* articulator that approximates the combination of the diaphragm and abdominal muscles. This grouping of muscles is necessary due to the simplicity of the lung model.

The larynx is modeled as 2 or optionally 11 tubes with adjacent tube sections coupled to each other. The 2 tube model represents the glottis and is very similar to Ishizaka and Flanagan's two mass model [130]. The 11 tube model extends this approach to modeling of the conus elasticus ligament. The two glottis tubes are also subdivided similar to the lungs in order to better represent the internal geometry of the glottis. This intercartilagenous glottis modeling enables simulation of some aspects of breathiness and whispering.

The nasal cavity is modeled as 14 tube sections which are also subdivided to better model viscous friction. Only the first segment representing the velum is articulated.

The pharyngeal and oral cavities are modeled as 27 tube sections and the dimensions of which are based on the model developed by Mermelstein [131]. However, Boersma reworks the model so that the original articulators such as jaw angle and tongue tip height are determined by the activations of 11 different muscles instead [125].

The aerodynamic equations are derived in [125]. He starts by deriving the equations for the continuity of mass flow for tube sections with varying length. He then derives the equations of motion from Newton's law incorporating modeling of the Bernoulli effect and viscous resistance. Next he applies a small pressure approximation to the equations of state relating

pressure to air density. Turbulence within the tract is also taken into account as both a resistive loss and as additive noise in the tube pressure proportional to the velocity squared. Boundary conditions are then laid out for each of the four possible tube terminations: closed boundary, open to atmosphere, two tube intersection, and tube branching. The resulting equations are then discretized to produce difference equations.

The Praat articulatory synthesis model meets all of the requirements laid out at the beginning of this section. It is one of the most biologically faithful simulators available excluding finite element approaches which are much more computationally demanding [132]. It takes into account the dynamics of the vocal tract walls and relies on muscles to articulate the tract as opposed to synergistic combinations of muscles like gestures. It is also open source and is actively supported by a community of researchers including the original developer. For these reasons I have chosen Praat as the simulator for this research.

### 3.1.2 Software Modifications

As discussed earlier, the articulatory synthesizer is one of many components available within Praat. Praat was designed to be primarily interacted with via a user interface, but it does support control of the system via a scripting language. Unfortunately, for the articulatory synthesizer this scripting language only allows open loop control of the vocal tract. In order to run a trial with the simulator one has to create what is called an *artword* which consists of the articulatory activation targets over the length of the trial. This scripting interface also only allows recording of the synthesized speech signals and does not enable saving of the articulator activations and area functions at each timestep. Therefore it is necessary to modify the software to perform the desired experiments.

All of the Praat software is written in C and is platform independent. The source code is available on github under the repository name praat doing phonetics by computer.[3] After looking through the source code, it was clear that the best approach was to extract the code for the simulator from Praat and to turn the articulatory synthesizer into a stand alone piece of software that could be modified to meet my needs. The software itself is not well

commented and utilizes a number of macros to enable object oriented style functionality while still using C instead of C++. In addition, even though arrays are indexed starting at zero in C, in this software, arrays are instead indexed starting at 1 and the first element is not used. These programming oddities make reading and understanding the code very difficult.

In order to save time in working with this software I decided that it needed to be ported over to C++ and restructured to enable programmatic control as opposed to GUI based control. This was a tedious process, but was necessary to enable quick modifications to the control of the vocal tract and access to logging of all relevant signals. I chose to use the X-Code development environment within OSX to make these changes.

## 3.2 Sensory-Motor Synergy Model

I chose to approach the problem of learning low-level controllers for a simulated vocal tract by learning sensory-motor synergies. I use a model similar to the one outlined by Todorov and Ghahramani [113] as reviewed in Section 2.2.4.2. I will first discuss the motivation for choosing this model and then show how it is related to dynamic factor analysis.

### 3.2.1 Motivation

As should be clear from the review of motor synergies, there is not *conclusive* evidence to support their existence in biological systems. The majority of research on motor synergies shows that recordings of EMG activity or joint positions during performance of a task can be reproduced by a small number of time varying synergies. Although the synergies may be capable of reproducing the original signals to within some percentage of variance, this approach does not ensure that a control system recruiting those synergies would be capable of performing the task adequately.

Moreover, the synergy hypothesis posits that the synergies may be recruited for a range of tasks and not specific to a single task with some minor variations. This task generalization property of the synergies could be very useful in transfer learning, enabling use of prior learned skill knowledge to speed up learning of a new skill. The problem is that the majority of motor

synergy studies focus on discovering synergies from recorded performances of a single constrained task. It is possible that these discovered synergies only correspond to a small range of movements necessary to perform one specific task. In other words, the structure in the task itself could be the source of the structure that the synergies are discovering, and not a reflection of the underlying control system structure. As discussed in Section 2.2.4.1, functional synergies, synergies that include task related variables and motor observations, provide a possible way around the task generalization problem. But, the functional synergy approach does not resolve the issue of ensuring adequate task performance using the synergies. Studies involving in vivo stimulation of premotor neural circuits in the spinal cord, although, do provide evidence for a neural basis for synergies. Despite all of the above concerns, collectively, the evidence for the synergy hypothesis suggests that it is plausible and merits further investigation.

Philosophically the sensory-motor synergy approach could be considered an extension of Barlow's original optimal sensory encoding hypothesis to include motor commands in addition to sensory signals [31]. In essence, what the optimal encoding principle assumes is that there is structure within the natural world and that humans have developed the ability to process information generated by interacting with this world in an optimal manner. Barlow hypothesized, and others have shown, that dimensionality reduction is a useful framework in understanding this process. Optimal encoding results in the structure of the world being reflected in the human nervous system.

I think that this insight can be extended to better understand the problem of motor control. Analogously, structure or redundancy is present in our physical body and in the daily tasks that we perform. Our motor system has developed a means of optimally performing these tasks via learning. And similar to perceptual systems, the structure in the world, our physical bodies, and the tasks we perform would then be reflected in our motor control system. Given that assumption, a natural way of representing these abilities is through the use of motor synergies derived using dimensionality reduction.

But, as we know from control theory and the study of human motor control, feedback is essential for most tasks. Therefore it is logical that a connection from action to perception should be included in our model. Todorov and Ghahramani's sensory-motor primitive approach makes this connection through the use of a latent variable model, where the hidden state is a linear

combination of sensory observations and motor commands over a discrete number of samples in the past [113]. In this model, future sensory observations and motor commands are predicted enabling control of the high dimensional system via control of the low-dimensional latent state. This model produces sensory-motor synergies and is found using unsupervised learning by applying expectation maximization as in the paper, or by applying a dimensionality reduction method to samples of both sensory observations and muscle activations as I propose.

This sensory-motor synergy approach enables application of RL techniques to systems with large state spaces because it reduces the dimensionality of the state space relieving issues caused by the curse of dimensionality and speeds up the time to finding an optimal policy. Because the method by which state space is reduced preserves the modes of the system, the main dynamics of the system are captured.

The sensory-motor synergy approach is particularly well suited to the problem of learning speech production. One reason for this is that speech is very structured indicating that dimensionality reduction methods may be appropriate to apply. This is reflected within the articulatory phonology framework's definition of gestures, which can be naturally represented within the sensory-motor synergy framework. In the context of speech sensory-motor synergies, the lower dimensional latent space, or factor space, provides an interesting means of defining gesture scores. Gestures may be represented as a subset of the factor space or as a distribution over the space. To enable the system to produce common speech sounds, a second layer of sensory-motor synergies could be added. Todorov and Ghahramani's model lends itself to building of such a hierarchy, where the first level hidden state $\mathbf{x}$ and low-level control signal $\mathbf{g}$ become part of the observable state at the next level. Autonomous construction of this hierarchy poses an interesting challenge that will be investigated as this research progresses, but it is conceivable that this approach may lead to the system learning a phonemic representation of speech comprised of combinations of gestures.

In summation, applying sensory-motor synergies to the problem of learning to produce speech is well motivated. The evidence that biological systems rely on a similar structures combined with the Barlowian principle of optimally encoding via dimensionality reduction, plus the need for feedback in motor control, and the natural decomposition of speech into synergistic

units produced by gestures all indicate that sensory-motor synergy approach is worth exploring.

### 3.2.2 Dynamic Factor Analysis

Todorov and Ghahramani describe their model, given in Equations 2.61 and 2.62, as a generalized form of factor analysis [113]. As discussed earlier, while this model bears resemblance to factor analysis, it is unclear how it could be considered a generalized form of factor analysis. The authors provide little discussion of the model derivation and make no reference to other works on which the model is based. In order to place the sensory-motor synergy approach in the context of the existing literature, I began looking for similar techniques. The sensory-motor primitive model reminded me of an autoregressive style process where the future observations are predicted by a number of past observations. In fact, this is very similar to a vector autoregressive (VAR) model

$$\mathbf{y}_t = \mathbf{B}\mathbf{Y}_t^{-p} + \mathbf{w}_t \tag{3.5}$$

where $\mathbf{y}_t$ is the $m$ dimensional observation at time $t$, $\mathbf{B}$ is a matrix, $\mathbf{Y}_t^{-p}$ is a vector of $p$ past observations, and $\mathbf{w}_t$ is a error term. This led me to discover a technique called dynamic factor analysis (DFA) which has the same base form as factor analysis but allows the common factors to change over time according to a first order autoregressive process influenced by noise [133].

$$\mathbf{x}_t = \mathbf{A}\mathbf{x_{t-1}} + \mathbf{B}\mathbf{v}_{t-1} \tag{3.6}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{x}_t + \mathbf{D}\mathbf{w}_t \tag{3.7}$$

where $\mathbf{x}_t$ and $\mathbf{y}_t$ represent the $k$ dimensional common factors and the $m$ dimensional observations at time $t$ respectively, and $\mathbf{v}$ and $\mathbf{w}$ are zero-mean Gaussian noise vectors with covariances $\mathbf{Q}$ and $\mathbf{R}$ respectively. In addition, it is assumed $\mathbf{w}$ and $\mathbf{v}$ are mutually uncorrelated. On the face of it, the DFA model does not appear to be the same as the model presented in [113]. However, following Deistler and Hannan [134], the two noise processes in Equations 3.6 and 3.7 can be represented as a higher dimensional orthogonal

white noise process $\boldsymbol{\epsilon}_t$ yielding

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{\mathbf{t-1}} + \mathbf{B}^*\boldsymbol{\epsilon}_{t-1} \tag{3.8}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{x}_t + \mathbf{D}^*\boldsymbol{\epsilon}_t \tag{3.9}$$

Then, following the derivation in [135], we can define vectors past and future observations and future errors

$$\mathbf{y}_t^{p-} = [\mathbf{y}_{t-1}^\top, \mathbf{y}_{t-2}^\top, \dots]^\top \tag{3.10}$$

$$\mathbf{y}_t^{f} = [\mathbf{y}_t^\top, \mathbf{y}_{t+1}^\top, \dots]^\top \tag{3.11}$$

$$\mathcal{E}_t^{f} = [\boldsymbol{\epsilon}_t^\top, \boldsymbol{\epsilon}_{t+1}^\top, \dots]^\top \tag{3.12}$$

We then perform some algebraic manipulation, taking advantage of the the common error vector $\boldsymbol{\epsilon}$ between the equations and the recursive definition of $\mathbf{x}_t$. Note that it is assumed that $\mathbf{D}^*$ is invertible. This results in the following formulation

$$\mathbf{x}_t = \mathbf{K}\mathbf{y}_t^{p-} \tag{3.13}$$

$$\mathbf{Y}_t^{f} = \mathbf{O}\mathbf{x}_t + \mathbf{E}\mathcal{E}_t^{f} \tag{3.14}$$

where

$$\mathbf{O} = [\mathbf{C}^\top, \mathbf{A}^\top\mathbf{C}^\top, \mathbf{A}^{2\top}\mathbf{C}^\top, \dots]^\top \tag{3.15}$$

$$\mathbf{K} = [\mathbf{B}^*\mathbf{D}^{*-1}, (\mathbf{A} - \mathbf{B}^*\mathbf{D}^{*-1}\mathbf{C})\mathbf{B}^*\mathbf{D}^{*-1}, (\mathbf{A} - \mathbf{B}\mathbf{D}^{*-1}\mathbf{C})^2\mathbf{B}^*\mathbf{D}^{*-1}, \dots] \tag{3.16}$$

$$\mathbf{E} = \begin{bmatrix} D^* & 0 & \cdots & 0 \\ CB^* & D^* & \ddots & 0 \\ CAB^* & \ddots & \ddots & 0 \\ \vdots & \cdots & CB^* & D^* \end{bmatrix} \tag{3.17}$$

Interestingly, we observe that $\mathbf{O}$ is actually the observability matrix for the modified state space Equations 3.8 and 3.9. When implementing this method, we do not actually have infinite future and past observation vectors $\mathbf{y}_t^f$ and $\mathbf{y}_t^{p-}$, so they are truncated to $f$ samples and $p$ samples of $\mathbf{y}$ respectively. This truncation implies that we are assuming the term $(\mathbf{A} - \mathbf{B}^*\mathbf{D}^{*-1}\mathbf{C})^p\mathbf{x}_{t-p} = 0$.

In this new formulation of DFA, we can see that Equations 3.13 and 3.14

very closely resemble the Equations 2.61 and 2.62 proposed by Todorov and Ghahramani [113] which are repeated here for convenience.

$$\mathbf{h}_t = \mathbf{B}\mathbf{i}_t + \mathbf{w}$$
$$\mathbf{o}_t = \mathbf{C}\mathbf{h}_t + \mathbf{v}$$

Notice the similarity between the definition past and future history vectors $\mathbf{p}_t$ and $\mathbf{f}_t$ in Equations 2.56 and 2.57, also repeated here, with the concatenated past and future observations $\mathbf{y}_t^{p-}$ and $\mathbf{y}_t^f$ in Equations 3.10 and 3.11.

$$\mathbf{p}_t = [\mathbf{u}_{t-p}^\top, \mathbf{y}_{t-p}^\top, \cdots, \mathbf{u}_{t-1}^\top, \mathbf{y}_{t-1}^\top]^\top$$
$$\mathbf{f}_t = [\mathbf{y}_t^\top, \mathbf{u}_{t+1}^\top, \mathbf{y}_{t+1}^\top, \cdots, \mathbf{u}_{t+f}^\top, \mathbf{y}_{t+f}^\top]^\top$$

One obvious difference though, is the absence of a noise term in Equation 3.13 in coparision to Equation 2.61. So, although these models are very similar they do not appear to be completely equivalent. I'm not sure how to interpret the implications of this model difference, but intuitively, the presence of a noise term in the projection from past observations to the hidden state implies that some error occurs. This error could be a result of the dimensionality reduction as in Todorov and Ghahramani's formulation, from the truncation of the past history vector as in the DFA model, or simply because the model does not adequately represent the process.

One solution to the DFA problem that presents itself in the concatenated form is to find an estimate for the matrix $\mathcal{F} = \mathbf{OK}$ as in $\mathbf{y}_t^f = \mathcal{F}\mathbf{y}_t^{p-} + \mathbf{E}$ using least squares regression [135]. To perform this regression multiple, $n$, samples of the observation histories are combined into matricies

$$\mathbf{Y}^f = [\mathbf{y}_{t_1}^f, \mathbf{y}_{t_2}^f, \cdots, \mathbf{y}_{t_n}^f] \tag{3.18}$$
$$\mathbf{Y}^{p-} = [\mathbf{y}_{t_1}^{p-}, \mathbf{y}_{t_2}^{p-}, \cdots, \mathbf{y}_{t_n}^{p-}] \tag{3.19}$$

Once the estimate $\tilde{\mathcal{F}}$ is obtained we need to obtain estimates for $\mathbf{O}$ and $\mathbf{K}$. This can be accomplished by performing singular value decomposition on the scaled estimate

$$\Gamma^{p-}\tilde{\mathcal{F}}\Gamma^f = \mathbf{USV}^\top \tag{3.20}$$

where $\Gamma^{p-} = \mathrm{Cov}(\mathbf{Y}^{p-})$ and $\Gamma^f = \mathrm{Cov}(\mathbf{Y}^f)$. This scaling is performed to determine the weights of the individual features. However, if desired $\Gamma^f$

and $\Gamma^{p-}$ can be made to be identity matrices while maintaining theoretical convergence guarantees of the DFA model. Alternatively, the features can be scaled by dividing by the square root of their individual variances. This method will cause the scaled features to all have unit variance and may be useful when features have different units or very different variances. Scaling is an important step that can change the results of the subspace model quite drastically. This is because the least squares regression will give more weight to larger signals and errors, meaning that higher variance features will be better reconstructed with respect to their variance and lower variance signals may not be well characterized.

Note that as with standard factor analysis, it is assumed that the observations are zero mean. If they are not, the observations can be mean centered by computing the mean and subtracting it from the observations before the analysis. Estimates of the concatenated DFA model matricies can then be found as

$$\tilde{\mathbf{O}} = \Gamma^{f^{-1/2}} \mathbf{U}_k \mathbf{S}_k^{1/2} \tag{3.21}$$

$$\tilde{\mathbf{K}} = \mathbf{S}_k^{1/2} \mathbf{V}_k^{\top} \Gamma^{p^{-^{-1/2}}} \tag{3.22}$$

where the subscript $k$ is the number of latent variables and indicates the $k^{th}$ order reduced dimensionality SVD factorization [135]. We refer to this as the subspace DFA solution or SDFA. Kapetanios and Marcellino show that this approach will discover the true factors asymptotically with the number of samples [135]. They also prove several other properties of this estimator.

It is important to point out that increasing the number of latent variables from $k$ to $j$ only appends $j-k$ rows to $\tilde{\mathbf{K}}$ and $j-k$ columns to $\tilde{\mathbf{O}}$ and does not change the first $k$ rows and columns of the respective matrices. This is because the SDFA method uses the right-singular-vectors and left-singular-vectors corresponding to the $k$ largest singular values of $\hat{\mathcal{F}}$ to create $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ respectively. To find synergies using the SDFA model we can simply take the $i^{\text{th}}$ row of $\tilde{\mathbf{K}}$ to obtain the input synergy $\tilde{\mathbf{k}}_i$ and the corresponding column of $\tilde{\mathbf{O}}$ to obtain the output synergy $\tilde{\mathbf{o}}_i$. The $i^{\text{th}}$ input and output synergies can also be represented in matrix form by reshaping the vector synergies enabling visualization of the mappings over the time histories. That transformation

is given as

$$\mathcal{K}_i = reshape(\tilde{\mathbf{k}}_i, m, p) \tag{3.23}$$

$$\mathcal{O}_i = reshape(\tilde{\mathbf{o}}_i, m, f) \tag{3.24}$$

where $reshape(\mathbf{a}, m, n)$ is a function that forms an $m \times n$ matrix from the $m\dot{n}$ length vector $\mathbf{a}$. Note that when the term synergy is used in the context of SDFA, it can refer to either the vector or matrix form, but the matrix form will primarily be utilized because it facilitates interpretation.

An alternative approach to solving the original DFA problem, Equations 3.6 and 3.7, is to apply maximum likelihood estimation (MLE). If desired, the concatenated form could then be obtained through equaitons 3.15-3.17. The two most popular methods of MLE for a DFA model are scoring and expectation-maximization (EM) [136, 137]. Scoring is essentially Newtons method applied to MLE. And EM attempts to maximize the log likelihood by instead maximizing the expected log likelihood of the model given the observations and the previous guess at the parameter values. This method is guaranteed to provide a better or equal estimate to the parameters at each iteration, but is susceptible to becoming trapped in local maximums. Both EM and scoring require an estimate of the internal state $\mathbf{x}_t$ that we do not have. To get around this, a Kalman filter can be used. However, the Kalman filter requires knowledge of the model. So, to perform MLE one must first make a guess at the parameter values, run the Kalman filter, and then update the parameter values using the state sequence given by the Kalman filter. This process is repeated until the likelihood is determined to have converged.

The DFA model is particularly popular within economics where techniques have been developed to handle very large datasets [138, 139]. In addition, one can slightly modify the DFA model, Equations 3.6 and 3.7, to allow for correlation in errors over time and/or to include what economists call explanatory variables which affect the observation but not the state [138].

It is worth taking another look at the probabilistic model that Todorov and Ghahramani [113] propose shown in Equation 2.58 and repeated here for convenience

$$P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t) = \int P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{h})P(\mathbf{h}|\mathbf{p}_t)d\mathbf{h}$$

In order to arrive at this expression we apply the definition of conditional probability

$$P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t) = \frac{P(\mathbf{f}_t, \mathbf{u}_t, \mathbf{p}_t)}{P(\mathbf{p}_t)}$$

then marginalizing over $\mathbf{h}$

$$P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t) = \int \frac{P(\mathbf{f}_t, \mathbf{u}_t, \mathbf{p}_t, \mathbf{h})}{P(\mathbf{p}_t)} d\mathbf{h}$$

and applying the conditional probability definition two more times

$$\begin{aligned} P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t) &= \int \frac{P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t, \mathbf{h})P(\mathbf{p}_t, \mathbf{h})}{P(\mathbf{p}_t)} d\mathbf{h} \\ &= \int P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t, \mathbf{h})P(\mathbf{h}|\mathbf{p}_t) d\mathbf{h} \end{aligned}$$

we arrive at a very similar expression. But, in order to obtain the exact same expression, we have to make the following assumption

$$P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{p}_t, \mathbf{h}) = P(\mathbf{f}_t, \mathbf{u}_t|\mathbf{h}) \tag{3.25}$$

meaning that $(\mathbf{f}_t, \mathbf{u}_t)$ are conditionally independent from $\mathbf{p}_t$ given $\mathbf{h}$.

Now, it is important to point out that the SDFA method still fits within the probabilistic framework outlined by Todorov and Ghahramani even though it lacks the error term in the input equation. In particular, since we assume that the noise in both models is zero mean, the expectations used to compute the hidden state and the control signal simplify to

$$\mathbf{h}_{\text{past}} = \mathbb{E}[\mathbf{h}|\mathbf{p}_t] = \mathbf{B}\mathbf{p}_t \qquad \mathbf{x}_t = \mathbb{E}[\mathbf{x}|\mathbf{Y}_t^{p-}] = \mathbf{K}\mathbf{Y}_t^{p-} \tag{3.26}$$

$$\mathbf{u}_t = \mathbb{E}[\mathbf{u}|\mathbf{h}_{\text{desired}}] = \mathbf{C}_{\mathbf{u}_t}\mathbf{h}_t \qquad \mathbf{u}_t = \mathbb{E}[\mathbf{u}|\mathbf{x}_{\text{desired}}] = \mathbf{O}_{\mathbf{u}_t}\mathbf{x}_{\text{desired}} \tag{3.27}$$

where the subscript in $\mathbf{C}_{\mathbf{u}_t}$ and $\mathbf{O}_{\mathbf{u}_t}$ denotes taking the rows of $\mathbf{C}$ and $\mathbf{O}$ corresponding to the prediction of the future control $\mathbf{u}_t$, and $\mathbf{x}_{\text{desired}}$ is the output of the low-level controller similar to Equation 2.59.

## 3.3 Experiments

From the previous section, it is clear that the SDFA method and Todorov and Ghahramani's method are very similar. Due to the ease of implementation of the SVD solution for the SDFA method, I have chosen to use it over Todorov and Ghahramani's method. We will now explore the use of the SDFA method in developing synergies for a few different data types. First, I will outline the method used to evaluate the quality of the various learned synergies. I will then discuss the application of SDFA to spectrogram features of human speech and then to data obtained from random stimulation of the vocal-tract model, including tract area function, articulatory activations, and sound spectrogram features. After reviewing these results, I will discuss how the vocal tract can be controlled using the derived sensory-motor synergies.

### 3.3.1 Evaluation Method

In general, it is difficult to evaluate the results of a synergy learning algorithm. One common approach is to look at the learned synergy weights and to make observations on what each synergy most corresponds to, such as movement around a particular joint or movement of a limb away from or towards the body. This type of analysis can provide insight into what the model has learned, but it is very open-ended and results will vary with the individual researcher interpreting the weights.

A more quantitative approach in studies looking at muscle synergies, is to look at the squared multiple correlation coefficient, $R^2$. It is often used to judge the goodness of fit of a set of synergies in representing a signal [66, 55]. For $n$-dimensional time varying signals, $n$ multiple correlation coefficients are computed. In order to arrive at a goodness of fit measure for the entire signal, these coefficients must be combined in some way. Commonly, the mean and variance of the coefficients is computed and is considered to be representative of the ability of the primitives or synergies to reconstruct the signal. But, it is not always clear how to weight the different quantities as they may have different units or may have differing importance in reconstruction of a signal. This is true in the case of speech and vocal-tract synergies.

Although I look at both the synergy weights and $R^2$ values, the primary approach I chose to evaluate the learned synergies is based on class separabil-

ity within the factor space $\mathbf{x}_t$. This metric is motivated by the goal of finding a lower dimensional space that affords creation of symbols, phonemes and broad phonetic categories, and enables control of the vocal tract for producing instances of these symbols. To meet these goals the factor space should have the following essential properties [1]:

- Localization - different phones appear in different areas of the factor space
- Continuity - slight variation of a phone should result in slight variation of the location in the factor space.

Together localization and continuity of the factor space imply that phonemes can be represented as classes of trajectories through the factor space. They also imply that similar phonemes should be clustered together in the factor space producing higher level classes of similar phones.

The broad idea behind the analysis method is to evaluate the synergy learning algorithm by using the learned synergies to analyze different phone classes in the factor space with respect to continuity and localization. If the factor space has both of these properties then the algorithm has accomplished the desired goal of dimensionality reduction while preserving meaningful structure within the signals. It would then be a good candidate for simplified control of the vocal tract and would indicate that this space could be useful for recognition as well.

As I am ultimately interested in speech production, I chose various speech sounds as the individual classes selecting four vowels and four fricatives so that separability between these two different phoneme classes could be observed as well. For the experiment involving human speech I produced five samples of each of the eight sounds each lasting 1 second. For the experiments involving the vocal tract model I specified the articulations manually to produce each of the eight sounds with the vocal tract producing one sample of each sound for 0.5-0.8 seconds each.

To specify the correct articulation to produce each sound, I first attempted to select the vocal tract model articulators that corresponded to the description of the desired articulation. Due to the large number of articulators,

[1] In these definitions, I use the term phone to refer to the instantaneous combination of the articulatory activations, vocal tract area function, and the acoustic signal being produced.

29, this was difficult, and therefore involved some trial and error to achieve the correct vocal tract shape and produce the correct sound. Vowels and fricatives that can be produced with relatively static vocal tract shape were chosen to enable localization of each of the sounds within the factor space. The main exceptions are the movement of the vocal folds produced during voicing and some small transient movements of the lower vocal tract due to increased pressure produced during creation of constrictions for frication.

The name, IPA number, IPA symbol, and a snapshot of the vocal tract area function using the Praat model for each of the eight sounds is shown in Figures 3.1-3.8.

## Fricatives

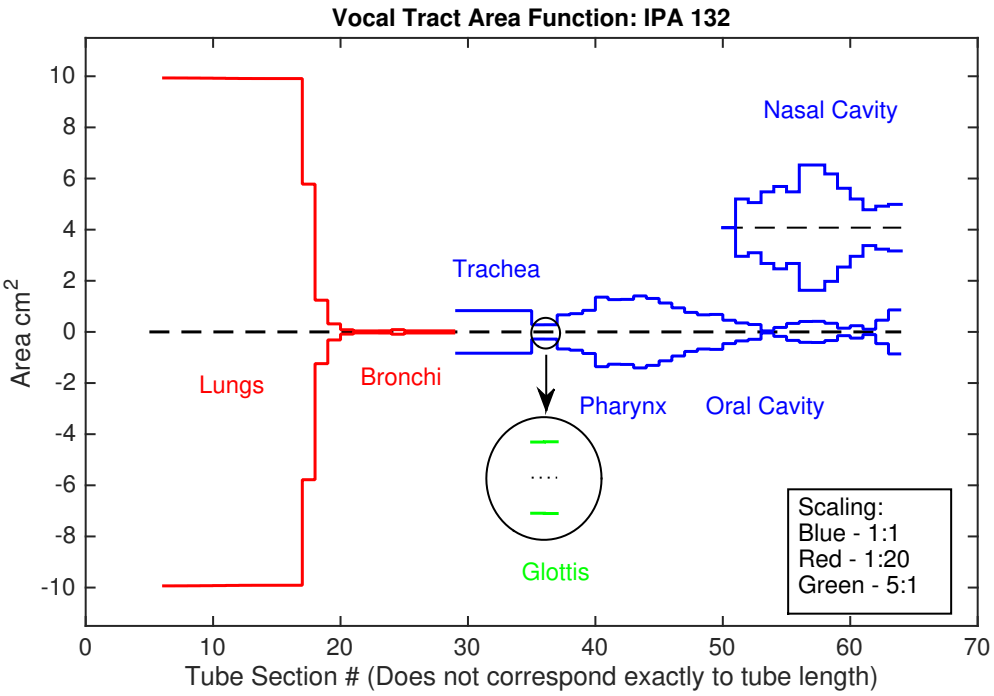


Figure 3.1: Voiceless alveolar sibilant **s** - as in pa**ss**

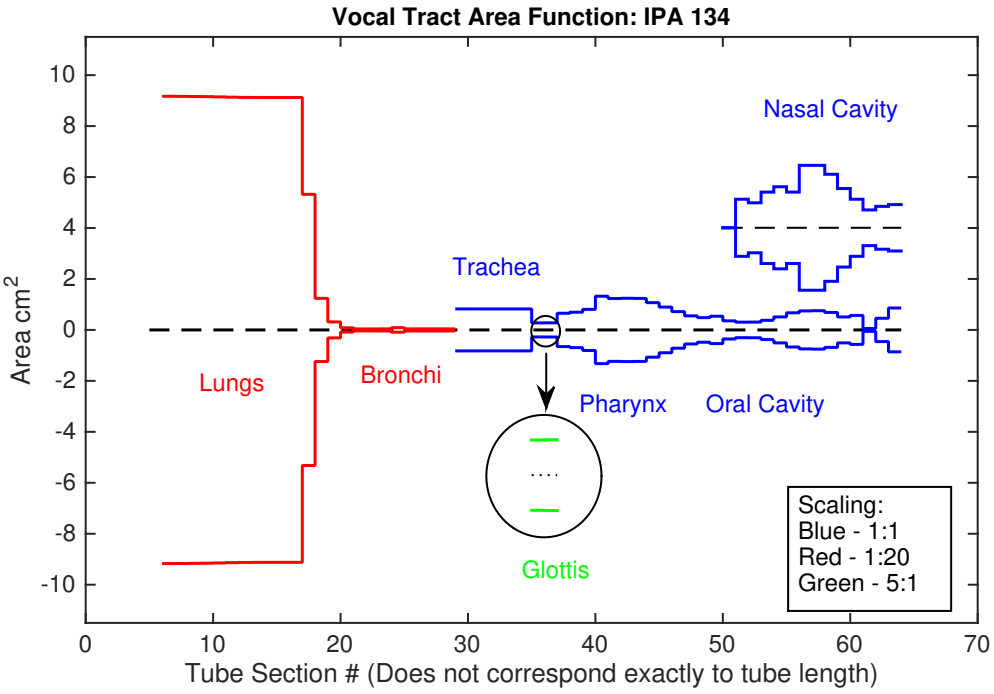


Figure 3.2: Voiceless palato-alveolar sibilant ʃ - as in **sh**ip

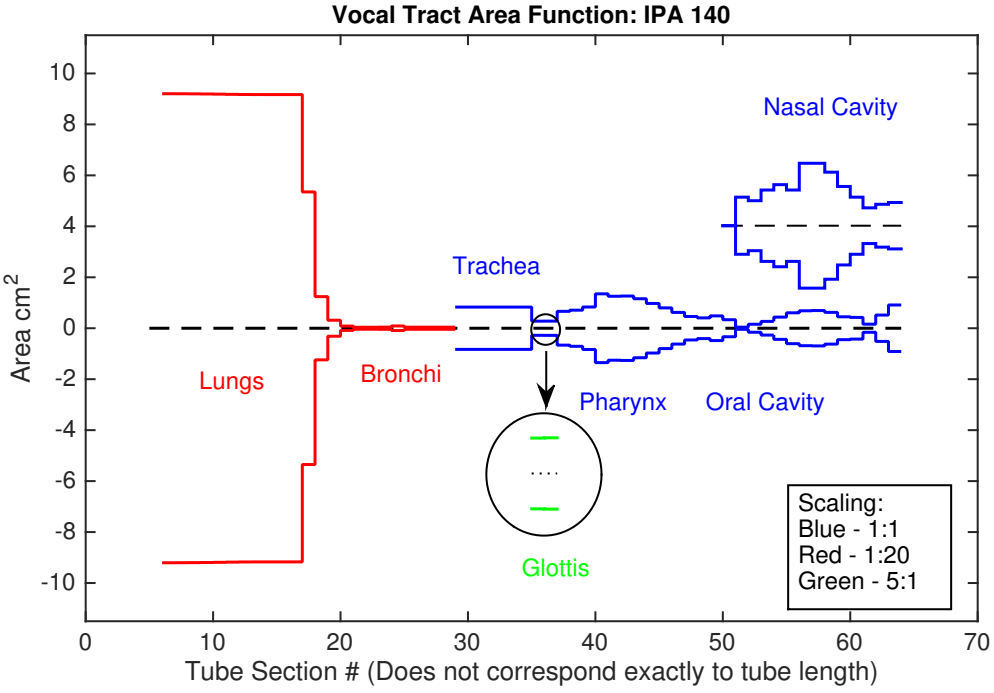


Figure 3.3: Voiceless velar fricative **x** - as in ye**ch**

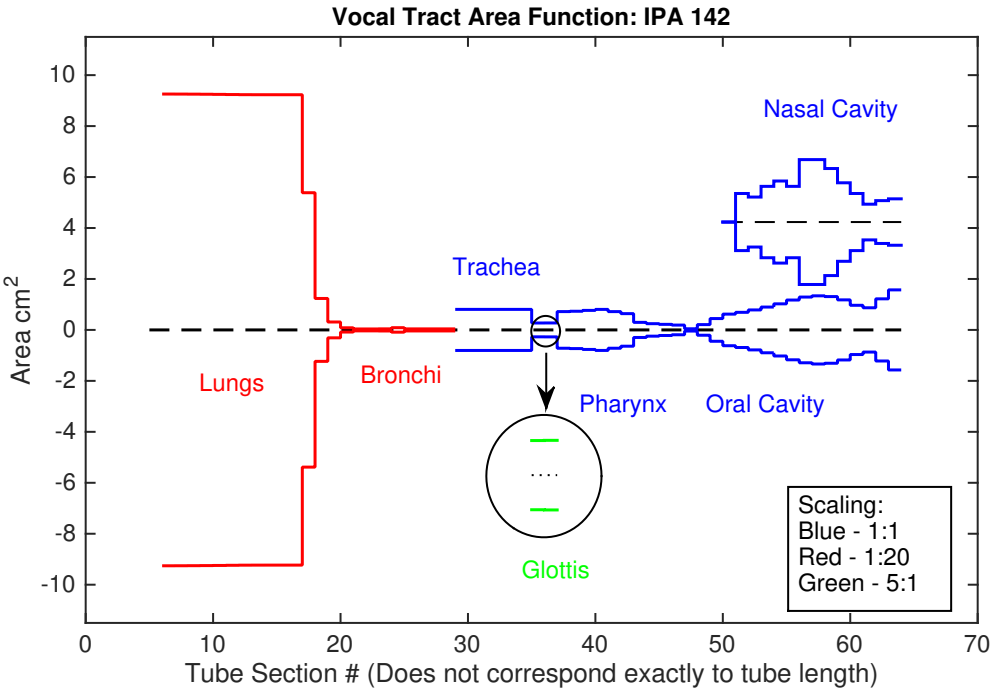


Figure 3.4: Voiceless uvular fricative **χ** - not found in english

## Vowels

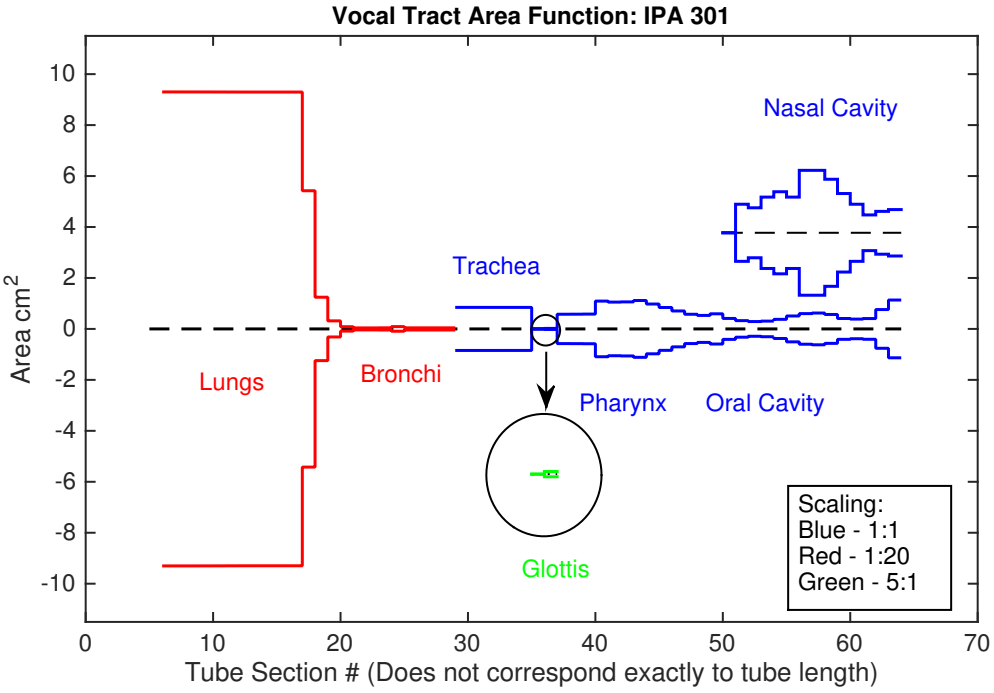


Figure 3.5: Close front unrounded vowel **i** - as in fr**ee**

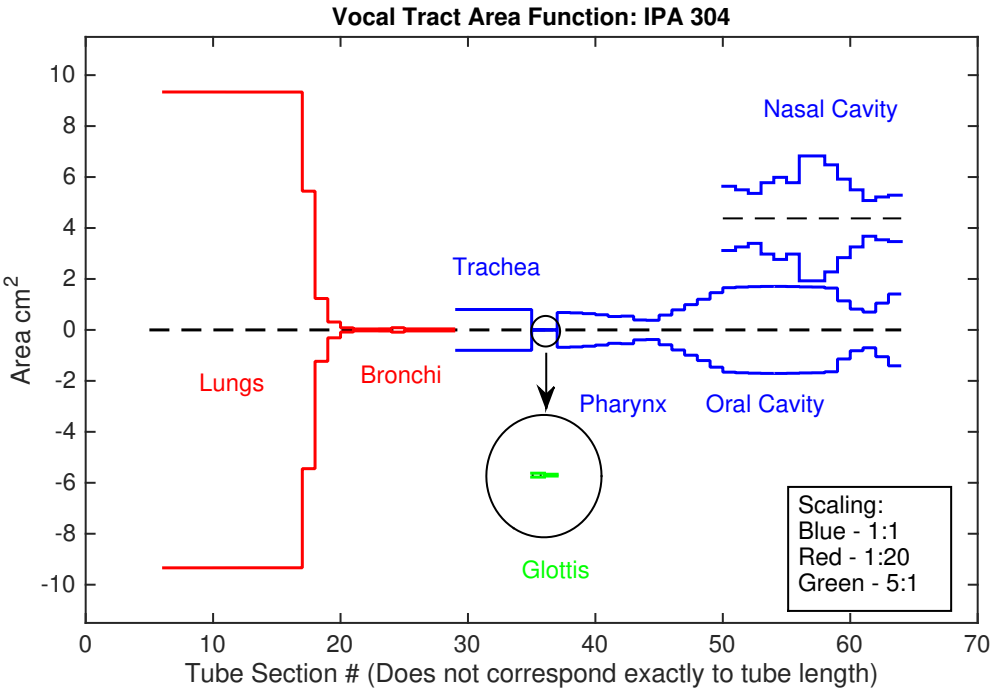


Figure 3.6: Open front unrounded vowel **a** - as in c**a**t

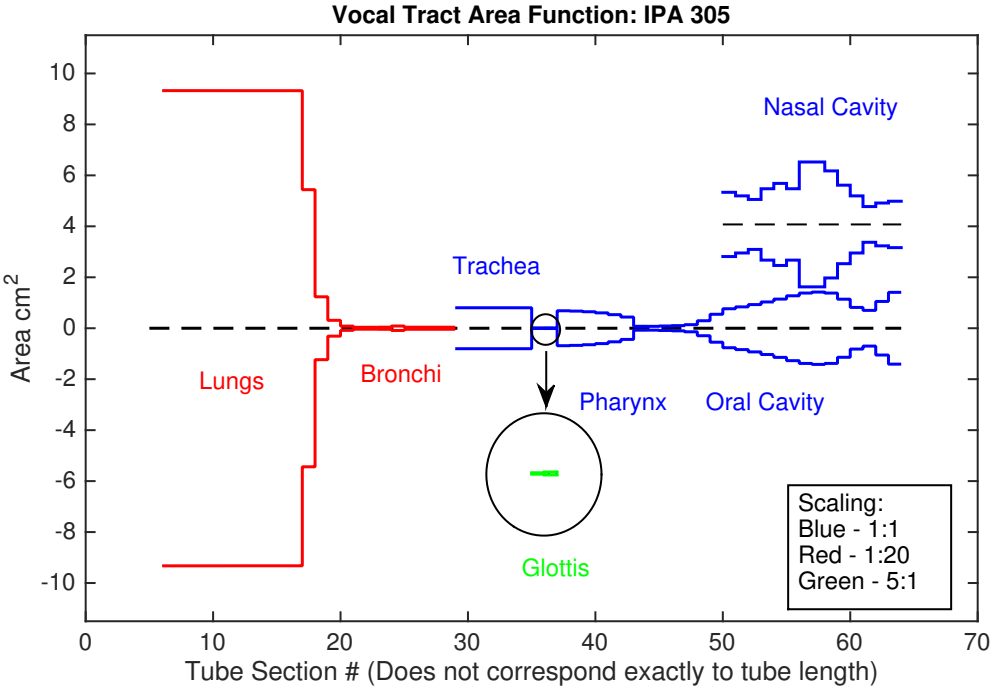


Figure 3.7: Open back unrounded vowel **ɑ** - as in h**o**t

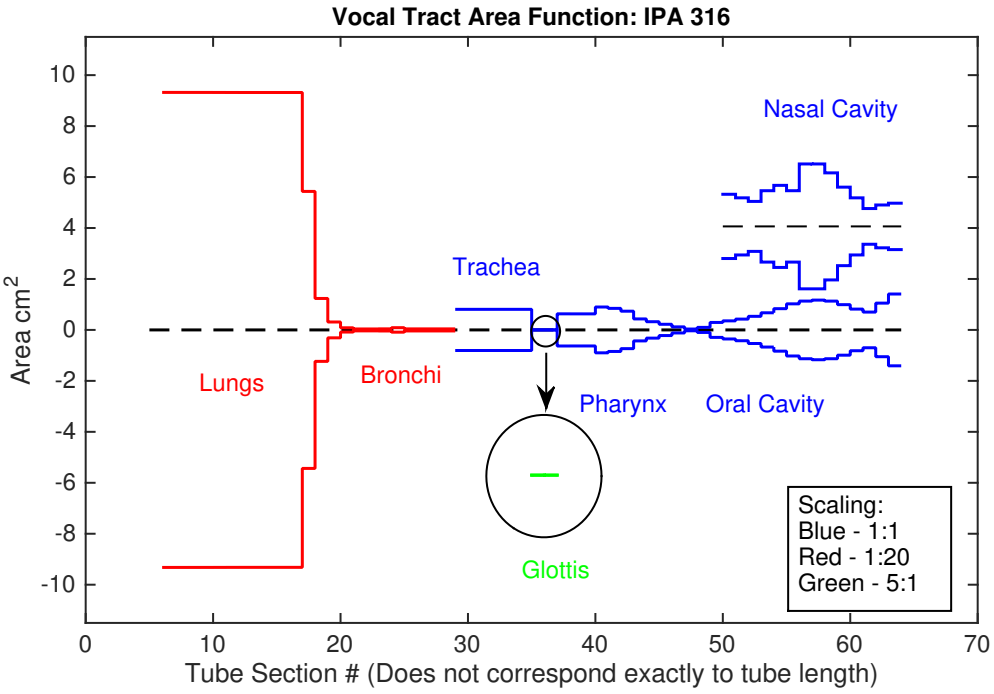


Figure 3.8: Close back unrounded vowel **ɯ** - as in g**oo**se with California accent

### 3.3.2 Human Speech Synergies

This first experiment was somewhat motivated by the work of Poritz in discovery of broad phonetic categories [140]. Recall that broad phonetic categories are categories of phones derived from features of the speech signal including voiced vs unvoiced speech, frication, plosion, etc. One possible set of classes is vowels, consonants, dipthongs, and semivowels. Poritz applied an autoregressive hidden markov model (AR-HMM) to LPC features obtained from readings of short paragraphs. He discovered that the different states of the HMM corresponded to broad phonetic categories, and that the transition matrix discovered some basic phonotactic rules.

I was interested in determining if the SDFA method could similarly discover broad phonetic categories. I first recorded 30 seconds of my own speech at 9000 Hz. I read at a moderate pace from a New Yorker Magazine article, getting through 77 words. I chose to use the log magnitude squared spectrum of the speech signal as observation features. I obtained this representation by first generating a spectrogram of the speech signal shown in Figure 3.9. The spectrogram was generated using a custom MATLAB function using a

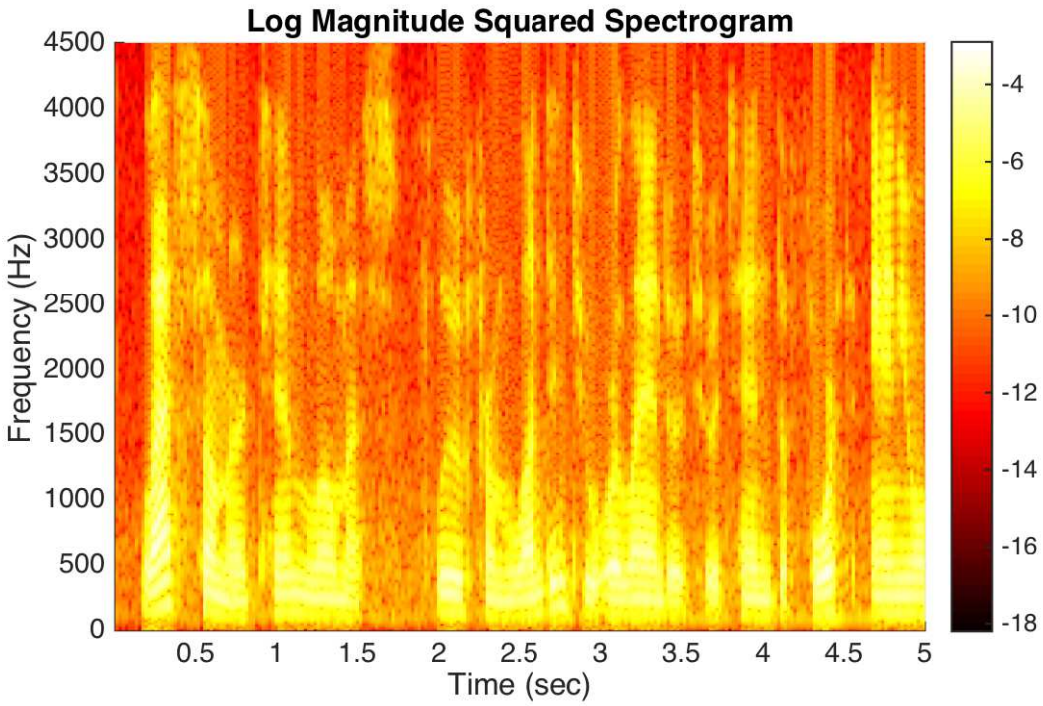


Figure 3.9: Five second sample spectrogram of speech recorded at 9,000 Hz using a Hamming window of length 20 *ms* and no overlap between successive windows.

Hamming window. Let $x(t)$ represent the speech signal. Then the STFT $\hat{\mathbf{X}}$ is obtained using Equations 2.40 and 2.41. Then the spectrogram is defined as

$$\mathbf{Z} = \log_{10}(|\hat{\mathbf{X}}|^2) \tag{3.28}$$

and we denote the column of $\mathbf{Z}$ with window centered at time $t$ as $\mathbf{z}_t$.

In order to form the actual observation variable used in the SDFA analysis, we actually subsample the vector $\mathbf{z}_t$. This is done to reduce the number of frequency bins and to reduce the computation time for finding the synergies. To stay consistent with our early notation in the SDFA section, Section 3.2, we will define the subsampled log magnitude squared frequency vector as our observation vector $\mathbf{y}_t$. We then must collect pairs of observation histories to form $\mathbf{Y}^{-p}$ and $\mathbf{Y}^{f}$. This can either be done by randomly sampling $n$ pairs of histories or by taking each possible history pair.

There are somewhat large number of parameters to this model, and it is difficult to say how each affects the generation of synergies and specifically the localization and continuity properties of the factor space that are of

primary interest. However, during initial experimentation it was determined that the choice of observation history lengths $f$ and $p$ has a great effect on the formation of the synergies. Therefore I have chosen to display the results from three configurations where only the values of $f$ and $p$ are varied in order to illustrate the effect these parameters have. For each of the configurations I let $f = p$ for simplicity. The parameters used when recording the speech along with the parameters used to generate the spectrogram are listed below and are common across all 3 configurations.

Common Configuration

- Signal
  - Length: $t = 30$ s
  - Sample rate: $f_s = 9,000$ Hz
- Spectrogram
  - Window length: 20 ms or $N = 180$ samples
  - Overlap: $l = 0$
  - Number of frequencies: $f_n = 119$

SDFA Configuration 1

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 2$ samp. OR $p_t = f_t = 0.04$ s
- Number of observations: $n = 1,496$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.10 and 3.11 show the weights of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. The input synergies appear very noisy and it is difficult to see any pattern across the frequency bins or time. The output synergies are a bit more interesting, with nearby frequency bins having very similar weights at each time. In other words, the weights

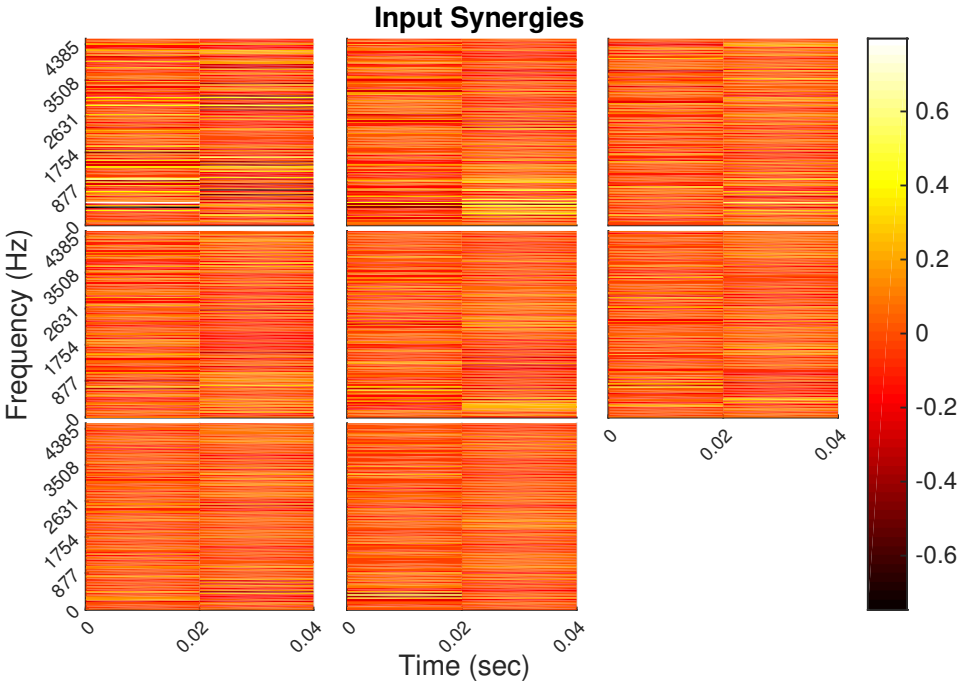


Figure 3.10: SDFA Configuration 1: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

of each synergy are fairly smooth with respect to frequency. In addition, individual output synergies appear to be discovering some low level features of the spectrogram. For example, synergy 1 is capturing a broadband amplitude decrease while synergy 2 emphasizes the lower frequency components. Subsequent synergy weights vary along the frequency axis more quickly and are more difficult to interpret, but may be capturing some sort of harmonics. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.7091 \pm 0.0812$.

Figures 3.12 and 3.13 both show scatter plots of the latent variable $\mathbf{x}_t$ for the eight different IPA phonemes shown in Figures 3.1 - 3.8. For each phoneme, five examples are plotted. Referring back to the earlier definition of localization, it is apparent In Figure 3.12 that this projection in the factor space exhibits the property as different IPA phonemes appear in different areas of the space. Looking more closely, see Figure 3.13, it can be observed that two distinct classes of sounds, or broad phonetic categories, emerge namely vowels and fricatives, indicated by $*$ and $\triangledown$ marks respectively. At the bottom of the figure there is an intermingling of the two sound classes

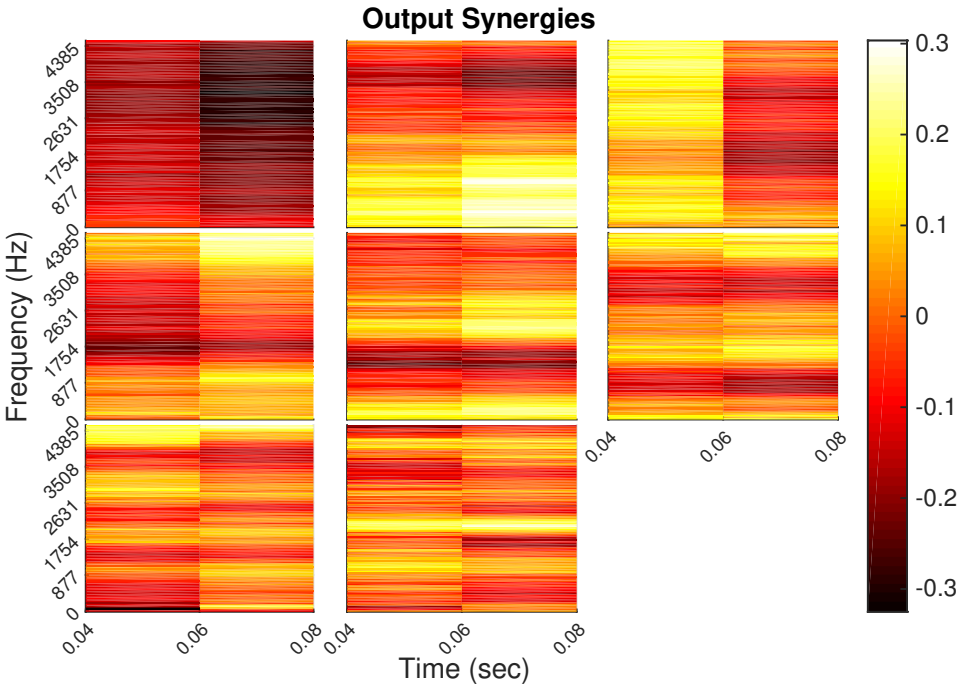


Figure 3.11: SDFA Configuration 1: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

that represent datapoints corresponding to moments of silence in the sample sounds. So, the factor space exhibits the desired continuity property as well.

It is difficult to show the full shape of this eight dimensional feature space on paper. However, by plotting the latent variables as points in a three dimensional space with each of the axes corresponding to a synergy we begin to see that the localization and continuity properties hold for more than just synergies 2 and 3. In addition, for some of the classes that overlapped in Figure 3.12, such as IPA #304 and #316, their is increased separation between in Figures 3.14 and 3.15. This means that the space could be useful for phoneme recognition purposes in addition to control.

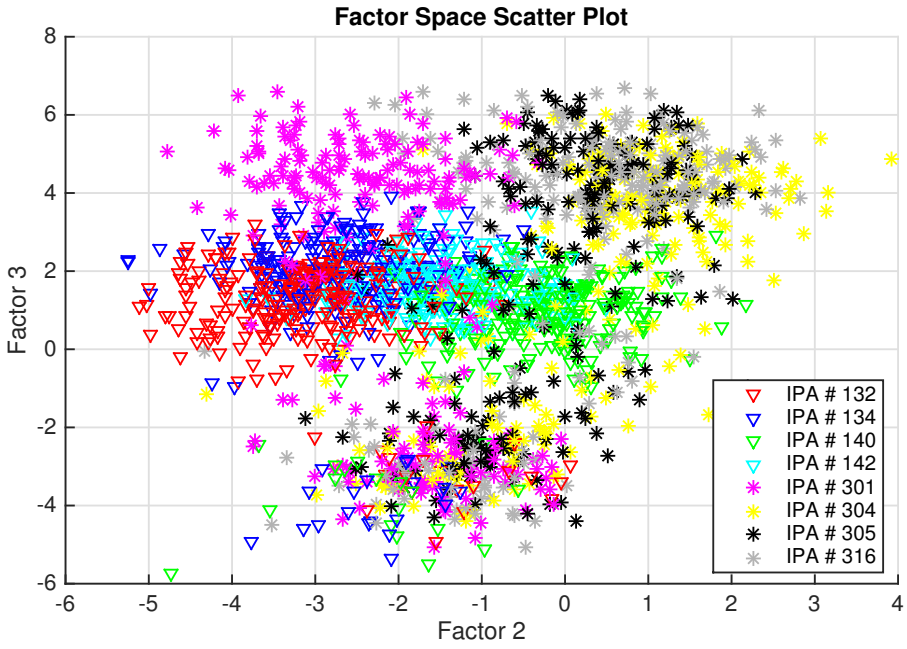


Figure 3.12: SDFA Configuration 1: Scatter plot of the latent variable $\mathbf{x}_t$ over synergy 2 vs. 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

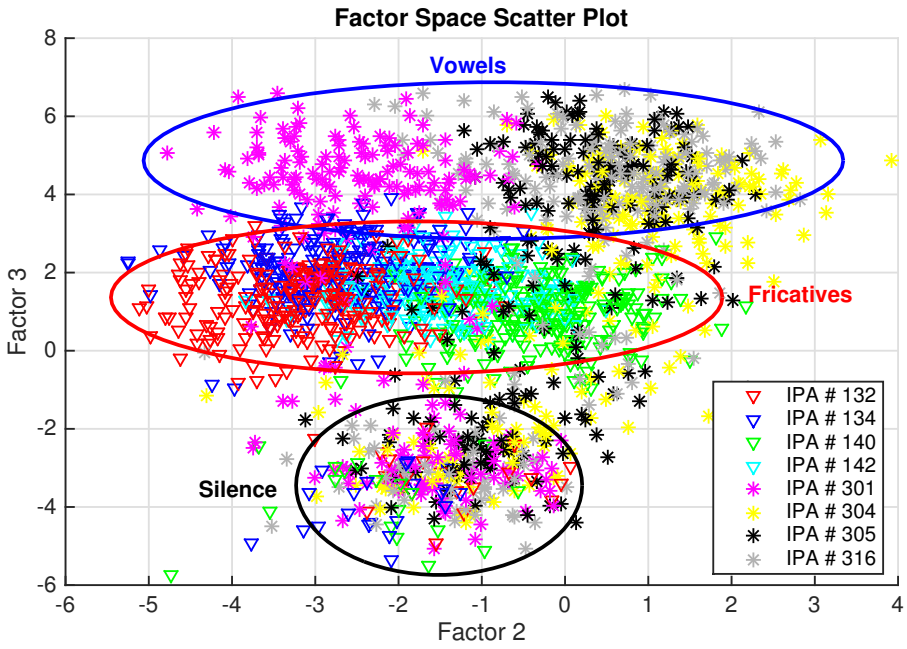


Figure 3.13: SDFA Configuration 1: Scatter plot of the latent variable $\mathbf{x}_t$ over synergy 2 vs. 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively. Clustering of vowels and fricatives within the space is indicated along with a third category corresponding to silence.

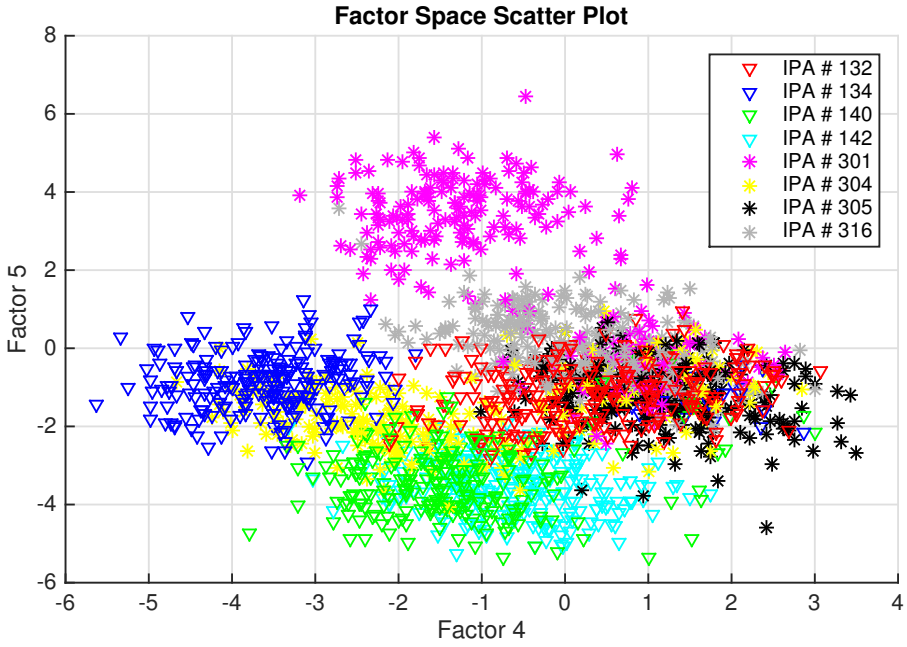


Figure 3.14: SDFA Configuration 1: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4 vs. 5 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

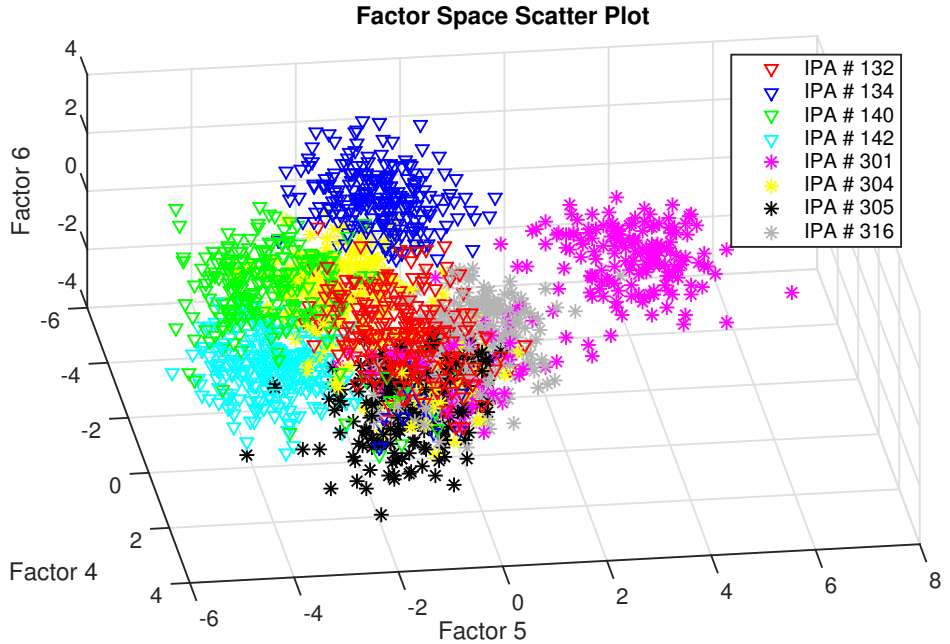


Figure 3.15: SDFA Configuration 1: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4, 5, and 6 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

SDFA Configuration 2

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 6$ samp. OR $p_t = f_t = 0.12$ s
- Number of observations: $n = 1,488$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.16 and 3.17 show the weights of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. The input synergies are again unstructured and very noisy. The increased future history, however, results in output synergies with increased structure. For example, synergy 1 captures the onset of a broadband sound around 0.14 seconds into the future. And synergy 3 shows a coordinated high frequency amplitude increase and a low frequency amplitude decrease. Some of the other output synergies indicate more complex responses with combinations of low frequency, high frequency, and broadband activity over time. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.6851 \pm 0.0649$.

Figures 3.18 and 3.19 both show scatter plots of the latent variable $\mathbf{x}_t$ for the eight different IPA phonemes shown in Figures 3.1 - 3.8. For each phoneme, five examples are plotted. While the individual IPA phoneme classes are somewhat localized within the space there is a good deal of overlap between classes especially with the vowels. There is some degree of continuity evidenced by the clustering of fricatives to the right of the graph and vowels to the left. However, this separation is not very distinct. There also does not appear to be a third cluster corresponding to silence as there was for configuration 1. Plotting of the latent variable over some of the other synergies does show slightly better localization, see Figure 3.20, but still has substantial overlap between IPA phoneme classes.

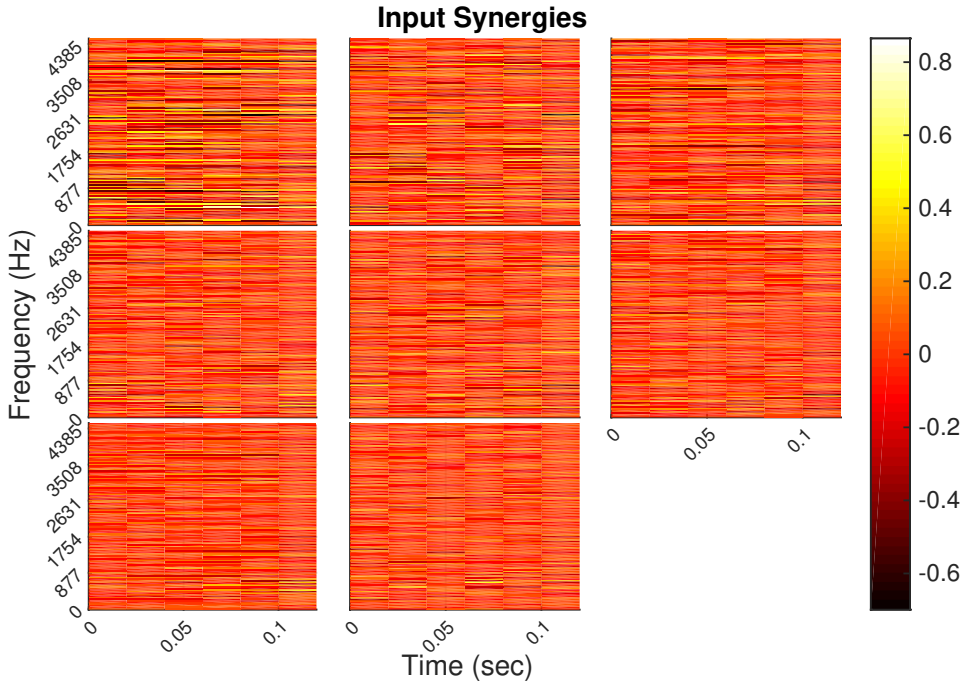


Figure 3.16: SDFA Configuration 2: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

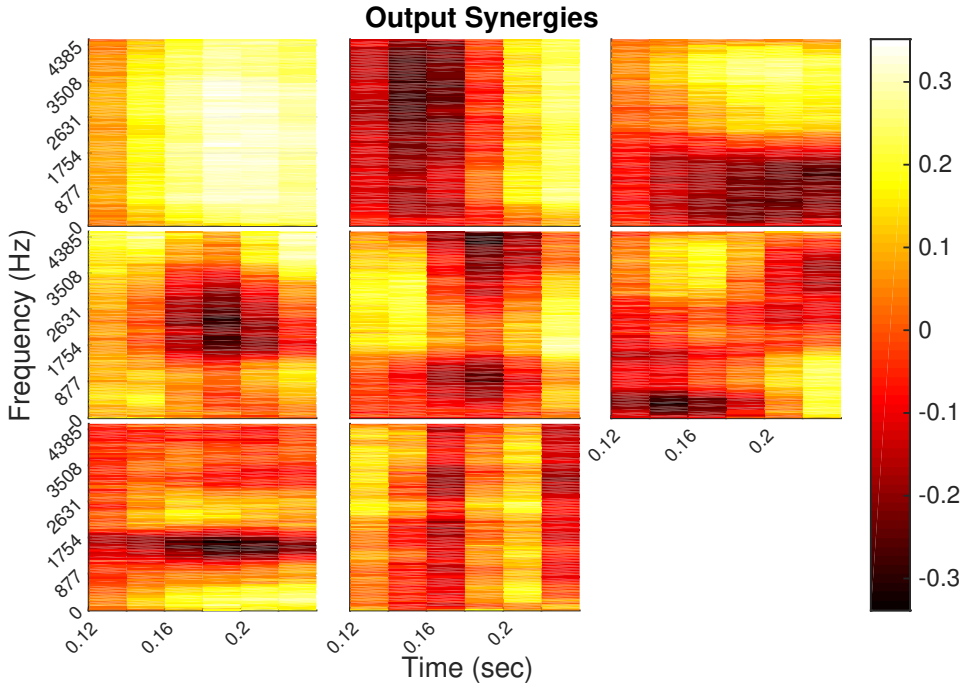


Figure 3.17: SDFA Configuration 2: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

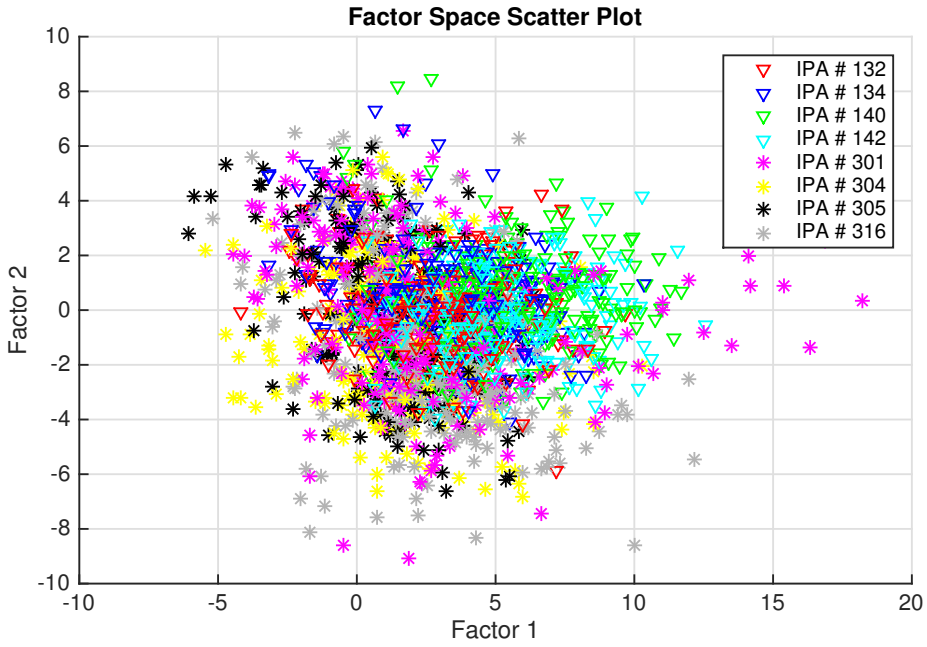


Figure 3.18: SDFA Configuration 2: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1 and 2 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

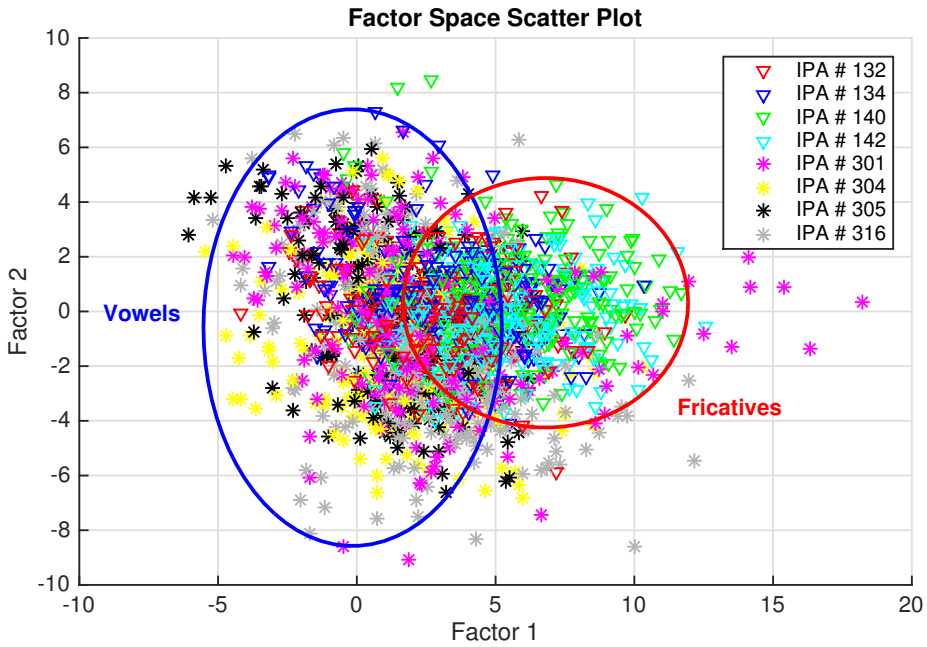


Figure 3.19: SDFA Configuration 2: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1 and 2 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively. Minor clustering of vowels and fricatives within the space is indicated.

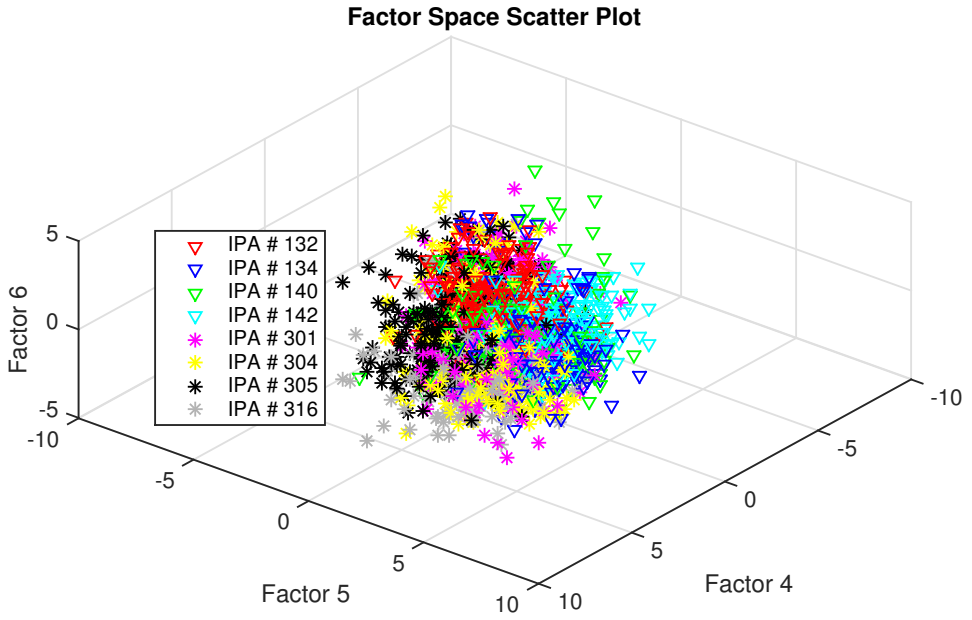


Figure 3.20: SDFA Configuration 2: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4, 5, and 6 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

SDFA Configuration 3

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 12$ samp. OR $p_t = f_t = 0.24$ s
- Number of observations: $n = 1,476$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.21 and 3.22 show the weights of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. Although a longer past history is used than in configuration 1, the input synergies are still very noisy. The increased future history, however, results in output synergies with a great deal of structure. For example, synergy 1 captures the onset of a broadband sound around 0.4 seconds into the future. And synergy 8 shows a low frequency amplitude increase around 0.4 seconds into the future. Some of the other output synergies indicate more complex responses with combinations of low frequency, high frequency, and broadband activity over time. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.7239 \pm 0.1122$.

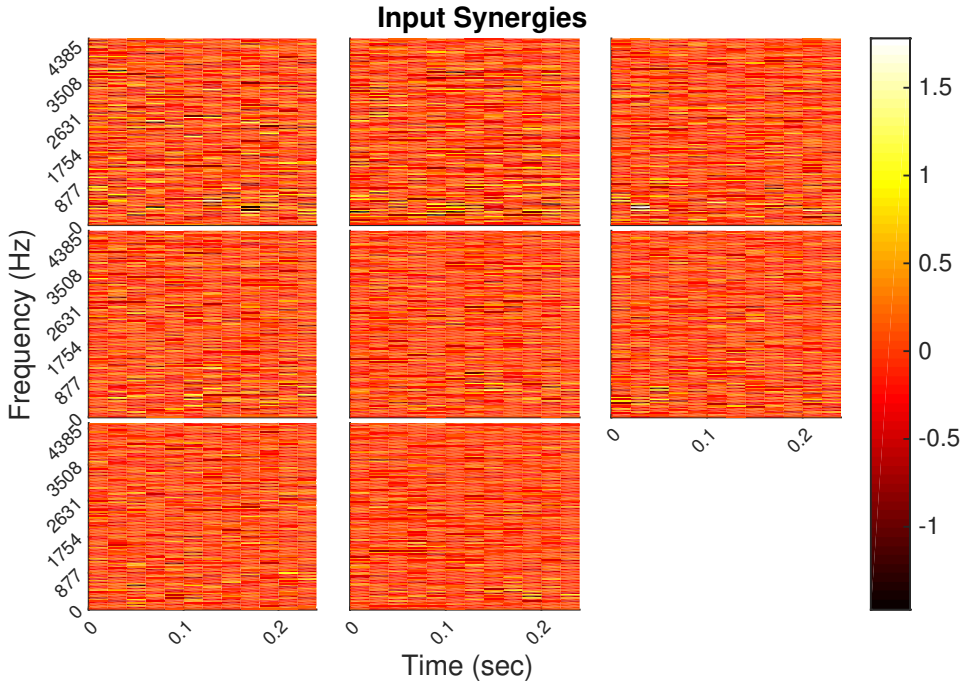


Figure 3.21: SDFA Configuration 3: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

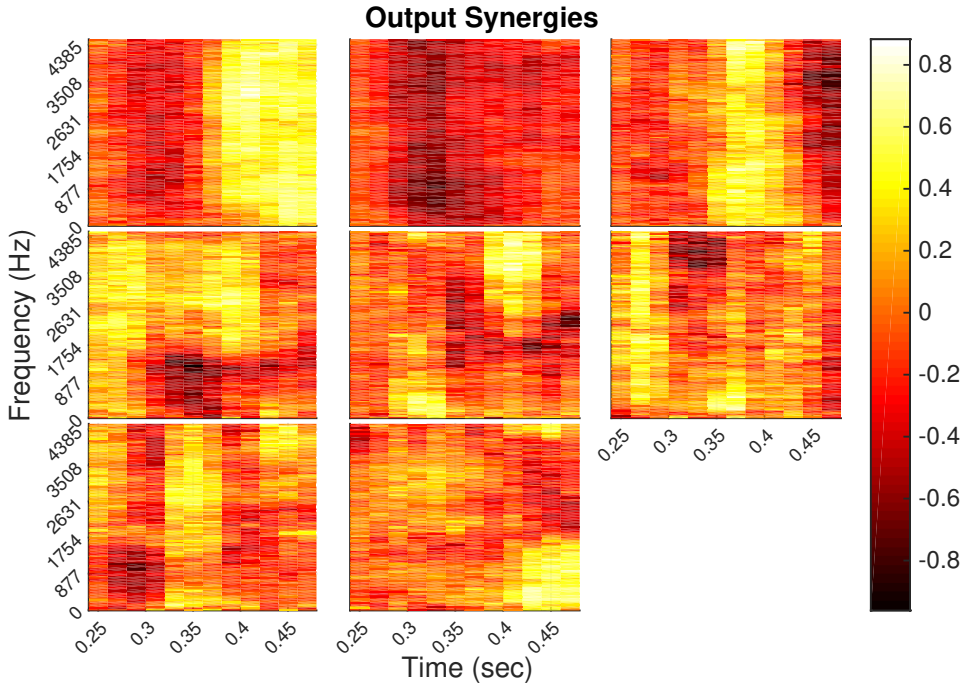


Figure 3.22: SDFA Configuration 3: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

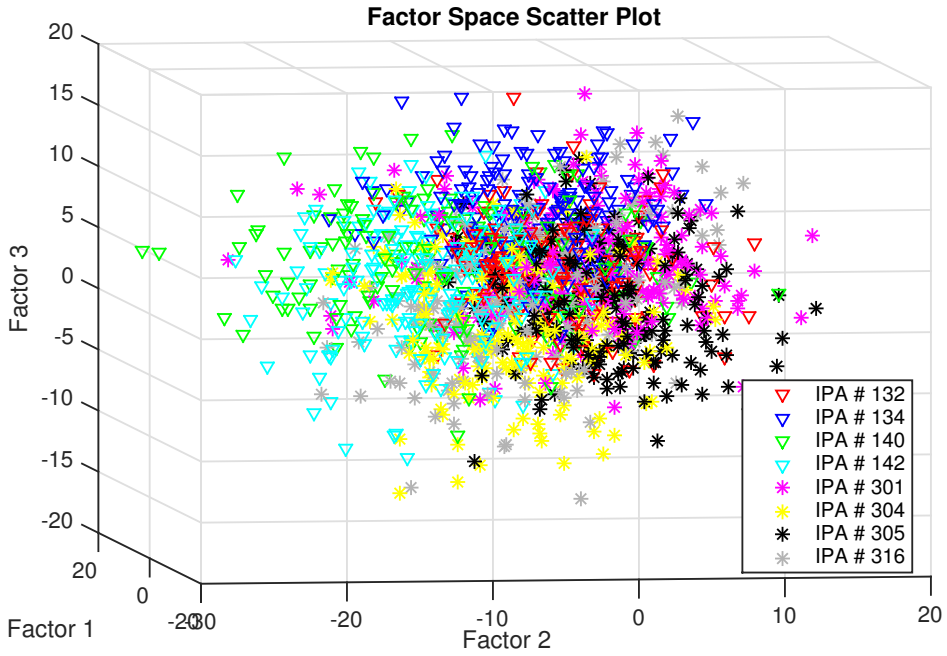


Figure 3.23: SDFA Configuration 3: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1, 2, and 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

Figures 3.23 and 3.24 both show scatter plots of the latent variable $\mathbf{x}_t$ for the eight different IPA phonemes shown in Figures 3.1 - 3.8. For each phoneme, five examples are plotted. It is difficult to separate the individual IPA phoneme classes within this space, meaning that the space does not exhibit localization very strongly. There still does appear to be some degree of continuity. Vowels tend to appear more on the right side of the figure and consonants on the left. The separation between these broad phonetic classes is minor and there is a great deal of overlap. There also does not appear to be a third cluster corresponding to silence as there was for configuration 1. Plotting of the latent variable over the other synergies does not reveal any more obvious separations either, see Figure 3.25.

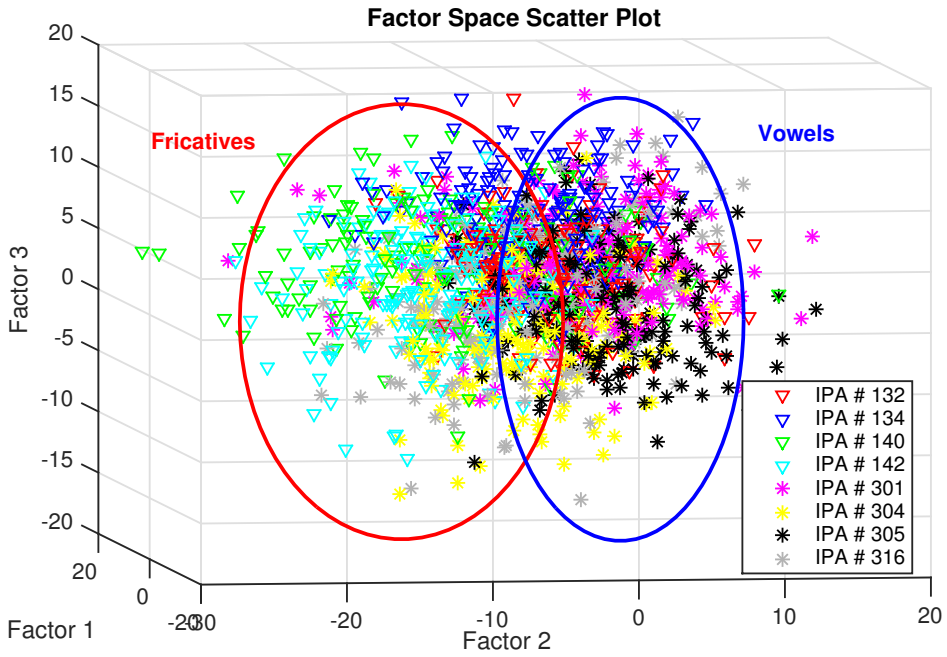


Figure 3.24: SDFA Configuration 3: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1, 2, and 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively. Minor clustering of vowels and fricatives within the space is indicated.

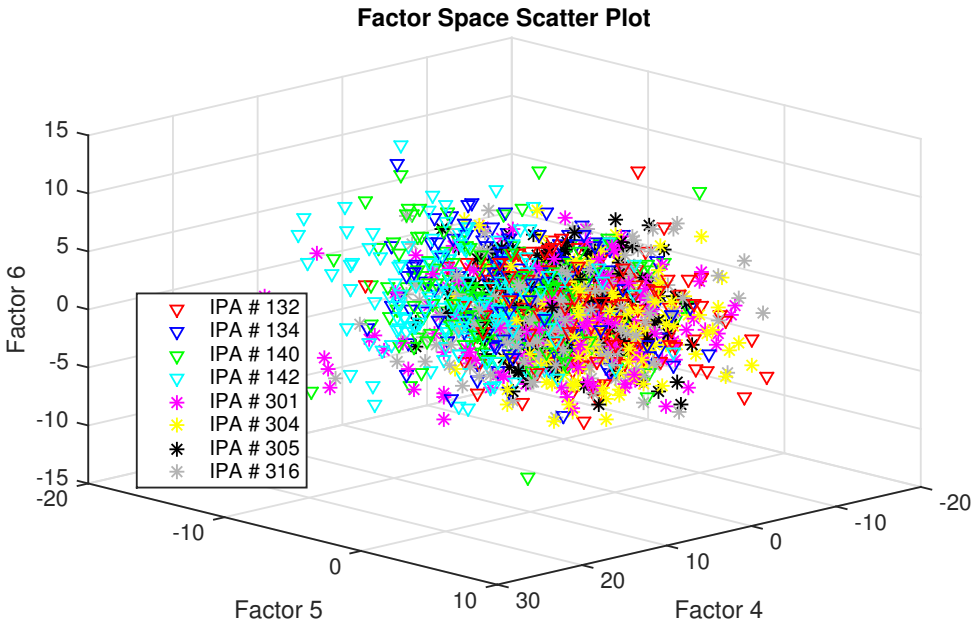


Figure 3.25: SDFA Configuration 3: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4, 5, and 6 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

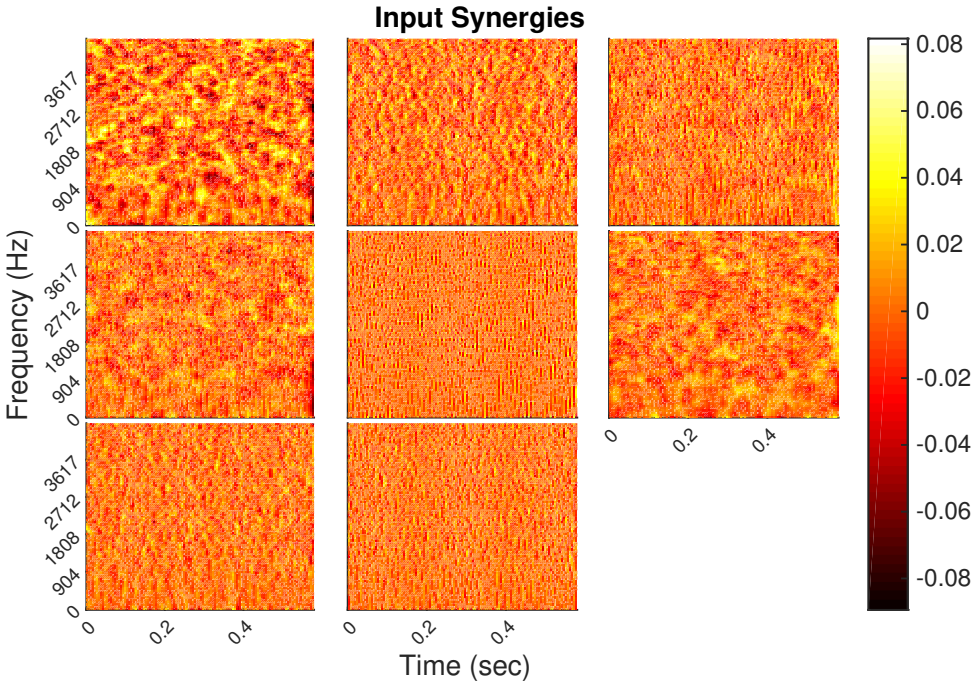


Figure 3.26: Textured SDFA Configuration: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

### DFT Window Length

Although I primarily was interested in investigating the effect of the history lengths $f$ and $p$ on the synergies I came across an interesting observation when initially selecting the DFT window length for generating the spectrogram. The input mapping exhibited very little structure in configurations 1, 2, and 3 which all used the same window length of 20 ms. However, if the window length is decreased the input mapping begins to obtain some structure. This effect is particularly evident with a long past history. Modifying configuration 1 by decreasing the window length to 5 ms and letting $p = 116$ and $f = 4$ results in the input mapping shown in Figure 3.26. The input synergies, while still random, show some degree of continuity over time and frequency. Also note how the different synergies are selective for different spatial frequencies.

These figures look very similar to multi-dimensional Brownian noise or what some refer to as Gaussian random fields. I am not sure what to make of this observation at this time, but this is certainly an intriguing result. A potential future line of investigation may be to relate this to the concept of

time-frequency reassignment which utilizes the spatial-temporal derivatives of the spectrogram and the phase of the DFT to improve the resolution of the magnitude spectrum [141].

### 3.3.3 Vocal Tract Synergies

The results from the human speech experiment are encouraging and indicate that this approach may be useful in developing a system that can learn to produce speech. The next step is therefore to apply this method to data obtained from the vocal tract simulator. The question is what features should be used. We know that the shape of the vocal tract by in large determines the sound that is produced. So it may be interesting to use the vocal tract area function and some acoustic component as features. And based on our review of integrated perceptual and motor primitives in Section 2.2.4, it may be worthwhile to include a motor component in our feature set as well. Therefore, in the following three sets of experiments, synergies will be learned from:

1. vocal tract area functions
2. vocal tract area functions and articulatory activations
3. vocal tract area functions, articulatory activations, and spectrograms

The sample data for these experiments was created by randomly articulating the vocal tract and recording the changing area function, articulatory activations, and audio signal. There are many ways one could choose to generate these random articulations. For example, one could simply randomly choose a new activation for each articulator at a set time step. However, since the articulators themselves are dynamical systems they will not instantaneously move to the new equillibrium position. Different articulators will respond to changes in the activation levels at different rates. Therefore, in order to sample the dynamics adequately, it follows that the time between articulatory activation changes should vary.

From the earlier discussion of sensory-motor primitives, see Section 2.2.4.2, we expect the synergies to learn something about the controller being used to generate the observations in addition to the dynamics of the system. So if the time between changes in articulatory activations varied, but all articulation

activations are changed at the same times, the model may learn that the vocal tract shape will change rather drastically and then settle into an equillibrium before changing again. Ultimately, we desire to use synergies to control the vocal tract to produce speech, and since speech is produced by fluid movement of the articulators this step response type behavior should be avoided. Additionally, that randomization method may make it more difficult for the model to learn the individual effects of each articulator. So, it follows that articulatory activations should be allowed to change at different times from one another. It also implies that changes in articulation should be somewhat smooth to avoid jerky movement of the tract.

Therefore the following method for generation of random articulatory activations was developed and used. See Figure 3.27 for an illustration of this process for a single articulator. A starting activation $a_i$ is chosen for the articulator from a standard uniform random distribution Uni$[0, 1]$. Then the time in seconds to hit the next activation for that articulator $t_j$ is chosen from a Gaussian distribution $\mathcal{N}(0.1, 0.25)$. The activation $a_j$ at this new time $t_j$ is then chosen from the same standard uniform distribution. The activations between these two points is then interpolated linearly such that $a_k = \frac{a_j - a_i}{t_j - t_i} t_k$ where $i < k < j$. This process is repeated until the length of the trial is reached. This is performed for each of the 29 actuators.

A total of $200 \times 0.5$ *sec* trials were run using the below simulation parameters. Note that the vocal tract area function and articulator values are sampled at a lower rate than the pressure wave at the lips. This is because the pressure wave is preprocessed into spectrogram features before being appended to the vocal tract features and must therefore have a higher sample rate to capture signal accurately.

Common Configuration

- Simulation Parameters
  - Simulation rate: $8,000$ Hz with $70 \times$ oversampling $= 560,000$ Hz
  - Trial Length: $t = 0.5$ s
  - Vocal Tract Sample Rate: $f_s = 50$ Hz
  - Sound Sample Rate: $F_s = 8,000$ Hz

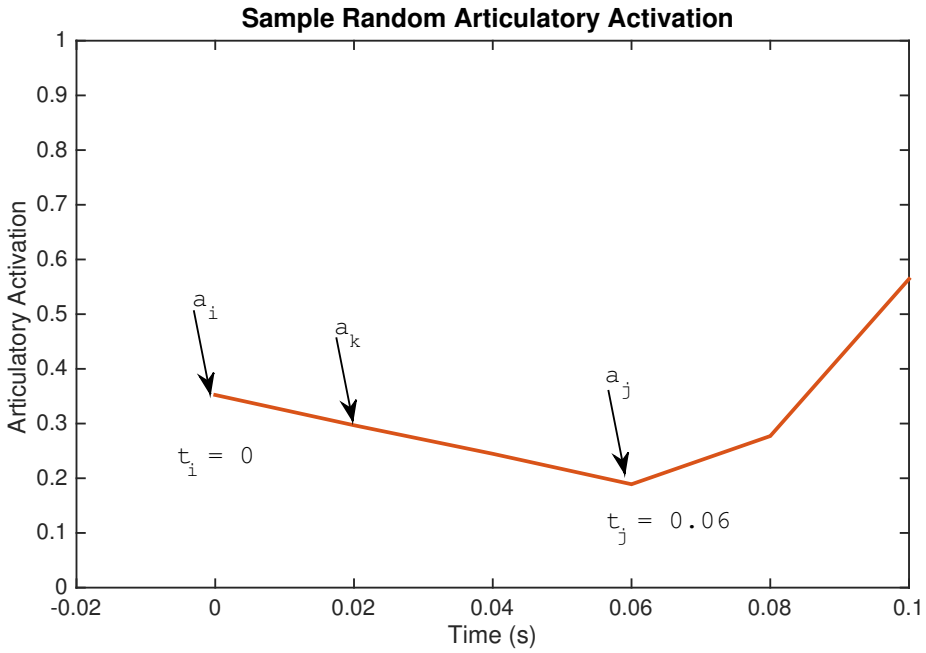


Figure 3.27: An example of what a random articulation for a single articulator may look like.

- Spectrogram Parameters
  - Window length: 20 ms or $N = 160$ samp.
  - Overlap: $l = 0$
  - Number of frequencies: $f_n = 117$

In the input and output mappings related to vocal tract area function, it is useful to note what part of the vocal tract the tube section #'s correspond to. Tubes 6-22 correspond to the lungs, 23-28 the bronchi, 29-34 the trachea, 35-36 the glottis, 37-63 the pharynx and oral cavity, and 64-77 the nasal cavity. In the Figures 3.1 - 3.8 the nasal cavity area function is plotted above the mouth instead of afterwards because although it is represented by tubes 64-77 in the software it is actually parallel to the oral cavity branching off at the velum at tube 50.

One other important factor that affects the performance of the SDFA algorithm is the choice of a scaling method. Initially I experimented with no scaling, but because the lungs have such a large variance in comparison to the rest of the vocal tract, they dominated the weighting of the synergies causing the rest of the vocal tract to be poorly represented. Therefore I

chose to use the individual feature variance scaling method, which consists of dividing each feature by its standard deviation. This remedied the problems with large variances, but introduced a problem with very small variance features. The nasal cavity features have a variance of around $1 \times 10^{-7}$ due to the stiffness passage's walls, which is quite small in comparison to the the rest of the vocal tract. The next smallest variance of any tubes are those that represent the glottis with a variance of $1.5 \times 10^{-5}$ or about two orders of magnitude larger. The issue is that now the nasal cavity features are given essentially equal weighting as the rest of the tract even though they are essentially unactuated as the only direct articulator is the velum. In addition, the nasal tract consist of the same number of tubes as the oral cavity whose importance is decreased due to the scaling of the nasal cavity areas. This distorts the synergies leads to worse results in terms of continuity and localization.This is a problem that warrants further investigation, but the approach I have taken is to set a minimum variance threshold, beyond which features are removed from the analysis. I have chosen that value to be $1 \times 10^{-6}$ which removes the nasal cavity sections from the analysis, but keeps the glottal sections.

#### 3.3.3.1 Vocal Tract Area Function Synergies

This experiment consists of applying the SDFA synergy learning algorithm to the vocal tract area function and evaluating the results in a similar manner as was done with human speech. In addition, as I am ultimately interested in control of the vocal tract using learned sensory-motor synergies, this experiment serves as a baseline for two subsequent experiments in which articulatory activations and acoustic features are added to the feature set. I have chosen to only show the results from the long configuration with $f = p = 12$ because the results from the tests with shorter histories are not well behaved. I think that there is a numerical issue that arises for the shorter histories during the least squares regression. This does not appear to be a result of having too few samples. The factor scatter plots from these tests show some of the fricatives being very far away from the rest of the phonemes and the weights of the input and output mappings are somewhat large in comparison to the more well behaved configurations.

Long Configuration

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 12$ samp. OR $p_t = f_t = 0.24$ s
- Number of observations: $n = 200$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.28 and 3.29 show the weigths of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. The input synergies are mostly unstructured as was the case with the speech experiments. However, it is important to note that the tube sections 0-5 and 65-88 have zero weight. Tubes 0-5 and 78-88 have zero weight because they are not used by the model in the specific configuration I have chosen to run these simulations. Tubes 65-77, the nasal cavity, have zero weight because although they are used in the simulation the variance of those tube sections is below the chosen threshold to be included in the analysis.

The output synergies, however, have some interesting structure. Synergy 1, for example, captures an increase in lung area. Synergy 2 captures a decrease in oral cavity area. Synergy 3 captures an increase in pharyngeal area. Upon closer inspection, in each of the synergies, clear patterns emerge showing differences between areas of the vocal tract that are actuated differently. For example, for the glottis each of the synergies shows distinctly different patterns with respect to the neighboring tube sections. Synergy 7 captures a large decrease in glottal size. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.5582 \pm 0.6125$.

The results of the output mappings are encouraging, but it is difficult to say anything definitive about what the algorithm has learned from simply looking at these mappings. However, the factor scatter plots, Figures 3.30 and 3.31, show that the model exhibits some localization. There is not a clear separation between vowels or fricatives but there is some grouping within the vowels indicating a low level of continuity. These plots are notably not as dense as the the similar plots from the human speech experiments. This is partially because there is only one test sample per IPA phoneme compared to the five for each phoneme for the human speech tests.

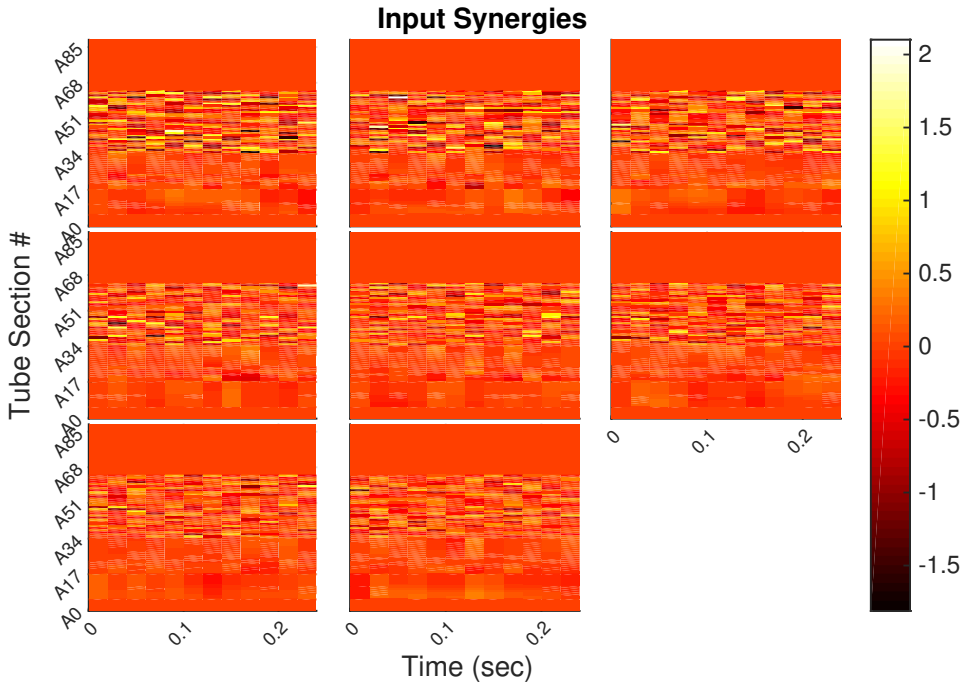


Figure 3.28: Area Function Long Configuration: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

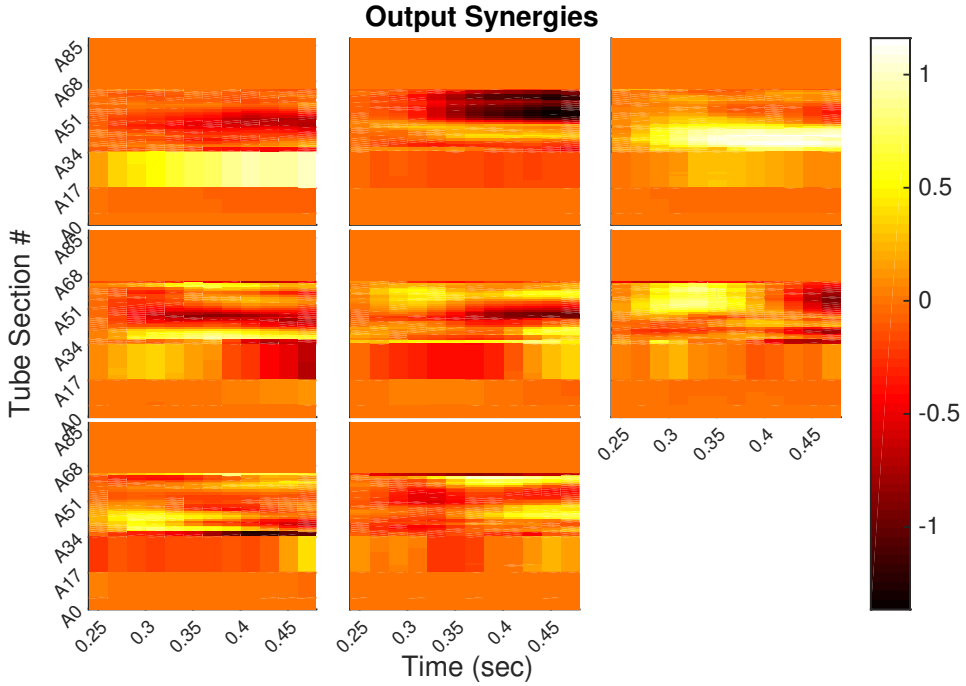


Figure 3.29: Area Function Long Configuration: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

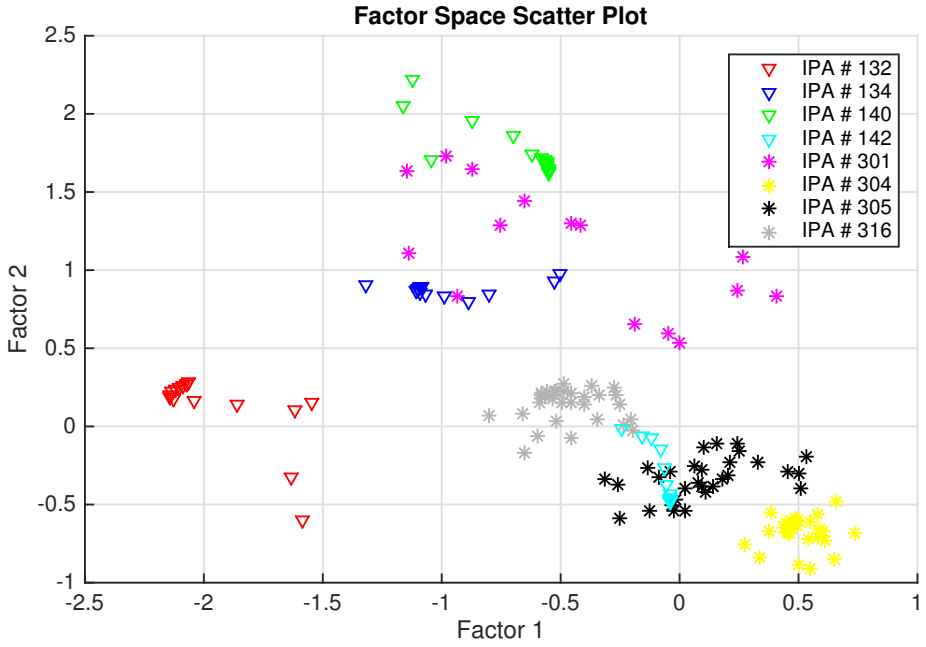


Figure 3.30: Area Function Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1 and 2 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

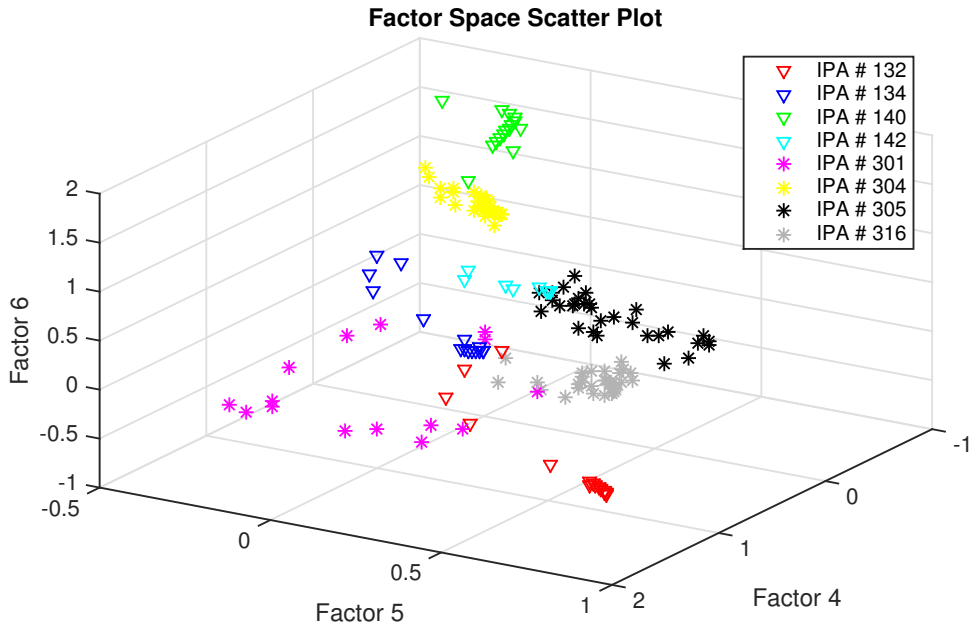


Figure 3.31: Area Function Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4, 5, and 6 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

#### 3.3.3.2 Vocal Tract Area Function and Articulation Synergies

This experiment is setup in a very similar fashion to the vocal tract area function synergies, with the exception being that articulator activations are included in the features provided to the SDFA algorithm. The articulations seem to partially resolve the issue with the shorter history configurations not being well behaved. The factor scatter plots for the short histories, $f = p = 3$, exhibit the same problem with some of the fricatives being very far away from the rest of the phonemes. However, this is resolved for the medium length case with $f = p = 6$. Only the results for the long configuration, $f = p = 12$, are shown because they adequately illustrate the improvement from the pure area function synergies and enable closer comparison.

Long Configuration

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 12$ samp. OR $p_t = f_t = 0.24$ s
- Number of observations: $n = 200$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.32 and 3.33 show the weights of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. The input synergies are mostly unstructured as was the case with the speech and vocal tract area function experiments. As with the pure area function synergies, tube sections 0-5 and 78-88 have zero weight because they are not used by the simulator, and the nasal cavity tubes 65-77 have zero weight because their variance is lower than the threshold.

The output synergies again, are more revealing. Synergy 1, for example, shows that the pharynx area increases as the oral cavity area decreases. In concert with these area function changes, articulators 15, 16, and 25 increase activations. These articulators correspond to the masseter, styloglossus, and genioglossus muscles. From our knowledge of the vocal tract model, we know that the masseter muscle closes the jaw, the styloglossus moves the tounge upward, and the genioglossus moves the tongue forward in the oral

cavity. The model has discovered that increasing these three articulator activations will decrease the area function in the oral cavity and increase the area function in the pharynx. This is remarkable. Todorov Ghahramani's experiments indicated that sensory-motor primtives could be learned for low dimensional dynamic systems, but this result indicates that it is possible with much higher dimensional systems using a different solution method [113].

The other synergies can be analyzed in a similar fashion and reveal equally interesting results. Synergy 2 has captured very strongly the connection between the lungs articulator, articulator 0, and the area function of the lungs. Simultaneously it identifies the relationship between the styloglossus and constriction in the oral cavity. It is important to point out that synergy 2 is capturing the coordination of the lungs and of the oral cavity in the same synergy. So as the volume of the lungs is increasing the area oral cavity area is decreasing and then increasing again. This type of coordination is exactly what we set out to discover. Synergy 6 identifies the connection between activation of the levator palatini and closing off the nasal cavity with the velum. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.4809 \pm 0.5133$.

The factor scatter plots, Figures 3.34 and 3.35, show that the model exhibits very strong localization evidenced by the separation between phones. In addition, the trajectories of the factors are fairly smooth, indicating that small changes in the vocal tract and articulation result in small changes in the factor space, meaning that continuity is also observed. The combination of these two properties implies that grouping along broad phonetic categories should be observed. This is indeed the case, as can be seen by the grouping of vowels and fricatives in the factor scatter plots.

So, the addition of articulator activation features to area function features results in a model that has both continuity and localization which leads to the grouping of like phones into broad phonetic categories. This indicates that the the combination of sensory and motor features in forming of synergies may create a model that better captures the dynamics of the system important for control.

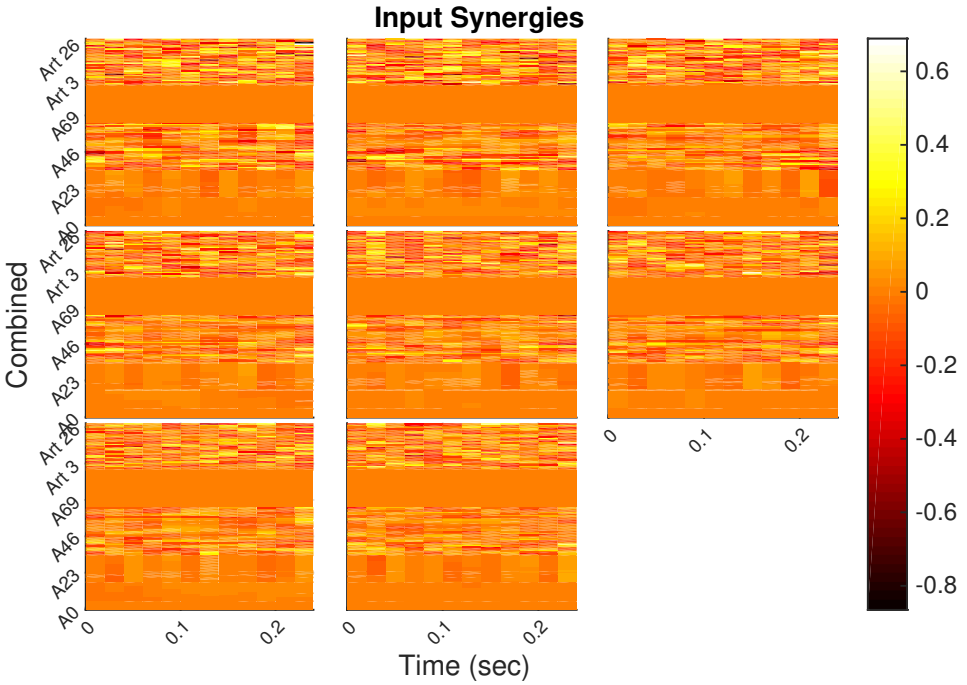


Figure 3.32: Area Function and Articulator Activation Long Configuration: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

#### 3.3.3.3 Vocal Tract Area Function, Articulation, and Spectrogram Synergies

The results of the vocal tract area function and articulation synergies experiment showed that the addition of motor features to sensory features enables the synergy learning algorithm to better identify broad phonetic categories. However, the area function is not the only sensory channel we have access to. The results of the human speech synergy experiments showed that acoustic features can be used to identify broad phonetic categories in structured speech. Therefore, including acoustic features in the model may lead to improved localization and continuity. This experiment tests this hypothesis by performing SDFA with vocal tract area function, articulatory activations, and spectrogram features. Two configurations, with medium and long history lengths, are evaluated.

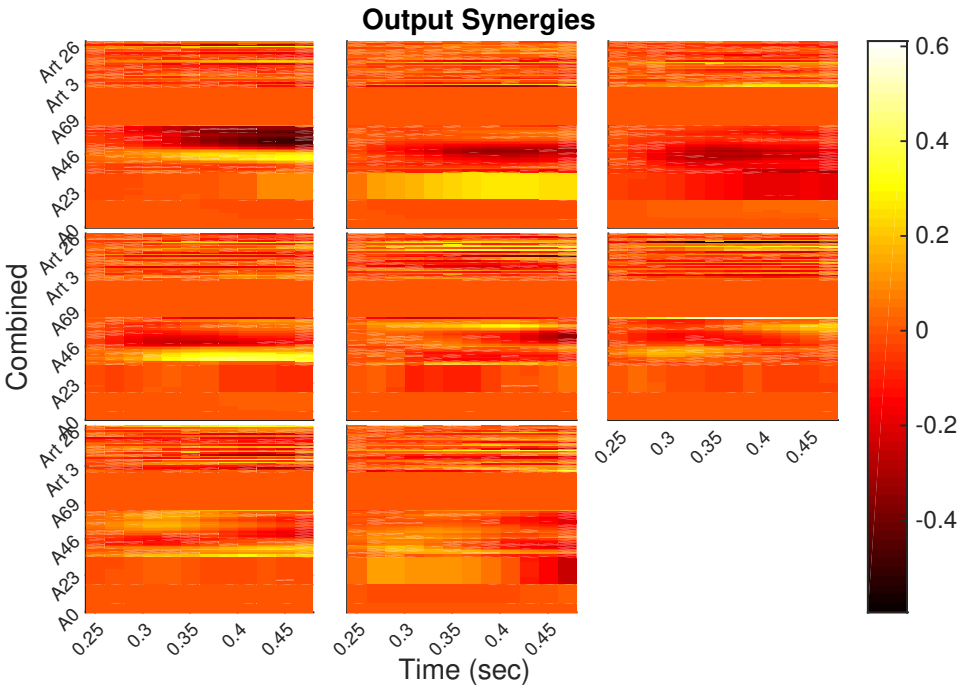


Figure 3.33: Area Function and Articulator Activation Long Configuration: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

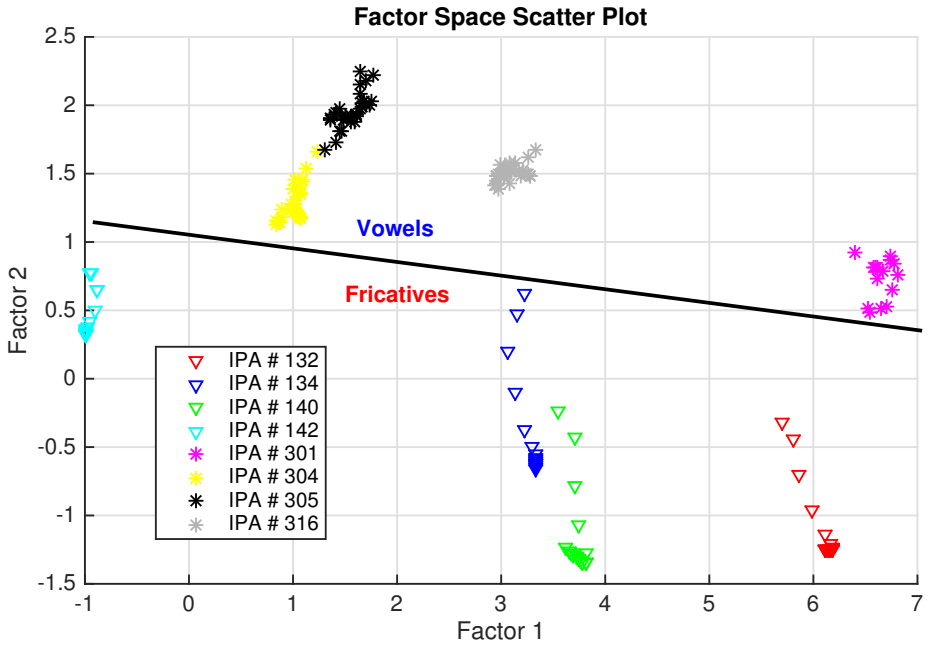


Figure 3.34: Area Function and Articulator Activation Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1 and 2 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

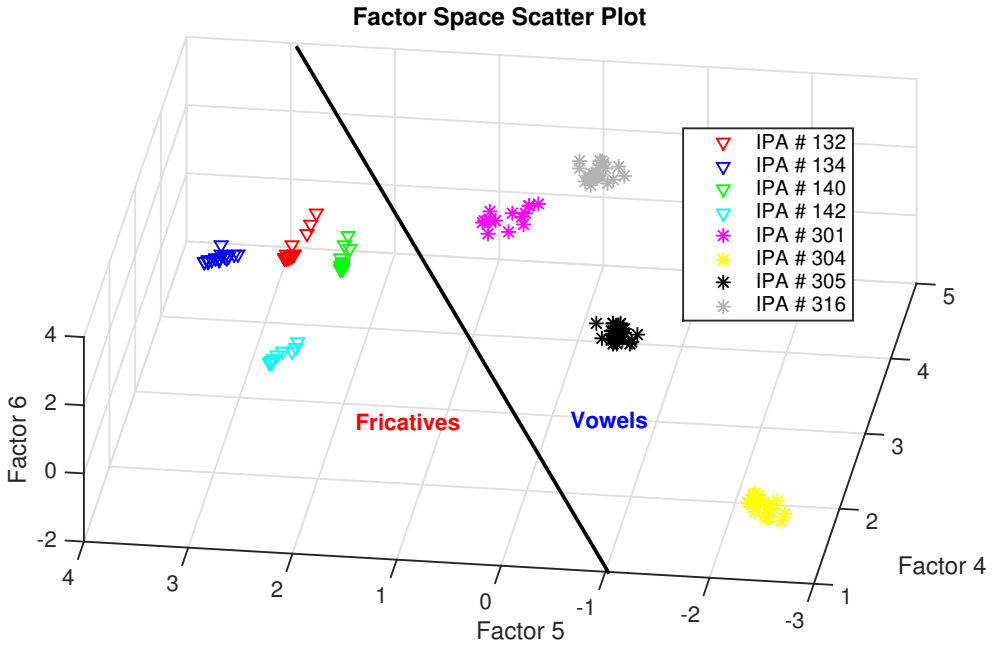


Figure 3.35: Area Function and Articulator Activation Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4, 5, and 6 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

Medium Configuration

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 6$ samp. OR $p_t = f_t = 0.12$ s
- Number of observations: $n = 200$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.36 and 3.37 show the weights of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. The input synergies are mostly unstructured as was the case with the other experiments. As with the other vocal tract synergies, tube sections 0-5 and 78-88 have zero weight because they are not used by the simulator, and the nasal cavity tubes 65-77 have zero weight because their variance is lower than the threshold.

The output synergies are much more structured, but are somewhat less interesting than the synergies without spectrogram features. The first two synergies show very little activation for the area and articulatory features,

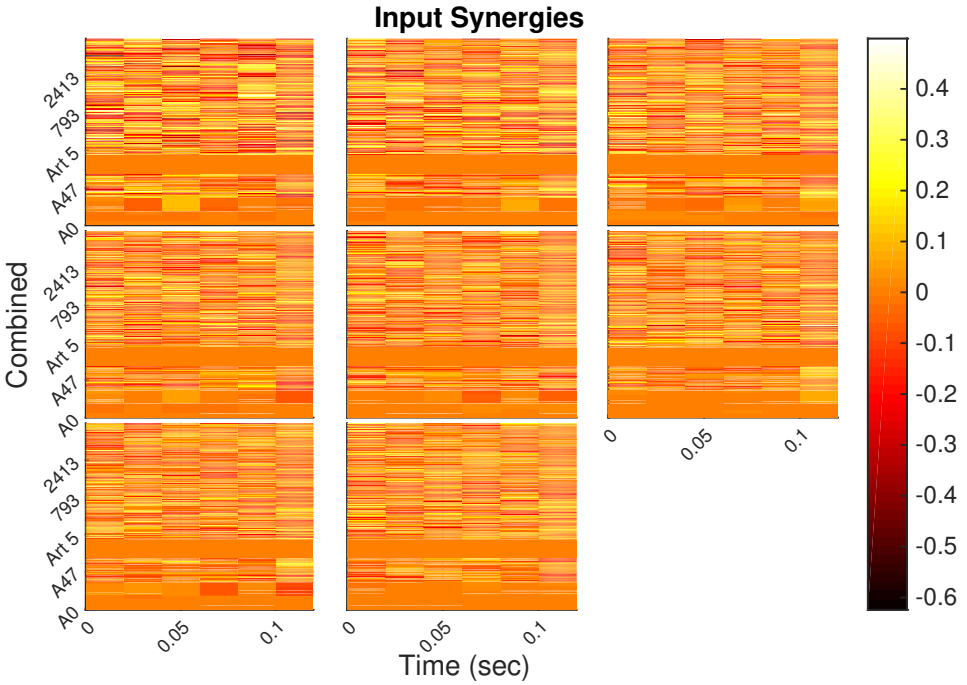


Figure 3.36: Area Function, Articulator Activation, and Spectrogram Medium Configuration: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

and decreasing broadband spectrogram features for both synergies. The following synergies are more interesting. Synergy 3 shows increasing area of the glottis and the pharynx accompanied by a slight decrease in mylohyloid activation which would cause the jaw to close slightly. The spectral features for synergy 3 start with broadband activation followed by a broadband decrease and then increase. It is notable that this broadband coordination over time is observed for each of the synergies shown, but unlike the human speech synergies, there does not appear to be much of a pattern along the frequency axis. This may be a numerical issue caused by having too few samples with actual sound being produced. Since the samples are generated via random articulation, and there is no biasing given to increase the chance of sounds being produced the majority of the trials produce little to no sound. This means that the spectral space is not being well sampled even though the vocal tract dynamics may be. This could be addressed by biasing the lungs articulator to force air out of the vocal tract, but that was not attempted in this experiment. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.6869 \pm 0.4182$.

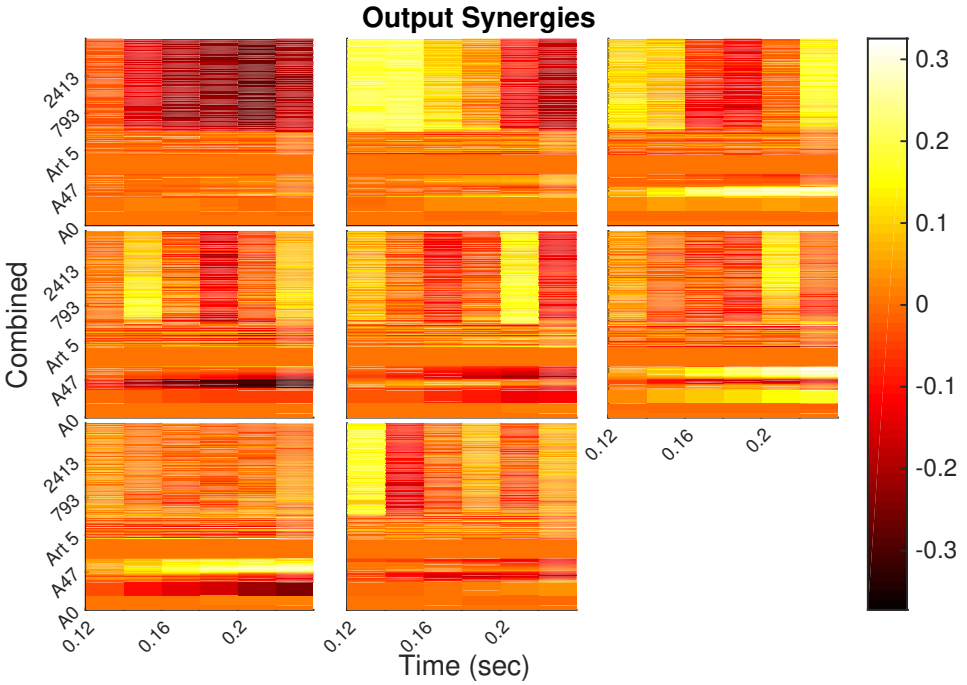


Figure 3.37: Area Function, Articulator Activation, and Spectrogram Medium Configuration: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

The factor scatter plots, figures 3.38 and 3.39, clearly exhibit both localization and continuity. In Figure 3.38 separation between vowels and fricatives is evident. IPA 301 does appear closer to the cluster of fricatives than the vowels, but it does not overlap with either. Looking at the factors 4,5, and 6 in Figure 3.39 reveals slightly different groupings. IPA 142 and 301 are grouped with the other vowels and fricatives respectively. Looking more closely at output synergies 5 and 6 in Figure 3.37, we can see that these factors are associated with opening of the jaw. This explains the alternative grouping because for all of the fricatives but IPA 142, the jaw is fairly closed, and for all the vowels but 301 the jaw is fairly open. What the model has identified is a natural grouping of phonemes produced with an open or a closed mouth.

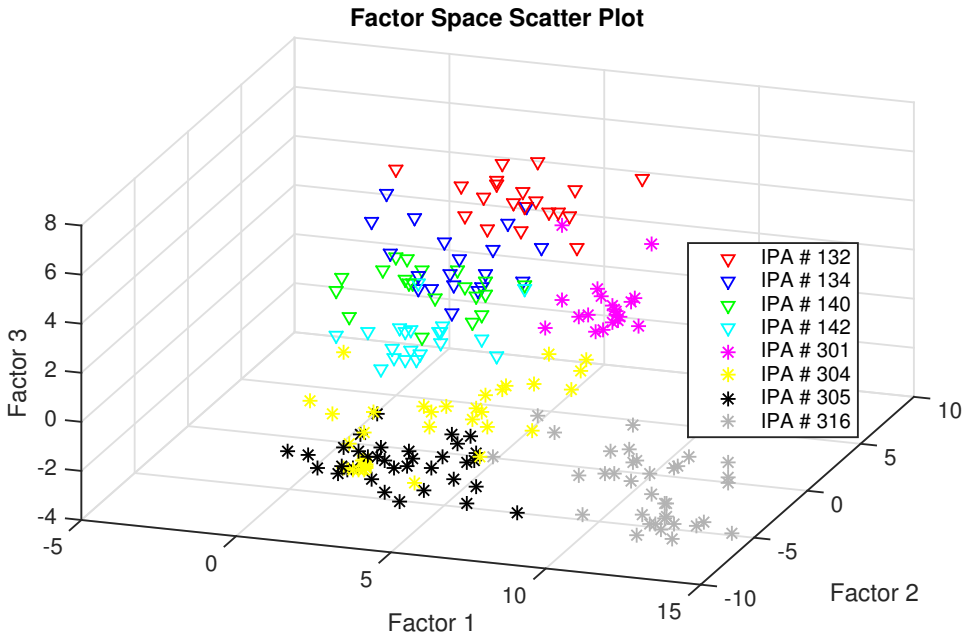


Figure 3.38: Area Function, Articulator Activation, and Spectrogram Medium Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1, 2, and 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

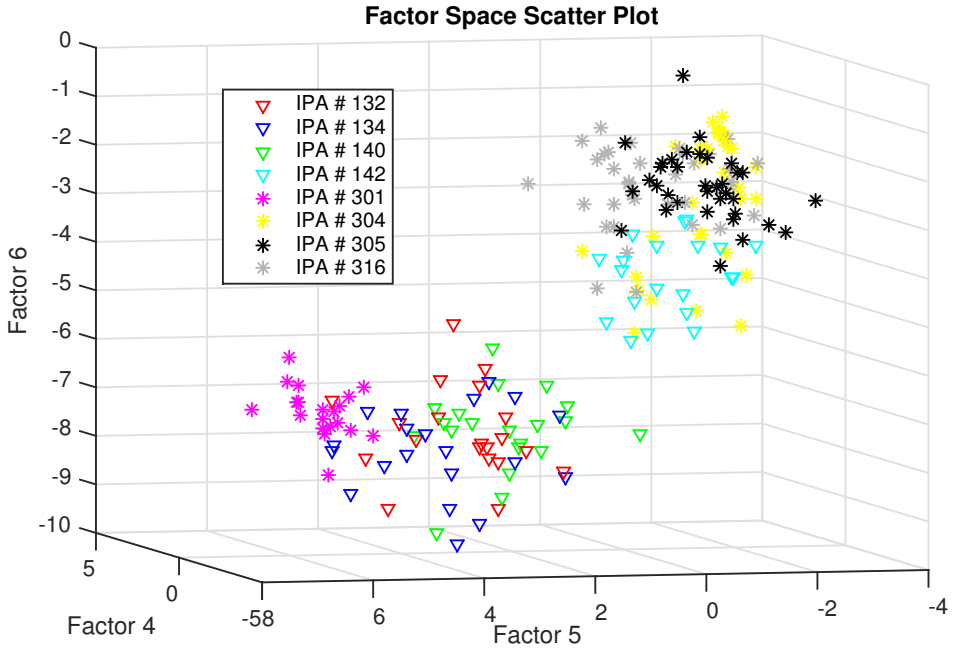


Figure 3.39: Area Function, Articulator Activation, and Spectrogram Medium Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 4, 5, and 6 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

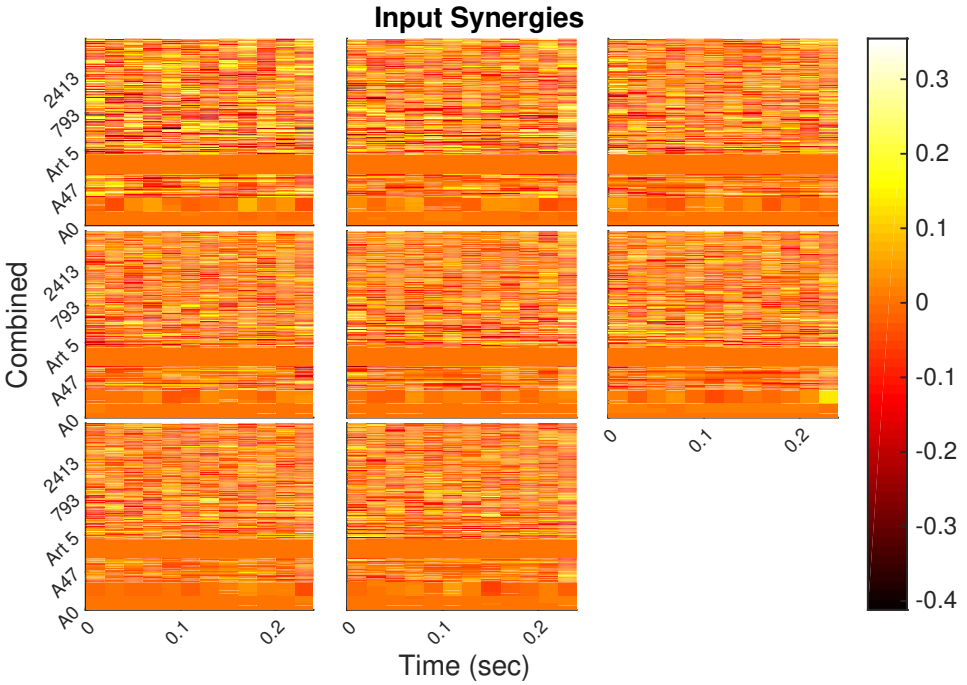


Figure 3.40: Area Function, Articulator Activation, and Spectrogram Long Configuration: Visualization of the input mapping $\tilde{\mathbf{K}}$ where each subplot is the $i^{th}$ reshaped input synergy $\mathcal{K}_i$ as in Equation 3.23. The synergy weights are unitless due to the normalization.

Long Configuration

- Number of synergies: $k = 8$
- Observation lengths: $p = f = 12$ samp. OR $p_t = f_t = 0.24$ s
- Number of observations: $n = 200$
- Normalization: Remove mean of each feature and divide individual features by their corresponding standard deviations

Figures 3.40 and 3.41 show the weights of the input and output synergies, composed by the $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{O}}$ matrices respectively. The input synergies are mostly unstructured as was the case with the other experiments. As with the other vocal tract synergies, tube sections 0-5 and 78-88 have zero weight because they are not used by the simulator, and the nasal cavity tubes 65-77 have zero weight because their variance is lower than the threshold.

This long configuration yields similar output mappings as the medium configuration. The first three output synergies are rather uninteresting with

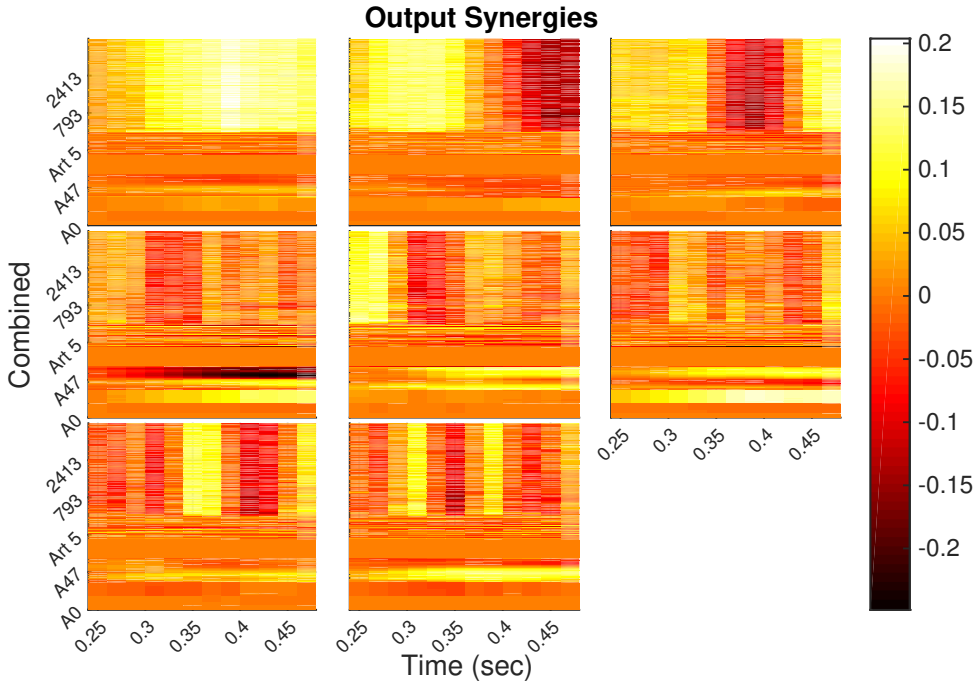


Figure 3.41: Area Function, Articulator Activation, and Spectrogram Long Configuration: Visualization of the output mapping $\tilde{\mathbf{O}}$ where each subplot is the $i^{th}$ reshaped output synergy $\mathcal{O}_i$ as in Equation 3.24. The synergy weights are unitless due to the normalization.

respect to the area function and articulation features. As with the medium configuration, the spectrogram features vary with time, but exhibit little structure along the frequency axis. Synergy 4 captures the relationship between the muscles articulating the jaw, the masseter and the mylohyoid, and decrease in oral cavity area. In coordination with the jaw closing, synergy 4 captures an increase in lung volume controlled by the decreasing of the lung articulator. Synergy 5 identifies the control of the velum via the levator palatini. Synergy 6 discovers that the oral cavity area can be increased by increasing the activation of the mylohyoid muscle.

So it appears that the addition of spectrogram features has somewhat decreased the structure in the output synergies, but that the algorithm is still able to identify relationships between articulators and area functions. It does not, however, seem to be able to identify any real connection to acoustic features. The multiple correlation coefficient mean and standard deviation for this configuration is $R^2 = 0.6569 \pm 0.3989$.

The factor scatter plots, Figures 3.42 - 3.44, for the long configuration exhibit both continuity and localization to some extent. Localization is stronger

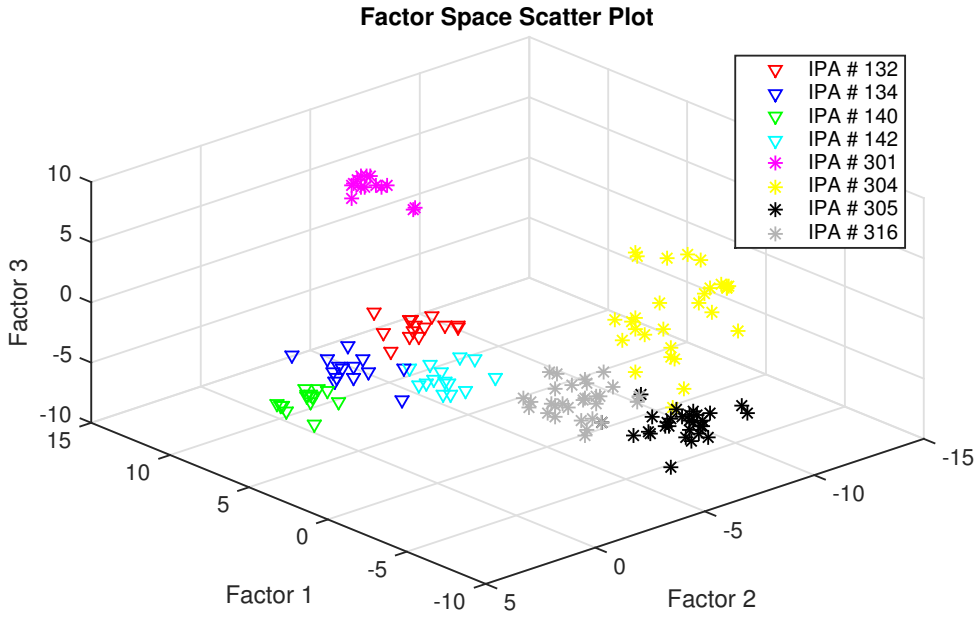


Figure 3.42: Area Function, Articulator Activation, and Spectrogram Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1, 2, and 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\bigtriangledown$ and $*$ respectively.

in the for the first three factors than it is for factors 6 and 7 as can be seen by the degree of overlap between phoneme classes. However, broad phonetic categories do still arise and are evident across factors 1, 2, and 3 as well as factors 6 and 7.

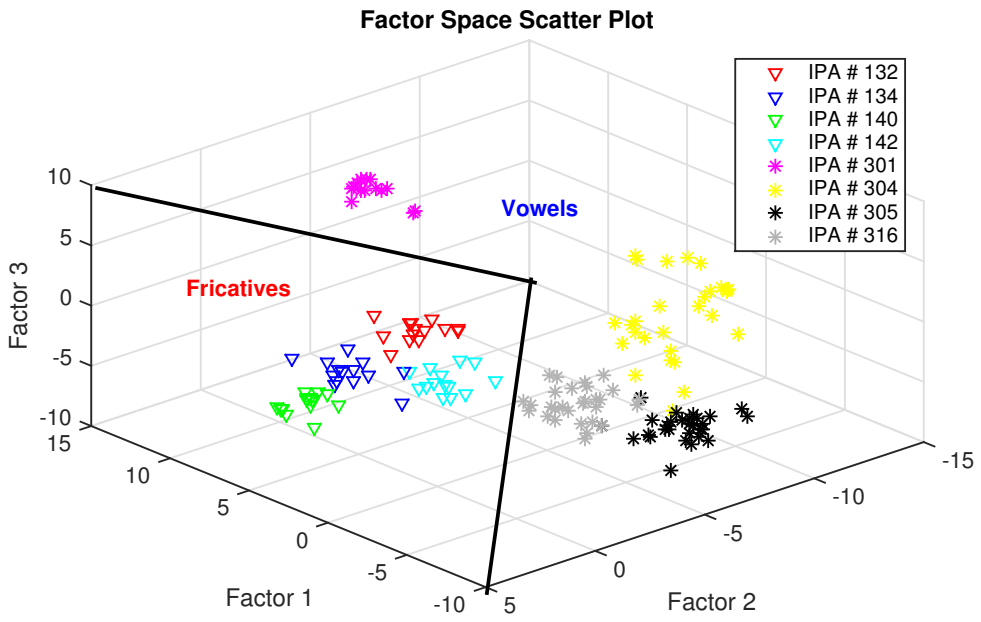


Figure 3.43: Area Function, Articulator Activation, and Spectrogram Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 1, 2, and 3 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

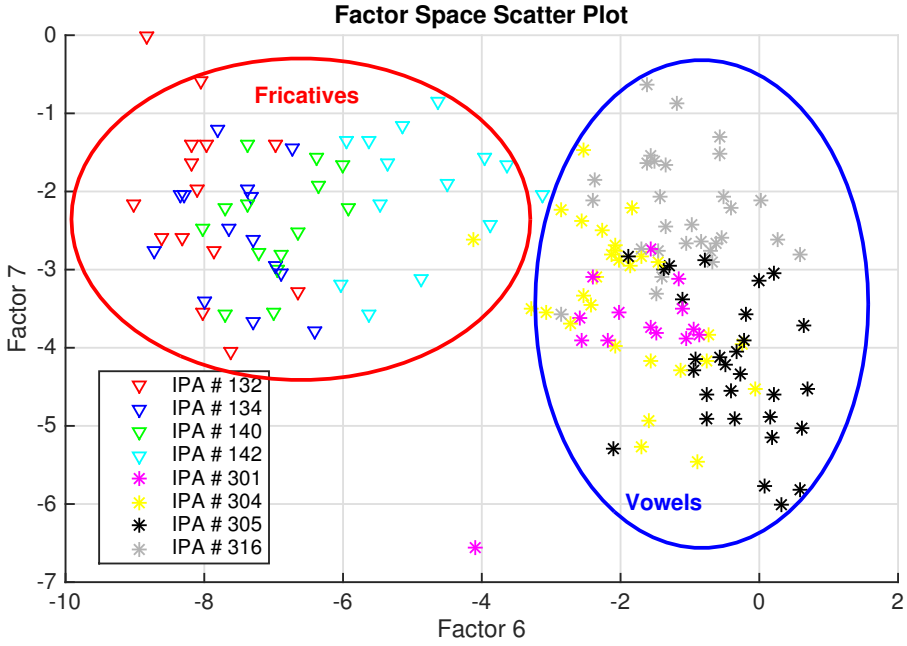


Figure 3.44: Area Function, Articulator Activation, and Spectrogram Long Configuration: Scatter plot of the latent variable $\mathbf{x}_t$ over synergies 6 and 7 for the eight IPA phonemes. Data points from fricatives and vowels are marked with $\triangledown$ and $*$ respectively.

### 3.3.4 Controlling the Vocal Tract with Synergies

Recall that a goal of developing vocal tract sensory-motor synergies is to enable bypassing of the curse of dimensionality in the task of learning to produce speech. The analysis of the learned vocal tract synergies revealed that these synergies capture coordination between muscles and tract shape which reflect the dynamics of the vocal tract model. The analysis also revealed that the learned synergies exhibit properties indicating that the factor space is a good low-dimensional approximation of the vocal tract model where control can be performed. In order to demonstrate that it is possible to use this synergy learning model to generate articulator outputs to the vocal tract, I implemented a simple controller based on the vocal tract area function and articulation synergies in Section 3.3.3.2.

The simulator is first initialized with an articulation which sets the initial shape of the vocal tract. Since no past history exists yet, it must be initialized somehow. I chose to initialize the past history by assuming that the initial state of the vocal tract model was held for the last $p$ time steps. Therefore the columns of $Y_o^{p-}$ are filled with the same initial vector $y_{t_o}$ which has been mean centered and scaled appropriately. The hidden factors $x_o$ are then computed using Equation 3.26. This is the low-dimensional representation of the vocal tract state. This state can then be fed to a controller that generates $x_{desired}$. For this simple demonstration the state is simply passed through without any modification.

$$x_{desired} = G(x_t) = x_t \tag{3.29}$$

The outputs are then generated according to Equation 3.27. The control $u_t$ is then rescaled and the mean is added back in to generate the actual articulator commands. These commands are then sent to the model which simulates the next timestep of the simulation generating a new observation vector $y_{t_1}$. The past history vector is then updated by removing the last column of $Y_o^{p-}$ shifting the other $p-1$ columns to the right and inserting $y_{t_1}$ in the first column to yield $Y_1^{p-}$. This process is repeated for the length of the trial.

I refer to this whole process as vocal tract sensory-motor synergy based control, because articulator outputs are being generated using the synergies and feedback from the model. However, the term control is used somewhat loosely here as no specific behavior is desired. It is more accurate to think

of the behavior of this system as a quasi step response.

I implemented this approach in C++ as part of the revised Praat simulator software. The computation of articulator outputs consists of matrix addition, subtraction, multiplication, and element wise division. I chose to use the open source GNU Scientific Library to perform these operations because it is fast and compatible with other software in the Language Acquisition and Robotics Laboratory. I have tested this approach with the vocal tract area function and articulation synergy long configuration, $p = f = 12$. This produces what can be described as oscillating movements of the vocal tract, with some intializations producing voiced sounds. This approach can also be used to control the vocal tract with individual synergies by setting all but one of the factors $x_t$ to zero. Again, some of these tests produced voicing.

Earlier on in my research I experimented with different methods for random articulation of the vocal tract as well as different versions of scaling in the SDFA algorithm. One of these early configurations used a randomized stimulation method that tended to create more drastic changes in articulator activations at the end of each trial. I was also experimenting with scaling only to account for a difference in the units of the features. Articulatory activations take on values between zero and one and tube areas are in units of $cm^3$. So I simply scaled the areas by the mean of area function standard deviations and the articulatory activations by the mean of their standard deviations. I also let $f = p = 13$ and restricted the analysis to 50 trials. This early configuration did not prove as useful for discriminating between IPA phonemes, but when used to control the vocal tract, it produced much more vowel like vocalizations. I think this is likely due to the fact that the scaling method weights features that have larger dynamic ranges more importantly in computation of the synergies. This created a biasing of the lungs which drive vocalization.

These results are all intriguing, but it is difficult to draw any conclusions. More research is needed in this area in order to determine how to use these synergies for true control. When designing a system that learns to speak, RL could be utilized to find a controller $G(x_t)$. The synergies could enable learning by using reduced dimensionality state $x_t$ instead of the original state $y_t$.

# CHAPTER 4

# CONCLUSION

This research brings us one step closer to developing a system that learns to produce speech. Due to the complexity and high dimensionality of the vocal tract standard reinforcement learning approaches can not be used to enable this learning. Yet, we know that biological systems have found a way to circumvent the curse of dimensionality. The study of perceputal processing provides the clue that optimal encoding via dimensionality reduction plays a key role. Research on motor control has indicated that use of muscle synergies may greatly simplify the control of complex dynamical systems.

There is also some evidence to support the idea that combining sensory and motor features is necessary to accurately characterize the system dynamics and may enable more efficient learning strategies. This hybrid synergy concept also aligns well with the concept of gestures in the articulatory phonology framework providing motivation for approaching the problem of learning to produce speech by first finding vocal tract sensory-motor synergies. In the SDFA formulation we can think of the input and output primitives as gestures because they represent the coordination of articulators and vocal tract shape. Points within the factor space correspond to vocal tract configurations. Trajectories within the factor space then can be considered analogous to gestural scores because they capture the combination and activation of the underlying synergies over time. Symbols, such as phonemes, can also be defined within this factor space as clusters of similar trajectories.

But, there is no guarantee that the SDFA algorithm will learn a model that is consistent with the hypothesis of synergies being analogous to gestures and factor trajectories analogous gestural scores. Therefore I set out to evaluate this hypothesis by performing a number of experiments using the SDFA algorithm and the vocal tract simulator. I primarily used two different methods of evaluating the quality of the learned synergies. I first analyze the patterns of coordination in the input and output synergies. Then I use the

learned synergies to analyze utterances of eight different phoneme by looking at the trajectories of the different symbols within the factor space. From this test, I determine if the factor space exhibits localization and continuity properties, which are important for making control feasible within the factor space, and for the formation of symbols as clusters of trajectories.

Before committing to the SDFA algorithm I first wanted to determine if it was a good candidate for learning primitives, as it had not been used for this purpose before. The early experiments by Poritz [140] showed that broad phonetic categories could be discovered by use of an autoregressive HMM on spectral features from recordings of human speech. The rationale was that if the SDFA method was capable of discovering similar categories, then it would indicate that the SDFA algorithm was worth further investigating.

I chose to apply the SDFA algorithm to spectrograms of human speech. The results of these experiments were very encouraging, especially for the short history configuration. Analysis of the different phonemes in the factor space revealed the discovery of three broad phonetic categories corresponding to vowels, fricatives, and silence. The space also clearly exhibits localization and continuity. Interestingly, the input primitives displayed little structure. This is somewhat surprising considering the fact that only the input primitives and not the output primitives are used to generate the factor scores. The output primitives show much more structure, with different primitives capturing broadband activity, low frequency components, high frequency components, and some harmonic components.

These results showed that the SDFA method is a good candidate for reducing the dimensionality of the system in a smart way. They also showed that the factor space can be used to define symbols. Overall, the results indicated that the SDFA method could be used to learn useful primitives from very high dimensional features. However, this experiment suffers from the same issue that most synergy studies do; the data upon which the synergies were trained is highly structured due to the task that is being performed, i.e. speaking English. Human speech is very structured and takes years of learning and practice to produce. The task of learning synergies for speech production is more difficult because it is less constrained.

In the vocal tract synergy learning experiments the system is required to learn from random excitation of the vocal tract model instead of from observations of speech being produced. I was interested in determining the

effect of combining sensory and motor features in synergy learning. I started out by applying the SDFA algorithm to learning vocal tract area function synergies. The learned output synergies captured a great deal of structure, identifying segments of the tract that move in concert. In addition, the factor space does exhibit some degree of localization and continuity although there is overlap between phoneme classes and no apparent broad phonetic categories. However, when the articulatory activations are included with the area functions, the resulting synergies are much more interesting. The output synergies capture the relationship between various articulatory muscles and tract shape. The factor space also exhibits strong localization and continuity with little overlap between phonemes. Broad phonetic categories emerge naturally in the factor space as well. So we can conclude that use of combined sensory-motor features enabled learning of synergies better suited to the task of speech production. In addition, the inclusion of articulator activations opened up the possiblity for using these synergies for performing control of the vocal tract.

Adding spectrogram features to the area function and articulatory activation features resulted in output synergies that still capture relationships between tract shape and articulatory muscles, but little correspondence with the spectral features. This is likely due to having too few samples where any sound is actually produced. This could be remedied by biasing the lungs articulatory muscle to cause air to exit the tract. However, it is not clear that it is necessary to include an acoustic features in the primitives at all.

Taken together, all of these results indicate that sensory-motor synergies offer a promising approach for learning to control a vocal tract to produce speech. As hypothesized, the area function and articulatory activation synergies capture coordinations between vocal tract shapes and articulatory muscles, closely resembling the concept of gestures in the articulatory phonology framework. This research extends that framework by showing that it is possible that these gestures are not preprogrammed or innate and instead that they can be learned learned through interaction with the world. The fact that the factor space exhibits strong localization and continuity in combination with the natural formation of broad phonetic categories within the space indicate that this model may be useful for performing control and for constructing symbols.

This is a somewhat unique result because very few studies have looked at

learning vocal tract synergies, and those that have rely on recordings of vocal tract shape during speech production. As pointed out earlier, human speech is highly structured so it is not surprising that primitives would emerge from this analysis. The problem is that there is no way to tell whether the structure that these synergies discovered reflects the structure in the underlying control units recruited for producing speech or whether they reflect the structure in the language.

In this research, synergies were discovered from random articulation of the vocal tract, meaning that the synergies constitute a compact representation of the vocal tract dynamics. Based on our understanding of dimensionality reduction, we can say that the individual synergies actually represent different modes of the vocal tract. One interesting aspect of these results is that the learned synergies generate a space where different phonemes show up as distinct and broad phonetic categories can be defined. This implies that phonemes and broad phonetic categories aren't just the result of an arbitrary taxonomy imposed by language, but actually represent physically different modes of the human vocal tract.

## 4.1 Future Work

The research presented in this thesis is a first step in developing a system that can learn to produce speech with little prior knowledge. I have laid out the motivation for approaching this as a dimensionality reduction problem and provided evidence that a vocal tract sensory-motor primitive representation may enable bypassing of the curse of dimensionality in reinforcement learning. Therefore, the next step is to validate this approach by using these synergies to perform reinforcement learning for producing speech. Although there is no textbook solution for how to implement this. Many decisions about what type of reinforcement learning to use, the appropriate cost functions, how to integrate the primitives into a RL framework, etc. must be made. Also, in order to produce a variety of sounds it is likely that some form of a hierarchical controller will be required. It is possible that forming a second layer of synergies incorporating acoustics and the activations of the lower-level primitives may be useful. The probabilistic formulation of DMPs may be useful in forming this second layer of synergies.

One problem that may arise is that the initial sensory-motor synergies may not be adequate for producing some sounds. It is possible that this would require creation of a new synergy. However, this raises a great deal of questions. How would this new synergy be learned? Would this affect the production of other sounds that have already been learned via reinforcement learning? This would essentially require development of an online synergy learning algorithm as opposed to the current batch based method. It would also require the development of an adaptive basis reinforcement learning method.

Another way of addressing this issue is to learn many different sets of synergies that approximate the dynamics of the vocal tract around different tract configurations, essentially creating a number of different locally linear models. This may enable better speech production while still keeping the dimensionality of the controller low. It does add the complications of determining how to initially learn these various models and how to choose which model is necessary for a given task, but I think it is worth further investigating.

In regards to the synergy learning algorithm, I am interested in trying out the expectation maximization approach to solving the DFA model. This may help shed some light as to why the output synergies are structured, but the input primitives are noisy. It may also be worthwhile to look at developing sensory-motor synergy learning algorithms that use a dynamic form of ICA or NMF instead of factor analysis.

I am also interested in extending this approach to other problem domains such as robotics, autonomous construction, and general motor control. More broadly I think this approach could be useful in the task of transfer learning. If we are interested in reusing aspects of learned behaviors, it makes a lot of sense to use synergies to develop more modular controllers. Synergies could also offer a way to make RL more understandable and make the connection between symbolic processing and action, that is difficult to do with standard RL techniques.

# REFERENCES


[1] A. M. Turing, "Computing Machinery and Intelligence," *Mind*, vol. 59, no. 236, pp. 433–460, 1950.

[2] A. NEWELL and H. A. SIMON, "Computer Science as Empirical Inquiry: Symbols and Search," *Communications of the ACM*, vol. 19, no. 3, pp. 113–126, 1976.

[3] P. Boersma and D. Weenink, "Praat: Doing Phonentics By Computer," 2016. [Online]. Available: http://www.praat.org/ Accessed March 5, 2016.

[4] G. Fant, *Acoustic Theory of Speech Production: With Calculations based on X-Ray Studies of Russian Articulations*. The Hague: Mouton Publishers, 1971, vol. 2.

[5] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Prentice-Hall, 1978.

[6] C. H. Coker, "A Model of Articulatory Dynamics and Control," *Proceedings of the IEEE*, vol. 64, no. 4, pp. 452–460, 1976.

[7] S. E. Levinson, *Mathematical models for speech technology*. John Wiley & Sons, 2005.

[8] "The International Phonetic Alphabet Number Chart," 2005. [Online]. Available: https://www.internationalphoneticassociation.org/sites/default/files/IPA_Number_chart_(C)2005.pdf Accessed Mar. 27, 2017.

[9] M. R. Portnoff, "A quasi-one-dimensional digital simulation for the time-varying vocal tract," Ph.D. dissertation, Massachusetts Institute of Technology, 1973.

[10] M. Sondhi, "Model for wave propagation in a lossy vocal tract," *The Journal of the Acoustical Society of America*, vol. 55, p. 1070, 1974.

[11] H. K. Dunn, J. L. Flanagan, and P. J. Gestrin, "Complex zeros of a triangular approximation to the glottal wave," *Journal of the Acoustical Society of America*, vol. 34, 1962.

[12] D. O'Shaughnessy, "Linear Predictive Coding." *IEEE Potentials*, vol. 7, no. 1, pp. 29–32, 1988.

[13] H. Wakita, "Direct Estimation of the Vocal Tract Shape by Inverse Filtering of Acoustic Speech Waveforms," *IEEE Transactions on Audio and Electroacoustics*, vol. 21, no. 5, pp. 417–427, 1973.

[14] M. Carreira-Perpinán, "A review of dimension reduction techniques," Department of Computer Science. University of Sheffield, Tech. Rep., 1997.

[15] L. J. P. Van Der Maaten, E. O. Postma, and H. J. Van Den Herik, "Dimensionality Reduction: A Comparative Review," *Journal of Machine Learning Research*, vol. 10, pp. 1–41, 2009.

[16] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 1, pp. 559–572, 1901.

[17] I. T. Jolliffe, *Principal Component Analysis, Second Edition*, 2002.

[18] I. K. Fodor, "A survey of dimension reduction techniques," *Technical Report UCRL-ID-148494, Lawrence Livermore National Laboratory*, 2002.

[19] C. Spearman, "General Intelligence, Objectively Determined and Measured," *The American Journal of Psychology*, vol. 15, no. 2, pp. 201–292, 1904.

[20] A. Montanari, "Factor Analysis." [Online]. Available: http://www2.stat.unibo.it/montanari/Didattica/Multivariate/FA.pdf Accessed Nov. 4, 2016.

[21] M. B. Richman, "Rotation of principal components," pp. 293–335, 1986.

[22] R. Cattell, *The scientific use of factor analysis in behavioral and life sciences*. Springer Science \& Business Media, 1978.

[23] K. Holzinger, "A simple method of factor analysis," *Psychometrika*, vol. 9, no. 4, pp. 257–262, 1944.

[24] S. J. Press, *Applied Multivariate Analysis: Using Bayesian and Frequentist Methods of Inference*, 2nd ed., 2012.

[25] D. Lawley and A. Maxwell, "Factor analysis as a statistical method," *Journal of the Royal Statistical Society Series D (The Statistician)*, vol. 12, no. 3, pp. 209–229, 1962.

[26] C. Jutten, "Calcul neuromimétique et traitement du signal: analyse en composantes indépendantes," Ph.D. dissertation, Grenoble INPG, 1987.

[27] C. Jutten and J. Herault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture," *Signal processing*, vol. 24, no. 1, pp. 1–10, 1991.

[28] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 10, no. 3, pp. 626–34, 1999.

[29] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, oct 1999.

[30] D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, no. 1, pp. 556–562, 2001.

[31] H. B. Barlow, "Possible Principles Underlying the Transformations of Sensory Messages," *W.A. Rosenblith (Ed.), Sensory Communication*, pp. 217–234, 1961.

[32] D. H. Hubel and T. N. Wiesel, "Receptive Fields of Single Neurones in the Cat's Striate Cortex," *Journal of Physiology*, vol. 148, no. 3, pp. 574–591, 1959.

[33] D. H. Hubel and T. N. Wiesel, "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex," *The Journal of Physiology*, vol. 160, no. 1, pp. 106–154, 1962.

[34] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst, "Spatial summation in the receptive fields of simple cells in the cat's striate cortex." *The Journal of physiology*, vol. 283, p. 53, 1978.

[35] S. Marĉelja, "Mathematical description of the responses of simple cortical cells," *Journal of the Optical Society of America*, vol. 70, no. 11, pp. 1297–1300, 1980.

[36] A. J. Bell and T. J. Sejnowski, "The "independent components"of natural scenes are edge filters," *Vision Research*, vol. 37, no. 23, pp. 3327–3338, 1997.

[37] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," pp. 607–609, 1996.

[38] B. a. Olshausen and D. J. Field, "Sparse coding with an incomplete basis set: a strategy employed by V1?" pp. 3311–3325, 1997.

[39] M. Turk and A. Pentland, "Face recognition using eigenfaces," *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 586–591, 1991.

[40] E. Wachsmuth, M. W. Oram, and D. I. Perrett, "Recognition of objects and their component parts: Responses of single units in the temporal cortex of the macaque," *Cerebral Cortex*, vol. 4, no. 5, pp. 509–522, 1994.

[41] A. Coates, A. Karpathy, and A. Y. Ng, "Emergence of Object-Selective Features in Unsupervised Feature Learning," *Neural Information Processing Systems*, pp. 1–9, 2012.

[42] C. Ding, "K-means clustering via principal component analysis," in *Proceedings of the twenty-first international conference on machine learning*, no. 2000. ACM, 2004, p. 29.

[43] D. Gabor, "Theory of Communication," *IEEE Radio and Communications Engineering*, vol. 93, no. 26, pp. 429–457, 1946.

[44] P. Smaragdis, "Redundancy reduction for computational audition, a unifying approach," Doctoral Dissertation, Massachusetts Institute of Technology, 2001.

[45] T. C. Andringa, "Continuity Preserving Signal Processing," Doctoral Dissertation, University of Groningen, 2002.

[46] A. J. Bell and T. J. Sejnowski, "Learning the higher-order structure of a natural sound," *Network: Computation in Neural Systems*, vol. 7, no. September 1995, pp. 261–266, 1996.

[47] N. Ahmed, T. Natarajan, and K. Rao, "Discrete Cosine Transform," *Computers, IEEE Transactions on*, vol. C-23, no. 1, pp. 90–93, 1974.

[48] V. Sanchez, P. Garcia, A. M. Peinado, J. C. Segura, and A. J. Rubio, "Diagonalizing Propexties of the Discrete Cosine Transforms," vol. 43, no. 11, 1995.

[49] M. S. Lewicki, "Efficient coding of natural sounds." *Nature neuroscience*, vol. 5, no. 4, pp. 356–63, 2002.

[50] H. Lee, P. Pham, Y. Largman, and A. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," *Advances in nerual infromation processing systems*, pp. 1096–1104, 2009.

[51] N. A. Bernstein, "The co-ordination and regulation of movements," pp. 23–25, 1967.

[52] J. Knierim, "Motor Units and Muscle Receptors," 1997. [Online]. Available: http://nba.uth.tmc.edu/neuroscience/s3/chapter01.html Accessed Nov. 1, 2016.

[53] T. Flash and B. Hochner, "Motor primitives in vertebrates and invertebrates," *Current Opinion in Neurobiology*, vol. 15, no. 6, pp. 660–666, 2005.

[54] M. C. Tresch and A. Jarc, "The case for and against muscle synergies," *Current Opinion in Neurobiology*, vol. 19, no. 6, pp. 601–607, 2009.

[55] A. D'Avella, "Muscle Synergies," pp. 2509–2512, 2009.

[56] E. Bizzi, F. A. Mussa-Ivaldi, and S. F. Giszter, "Computations underlying the execution of movement: a biological perspective." *Science (New York, N.Y.)*, vol. 253, no. 5017, pp. 287–91, 1991.

[57] S. F. Giszter, F. a. Mussa-Ivaldi, and E. Bizzi, "Convergent force fields organized in the frog's spinal cord." *The Journal of Neuroscience*, vol. 13, no. 2, pp. 467–491, 1993.

[58] F. A. Mussa-Ivaldi, S. F. Giszter, and E. Bizzi, "Motor-space coding in the central nervous system." *Cold Spring Harbor symposia on quantitative biology*, vol. 55, pp. 827–835, 1990.

[59] E. Bizzi, S. F. S. Giszter, E. Loeb, F. a. F. Mussa-ivaldi, and P. Saltiel, "Modular organziation of motor behavior in the frog's spinal cord," *Trends in Neuroscience*, vol. 18, pp. 442–446, 1995.

[60] T. G. Brown, "On the nature of the fundamental activity of the nervous centres; together with an analysis of the conditioning of rhythmic activity in progression, and a theory of the evolution of function in the nervous system," *The Journal of Physiology*, vol. 48, no. 1, pp. 18–46, mar 1914.

[61] E. Marder and D. Bucher, "Central pattern generators and the control of rhythmic movements," *Current Biology*, vol. 11, no. 23, pp. 986–996, 2001.

[62] D. M. Wilson, "The central nervous control of flight in a locust," *Journal of Experimental Biology*, vol. 38, no. 2, pp. 471–490, 1961.

[63] D. M. Wilson, "Central nervous mechanisms for the generation of rhythmic behaviour in arthropods." *Symposia of the Society for Experimental Biology*, vol. 20, pp. 199–228, 1966.

[64] D. M. Wilson and R. J. Wyman, "Motor Output Patterns during Random and Rhythmic Stimulation of Locust Thoracic Ganglia," *Biophysical Journal*, vol. 5, no. 2, pp. 121–143, 1965.

[65] A. E. Patla, "Some Characteristics of EMG Patterns During Locomotion," *Journal of Motor Behavior*, vol. 17, no. 4, pp. 443–461, 1985.

[66] M. C. Tresch, P. Saltiel, and E. Bizzi, "The construction of movement by the spinal cord." *Nature neuroscience*, vol. 2, no. 2, pp. 162–167, 1999.

[67] A. D'Avella and E. Bizzi, "Shared and specific muscle synergies in natural motor behaviors." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 8, pp. 3076–3081, 2005.

[68] F. L. Moro, N. G. Tsagarakis, and D. G. Caldwell, "On the kinematic motion primitives (kMPs) - theory and application," *Frontiers in Neurorobotics*, vol. 6, no. OCT, pp. 1–18, 2012.

[69] R. Urtasun, P. Glardon, R. Boulic, D. Thalmann, and P. Fua, "Style-based motion synthesis," *Computer Graphics Forum*, vol. 23, no. 4, pp. 799–812, 2004.

[70] J. Woo, F. Xing, J. Lee, M. Stone, and J. L. Prince, "Determining functional units of tongue motion via graph-regularized sparse non-negative matrix factorization," *Med Image Comput Comput Assist Interv.*, vol. 8674 LNCS, no. PART 2, pp. 146–153, 2014.

[71] E. Rückert and A. D'Avella, "Learned parametrized dynamic movement primitives with shared synergies for controlling robotic and musculoskeletal systems," *Frontiers in computational neuroscience*, vol. 7, no. October, p. 138, 2013.

[72] S. Schaal, "Dynamic movement primitives—A framework for motor con- trol in humans and humanoid robots," *Proc. Int. Symp. Adaptive Motion Animals*, no. 1, 2003.

[73] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors." *Neural computation*, vol. 25, no. 2, pp. 328–73, 2013.

[74] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and Generalization of Motor Skills by Learning from Demonstration," *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, pp. 1293–1298, 2009.

[75] K. Muelling, J. Kober, and J. Peters, "Learning Table Tennis with a Mixture of Motor Primitives," in *International Conference on Humanoid Robots*, 2010, pp. 411–416.

[76] A. Paraschos, C. Daniel, J. Peters, and G. Neumann, "Probabilistic Movement Primitives," *Neural Information Processing Systems*, pp. 1–9, 2013.

[77] A. Feldman, "Functional tuning of the nervous system with control of movement or maintenance of a steady posture. II. Controllable parameters of the muscle," pp. 565–578, 1966.

[78] R. Shadmehr, "The Equilibrium Point Hypothesis for Control of Movements," *Baltimore, MD: Department of Biomedical Engineering, Johns Hopkins University*, 1998.

[79] J. H. Challis, "Lecture 5 - Equaitons of Motion II: Basics," 2010. [Online]. Available: http://www.personal.psu.edu/jhc10/KINES574/Lecture5.pdf Accessed Nov 1, 2016.

[80] R. Shadmehr and M. A. Arbib, "A mathematical analysis of the force-stiffness characteristics of muscles in control of a single joint system," *Biological Cybernetics*, vol. 66, no. 6, pp. 463–477, 1992.

[81] T. Flash, "The control of hand equilibrium trajectories in multi-joint arm movements," *Biological Cybernetics*, vol. 57, no. 4-5, pp. 257–274, 1987.

[82] H. Gomi and M. Kawato, "Equilibrium-point control hypothesis examined by measured arm stiffness during multijoint movement." *Science (New York, N.Y.)*, vol. 272, no. 5258, pp. 117–120, 1996.

[83] P. L. Gribble, D. J. Ostry, V. Sanguineti, and R. Laboissiere, "Are complex are signals required for human arm movement?" *Journal of neurophysiology*, vol. 79, no. 3, pp. 1409–1424, 1998.

[84] F. A. Mussa-Ivaldi, "Motor Primitives, Force-Fields and the Equilibrium Point Theory," *From Basic Motor Control to Functional Recovery*, no. 1, pp. 392–398, 1999.

[85] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 1, pp. 43–53, 1987.

[86] S. Schaal and N. Schweighofer, "Computational motor control in humans and robots," *Current Opinion in Neurobiology*, vol. 15, no. 6, pp. 675–682, 2005.

[87] Spong, "Robot dynamics and control," *Automatica*, vol. 28, no. 3, pp. 655–656, 1992.

[88] J. P. Scholz and G. Schöner, "The uncontrolled manifold concept: Identifying control variables for a functional task," *Experimental Brain Research*, vol. 126, no. 3, pp. 289–306, 1999.

[89] M. M. L. Latash, M. M. F. Levin, J. P. Scholz, and G. Schöner, "Motor Control Theories and Their Applications," *Medicina (Kaunas, ...*, vol. 46, no. 6, pp. 382–92, 2010.

[90] D. A. Rosenbaum, "Human movement initiation: Specification of arm, direction, and extent." *Journal of Experimental Psychology*, vol. 109, no. 4, pp. 444–474, 1980.

[91] A. P. Georgopoulos, J. F. Kalaska, R. Caminiti, and J. T. Massey, "On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex." *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 2, no. 11, pp. 1527–1537, nov 1982.

[92] K. a. Thoroughman and R. Shadmehr, "Learning of action through adaptive combination of motor primitives." *Nature*, vol. 407, no. 6805, pp. 742–7, 2000.

[93] A. P. Georgopoulos, J. Ashe, N. Smyrnis, and M. Taira, "The Motor Cortex and the Coding of Force," *Science*, vol. 256, no. 5064, pp. 1692–1695, 1992.

[94] E. Todorov and M. I. Jordan, "Optimal feedback control as a theory of motor coordination," *Nat. Neurosci.*, vol. 5, no. 11, pp. 1226–1235, 2002.

[95] G. Schoner and J. A. Kelso, "Dynamic pattern generation in behavioral and neural systems," *Science*, vol. 239, no. 4847, pp. 1513–1520, 1988.

[96] Y. Tseng, J. P. Scholz, and G. Schöner, "Goal-equivalent joint coordination in pointing: affect of vision and arm dominance." *Motor control*, vol. 6, no. 2, pp. 183–207, 2002.

[97] J. P. Scholz, D. Reisman, and G. Schöner, "Effects of varying task constraints on solutions to joint coordination in a sit-to-stand task," *Experimental Brain Research*, vol. 141, no. 4, pp. 485–500, 2001.

[98] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational Space Control: A Theoretical and Empirical Comparison," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 737 –757, 2008.

[99] W. L. Nelson, "Physical principles for economies of skilled movements," *Biological Cybernetics*, vol. 46, no. 2, pp. 135–147, 1983.

[100] N. Hogan, "An organizing principle for a class of voluntary movements." *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 4, no. 11, pp. 2745–2754, 1984.

[101] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model." *The Journal of neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[102] D. M. Wolpert, "Computational approaches to motor control," *Trends in Cognitive Sciences*, vol. 1, no. 6, pp. 1–5, 1997.

[103] D. M. Wolpert, Z. Ghahramani, and M. I. Jordan, "Are arm trajectories planned in kinematic or dynamic coordinates? An adaptation study," *Experimental Brain Research*, vol. 103, no. 3, pp. 460–470, 1995.

[104] E. Todorov, "Optimality principles in sensorimotor control." *Nature neuroscience*, vol. 7, no. 9, pp. 907–15, 2004.

[105] D. M. Wolpert, Z. Ghahramani, and M. I. Jordan, "Perceptual distortion contributes to the curvature of human reaching movements," *Experimental Brain Research*, vol. 98, no. 1, pp. 153–156, 1994.

[106] J. R. Flanagan and a. K. Rao, "Trajectory adaptation to a nonlinear visuomotor transformation: evidence of motion planning in visually perceived space." *Journal of neurophysiology*, vol. 74, no. DECEMBER, pp. 2174–2178, 1995.

[107] R. Shadmehr and F. a. Mussa-Ivaldi, "Adaptive representation of dynamics during learning of a motor task," *The Journal of Neuroscience*, vol. 14, no. 5, pp. 3208–3224, 1994.

[108] J. R. Lackner and P. Dizio, "Rapid adaptation to Coriolis force perturbations of arm trajectory," *J Neurophysiol*, vol. 72, no. 1, pp. 299–313, 1994.

[109] P. N. Sabes and M. I. Jordan, "Obstacle avoidance and a perturbation sensitivity model for motor planning." *Journal of Neuroscience*, vol. 17, no. 18, pp. 7119–7128, 1997.

[110] L. Busnoniu, R. Babuska, B. D. Schutter, D. Ernst, L. Busoniu, R. Babuska, B. D. Schutter, and D. Ernst, "Reinforcement learning and dynamic programming using function approximators," p. 260, 2010.

[111] S. a. Chvatal, G. Torres-Oviedo, S. a. Safavynia, and L. H. Ting, "Common muscle synergies for control of center of mass and force in nonstepping and stepping postural behaviors." *Journal of neurophysiology*, vol. 106, no. 2, pp. 999–1015, 2011.

[112] C. Alessandro, I. Delis, F. Nori, S. Panzeri, and B. Berret, "Muscle synergies in neuroscience and robotics: from input-space to task-space perspectives," *Frontiers in computational neuroscience*, vol. 7, no. April, pp. 1–16, 2013.

[113] E. Todorov and Z. Ghahramani, "Unsupervised learning of sensory-motor primitives," *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No.03CH37439)*, vol. 2, pp. 1750–1753, 2003.

[114] P. Pastor, M. Kalakrishnan, L. Righetti, and S. Schaal, "Towards Associative Skill Memories," in *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*, 2012, pp. 309–315.

[115] J. Kober, B. Mohler, and J. Peters, "Learning perceptual coupling for motor primitives," *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 834–839, 2008.

[116] B. Gick and I. Stavness, "Modularizing speech," *Frontiers in Psychology*, vol. 4, no. 16, pp. 5985–5991, 2013.

[117] C. P. Browman and L. Goldstein, "Articulatory gestures as phonological units," *Phonology*, vol. 6, no. 2, pp. 201–251, 1989.

[118] E. Saltzman, H. Nam, J. Krivokapic, and L. Goldstein, "A task-dynamic toolkit for modeling the effects of prosodic structure on articulation," in *Proceedings of the 4th international conference on speech prosody*, 2008, pp. 175–184.

[119] V. Ramanarayanan, L. Goldstein, and S. S. Narayanan, "Spatio-temporal articulatory movement primitives during speech production: extraction, interpretation, and validation." *The Journal of the Acoustical Society of America*, vol. 134, no. 2, pp. 1378–94, 2013.

[120] F. H. Guenther, "Speech Sound Acquisition , Coarticulation , and Rate Effects in a Neural Network Model of Speech Production," *Psychological Review*, vol. 102, no. 3, pp. 594–621, 1995.

[121] J. A. Tourville and F. H. Guenther, "The DIVA model : A neural theory of speech acquisition and production," *Language and Cognitive Processes*, vol. 26, no. 7, pp. 952–981, 2011.

[122] S. Maeda, *Compensatory Articulation During Speech: Evidence from the Analysis and Synthesis of Vocal-Tract Shapes Using an Articulatory Model*. Dordrecht: Springer Netherlands, 1990, pp. 131–149.

[123] D. Plaut and C. Kello, "The emergence of phonology from the interplay of speech comprehension and production: A distributed connectionist approach," *The emergence of language*, no. Grant 9720348, pp. 1–25, 1999.

[124] A. S. Warlaumont, G. Westermann, E. H. Buder, and D. K. Oller, "Prespeech motor learning in a neural network using reinforcement," *Neural Networks*, vol. 38, pp. 64–75, 2012.

[125] P. Boersma, "Functional Phonology," pp. 1–493, 1998.

[126] S. Narayanan and S. Lee, "An approach to real-time magnetic resonance imaging for speech production," *The Journal of the Acoustical Society of America*, vol. 115, no. 4, pp. 1771–1776, 2004.

[127] T. Schultz and M. Wand, "Modeling coarticulation in EMG-based continuous speech recognition," *Speech Communication*, vol. 52, no. 4, pp. 341–353, 2010.

[128] P. Boersma, "An Articulatory Synthesizer for the Simulation of Consonants," *European Conference on Speech Communication and Technology*, vol. 3, no. September, pp. 1907–1910, 1993.

[129] P. Boersma, "Interaction between glottal and vocal-tract aerodynamics in a comprehensive model of the speech apparatus," *Proceedings of the XIIIth International Congress of Phonetic Sciences ICPhS'95*, pp. 430–433, 1995.

[130] J. L. Ishizaka, Kenzo and Flanagan, "Synthesis of voiced sounds from a two-mass model of the vocal cords," *Bell Labs Technical Journal*, vol. 51, no. 6, pp. 1233—-1268, 1972.

[131] P. Mermelstein, "Articulatory model for the study of speech production," *Journal of the Acoustical Society of America*, vol. 53, no. 4, pp. 1070—-1082, 1973.

[132] K. V. D. Doel, F. Vogt, R. E. English, and S. Fels, "Towards Articulatory Speech Synthesis with a Dynamic 3D Finite Element Tongue Model," in *Proceedings of the International Seminar on Speech Production*, 2006.

[133] J. Stock and M. Watson, "Dynamic factor models," *Oxford Handbook of Economic Forecasting*, no. July, pp. 1–43, 2010.

[134] E. J. Hannan and M. Deistler, *The statistical theory of linear systems*. SIAM, 2012.

[135] G. Kapetanios and M. Marcellino, "A parametric estimation method for dynamic factor models of large dimensions," *Journal of Time Series Analysis*, no. 5620, 2009.

[136] M. W. Watson and R. F. Engle, "Alternative algorithms for the estimation of dynamic factor, mimic and varying coefficient regression models," pp. 385–400, 1983.

[137] B. Jungbacker and S. J. Koopman, "Likelihood-based dynamic factor analysis for measurement and forecasting," *The Econometrics Journal*, vol. 18, pp. n/a–n/a, 2015.

[138] A. Zurr, R. J. Fryer, I. T. Jolliffe, R. Dekker, and J. J. Beukema, "Estimating common trends in multivariate time series using dynamic factor analysis," vol. 53, no. 9, pp. 1689–1699, 2013.

[139] C. Heaton, "Factor Analysis of High Dimensional Time Series," Unpublished Doctoral Dissertation, University of New South Wales, 2008.

[140] A. B. Poritz, "Linear predictive hidden markov models and the speech signal," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82.*, vol. 7, 1982, pp. 1291–1294.

[141] K. R. Fitz and S. A. Fulop, "A Unified Theory of Time-Frequency Reassignment," 2009, unpublished.