LA-UR 96-3518

RECEIVED
DEC 2 6 1996
OSTI

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

TITLE: **A MAXIMUM LIKELIHOOD APPROACH TO ESTIMATING ARTICULATOR POSITIONS FROM SPEECH ACOUSTICS**

AUTHOR(S): John Hogden

SUBMITTED TO:

External Distribution -Hard Copy

MASTER

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive royalty-free license to publish or reproduce the published form of this contribution or to allow others to do so, for U.S. Government purposes.

The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy.

Los Alamos

Los Alamos National Laboratory
Los Alamos New Mexico 87545

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

A maximum likelihood approach to estimating articulator positions from speech acoustics

John Hogden
CIC-3, MS B265
Los Alamos National Laboratory
Los Alamos, NM 87545

Running head: Estimating articulator positions

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

A. Specific Aims

This proposal presents an algorithm called maximum likelihood continuity mapping (MALCOM) which recovers the positions of the tongue, jaw, lips, and other speech articulators from measurements of the sound-pressure waveform of speech. MALCOM differs from other techniques for recovering articulator positions from speech in three critical respects: it does not require training on measured or modeled articulator positions, it does not rely on any particular model of sound propagation through the vocal tract, and it recovers a mapping from acoustics to articulator positions that is linearly, not topographically, related to the actual mapping from acoustics to articulation. The approach categorizes short-time windows of speech into a finite number of sound types, and assumes the probability of using any articulator position to produce a given sound type can be described by a parameterized probability density function. MALCOM then uses maximum likelihood estimation techniques to: 1) find the most likely smooth articulator path given a speech sample and a set of distribution functions (one distribution function for each sound type), and 2) change the parameters of the distribution functions to better account for the data. Using this technique improves the accuracy of articulator position estimates compared to continuity mapping -- the only other technique that learns the relationship between acoustics and articulation solely from acoustics.

B. Background

Research has demonstrated that, in some cases, speech acoustics (e.g. digitized speech samples) can be used to recover the positions of the speech articulators (e.g. the tongue & lips) (Hogden et al., submitted; Ladefoged, Harshman, Goldstein & Rice, 1978; Papcun et al., 1992). This is an important finding because techniques for recovering articulator positions from acoustics have several potential applications. For example, computer speech recognition is performed more accurately when the computer is provided with information about both articulator positions and acoustics, even when the articulator positions are estimated from speech (Zlokarnik, 1995). Furthermore, by providing real-time displays of articulator positions, it may be possible to help teach the hearing impaired to speak, to provide better foreign language instruction, and surprisingly, to help dyslexics learn to read. In addition, we may be able to use the relationship between articulator positions and acoustics to improve speech synthesis and speech coding.

The theory of linear prediction (Markel & Gray, 1976; Wakita & Gray, 1975) shows that, given certain strict (and at least partially inaccurate) assumptions about the characteristics of vocal tracts and the propagation of sound through acoustics tubes, we can derive equations that allow us to recover the shape of the vocal tract from speech acoustics for some speech sounds. However, not only is linear prediction theoretically incapable of recovering vocal tract shapes for many common speech sounds (e.g. nasals & fricatives), but when the assumptions underlying linear prediction are relaxed to make more realistic models of speech production, the relationship between acoustics and articulation becomes mathematically intractable.

Because a simple mathematical formula going from acoustics to articulation has not been found, some even argue that such a formula can not exist (Schroeter & Sondhi, 1994; Sondhi, 1979), most techniques for recovering the articulator positions require that we first learn the mapping from acoustics to articulation from a data set consisting of simultaneously collected measurements of articulator positions and speech sounds (Hogden et al., submitted; Ladefoged et al., 1978; Papcun et al., 1992). This approach also has problems. While it is easy to collect recordings of speech, it is very difficult to obtain measurements of articulator positions while simultaneously recording speech. In fact, with

the current technology, it is impossible to measure some potentially important information about articulator positions (e.g. the three dimensional shape of the tongue) while also recording speech sounds. This has lead some researchers to use articulatory synthesizers to create speech sounds, and then learn the mapping from the synthesized speech to the articulatory model parameters (Atal, Chang, Mathews & Tukey, 1978; Boe, Perrier & Bailly, 1992; Rahim, Goodyear, Kleijn, Schroeter & Sondhi, 1993; Rahim, Kleijn, Schroeter & Goodyear, 1991; Schroeter & Sondhi, 1994; Stevens & House, 1955). However, currently available articulatory synthesizers make many simplifying assumptions that can lead to marked differences between synthesized and actual speech, and also call into question the accuracy of the acoustic/articulatory mapping derived from articulatory models.

One technique for recovering articulator positions from speech sounds, continuity mapping (Hogden, 1991; Hogden, Rubin & Saltzman, in press; Hogden, Saltzman & Rubin, 1993), does not require articulator position measurements. Continuity mapping (CM) finds a mapping from acoustics to articulation using only the information available in recordings of speech -- eliminating the need to collect articulator position data. Unfortunately, continuity mapping only recovers topologically accurate information about the articulator positions, i.e. in comparing two acoustic segments, we can determine which acoustic segment was created with the tongue further forward, but we can not determine how much further forward.

This proposal describes an improvement on the continuity mapping technique that allows us to more accurately determine the relationship between articulation and acoustics without ever having to measure articulator positions. This new technique is called maximum likelihood continuity mapping (MALCOM) because it combines maximum likelihood estimation techniques with continuity mapping. Because a continuity map will typically be used as the starting point for the MALCOM algorithm, the following description of MALCOM starts by describing continuity mapping. Next, the maximum likelihood approach to improving the continuity map is described. Finally continuity mapping and MALCOM will be quantitatively compared to demonstrate that MALCOM recovers articulator positions more accurately than continuity mapping.

C. Creating Continuity Maps

The essential steps of the CM algorithm are 1) categorize short-time windows of speech into a finite number of categories -- replacing the speech signal with a sequence of codes which give the categories associated with the successive windows of speech; 2) estimate the temporal distances between the codes in the quantized speech; 3) position the codes in a spatial representation so that the distances between the codes in the spatial representation are monotonically related to the temporal distances between the codes in the quantized speech. The purpose behind these steps has been described in the previously referenced papers so will only be briefly reviewed here.

As stated above, the first step of the continuity mapping algorithm is to replace the speech signal with a sequence of codes representing the signal. This step is illustrated in Figure 1, which shows each of three overlapping windows of speech being replaced by a single number (called a code). The codes tell which sound category each speech window belongs to. Such a transformation can be performed using any of a number of algorithms. For example, the space of possible acoustic windows could be evenly divided and each section of the space could be given a corresponding code. Typically, some sort of spectral processing (e.g. the cepstrum analysis described in section F1 of this paper) is performed on each window of speech before the sounds are categorized, and a more efficient

categorization technique, like the frequency-sensitive vector quantization algorithm described in section F2 below, is used.

Although it is important to keep in mind that the encoding of the speech signal is done purely on the basis of acoustics -- no articulatory measurements are used -- the continuity mapping algorithm can be explained more easily by noting that *if* we had access to articulator information, we could make a map like Figure 2. The axes of Figure 2 correspond to (computer modeled) positions of the tongue body, so each position in the figure represents a position of the tongue. Figure 2 is divided into a set of numbered regions (called isocode regions), in which all the tongue positions in the region labeled 1 produce sounds that get encoded as type 1, all the tongue positions in the region labeled 2 produce sounds that get encoded as type 2, etc.

Notice that Figure 2 is a type of look-up table. For example, if we knew that a sound of type 1 was being produced, we could conclude that the sound was being produced using one of the tongue positions in region 1. Without further information, our best guess at the tongue position that created a sound of type 1 is the centroid of region 1. Thus, from the information available in Figure 2 we could make a simple table listing each sound type and the corresponding best guess of the tongue position -- providing a simple way to estimate articulator positions from acoustics.

Of course, Figure 2 was constructed using both articulatory and acoustic measurements and continuity mapping does not use articulatory measurements. Instead, continuity mapping makes use of the fact that the distances between the regions in Figure 2 can be determined from acoustics alone. For example, it is possible to determine that region 1 is closer to region 9 than it is to region 19 even without articulatory measurements. We can draw this conclusion from the encoded acoustics by noticing that code 1 is frequently seen right before or after code 9 but is never seen right before or after code 19. The reason for this is simple: in order for the tongue to move from region 1 to region 19, it has to move through intermediate regions, but there are no regions between region 1 and region 9. Since code 1 and code 9 are often adjacent in the encoded speech, we can conclude that we only need to cross one isocode region boundary to get from isocode region 1 to isocode region 9. In fact, we can calculate the average number of isocode region boundaries we need to cross to get between any to isocode regions by looking only at the sequence of codes in the encoded speech acoustics. Thus, we get information about the positions of the regions in Figure 2 from acoustics alone. Based on this reasoning, the second step of continuity mapping is to make the measurement of the distance between the regions based on the encoded acoustics signals found in step 1.

In general, the relative positions (although not the correct rotation) of any set of points can be derived if we know the distances between all pairs of points. A well-known algorithm called multidimensional scaling (MDS) already exists for performing this analysis. In fact, using a nonmetric variant of multidimensional scaling, we only need ordinal level information about the interpoint distances, i.e. relative positions of a set of points can be recovered from distance measurements that have been transformed by any monotonic function (Kruskal, 1964a; Kruskal, 1964b; Shepard, 1980). Thus, even though the average distance between the region centroids is only approximately monotonically related to the distance between the centroids, we can recover the relative positions of the region centroids by using multidimensional scaling on the distances estimated in step 2.

Figure 3 shows a continuity map made from acoustics produced by allowing the tongue to move through the positions in Figure 2. To show the relationship between the continuity map and Figure 2, the continuity map has been rotated and reflected to maximize the

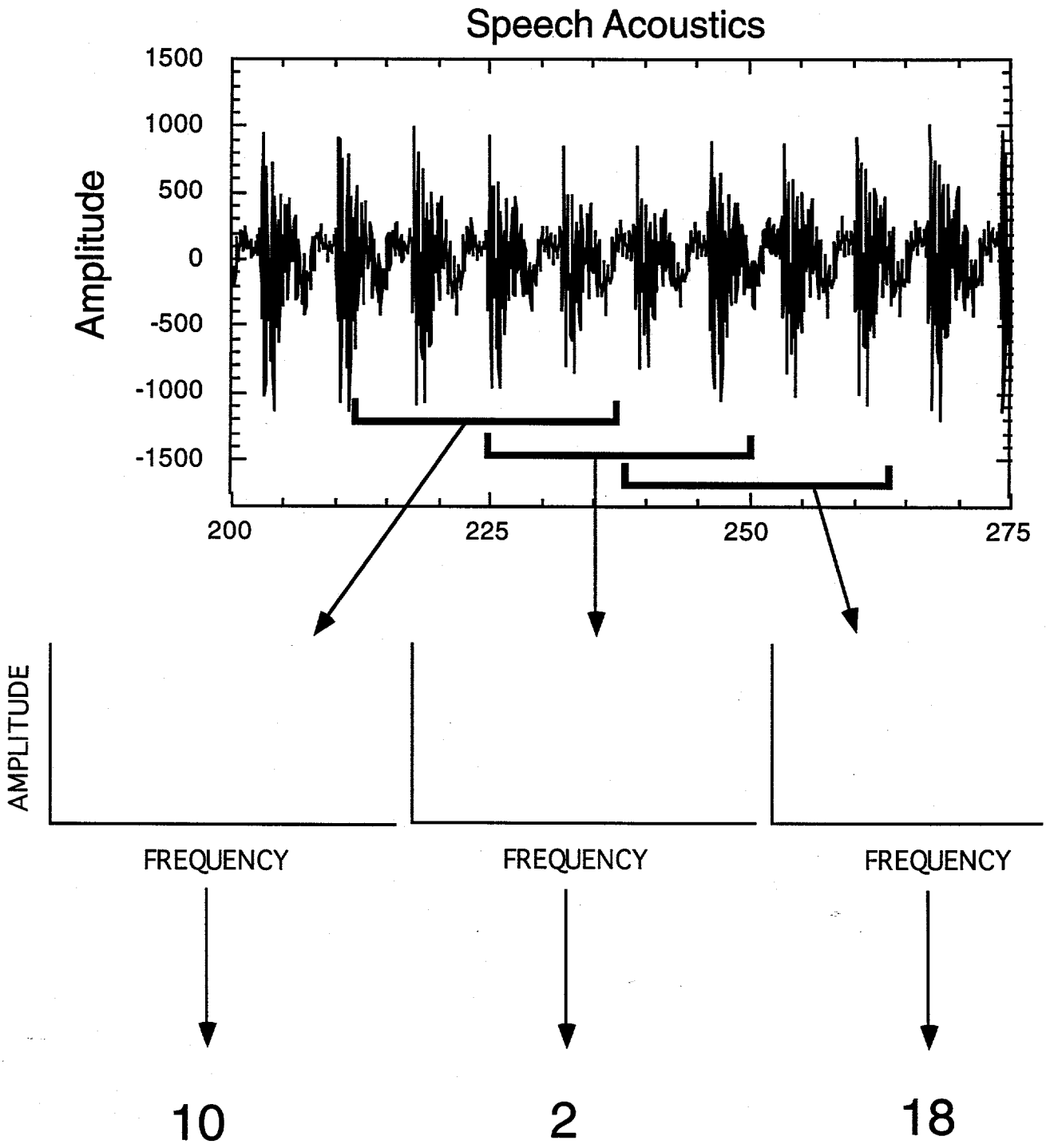
Speech Acoustics
Amplitude
1500
1000
500
0
-500
-1000
-1500
200
225
250
275
AMPLITUDE
FREQUENCY
FREQUENCY
FREQUENCY
10
2
18

Figure 1

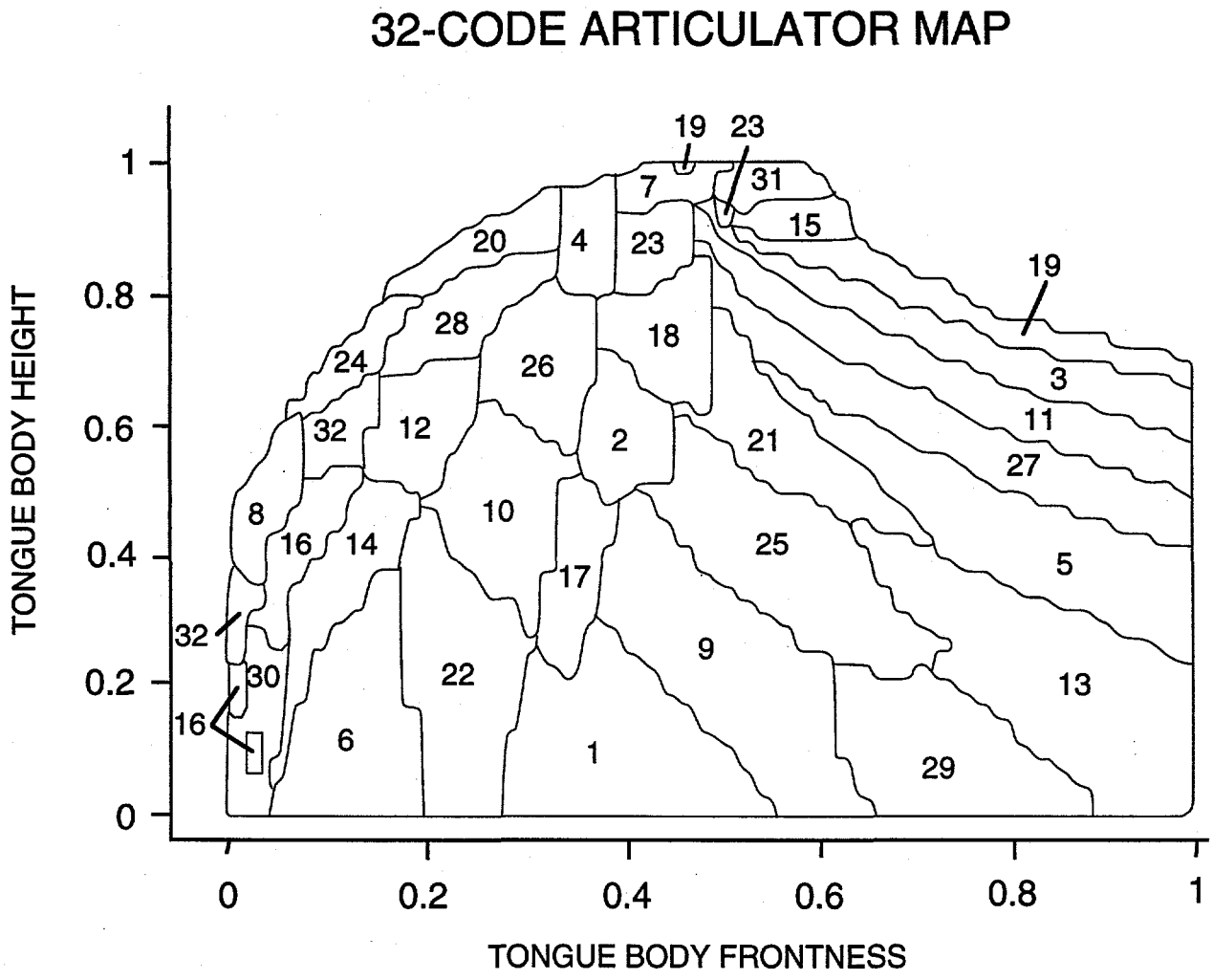
32-CODE ARTICULATOR MAP
TONGUE BODY HEIGHT
TONGUE BODY FRONTNESS
0
0.2
0.4
0.6
0.8
1
19
23
7
31
15
20
4
23
19
28
18
24
26
3
32
12
2
21
11
27
8
10
16
14
25
5
17
32
9
30
22
13
16
6
1
29

Figure 2

similarity between the continuity map and the positions of the centroids in Figure 2. The positions of the codes in Figure 3 are topologically related to the centroids of the regions in Figure 2 and so give information about the articulator positions, but the continuity map is non-uniformly scaled compared to the map shown in Figure 2. This non-uniform scaling is evident in the fact that, in the continuity map, the distance between codes 11 and 18 is almost the same as the distance between code 18 and 26. However, in the articulator space plotted in Figure 2, the distance between the centroids of regions 18 and 26 is much smaller than the distance between the centroids of regions 18 and 11. This type of distortion comes about because the number of region boundaries between region 18 and 11 is the same as the number of region boundaries between regions 18 and 26 (namely 1). So the continuity mapping algorithm, which estimates the distances by the number of region boundaries crossed, places 18 and 11 closer together than they should be.

D. Maximum Likelihood Continuity Mapping

Because of the distortion in the continuity map, if we use the positions of the codes in the continuity map to estimate the positions of the articulators from acoustics, the articulator trajectories will be unrealistic. Actual articulator trajectories move along smooth paths, with energy below 15 Hz or so (Muller & McLeod, 1982; Nelson, 1977). In contrast, the continuity map can make a small articulatory distance look like a large distance, and therefore gives articulatory trajectories that move faster than actual articulators can move.

We can minimize the effect of the continuity map distortion by requiring that the articulator paths estimated using the continuity map have all their energy below 15 Hz -- like actual articulator trajectories. One way to smooth the articulator trajectories is to simply use a low-pass filter. However, assuming that the articulator positions shown in Figure 2 are all used about equally often, we see that the variance of the articulator positions that produce sound type 15 is much smaller than the variance of the articulator position that produce sound type 9. Using a low-pass filter weights all of the articulator mean estimates equally, even though they vary in accuracy.

Instead of smoothing the paths by low-pass filtering, ideally we would require that the smooth path stays close to acoustically estimated mean articulator positions when the variance around the articulator mean is small, but can be further away from the mean position when the variance around the mean is large. Maximum likelihood estimation (Duda & Hart, 1973) gives us a way of implementing this requirement. To use the maximum likelihood approach, we will make the assumption that the articulator positions used to produce a sound of any given type are distributed in a parameterized probability density function. For the purposes of illustration, we will explicitly derive the procedure for multivariate Gaussian distributions parameterized by means and covariance matrices. However, it should be noted that relatively slight modifications would need to be made in order to use different classes of probability density functions, such as mixtures of Gaussians. Given the parameterized probability density functions, our goal will be twofold: 1) to estimate the parameters used to describe the probability density functions (the mean and covariance matrix associated with each sound type), and 2) using the mean and covariance estimates, find the most likely smooth articulatory trajectory given any sequence of sound types.

As will be shown, given any set of estimated means and covariances, we can improve the estimates of the means and covariance matrices by iteratively repeating three steps. The first step is to use the current estimates of the means and covariances to find the most likely smooth articulatory trajectories for a large set of speech samples. The second step is to

change the estimates of the means and covariances to maximize the probabilities of the paths estimated in step 1. The third step is to scale the solution to prevent a degenerate solution in which all the probability densitiy functions have the same mean. Thus, we can start by using the positions of codes in the continuity map as estimates of the means of the corresponding Gaussian distributions, set the initial estimates of the covariances to 1, and then iteratively improve these estimates

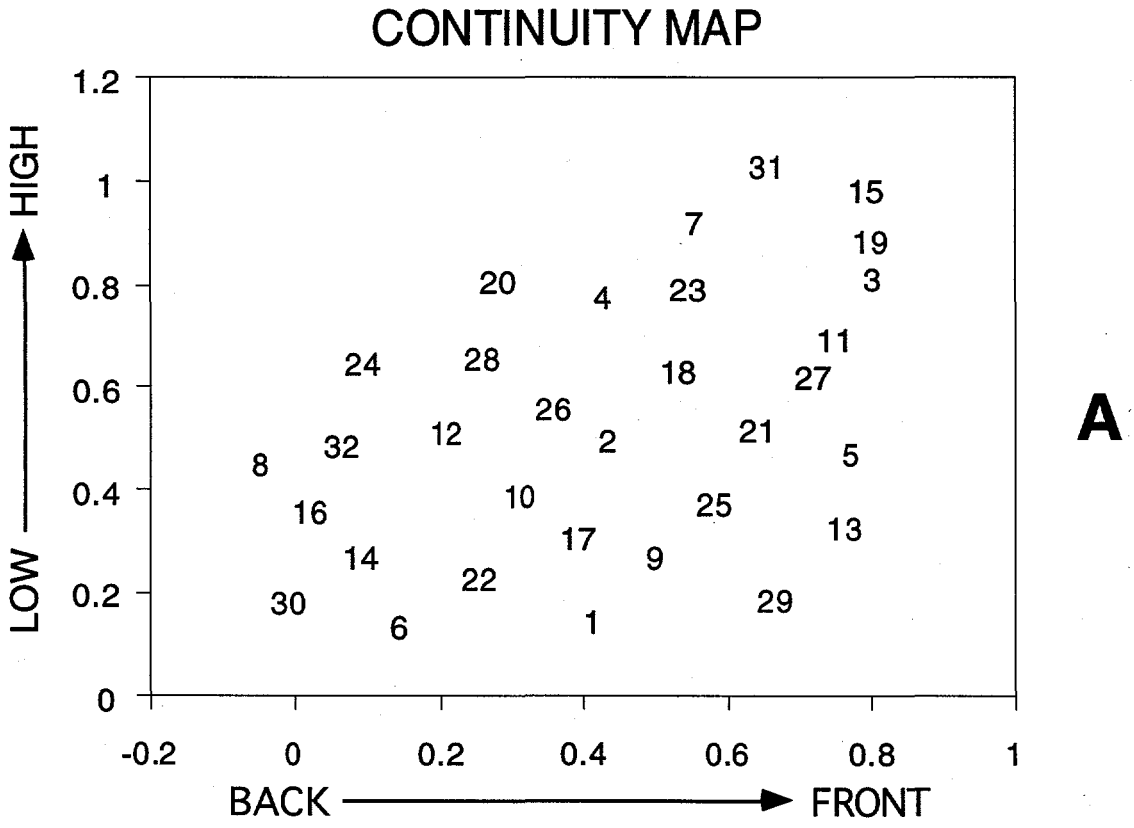

Figure 3

using the maximum likelihood approach described below. Since the technique for finding the most likely smooth articulatory trajectory is essential for estimating the means and covariances, we will start by describing the trajectory estimation technique, then describe how to improve the mean and covariance estimates.

D.1 Finding Most Likely Articulatory Trajectories.

Assume that the distribution of articulator positions that produce sounds quantized by code c is a multivariate Gaussian characterized by the equation:

$$P_c[\mathbf{x}] = \frac{1}{(2\pi)^{d/2}|\sigma(c)|^{1/2}} \exp\left\{-\frac{1}{2}[\mathbf{x}-\mu(c)]' \sigma^{-1}(c)[\mathbf{x}-\mu(c)]\right\}$$

Where:

d is the number of dimensions in the articulator space (i.e. the number of articulators).

$\mu(c)$ is a vector giving the mean of all the articulator positions used to produce sounds quantized with vector quantization code c. For example, $\mu_i(c)$ may give the mean lower lip position used to create sounds quantized as code c.

$\sigma(c)$ is the covariance matrix of the multivariate Gaussian distribution of articulator positions that produce sounds quantized with code c.

$\mathbf{x}$ is a vector describing an articulator position.

Assuming that the articulator positions used at different times are independent, an assumption that will be relaxed by the smoothness constraints discussed below, the probability of a path through articulator space, $\mathbf{X} = [\mathbf{x}(0) \quad \mathbf{x}(1) \quad \ldots \quad \mathbf{x}(n)]$, given the observed quantized speech signal, $C = [c(0) \quad c(1) \quad \ldots \quad c(n)]$, is:

$$P[\mathbf{X}] = \prod_{t=0}^{n} P_{c(t)}[\mathbf{x}(t)]$$

If we know the mapping from acoustics to articulation (the $\mu(c)$ and $\sigma(c)$ parameters), then it is possible to find the most probable articulator path given a sequence of codes. To do this we find the $\mathbf{X}$ that maximizes $P[\mathbf{X}]$, or equivalently, that sets the gradient of $\log P[\mathbf{X}]$ to 0. Since

$$\log P[\mathbf{X}] = \log \prod_{t=0}^{n} P_{c(t)}[\mathbf{x}(t)] = \sum_{t=0}^{n} \log P_{c(t)}[\mathbf{x}(t)]$$

we write:

$$\nabla \log P[\mathbf{X}] = \nabla \sum_{t=0}^{n} \log P_{c(t)}[\mathbf{x}(t)] = \sum_{t=0}^{n} \nabla \log P_{c(t)}[\mathbf{x}(t)]$$

If we use $\mu[c(t)]$ and $\sigma[c(t)]$ to denote the Gaussian distribution parameters that correspond to the code observed at time *t*, then
The individual components of the summation are:

$$\nabla \log P_{c(t)}[\mathbf{x}(t)] = \nabla\left\{-\log\left(\frac{1}{(2\pi)^{d/2}\left|\sigma[c(t)]\right|^{1/2}}\right)+\left\{-\frac{1}{2}\{\mathbf{x}(t)-\mu[c(t)]\}^{t}\sigma^{-1}[c(t)]\{\mathbf{x}(t)-\mu[c(t)]\}\right\}\right\}$$

$$= \nabla\left\{-\frac{1}{2}\{\mathbf{x}(t)-\mu[c(t)]\}^{t}\sigma^{-1}[c(t)]\{\mathbf{x}(t)-\mu[c(t)]\}\right\}$$

$$= \sigma^{-1}[c(t)]\{\mu[c(t)]-\mathbf{x}(t)\}$$

Thus, the gradient of the probability associated with a path is:

$$\nabla \log P[\mathbf{X}] = \sum_{t=0}^{n}\sigma^{-1}[c(t)]\{\mu[c(t)]-\mathbf{x}(t)\}$$

from this it can be seen that the most likely path is:

$$[\mu[c(0)] \quad \mu[c(1)] \quad \ldots \quad \mu[c(n)]]$$

However, this path is discontinuous, not smooth like actual articulator trajectories. To get around this problem we consider only those paths that have all their energy below some cut-off frequency (say 15 Hz, since actual articulator paths have very little energy above 15 Hz).

The constraint that the path have all of its energy below the cut-off frequency is equivalent to requiring that the path lie on a hyperplane composed of the axes defined by low frequency sine and cosine waves. We know from the theory of constrained optimization (Marsden & Tromba, 1981) that the most probable smooth path is the path for which $\mathbf{X}$ lies on the hyperplane $\nabla \log P[\mathbf{X}]$ is perpendicular to the hyperplane. Thus, the most probable smooth path is the path for which $\nabla \log P[\mathbf{X}]$ has no components with energy below the cut-off frequency.

This can be understood geometrically by recognizing that setting $\log P[\mathbf{X}]$ = *constant* defines an ellipsoid, with $\nabla \log P[\mathbf{X}]$ normal to the ellipsoid. In figure below, the axes of the space are intended to be the cosine axes used to represent $\mathbf{X}$ in the Fourier domain, so each point in the figure represents a complete path through articulator space. For ease of exposition, assume that we want the articulator path to have energy at frequency f_1 but not at f_2. To find the most probable path having no energy at frequency f_2 we find the point on the hyperplane -- the $\cos(2\pi f_1 t)$ axis -- at which the projection of the gradient onto the $\cos(2\pi f_1 t)$ axis (and other allowable frequencies) is zero. When the projection of the gradient is 0, we can not increase the probability of the path without adding components at higher frequencies.

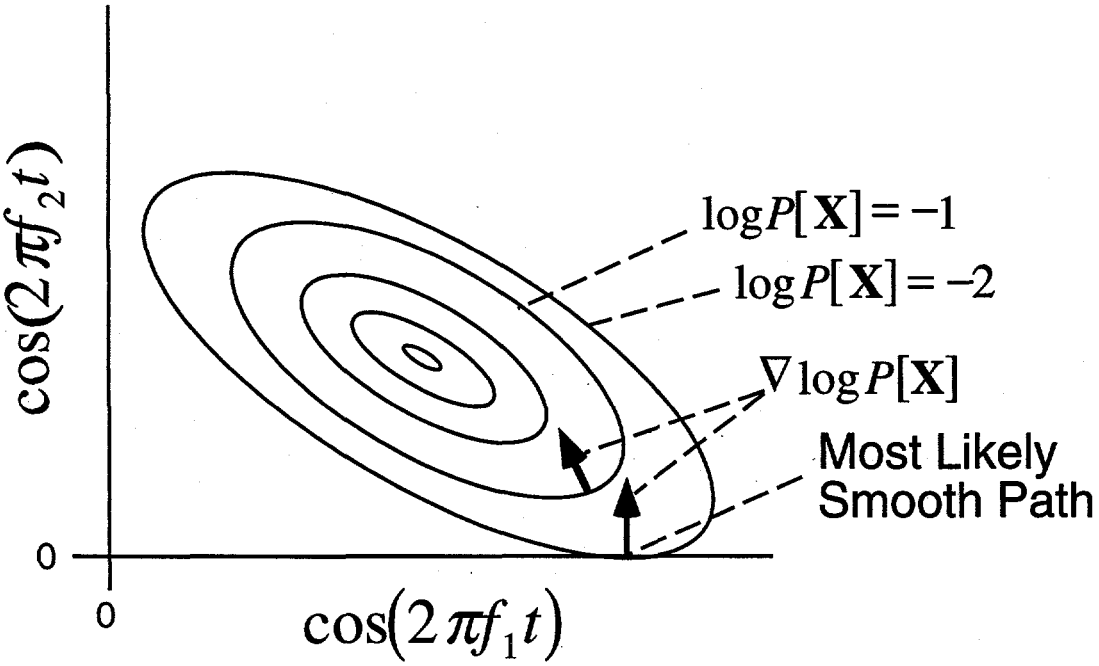

Figure 4

In fact, if we start at any point on the $\cos(2\pi f_1 t)$ axis and move in the direction of the gradient projected onto the $\cos(2\pi f_1 t)$ axis until the projected gradient is 0, we will eventually end up at the point representing the most probable smooth path.

This suggests the following algorithm for finding the most probable smooth path:

1) low-pass filter the path $[\mu[c(0)] \quad \mu[c(1)] \quad \ldots \quad \mu[c(n)]]$ to get an initial estimate of the most likely smooth path.

2) find the gradient of the log probability of the smooth path.

3) low-pass filtered the gradient.

4) add the low-pass filtered gradient times some small constant to the path to get a better estimate of the most likely smooth path.

5) repeat steps 2 - 4 until the algorithm converges.

There are also a variety of standard numerical algorithms that can be used to maximize functions. Using one of these algorithms can speed up the process of finding the most

likely smooth path. One of these techniques, the conjugate gradient algorithm, is used in the current implementation.

D.2 Improving the Mean and Covariance Estimates

Now that we have a technique for finding the probability of a path through articulator space, as well as a technique for finding the most likely smooth path, it is possible to estimate the model parameters, $\mu(c)$ and $\sigma(c)$.

Let n_u be the number of codes used to quantize the speech in utterance u.

Let N be the number of utterances

Let $C_u = \left[c_u(0) \quad c_u(1) \quad \ldots \quad c_u(n_u)\right]$ be the sequence of codes used to quantize the speech in utterance u.

Let $\mathbf{X}_u^{(i)} = \left[\mathbf{x}_u(0) \quad \mathbf{x}_u(1) \quad \ldots \quad \mathbf{x}_u(n_u)\right]$ be an estimate of the path through articulator space used to produce the sounds of utterance u. The i is used to indicate the path found by iteration i of the algorithm described below.

Let $\Phi^{(i)}$ be the set of $\mu^{(i)}(c)$ and $\sigma^{(i)}(c)$ parameters for all c. Once again, i is used to indicate how many iterations of the algorithm have been run.

Let $P\left[\mathbf{X}_u^{(i)} \middle| C_u, \Phi^{(i)}\right]$ be the probability of the smooth path $\mathbf{X}_u^{(i)}$ given C_u and $\Phi^{(i)}$.

The likelihood of all the paths is given by:

$$L(i) = \prod_{j=1}^{N} P\left[\mathbf{X}_j^{(i)} \middle| C_j, \Phi^{(i)}\right]$$

To get the best estimate of $\Phi^{(i)}$, we need to start with a reasonable estimate of $\Phi^{(i)}$ and then improve the estimate with the algorithm given below. A continuity map can be used to get a reasonable first estimate of the means, and identity matrices can be used as initial estimates of the covariance matrices.

$L(i)$ can be maximized by iteratively repeating three steps:

1) find $\mathbf{X}_u^{(i+1)}$ such that $P\left[\mathbf{X}_u^{(i+1)}\middle|C_u,\Phi^{(i)}\right]$ is maximized for all u.

Notice that the technique for doing this has already been described in section D.1 and that step 1 will increase $L(i)$ because each of the terms in the product will either increase or stay the same.

2) find $\Phi^{(i+1)}$ to maximize $\prod_{j=1}^{N} P\left[\mathbf{X}_j^{(i+1)}\middle|C_j,\Phi^{(i+1)}\right]$

Step 2 can be done simply by re-estimating the means and variances using the estimated smooth paths as the articulator data, as in:

$$\mu^{(i)}(k) = \sum_{\forall u,t:c_u(t)=k} \mathbf{x}_u^{(i)}(t) \Big/ K$$

where K is the total number of times code k is used to quantize speech sounds. This is the standard equation used to calculate means. The covariances can also be calculated using the standard equation.

3) to prevent the degenerate solution in which all the means are identical, set the variance of the means to 1.

E. Testing Materials

In order to determine how well MALCOM recovered articulator positions compared to CM, we recorded speech produced by a Swedish speaker at the same time the speaker's tongue, jaw, and lip positions were being measured. Although articulator measurements were made, the articulator measurements were not used for determining the mapping from acoustics to articulation. Both the CM and MALCOM techniques recover the mapping between acoustics and articulation from acoustics signals alone. The articulator measurements were only made to allow a comparison between recovered articulator positions and actual articulator positions.

E.1 Speech Samples

The speaker produced utterances containing two vowels spoken in a /g/ context with a continuous transition between the vowels, as in /guog/. The vowels in the utterances are all pairs of 9 Swedish vowels (/i/, /e/, /œ/, /a/, /o/, /u/, and the front rounded vowels /y/, /ˈ/, and /P/), as well as the English vowel /E/, for a total of 90 utterances (Fant, 1973). The data set includes 180 productions of /g/ and 18 productions of each vowel, since each vowel was produced before and after each of the other 9 vowels.

E.2 Acoustic Data

The speech sounds produced by the speaker were digitized using the Haskins Laboratories speech processing system (Whalen, Wiley, Rubin & Cooper, 1990). The speech was

sampled at 20 kHz with 12 bits/sample accuracy, after filtering out frequencies above 10 kHz and pre-emphasizing.

The boundaries of each token were found by examining the sound pressure versus time waveform. For the studies reported here, an effort was made to include as much of each token's acoustic signal as possible, even the very low amplitude portions of the acoustic signal corresponding to /g/ closure. The average token length is 833 ms.

E.3 Articulatory Data

Articulator positions were measured using a three-transmitter electromagnetic midsagittal articulometer (EMMA) like that described by Perkell et al. (Perkell et al., 1992). The EMMA system consists of three transmitter coils mounted on a plastic frame which is placed on the subjects head, and receiver coils that can be glued to the articulators. Each of the transmitters produces an alternating electromagnetic field but the frequency of oscillation is different for each coil. The positions of the coils can be inferred from the voltages induced in them by the transmitter coils, since the induced voltage varies with the distance between the transmitters and the receivers.

The voltage induced in a receiver coil is also function of the alignment of the receiver coil with respect to the electromagnetic fields produced by the transmitters, such that rotating the receiver coils can cause errors in the coil positions measurements. Because the tongue tilts during some articulations (Stone & Lele, 1992) the positions of the receiver coils glued to the tongue cannot be determined as accurately as those glued to the jaw and lips, which are less likely to tilt. Perkell et al. (1992) estimate that the receiver coils positions can be measured within about 0.5 mm for lip and jaw positions, and within about 1.0 mm for tongue placements.

Articulator position measurements were made 625 times per second. To decrease measurement noise, the measured articulatory trajectories were smoothed using a low-pass filter to remove frequencies above 20 Hz. Notice that a 20 Hz cut-off frequency is 5 Hz higher than needed to insure that the articulator motions are accurately measured, leaving room for error, but is also low enough to eliminate most random noise.

Receiver coils were placed on the tongue tip (TT), tongue dorsum (TD), tongue body (TB), tongue rear (TR), lower lip (LL), upper lip (UL), jaw (JA), upper incisors, and the bridge of the nose. The approximate placements of the receiver coils are displayed in Figure 5. The coils on the nose and upper incisors were used for correction of head movements. Two receivers attached to a plate were used to record the occlusal plane by having the subject bite down on the plate while recording. All data were subsequently corrected for head movements, and then rotated and translated to bring the occlusal plane into coincidence with the x axis. Fourteen parameters, the x and y positions of the receivers on the tongue, jaw, and lips, were used to describe each articulator configuration.

Notice that articulator motions with energies below 15 Hz can be completely described by specifying the articulator positions 30 times/second (Oppenheim, Willsky & Young, 1983). Therefore, by multiplying the total duration of the data set by 30 we determine that the receiver coil positions can be described by approximately 2,250 14-dimensional vectors, where each vector gives the x and y positions of each of the seven coils.

Approximate Placement of EMMA Receiver Coils

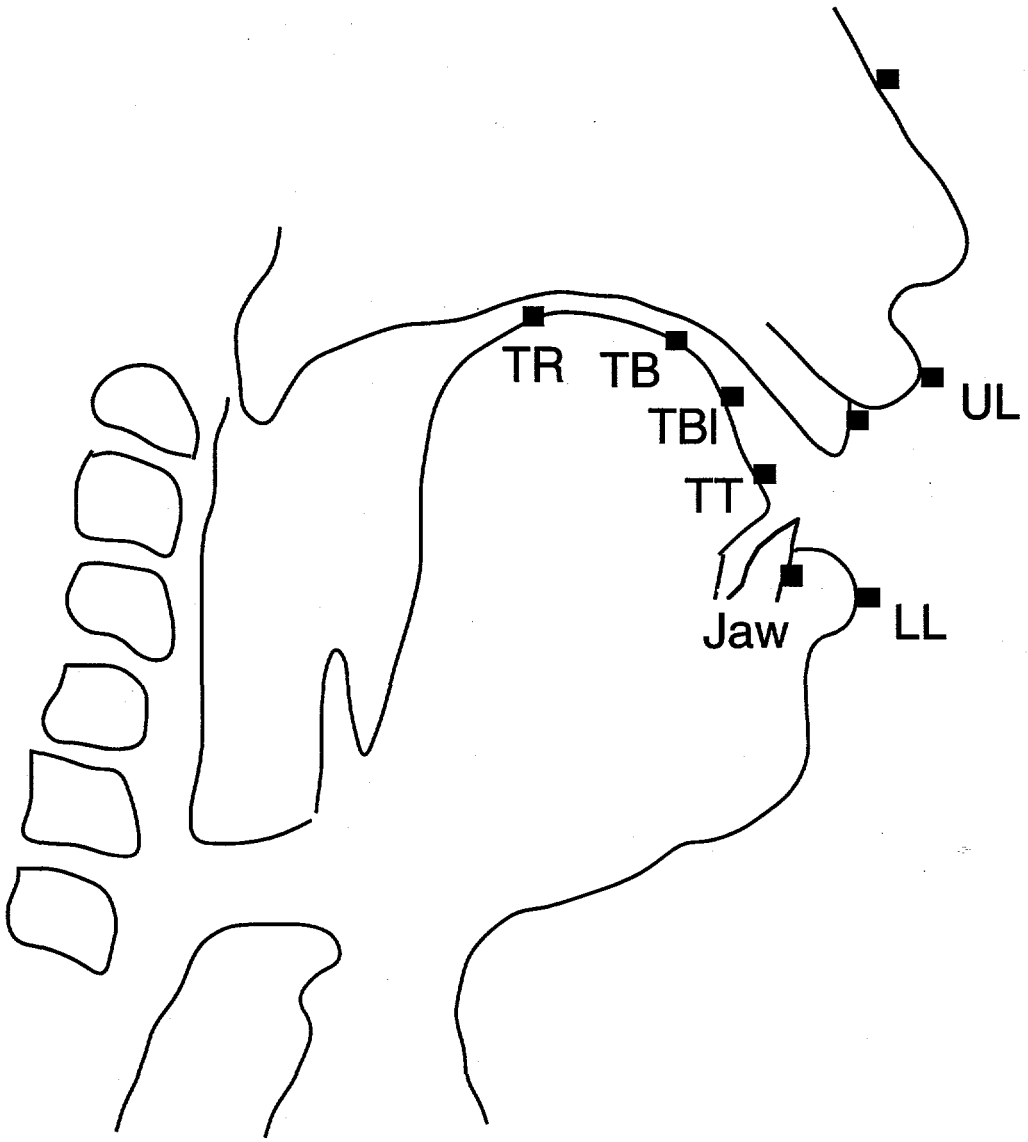

Figure 5

F. Continuity Mapping

Below, I describe the processing steps used to make a continuity map. The maps created using this technique are used to initialize MALCOM and to allow me to compare CM to MALCOM. In particular, I describe how the speech signals collected for the test were processed to facilitate good categorization of the speech windows. I also describe how the speech windows were categorized and how to estimate the distances between any two codes in an encoded speech signal. These techniques comprise the best know way to make an continuity map.

F.1 Acoustic Processing

Vocal tract transfer functions were estimated 625 times per second (one transfer function calculated at each time the articulator positions were measured) from 32 cepstrum coefficients of the corresponding 25.6 ms, Hamming windowed portion of the speech signal. Using cepstrum coefficients obtained by similar procedures to re-synthesize sounds results in "very high quality, natural sounding speech" (Oppenheim, 1969; Quartieri, 1979) -- suggesting that the cepstrum coefficients retain much of the information in the speech signal. To reduce the computational load, only frequencies below 5 kHz were used for further analysis. The result of the preprocessing was a sequence of smoothed spectral slices of the acoustic speech signal, with each slice represented by a vector composed of 128 energy measurements. The spectral slices were then normalized by setting the total energy of each slice to one.

F.2 Vector quantization of the acoustic signals

The spectral slices (hereafter called acoustic vectors) were categorized using vector quantization (VQ) (Linde, Buzo & Gray, 1980). The categorization is performed by finding the shortest Euclidean distance between the acoustic vectors and each of a small set of numbered *reference vectors* (a full set of numbered reference vectors is called a *codebook*). If an acoustic vector is found to be closest to reference vector 13, for example, it is said to belong to category 13. The number of the reference vector, "13" in this case, is often called a code, so a vector belonging to sound category 13 is quantized by replacing it with code 13. Equivalently, we say that the vector is *encoded* by code 13. We also use the word *decode* to mean that code 13 in an encoded speech sample is being replaced by reference vector 13.

A variation of the frequency-sensitive competitive learning (FSCL) algorithm (Ahalt, Krishnamurthy, Chen & Melton, 1990) was used to create VQ codebooks. In this variation, the reference vectors were initialized with small random numbers. After initialization, the reference vectors were moved to minimize the *distortion*, or error, that would be caused by replacing each data vector with the most similar reference vector. The reference vectors were moved to minimize distortion by iteratively repeating two steps. In the first step, each data vector is categorized by finding the reference vector which minimizes the value of

$$distortion = N_c \sum_i \left(d_i - r_{ci}\right)^2$$

where N_c is the number of times the code has already been used to represent data vectors, d_i is the i^{th} element of the data vector, and r_{ci} is the i^{th} element of reference vector c. The second step of the learning is to replace each reference vector, r_c, by the mean of all the data vectors (in the training set) that were encoded as c. For example, reference vector 1 would be replaced by the mean of all the data vectors that were quantized as code 1. The N_c factor provides a pressure for the codes to be used about equally often. This is because if a code has not been used many times, the distortion for that code will tend to be lower, making it more likely that the code will be used in the future. So, if during training, two codes are equally distant from a data point, then the code which has been used less often will be chosen to represent the data point. The N_c factor is only used during training, not when quantizing a new data set. As stated above, for quantizing a data set, the smallest Euclidean distance measure is used to determine which code will replace a segment of acoustics.

Based one a previous experiment, 256 codes were used to encode the speech data. Using this number of codes, the centroids of the isocode regions are good estimates of the actual articulator positions.

F.3 Estimating Distances Between Codes

The second step of the CM algorithm is to estimate the distances between pairs of codes from the vector quantized speech signal. As described above, essentially we are trying to estimate the number of isocode region boundaries that must be crossed when traveling between codes i and j. The method of counting the distances between pairs of codes described in this paper worked better than previously published methods for this data. To see how these distance measurements are calculated, notice that the articulatory synthesizer used to create Figure 2 could produce a signal which would be quantized as: 1, 9, 9, 9, 17, 22, 1, 9. From this sequence of codes I calculate the minimum number of times I see a change from one code to a different code between each pair of codes. In the example sequence, the minimum number of code transitions between code 1 and code 9 is 1 -- the only transition is the change from code 1 to code 9. Notice that, even though there are three codes between code 1 and code 17, there are only two transitions between code 1 and code 17 -- the transition from 1 to 9 and the transitions from 9 to 17. Notice also that the minimum number of transitions between code 1 and code 22 is one -- the fact that code 22 appears before the last example of code 1 is irrelevant. In addition to finding the minimum number of transitions between codes, we also calculate the number of times we observe that minimum number of transitions. So in the code sequence given above, the minimum number of transitions between code 1 and code 9 is one, and there are two times that we see the minimum number of transitions between code on and code 9 -- once at the beginning of the sequence and once at the end of the sequence.

Given code sequences corresponding to several utterances, we calculate the minimum distances between each code pair for each utterance separately, then combine the estimates using a weighted average. So if we found that, for an utterance, the minimum distance between code 1 and code 9 was 1, and also found that there were two times within the utterance when the minimum distance was observed between code 1 and code 9 (like in the example above), then we would weight the minimum distance between codes 1 and 9 by 2.

F.4 Multidimensional Scaling

The last step in the continuity mapping algorithm is to estimate the relative positions of the centroids of the isocode regions. I used nonmetric MDS on the distance estimates

calculated in the last section to get a continuity map in which the N vector quantization codes are placed in a low-dimensional space. The number of dimensions in the space can be varied; we used spaces with between 1 and six dimensions, inclusive. The distances between the VQ codes in this space are monotonically related to the number of isocode region boundaries that must be crossed to travel between the corresponding isocode regions.

F.5 Evaluating Continuity Mapping

The position of a code in a continuity map should provide information about the mean of the articulator positions that produce the code. However, the continuity map may be rotated, reflected, scaled, or topologically transformed in some other way compared to the actual mean articulator positions. This makes comparisons between continuity maps and mean articulator positions difficult (which is one reason for using MALCOM instead).

One way to determine whether the positions of codes in a continuity map supply information about the mean articulator positions is to see whether equations can be constructed relating code positions to mean positions. Because the continuity map is only topographically related the mean positions, the equations relating continuity map positions to mean articulator positions can theoretically be very complex; however, in order for the mean articulator position estimates to be useful, we hope that the equations are simple. I will consider only linear functions of the form:

$$\hat{A}_{ic} = \sum_{d=1}^{D} \alpha_{id} M_{dc} + k_i \quad \text{with } \varepsilon_{ic} = A_{ic} - \hat{A}_{ic}$$

where:

$\hat{A}_{ic}$ is the mean position of the receiver coil i for sounds of type c as estimated by the linear equation,

A_{ic} is the actual mean position of the receiver coil i for sounds of type c,

D is the number of dimensions in the continuity map,

M_{dc} is the position of code c on the d^{th} dimension of the continuity map, and

ε_{ic} is the error term.

The other parameters, α_{id} and k_i, are values that will minimize the sum of the squared error terms. An equation of this form is particularly interesting because, in solving for the unknown α_{id} and k_i values, we are finding axes in the continuity map that correspond most closely to the articulator positions -- essentially compensating for the fact that the continuity map can be rotated, scaled, of reflected with respect to the articulator positions.

Using standard multiple regression techniques (Neter, Wasserman & Kutner, 1985) we can find the α_{id} and k_i values that minimize the sum of the squared error terms. Multiple regression also gives a quantitative measure of the extent to which the equation is accurate, namely, the multiple regression R value. The multiple regression R is the correlations between $\hat{A}_{ic}$ and A_{ic}.

Figure 6 shows the multiple correlation r values obtained when trying to relate the positions of codes in the continuity map to the mean articulator positions of three key articulators -- the tongue rear (*x* and *y* positions), the tongue tip (*y* position) and the upper lip (*y* position). From Figure 6 we see that a four dimensional continuity map is sufficient to capture much of the information about the mean articulator positions, and that continuity maps with more than four dimensions do no better than a four dimensional continuity map.

G. MALCOM Implementation

G.1 Implementation

The acoustic processing and frequency sensitive vector quantization techniques used to implement continuity mapping were also used to implement MALCOM. In fact, the positions of the codes in the continuity maps were used as the initial estimates of the mean articulator positions for the MALCOM; the one dimensional continuity map was used to initialize the 1-dimensional maximum likelihood continuity map, the two-dimensional continuity map was used to initialize the two-dimensional maximum likelihood continuity map, etc.

Although the theory behind MALCOM allows articulator mean and covariance values to be estimated, I have only tried to estimate the mean positions. The covariance matrices are set to identity matrices.

Instead of using the simple gradient ascent algorithm described in section D.1 to calculate the most likely smooth articulator paths, I used the conjugate gradient method (Press, Flannery, Teukolsky & Vetterling, 1988) to perform the maximization. The conjugate gradient algorithm requires the user to supply a function that return the gradient of the function to be maximized. However, in order to make sure that only smooth solutions were considered, the function returning the gradient actually returned the low-pass filtered gradient.

G.2 Evaluating Maximum Likelihood Continuity Mapping

The maps generated by MALCOM were evaluated using the same techniques used to evaluate the continuity maps. Figure 7 show the multiple regression R values obtained for the maximum likelihood continuity maps.

As with the continuity maps, there is little or nothing to be gained by using more than a four dimensional solution for this data set. The main different between the maximum likelihood continuity map and the continuity map is the accuracy of the mean position estimates. Clearly, the MALCOM is doing a much better job of recovering the means of the articulator distributions.

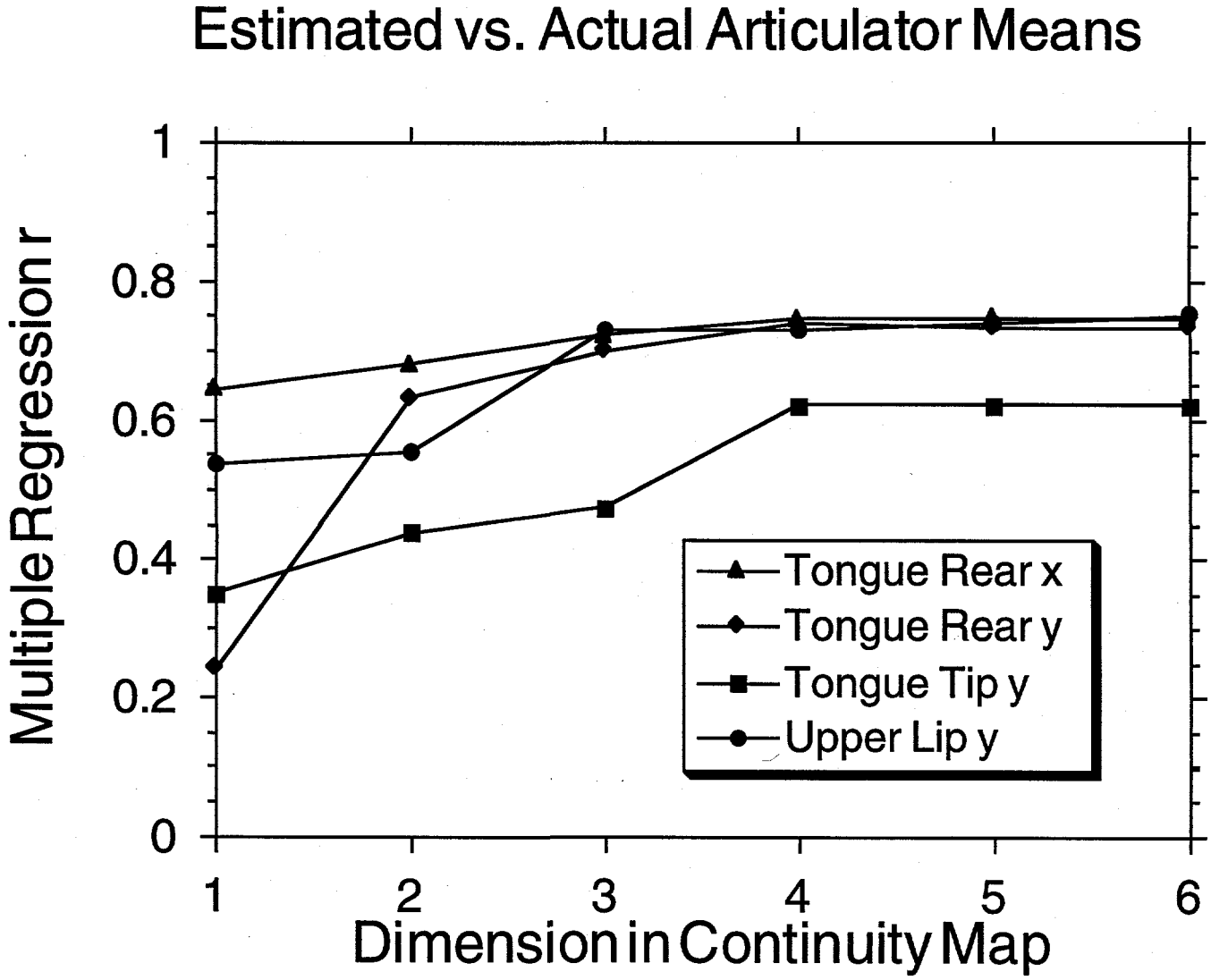
Estimated vs. Actual Articulator Means
Multiple Regression r
1
0.8
0.6
0.4
0.2
0
1
2
3
4
5
6
Dimension in Continuity Map
Tongue Rear x
Tongue Rear y
Tongue Tip y
Upper Lip y

Figure 6

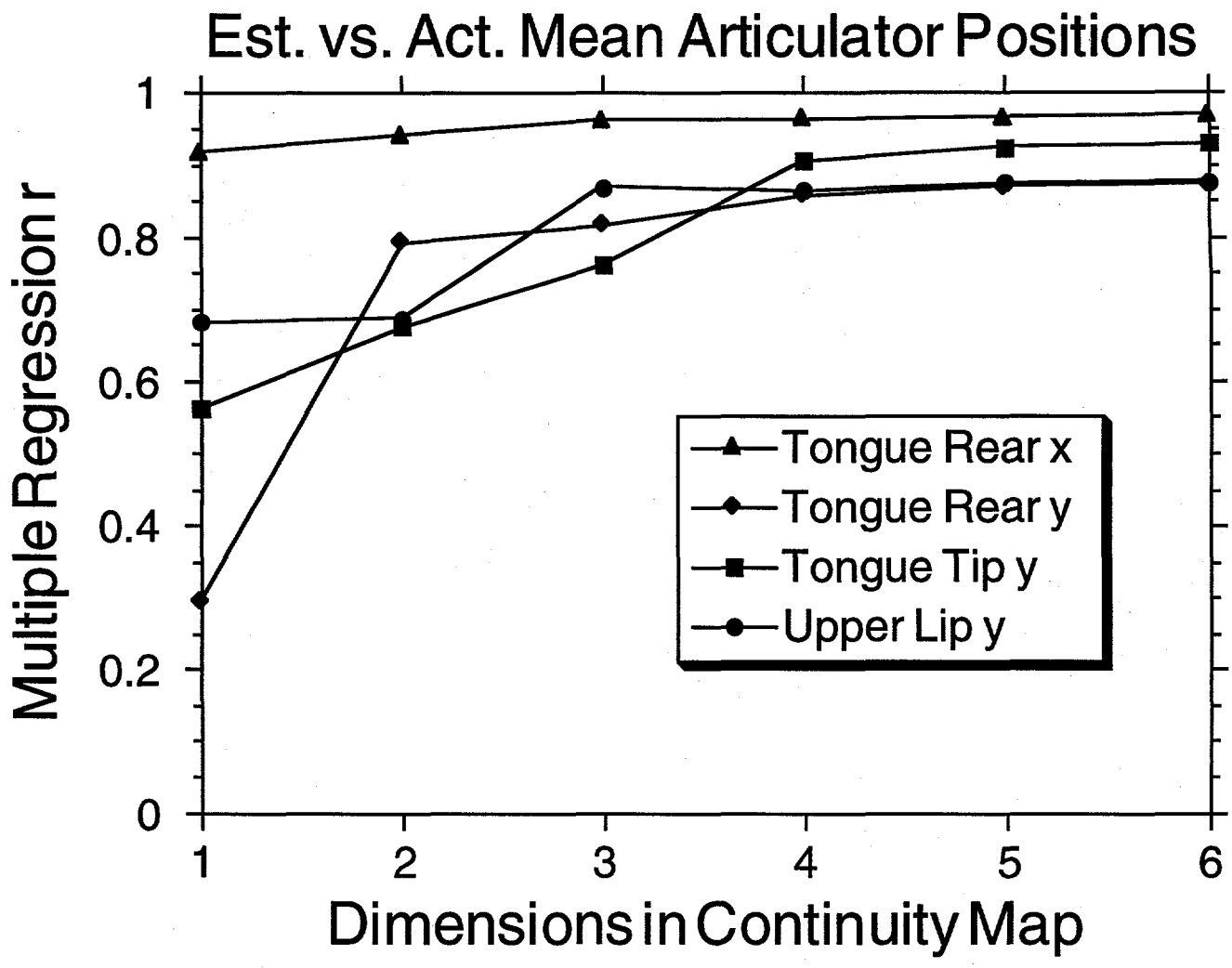
Est. vs. Act. Mean Articulator Positions
Multiple Regression r
1
0.8
0.6
0.4
0.2
0
1
2
3
4
5
6
Dimensions in Continuity Map
Tongue Rear x
Tongue Rear y
Tongue Tip y
Upper Lip y

Figure 7

H. Conclusion

An algorithm that recovers the positions of key articulator from speech acoustics has been presented. The algorithm does not need articulator measurements at any time during training and does not make strict assumptions about the propagation of sound through the vocal tract. The major assumption underlying the technique are that speech sounds produced sufficiently close together in time must have been created by similar articulator configuratios because the articulators move continuously (where continuously is meant in the mathematical sense: articulators do not move from one location to another without occupying intermediate positions. A second assumption is that the distribution of articulator psoitions that produce a given sound type is approximately Gaussian. This technique has been shown to recover the mean articulator positions more accurately than the only other technique with similar qualities -- continuity mapping.

References

Ahalt, S., Krishnamurthy, A., Chen, P., & Melton, D. (1990). Competitive learning algorithms for vector quantization.Neural Networks, 3, 277-290.

Atal, B. S., Chang, J. J., Mathews, M. V., & Tukey, J. W. (1978). Inversion of articulatory-to-acoustic transformation in the vocal tract by a computer-sorting technique. Journal of the Acoustical Society of America, 63(5), 1535-1555.

Boe, L. J., Perrier, P., & Bailly, G. (1992). The geometric vocal tract variables controlled for vowel production: proposals for constraining acoustic-to-articulatory conversion. Journal of Phonetics, 20, 27-38.

Duda, R., & Hart, P. (1973). Pattern Classification and Scene AnalysisNew York: John Wiley & Sons.

Fant, G. (1973). Chapter 1: The acoustics of speech, Speech Sounds and Features, . Cambridge, MA: MIT Press.

Hogden. (1991). Low-dimensional phoneme mapping using a continuity constraint. Unpublished Doctoral Dissertation, Stanford University.

Hogden, J., Rubin, P., & Saltzman, E. (in press). An unsupervised method for learning to track tongue position from an acoustic signal.Bulletin de la Communication Parlee

Hogden, J., Saltzman, E., & Rubin, P. (1993,). Tracking moving objects with unsupervised neural networks. Paper presented at the World Conference on Neural Networks, Portland, Oregon.

Hogden, J., Zlokarnik, I., Lofquist, A., Gracco, V., Rubin, P., & Saltzman, E. (submitted). Accurate recovery of articulator positions from acoustics -- new conclusions based on human data.Journal of the Acoustical Society of America

Kruskal, J. (1964a). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika, 29(1), 1-26.

Kruskal, J. (1964b). Nonmetric multidimensional scaling: a numerical method. Psychometrika, 29(2), 115-129.

Ladefoged, P., Harshman, R., Goldstein, L., & Rice, L. (1978). Generating vocal tract shapes from formant frequencies.Journal of the Acoustical Society of America, 64(4), 1027-1035.

Linde, Y., Buzo, A., & Gray, R. (1980). An algorithm for vector quantizer design. IEEE Transactions on Communications, COM-28 84-95.

Markel, J., & Gray, A. (1976). Linear Prediction of Speech New York: Springer-Verlag.

Marsden, J., & Tromba, A. (1981). Vector Calculus (2 ed.). San Francisco: W. H. Freeman and Company.

Muller, E., & McLeod, G. (1982). Perioral biomechanics and its relation to labial motor control. Journal of the Acoustical Society of America, 71(Suppl. 1), S38.

Nelson, W. (1977). Articulatory feature analysis -- I. Initial processing considerations. Memorandum, Bell Laboratories

Neter, J., Wasserman, W., & Kutner, M. (1985). Applied Linear Statistical Models: Regression, Analysis of Variance, and Experimental Design(second edition ed.). Homewood, Illinois: Richard D. Irwin, Inc.

Oppenheim, A. (1969). Speech analysis-synthesis system based on homomorphic filtering. Journal of the Acoustical Society of America, 45(2), 458-465.

Oppenheim, A., Willsky, A., & Young, I. (1983). Signals and Systems. Englewood Cliffs, New Jersey: Prentice-Hall Inc.

Papcun, G., Hotchberg, J., Thomas, T., Laroche, F., Zacks, J., & Levy, S. (1992). Inferring articulation and recognizing gestures from acoustics with a neural network trained on X-ray microbeam data.Journal of the Acoustical Society of America, 92(2), 688-700.

Perkell, J., Cohen, M., Svirsky, M., Matthies, M., Garabieta, I., & Jackson, M. (1992). Electromagnetic midsagittal articulometer systems for transducing speech

articulatory movements. Journal of the Acoustical Society of America, 96(6), 3078-3096.

Press, W., Flannery, B., Teukolsky, S., & Vetterling, W. (1988). Numerical Recipes in C: The Art of Scientific Computing Cambridge: Cambridge University Press.

Quartieri, T. (1979). Minimum and mixed phase speech analysis-synthesis by adaptive homomorphic deconvolution. IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-27(4), 328-335.

Rahim, M., Goodyear, C., Kleijn, W., Schroeter, J., & Sondhi, M. (1993). On the use of neural networks in articulatory speech synthesis. Journal of the Acoustical Society of America, 93(2), 1109-1121.

Rahim, M. G., Kleijn, W. B., Schroeter, J., & Goodyear, C. C. (1991). Acoustic to articulatory parameter mapping using an assembly of neural networks Proceedings of the International Conference on Acoustics, Speech, and Signal Processing 485-488.

Schroeter, J., & Sondhi, M. (1994). Techniques for estimating vocal-tract shapes from the speech signal. IEEE Transactions on Speech and Audio Processing, 2(1), 133-150.

Shepard, R. (1980). Multidimensional scaling, tree-fitting, and clustering. Science, 210(4468), 390-398.

Sondhi, M. (1979). Estimation of vocal tract areas: the need for acoustical measurements. IEEE trans. ASSP, 27(3), 268-273.

Stevens, K., & House, A. (1955). Development of a quantitative description of vowel articulation. Journal of the Acoustical Society of America, 27(3), 484-493.

Stone, M., & Lele, S. (1992,). Representing the tongue surface with curve fits. Paper presented at the International Conference on Spoken Language Processing, Banf, Alberta Canada.

Wakita, H., & Gray, A. (1975). Numerical Determination of the lip impedance and vocal tract area functions. IEEE trans. ASSP, 23(6), 574-580.

Whalen, D., Wiley, E., Rubin, P., & Cooper, F. (1990). The Haskins Laboratories' pulse code modulation (PCM) system. Behavioral Research Methods, Instruments, & Computers, 22(6), 550-559.

Zlokarnik, I. (1995). Adding articulatory features to acoustic features for automatic speech recognition. Journal of the Acoustical Society of America, 97(5 pt. 2), 3246(A).