Vocal Tract Inversion by Cepstral Analysis-by-Synthesis using Chain Matrices

Sankaran Panchapagesan and Abeer Alwan

Department of Electrical Engineering
University of California, Los Angeles, U.S.A.
panchap@ee.ucla.edu, alwan@ee.ucla.edu

Abstract

Acoustic-to-articulatory inversion for vowels is performed by cepstral analysis-by-synthesis, using chain-matrix calculation of vocal tract (VT) acoustics and the Maeda articulatory model. The derivative of the VT chain matrix with respect to the area function was calculated in a novel efficient manner, and used in the BFGS quasi-Newton method for optimizing a distance measure between input and synthesized cepstral features over the entire articulatory trajectory. The optimization is initialized by a fast search of an articulatory codebook with a bin structure in formant space and the cost function also includes regularization and continuity terms to obtain realistic inverted VT shapes and smooth articulatory trajectories. Inversion is evaluated on the three diphthongs /ai/, /oi/ and /au/ of two speakers, one male and one female, from the University of Wisconsin X-ray microbeam (XRMB) database, and good agreement was achieved between inverted midsagittal vocal tract outlines and measured XRMB tongue and lip pellet positions, with an average relative error of less than 3% in the first three formants.

Index Terms: Acoustic-to-articulatory inversion, Analysis-by-Synthesis, chain matrix, Maeda articulatory model.

1. Introduction

Acoustic-to-articulatory inversion or vocal tract (VT) inversion is the problem of obtaining the VT shapes that produced a given speech signal. Potential benefits include the use of inverted articulatory parameters for efficient speech coding and improved speech recognition, and aided language learning. Data-driven inversion methods based on Artificial Neural Networks, Mixture Density Networks, or Kalman filters, have become popular in recent years [1]. Here we focus instead on analysis-by-synthesis methods where inversion is performed by adjusting the parameters of an articulatory synthesizer to match acoustic features computed from the input speech [2, 3, 5].

For vowels, the first three formant frequencies are important for the perception of vowel quality, and acoustic distance measures between natural and synthesized formants are often minimized [7, 8]. Among a set of spectral distance measures, a cepstral distance was found, in [3], to give best performance when inverted articulatory parameters were used for vowel recognition.

The two main difficulties in acoustic-to-articulatory inversion are the well-known non-unique nature of the mapping [6, 2] and local minima in the optimization. The non-uniqueness is usually resolved by including regularization and continuity terms in the optimization cost function and by using articulatory models to constrain the VT area function [7, 5]. The problem of local optima is addressed by using articulatory codebooks to provide good initial points for the optimization [2, 5]. Further optimization has been performed by methods such as direct search [5, 7], gradient based [3], and variational [8] methods.

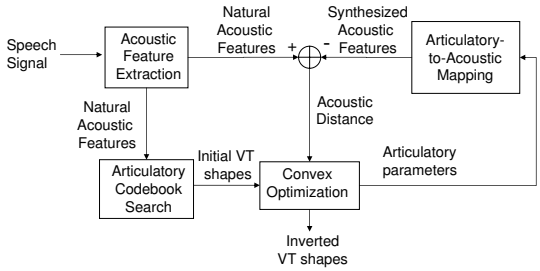

Figure 1: VT inversion using analysis-by-synthesis.

In this paper, we perform VT inversion for vowels by cepstral analysis-by-synthesis using chain-matrix calculation of VT acoustics [12] and the Maeda articulatory model [10]. We develop efficient methods for codebook search and subsequent optimization, and investigate the form of the cost function needed to balance the desired simultaneous goals of good match between input and synthesized acoustic features, realistic inverted VT configurations, and smooth articulatory trajectories. The inversion method is evaluated on two speakers from the X-Ray Microbeam Database [11], and speaker adaptation of the articulatory model is addressed.

In Sec. 2 we describe the different components of our inversion method including articulatory-to-acoustic mapping using chain matrices, the cost function to be minimized, and fast codebook search, and derive the gradient of the cost function for efficient optimization. We present inversion results in Sec. 3 and conclusions and directions for future work in Sec. 4.

2. The Inversion Method

2.1. VT Inversion by Analysis-by-Synthesis

Fig. 1 shows a block diagram of the different steps typically involved in VT inversion using analysis-by-synthesis.

For vowels, the first three formants would be good acoustic features for inversion since they are important for the perception of vowel quality. However, formant estimation can be difficult for high-pitched talkers, consonants and semi-vowels. We perform VT inversion by matching the truncated cepstrum, which is equivalent to matching the smoothed log spectral envelope. The first 20 cepstral coefficients were used, excluding the zeroth cepstral coefficient as it is only related to signal energy. De-emphasis of spectral tilt and formant bandwidths, log spectral weighting, and Mel frequency warping can all be captured in a linear weighting matrix on cepstra [4, 3]. By minimizing the cepstral distance measure, given in Sec. 2.3, formant peaks in the spectrum are effectively matched and explicit formant estimation is not necessary. Computationally too, DFT cepstra are more efficiently calculated and optimized than formants.

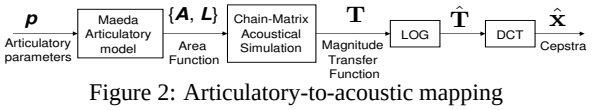

Figure 2: Articulatory-to-acoustic mapping

2.2. The Articulatory-to-Acoustic Mapping

Fig. 2 shows the block diagram of the articulatory-to-acoustic mapping using the Maeda articulatory model where the VT area function is described using seven parameters: jaw position, tongue body shape and position, tongue tip position, lip height and width, and larynx height [10].

In the chain matrix method for computing VT acoustics, the Laplace/Fourier transforms of pressure, P, and volume velocity, U, at the input and output of an acoustic tube, for a linear wave, are related by [12]:

$$\begin{pmatrix} P_{out} \\ U_{out} \end{pmatrix} = \begin{pmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{pmatrix} \begin{pmatrix} P_{in} \\ U_{in} \end{pmatrix} \tag{1}$$

where the matrix formed by $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ and $\mathcal{D}$ is called the chain matrix (CM), with a transfer function:

$$H(s) = \frac{U_{out}(s)}{U_{in}(s)} = \frac{1}{(\mathcal{A} - \mathcal{C}Z_L)} \tag{2}$$

where Z_L is the output radiation impedance. For a vowel sound, the VT is approximated as the concatenation of N uniform cylindrical tubes starting at the glottis and ending at the lips and the overall CM is:

$$K = K_N \cdot K_{N-1} \cdot \cdots \cdot K_1 \tag{3}$$

where K_n is the CM of the nth tube. For the Sondhi lossy VT model, the elements of K_n at a given angular frequency ω are given by ([12]):

$$\mathcal{A}_n = \cosh \frac{\sigma L_n}{c} \qquad \mathcal{B}_n = -\frac{\rho c \gamma}{A_n} \sinh \frac{\sigma L_n}{c} \tag{4}$$

$$\mathcal{C}_n = -\frac{A_n}{\rho c \gamma} \sinh \frac{\sigma L_n}{c} \qquad \mathcal{D}_n = \cosh \frac{\sigma L_n}{c} \tag{5}$$

where A_n (not to be confused with the CM parameter $\mathcal{A}_n$) is the area and L_n is the length of the nth tube, ρ is the density of air, c is the speed of sound in air, and γ and σ are only functions of ω and do not depend on A_n and L_n. Formulae for calculating γ and σ and other details are found in [12].

The magnitude of the VT transfer function is:

$$T(f) = |H(f)| \tag{6}$$

Let $\mathbf{T} = [T(f_0)\ T(f_1)\ \ldots\ T(f_{N_f})]^T$, where $f_i = i \cdot \frac{F_{max}}{N_f}$, $i = 0, 1, \ldots, N_f$ with $F_{max} = f_s/2$. The formants of the VT can be computed from the roots of the LP polynomial fitted to the $T(f_i)$. If we denote $\hat{\mathbf{T}} = \log(\mathbf{T})$, with elementwise logarithm of the vector $\mathbf{T}$, then the DFT cepstrum $\hat{\mathbf{x}}$ of $T(f)$ can be expressed as a DCT because of the symmetry of $T(f)$:

$$\hat{\mathbf{x}} = C \cdot \hat{\mathbf{T}} \tag{7}$$

where C is a DCT matrix.

Since the shape of the computed transfer function for vowels is very well fitted by an LP all-pole model, the synthesized LPC cepstrum is very well approximated by the above DFT cepstrum. The difference between log spectra described by liftered LPC and DFT cepstra was verified to be negligible for $f_s = 8000Hz$ and a DFT size as low as 64 ($N_f = 33$). Therefore, the synthesized DFT cepstrum may be used to match either DFT or LPC natural cepstra for vowels.

The most computationally intensive step in the synthesis is the calculation of the VT chain matrix using Equations 3 to 5 since there may be up to $N = 30$ sections in the area function, and $T(f)$ may be desired at $N_f = 30$ or more frequency points depending on the sampling rate and frequency resolution.

2.3. The Optimization Cost Function

The objective function to be minimized (E) is the sum of acoustic (E_{acou}), regularization (E_{reg}) and geometric continuity (E_{geo}) terms [7, 5, 8]:

$$E = E_{acou} + c_{reg} E_{reg} + c_{geo} E_{geo} \tag{8}$$

where c_{reg} and c_{geo} are weights. We use:

$$E_{acou} = \sum_{t=1}^{T} \left([\hat{\mathbf{x}}(t) - \hat{\mathbf{x}}_{in}(t)]^T W_{cep} [\hat{\mathbf{x}}(t) - \hat{\mathbf{x}}_{in}(t)] \right)^{\gamma} \tag{9}$$

$$E_{reg} = \sum_{t=1}^{T} [\mathbf{p}(t) - \mathbf{p}_0(t)]^T W_{par} [\mathbf{p}(t) - \mathbf{p}_0(t)] \tag{10}$$

$$E_{geo} = \sum_{t=1}^{T-1} ||\mathbf{p}(t+1) - \mathbf{p}(t)||^{2\eta} \tag{11}$$

where $\{\mathbf{p}(t),\ 1 \le t \le T\}$ is the articulatory vector sequence being optimized, $\{\hat{\mathbf{x}}_{in}(t),\ 1 \le t \le T\}$ and $\{\hat{\mathbf{x}}(t),\ 1 \le t \le T\}$ are the target and synthesized cepstral sequences, W_{cep} is the cepstral weighting matrix, γ and η are exponents, $\{\mathbf{p}_0(t),\ 1 \le t \le T\}$ is a "regularizing" sequence, and W_{par} is an articulatory parameter weighting matrix. As discussed in Sec. 2.1, W_{cep} incorporates the operations of liftering (we used the raised sine lifter, given in Eq. (3) of [4] for example), log-spectral emphasis between 250 Hz and 3000 Hz, and Mel warping.

The values of c_{reg}, c_{geo}, γ, $\mathbf{p}_0(t)$, W_{par} and η may be chosen to better achieve the competing goals of acoustic match, realistic VT shapes and smooth articulatory trajectories. Increasing values of γ and η result in lower maximum values across frames of the acoustic and geometric distances respectively. We used $\gamma = 3$ and $\eta = 2$. E_{reg} helps in eliminating unrealistic VT shapes during codebook search by discouraging VT configurations farther from the mean position (nominally $\mathbf{p}_0(t) = 0$ for the Maeda model), which are less likely to occur. In the subsequent optimization we use $\mathbf{p}_0(t) = \mathbf{p}_{init}(t)$, the initial sequence obtained from the codebook search, since we are more interested in improving the acoustic match close to the initial sequence. W_{par} and $\mathbf{p}_0(t)$ may be used to place constraints on the articulatory parameters, either based on phonetic considerations or to improve results for a specific speaker.

2.4. Codebook construction and fast search

We followed the method of codebook construction using log formant bins described in [4]. Using 2×10^6 random pairs of articulatory and acoustic vectors, a minimum log formant distance corresponding to 20% relative error, and an ∞-norm of 1 in articulatory space, we arrived at a codebook with around 82,000 vectors. The bin structure of the codebook in the formant domain can also be exploited for efficient search since it is equivalent to a (non-binary) tree organization in acoustic space. The cepstral centroids of the bins are used to first identify the bin containing an input cepstral vector $\hat{\mathbf{x}}_{in}(t)$, and the search at time t then continues only in it and neighbouring bins. Since the cepstral centroids were observed to retain formant peak information clearly, and there is some redundancy in articulatory space between adjacent bins in the codebook, further refinement of cepstral clusters was not considered necessary, and search results were satisfactory. Since the cost function includes E_{geo}, the codebook search involves dynamic programming (DP) [5]. For the DP search, we used two kinds of pruning. At each time t, from the identified bins for $\hat{\mathbf{x}}_{in}(t)$, only the best n_1 codevectors according to $E_{acou} + E_{reg}$ were considered for the DP iteration, and after the iteration, only n_2 codevectors were retained for the next iteration. Good search results were obtained even with $n_1 = 200$ and $n_2 = 20$, for a fraction of the original search time.

Further optimization is needed after codebook initialization to obtain both a better acoustic match with the input speech, and smoother articulatory trajectories, since there is a trade-off between the acoustic and articulatory resolutions and the size of the codebook.

2.5. Convex optimization of the cost function

We developed an efficient way of calculating the derivative of the CM of the VT with respect to the area function, since the computation of the VT CM is the most expensive step in synthesis as noted at the end of Sec. 2.2. This was then used in the BFGS quasi-Newton method to optimize the cost function [13]. The BFGS method has better (superlinear) asymptotic convergence than some other methods used in the past for optimization of area functions. The direct search methods of [5, 7] and the iteration in the variational approach of [8] which appears to be a type of fixed point method, have linear convergence. The BFGS method requires $\frac{\partial E}{\partial \mathbf{p}}$, the gradient of the cost function with respect to articulatory parameters (time dependence is ignored for the sake of clarity). $\frac{\partial E_{reg}}{\partial \mathbf{p}}$ and $\frac{\partial E_{geo}}{\partial \mathbf{p}}$ can easily be calculated from Equations 10 and 11. The functional dependencies in computing E_{acou} are: $\mathbf{p} \rightarrow \{\mathbf{A}, \mathbf{L}\} \rightarrow \mathbf{T} \rightarrow \hat{\mathbf{x}} \rightarrow E_{acou}$. $\frac{\partial E_{acou}}{\partial \mathbf{p}}$ can be computed by applying the chain rule. $\frac{\partial E_{acou}}{\partial \hat{\mathbf{x}}}$ and $\frac{\partial \hat{\mathbf{x}}}{\partial \mathbf{T}}$ are relatively straightforward to calculate from Equations 9 and 7, and $\frac{\partial \mathbf{A}}{\partial \mathbf{p}}$ and $\frac{\partial \mathbf{L}}{\partial \mathbf{p}}$ can be calculated from the equations of the Maeda model (where the notation $\frac{\partial \mathbf{x}}{\partial \mathbf{y}}$ is used to denote the matrix of partial derivatives $\left[\frac{\partial \mathbf{x}(i)}{\partial \mathbf{y}(j)}\right]$ when $\mathbf{x}$ and $\mathbf{y}$ are both vectors). We focus on the step $\{\mathbf{A}, \mathbf{L}\} \rightarrow \mathbf{T}$.

2.6. Chain matrix derivatives with respect to area function

By Equations 2 and 6, $\mathbf{T}$ depends on the CM parameters $\mathcal{A}$ and $\mathcal{C}$ of the VT and the radiation impedance Z_L. Therefore, to compute $\frac{\partial \mathbf{T}}{\partial \mathbf{A}}$ and $\frac{\partial \mathbf{T}}{\partial \mathbf{L}}$, we need to compute the derivatives of $\mathcal{A}$ and $\mathcal{C}$, which are given by Equations 3 to 5, with respect to $\{\mathbf{A}, \mathbf{L}\}$. The CM of each section depends only on its own area and length, and not on those of other sections. This simplifies the derivative calculation from Equation 3:

$$\frac{\partial K}{\partial A_n} = [K_N \cdots K_{n+1}] \cdot \frac{\partial K_n}{\partial A_n} \cdot [K_{n-1} \cdots K_1] \qquad (12)$$

If we define:

$$P_n = K_{n-1} K_{n-2} \cdots K_1, \;\; 2 \leq n \leq N \qquad (13)$$

$$Q_n = K_N K_{N-1} \cdots K_{n+1}, \;\; 1 \leq n \leq N-1 \qquad (14)$$

and let:

$$P_1 = Q_N = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad (15)$$

then:

$$\frac{\partial K(\mathbf{A})}{\partial A_n} = Q_n \cdot \frac{\partial K_n}{\partial A_n} \cdot P_n, \;\; 1 \leq n \leq N \qquad (16)$$

From Equations 4 and 5, we can show:

$$\frac{\partial \mathcal{A}_n}{\partial A_n} = 0 \qquad \frac{\partial \mathcal{B}_n}{\partial A_n} = -\frac{1}{A_n} \cdot \mathcal{B}_n \qquad (17)$$

$$\frac{\partial \mathcal{C}_n}{\partial A_n} = \frac{1}{A_n} \cdot \mathcal{C}_n \qquad \frac{\partial \mathcal{D}_n}{\partial A_n} = 0 \qquad (18)$$

Therefore, $\frac{\partial K_n}{\partial A_n}$ is very easily obtained from the elements of K_n. Similarly, many of the quantities involved in the calculation of $\frac{\partial K_n}{\partial L_n}$ are already available from the calculation of K_n.

Computational efficiency: A count of the real multiplications involved shows that computation of $\frac{\partial E_{acou}}{\partial \mathbf{p}}$ using the above analytical calculation of $\frac{\partial \mathbf{T}}{\partial \mathbf{A}}$ and $\frac{\partial \mathbf{T}}{\partial \mathbf{L}}$ is around 2.4 times as efficient as a finite-difference approximation for the 7 parameter Maeda model. If the analytical calculation was used in conjunction with an articulatory model with more parameters (like the Mermelstein model [9]), the advantage in efficiency would be higher.

3. Results

The inversion method was evaluated on two speakers, one female and one male, from the X-ray microbeam (XRMB) database [11]. For the female speaker ("JW46") the VT external outline (palate and rear pharyngeal wall) was similar in scale and shape to that of the Maeda model, and the model was used without any adaptation. For the male speaker ("JW11"), limited adaptation of the Maeda model was performed by overall scaling of the VT, and modifying the palate and pharyngeal wall outlines according to the measured outlines provided in the database. A speaker-specific codebook was therefore constructed. Detailed adaptation of the Maeda model would also involve separate scaling factors for the oral and pharyngeal regions and modifying the coefficients used to convert midsagittal widths to cross-sectional areas and the basis vectors used to compute the tongue outline from the jaw and tongue parameters [14, 10]. This would be a topic of future work.

We evaluated the inversion on three diphthongs: /ai/, /au/ and /oi/, taken from words of the form /sVd/ where V is the vowel, as contained in task #13 of the XRMB database. We downsampled speech signals to 8kHz, and computed 20 linear prediction cepstral coefficients (LPCCs) from 20ms frames with an LPC order of 10, for inversion. Frames were centered around times at which XRMB pellet positions were measured, with a frame rate around 146Hz. A lower frame rate would probably suffice and will be explored in the future. Synthetic cepstra were computed using the transfer function at 33 frequency points between 0 and 4kHz (included). Natural formants were manually extracted using the LPC analysis, for acoustic evaluation of inversion results.

The values of c_{reg}, c_{geo}, $\mathbf{p}_0(t)$, and W_{par} in the cost function were carefully chosen, as discussed in Sec. 2.3, to achieve a balance between the simultaneous goals of acoustic match (E_{acou}), realistic inverted VT shapes (E_{reg}) and smooth articulatory trajectories (E_{geo}). E_{geo} alone was found to be insufficient to resolve the non-uniqueness of the acoustic-to-articulatory mapping for /au/ and /oi/, where more than one articulatory trajectory was observed to correspond to the same trajectories for the first three formants. The codebook search often selected VT configurations with an elongated larynx and wide open mouth where lip-rounding was expected. A minimum value of c_{reg}, along with greater penalty weight (between 5 and 10) on the larynx height parameter relative to other parameters in W_{par} was found to be necessary to obtain alternate, more realistic articulatory trajectories with lip rounding, which are closer to the measured pellet positions. Adding a bias to the larynx height regularization parameter to shorten the larynx was also found to be necessary for JW11 and useful for JW46. Increasing the penalty on the jaw parameter (to 3) in W_{par} also improved results for JW11.

The strategy for improving acoustic match and smoothening articulatory trajectories after codebook search was as follows. The parameter trajectories obtained from codebook lookup are first smoothed using a short hamming window as this helps to reduce E_{geo} without affecting E_{acou} much. In the BFGS optimization, a smaller value of c_{reg} and a larger value of c_{geo} were used compared to their corresponding values in codebook search, and the articulatory parameter sequence after codebook search and smoothing was used for regularization as mentioned in Sec. 2.3. The BFGS iterations were stopped when the decrease in acoustic cost fell below a threshold (1%). The total inversion time for a speaker, with 3 diphthongs of around 30 frames each, was around 45-55 seconds in Matlab running on

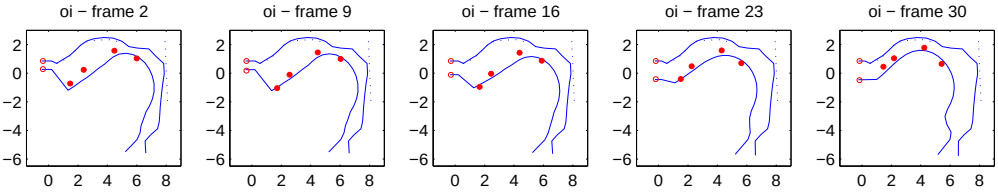

Figure 3: Speaker JW46, /oi/ from ‘soyed’ - Measured XRMB tongue (solid circles) and shifted lip (empty circles) pellet positions plotted against inverted VT outlines (solid lines). Measured palate and pharyngeal outlines are plotted using dotted lines.

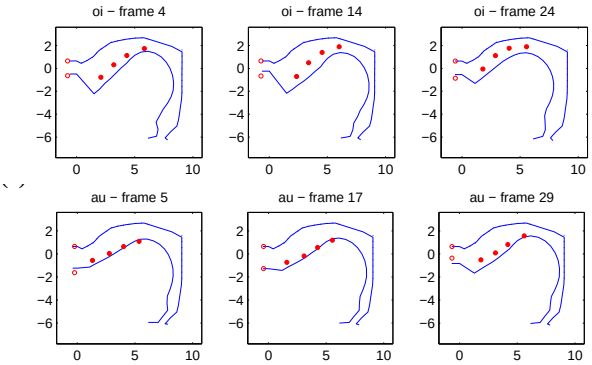

Figure 4: Speaker JW11, (a) /oi/ from ’soyed’ (b) /au/ from ‘saud’ - Measured XRMB tongue (solid circles) and shifted lip (empty circles) pellet positions plotted against inverted VT outlines (solid lines) for three frames.

an AMD Athlon 4.2GHz processor, with each codebook search taking around 1 second for $n_1 = 500$ and $n2 = 50$. After optimization, inverted articulatory trajectories varied smoothly as would be expected for a human talker, and the average relative errors in the first three formants for the three diphthongs were around 3% and 2% for JW46 and JW11 respectively.

Measured XRMB gold pellet positions are plotted against the VT outlines obtained from inversion and shown for five evenly spaced frames from /oi/ of JW46 in Fig. 3 and for three evenly spaced frames from /oi/ and /au/ of JW11 in Fig. 4. For the comparison, the model outlines were shifted so that the model and measured palate ends behind the teeth are aligned. Lip pellets were shifted vertically by the approximate height between them during a token of /m/ for the speaker, and horizontally averaged and shifted by an ad hoc speaker-specific distance. The match between inverted VT outlines and measured pellet positions is observed to be good for JW46. For JW11, although the acoustic match was excellent and the inverted VT outline approximately followed the curve of tongue pellet positions, the pellets were found to lie slightly away from the inverted outline for all three diphthongs. This could be due to lack of appropriate constraints in the regularization or inadequate adaptation of the Maeda model to the speaker, and needs to be investigated. Another direction for future work would be towards a mapping from XRMB pellet positions to articulatory parameters, so that better constraints could be easier learned and inverted articulatory parameter trajectories could be compared with actual ones.

4. Conclusions

VT inversion was performed for vowels by cepstral analysis-by-synthesis using chain matrices and the Maeda articulatory model. The contribution of this paper is in developing efficient techniques for the codebook search and convex optimization stages, using respectively, the bin structure of the codebook in log-formant space, and an analytical calculation of the derivative of the VT chain matrix with respect to the area function. The form of the optimization cost function was carefully chosen to obtain a balance between acoustic match, realistic VT shapes, and smooth articulatory trajectories. The inversion method was evaluated on diphthongs of two speakers from the X-ray microbeam database and limited adaptation of the Maeda model was performed for one speaker. A minimum weight on the regularization term, and related constraints on the articulatory parameters were found to be necessary to obtain realistic VT shapes for /au/ and /oi/. Good geometric match was achieved between inverted midsagittal VT outlines and measured XRMB tongue and lip pellet positions, with an average relative error of less than 3% between the first three synthesized and natural formants. Future work would be aimed at extending the inversion method to other speech sounds such as fricatives, and detailed adaptation of the articulatory model to different speakers.

5. Acknowledgments

We thank John Westbury for sharing the Microbeam database, which was supported (in part) by R01 DC 00820 from the NIDCD. This work was supported in part by the NSF.

6. References

[1] K. Richmond, “Estimating Articulatory Parameters from the Acoustic Speech Signal,” Ph.D. Thesis, U. Edinburgh, 2001.

[2] B. S. Atal et al., “Inversion of articulatory-to-acoustic transformation in the vocal tract by a computer-sorting technique,” *JASA*, Vol.63, No.5 , pp.1535-1555, 1978.

[3] K. Shirai and T. Kobayashi, “Estimating articulatory motion from speech wave,” *Speech Communication* 5, pp.159-170, 1986.

[4] J. Schroeter, P. Meyer, and S. Parthasarathy, “Evaluation of improved articulatory codebooks and codebook access distance measures,” *Proc. IEEE ICASSP 1990*, pp.393-396.

[5] J. Schroeter and M. M. Sondhi, “Techniques for estimating vocal tract shapes from the speech signal,”’ *IEEE Trans. SAP*, Vol.2, No.1, pp.133-150, Jan 1994.

[6] M. R. Schroeder, “Determination of the geometry of the human vocal tract by acoustic measurements,” *J. Acoust. Soc. Am. 41* (4, Part 2), pp.1002-1010, 1967.

[7] V. N. Sorokin, “Determination of vocal tract shape for vowels,” *Speech Communication* **11**, 1992, pp.71-85.

[8] S. Ouni and Y. Laprie, “Modeling the articulatory space using a hypercube codebook for acoustic-to-articulatory inversion,” *J. Acoust. Soc. Am.* **118** (1), pp.444-460, July 2005.

[9] P. Mermelstein, “Articulatory model for the study of speech production,” *JASA*, vol. 53, no. 4, pp.1070-1082, 1973.

[10] S. Maeda, “Compensatory articulation during speech: evidence from the analysis and synthesis of vocal tract shapes using an articulatory model,” in *Speech Production and Speech Modeling* (W. J. Hardcastle, A. Marchal, eds.), pp.131-149, Kluwer, 1990.

[11] J. R. Westbury, *X-ray Microbeam Speech Production Database User’s Handbook*, Version 1.0, June 1994.

[12] M. M. Sondhi and J. Schroeter, “A Hybrid Time-Frequency Domain Articulatory Speech Synthesizer,” *IEEE Trans. ASSP*, Vol. ASSP-35, No.7, pp.955-967, July 1987.

[13] M. S. Gockenbach, Online lectures on Numerical Optimization, Spring 2005: `http://www.math.mtu.edu/~msgocken/ma5630spring2005/lectures.html`

[14] B. Mathieu, Y. Laprie, “Adaptation of Maeda’s model for acoustic to articulatory inversion,” *Eurospeech 1997*, pp.2015-2018.