HMM-BASED SEQUENCE-TO-FRAME MAPPING FOR VOICE CONVERSION

Yu Qiao, Daisuke Saito, and Nobuaki Minematsu

The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, 113–0033, Japan
{qiao, dsk_saito, mine}@gavo.t.u-tokyo.ac.jp

ABSTRACT

Voice conversion can be reduced to a problem to find a transformation function between the corresponding speech sequences of two speakers. Perhaps the most voice conversions methods are GMM-based statistical mapping methods [1, 2]. However, the classical GMM-based mapping is frame-to-frame, and cannot take account of the contextual information existing over a speech sequence. It is well known that HMM yields an efficient method to model the density of a whole speech sequence and has found great successes in speech recognition and synthesis. Inspired by this fact, this paper studies how to use HMM for voice conversion. We derive an HMM-based sequence-to-frame mapping function with statistical analysis. Different from previous HMM-based voice conversion methods [3, 4, 5] that used forced alignment for segmentation and transform frames aligned to a state with its associated linear transformation, our method has a soft mapping function as a weighted summation of linear transformations. The weights are calculated as the HMM posterior probabilities of frames. We also propose and compare two methods to learn the parameters of our mapping functions, namely least square error estimation and maximum likelihood estimation. We carried out experiments to examine the proposed HMM-based method for voice conversion.

Index Terms: Voice conversion, sequence-to-frame mapping, HMM, speech synthesis

1. INTRODUCTION

Voice conversion (VC) aims at transforming a speaker's voice to make it sound like another speaker's without changing the linguistic contents. VC has many important applications in practice, and is receiving more and more attentions nowadays. Since utterances of two speakers differ from each other in many aspects, such as speech rate, duration, pitch, formant frequencies and speaking style etc., an ideal VC technique should take account of all these aspects. However, this is difficult in practice, some of these features are difficult to calculate and some are difficult to convert. For this reason, many VC techniques focus on the transformation of spectral features, and only conduct simple modifications for prosody features such as f0.

The GMM-based statistical mapping technique proposed by Stylianou et al. [1] has been widely used to convert spectral features between different speakers. These techniques make use of GMM to model the densities of source cepstral vectors [1] or joint cepstral vectors [6]. The mapping function is a weighted summation of linear transformations for each Gaussian component while the weights are calculated as posterior probabilities of source vectors. The parameters of the linear transformations are estimated by minimizing squared errors. The efficiency of GMM-based mapping and its advantage to other spectral conversion methods such as mapping codebooks and artificial neural network, have been demonstrated in many previous studies [1, 6, 2, 7]. However, GMM only describes the density of frame vectors and cannot take account of the contextual (dynamic) information. Although one can incorporate delta or delta-delta features into GMM, these features still only provide local dynamic information. On the other hand, HMM is a density model for sequences and the transition probabilities of HMM allow it to take account of the dynamics in speech. This paper studies an HMM-based mapping method for voice conversion. We deduce the formulas for sequence-to-frame mapping based on HMM by using statistical analysis. We use least square error (LSE) and maximum likelihood (ML) criteria to estimate the parameters of the mapping function. We find that the LSE estimation has a closed form solution, while the ML estimation leads to a nonlinear optimization problem. For this reason, we develop an EM-based algorithm for the ML estimation of HMM-based mapping. We conduct experiments to examine the performances of LSE estimation and ML estimation for HMM-based voice conversion. The results show the usefulness of the proposed method.

HMM has been applied to voice conversion in previous studies [3, 4, 5]. Kim et al. [3] introduced a hidden Markov VQ model for voice conversion, where the mapping function is determined by the codebook and the optimal states of a source utterance. Different from this method, we use normal HMM and our mapping function is a weighted summation of several linear transformations. Duxans et al. [4] used HMM to model the densities of source vectors and joint vectors, and estimated a linear transformation for each state of HMM to convert an input utterance. In [5], Wu et al. proposed duration-embedded DeBi-HMM for expressive voice conversion. Unlike the methods in [4] and [5] where the mapping functions only depend on the optimal states obtained by forced alignment, the mapping function of our method is derived by combining the linear transformations of different states using weights of posterior probabilities of states. We hope that this 'soft' mapping function can partly deal with the problem of spectral jumps at the boundaries of segments resulted from forced alignment [3, 4].

2. HMM-BASED VOICE CONVERSION

Voice conversion requires to find a mapping from an utterance of a source speaker to that of a target speaker. Let $Y = F(X)$ denote the mapping function where X, Y represent speech sequences of source and target speakers, respectively. Let $X = [x_1, x_2, ..., x_T]$, where x_t $(1 \leq t \leq T)$ represents a d-dimensional frame vector. However, to find a direct mapping between two sequences is very difficult. This is because that, a sequence usually contains a large number of elements and the length of sequences X and Y can be different. Therefore, many researchers reduced the sequence mapping to a frame-to-frame conversion problem, which is denoted by $y_t = f(x_t)$. A popular approach of this kind is to make use of the GMM-based statistical mapping, where GMM is used to model the density of frame vectors [1] of a source speaker or joint vectors of source and target speakers [2], and the final mapping function is the weighted combination of linear transformations estimated for each Gaussian component. In a previous work, we proposed a method called Mixture of Probabilistic Linear Regressions (MPLR) [8], which unifies the two GMM-based voice conversion techniques [1, 2] and leads to a better method for estimating mapping parameters. Although the frame-to-frame mapping is simple, it only considers the current frame x_t for conversion and doesn't take account of the contextual (dynamic) information to derive a mapping function, which plays an important role for speech perception.

For speech signals, GMM is a density model of frame vectors. And GMM-based mapping is a frame-to-frame conversion, which cannot take account of the contextual information over a speech sequence. Partially for this reason, it is observed that the classical GMM-based mapping usually generates overly smoothed utterances [7]. To overcome this problem, Toda et al.[7] took consideration of the dynamic features with a trajectory model and alleviate the overly smoothing problem by considering a global variance feature. In this paper, we try to solve this problem by using HMM. Different from GMM, HMM provides a probability model for sequences and accounts for the dynamic information by using transition probabilities. The effectiveness and efficiency of HMM has been demonstrated in both speech recognition and speech synthesis. Perhaps the most simplest idea for applying HMM to voice conversion is to 1) prepare a transformation for each state, 2) determine the optimal state of each frame of an input utterance to be converted with forced alignment (Viterbi decoding), and 3) convert each frame vector by the transformation associated with its optimal state. This idea was adopted by previous works [3, 4, 5]. However, forced alignment gives a hard segmentation of the speech sequence. And this usually leads to spectral jumps at the boundaries of segments, and diminishes the smoothness of converted speech. In this paper, we deal with this problem by introducing a 'soft' mapping function. This soft mapping function is a weighted summation of the linear transformations of all states, where the weights

2.1. HMM-based sequence-to-frame mapping

We use an ergodic HMM with K states to model the density of speech sequence of source speaker. Let $p(x|s)$ denote a state-observation probability of frame vector x given state s, and $p(s'|s)$ represent a state-transition probability from state s to s'. In HMM, the joint probability of speech sequence $X = [x_1, x_2, ..., x_T]$ and its corresponding state sequence $S = [s_1, s_2, ..., s_T]$ $(1 \leq s_t \leq K)$ can be calculated by,

$$p(X, S) = P(X|S)P(S) = \prod_{t=1}^{T} p(x_t|s_t)p(s_1)\prod_{t=2}^{T} p(s_t|s_{t-1}). \tag{1}$$

Given state s and source vector x, we assume that target vector y has the following linear-Gaussian distribution,

$$p(y|s, x) = N(y|B_s x + b_s, \Sigma_s), \tag{2}$$

where B_s, b_s denote the linear transformation parameters, and Σ_s represents the covariance matrix of the above linear-Gaussian distribution. Then the expectation (mean) of y is given by $E_{p(y|s,x)}[y] = B_s x + b_s$.

With the HMM of source sequence X, we can calculate the conditional probability of the t-th target vector y_t given sequence X as

$$p(y_t|X) = \sum_{S \in \mathbb{S}} p(y_t, S|X) = \sum_{S \in \mathbb{S}} p(y_t|S, X)p(S|X), \tag{3}$$

where $\mathbb{S}$ is the set of possible state sequences for X.

When state s_t is given, we assume that target vector y_t only depends on its corresponding source vector x_t. This allows us to make the following simplification,

$$p(y_t|S, X) = p(y_t|s_t, x_t) = N(y_t|B_{s_t} x_t + b_{s_t}, \Sigma_{s_t}). \tag{4}$$

Under this assumption, we can deduce the probability of Eq. 3 as

$$\begin{aligned} &\sum_{S \in \mathbb{S}} p(y_t|S, X)p(S|X) = \sum_{s_t} p(y_t|s_t, x_t) \sum_{S^{/s_t}} p(S^{/s_t}|X) \\ &= \sum_{k=1}^{K} p(y_t|s_t = k, x_t)p(s_t = k|X), \end{aligned} \tag{5}$$

where $S^{/s_t} = s_1, ...s_{t-1}s_{t+1}...s_T$. Noted that posterior probability $p(s_t = k|X)$ can be calculated efficiently by the famous backward and forward algorithm of HMM [9]. With Eq. 5, the mapping of sequence X to frame y_t can be estimated by

$$f_{\text{HMM}}(X, t) = E_{p(y_t|X)}[y_t] = \sum_{k=1}^{K} p(s_t = k|X)(B_k x_t + b_k). \tag{6}$$

2.2. Estimation of mapping parameters

In this section, we discuss how to calculate the parameters of HMM-based mapping function of Eq. 6 from a set of training sequence pairs $(X_n, Y_n)_{n=1}^{N}$, where source sequence $X_n = [x_1^n, ..., x_{T_n}^n]$ and target sequence $Y_n = [y_1^n, ..., y_{T_n}^n]$. We assume that X_n, Y_n have been aligned by dynamic time warping, and thus both have the same length denoted by T_n. We can train an HMM from the utterances of source speaker by the well known Baum-Welch algorithm [9] at first. And posterior probability $p(s_t^n = k|X_n)$ (s_t^n denotes the state of frame x_t^n in X_n) can be calculated with the backward and forward algorithm of HMM. Then the problem here is how to estimate the transformation parameters $\{B_s, b_s, \Sigma_s\}$ for state s. In the following, we describe two approaches for estimating these parameters. One is least square error (LSE) estimation and the other is maximum likelihood (ML) estimation. For convenience, we introduce notation $r_{t,k,n} = p(s_t^n = k|X_n)$.

2.2.1. *Least square estimation*

The objective function of least square estimation is,

$$\begin{aligned} &\min_{\{B_k, b_k\}} \sum_{n=1}^{N} \sum_{t=1}^{T_n} |f_{\text{HMM}}(X_n, t) - y_t^n|^2 \\ &= \sum_{n=1}^{N} \sum_{t=1}^{T_n} |\sum_{k=1}^{K} r_{t,k,n}(B_k x_t^n + b_k) - y_t^n|^2. \end{aligned} \tag{7}$$

This is a linear optimization problem, which can be solved directly. For simplicity, we introduce argument vector $\hat{x} = [x^T, 1]^T$ and set $A_k \hat{x}_t^n = B_k x_t^n + b_k$. Further, the following notations are used $X_k^n = [r_{1,k,n}\hat{x}_1, r_{2,k,n}\hat{x}_2, ..., r_{T_n,k,n}\hat{x}_{T_n}]$, $X_k = [X_k^1, ..., X_k^N]$, $\mathbb{X} = [X_1^\top, X_2^\top, ..., X_K^\top]^\top$, $Y_n = [y_1, y_2, ..., y_{T_n}]$, and $\mathbb{Y} =$

$[Y_1, Y_2, ..., Y_N]$, where '$\top$' denotes matrix transpose. The optimal matrices $\{A_k^*\}$ for Eq. 7 are given by

$$[A_1^*, A_2^*, ..., A_K^*] = \mathbb{Y}\hat{\mathbb{X}}^\top(\hat{\mathbb{X}}\hat{\mathbb{X}}^\top)^{-1}. \tag{8}$$

However, this is very computationally expensive, since matrix $\hat{\mathbb{X}}$ has a size of $K(d+1) \times \sum_n T_n$. To overcome this limitation, we use the following decomposition method. Remind $\sum_k r_{t,k,n} = 1$ and $r_{t,k,n} > 0$. According to Jensen's inequality, we have $|\sum_k r_{t,k,n}(y_t^n - A_k\hat{x}_t^n)|^2 \leq \sum_k r_{t,k,n}|y_t^n - A_k\hat{x}_t^n|^2$. Therefore, Eq. 7 can be approximated by the following upper bound,

$$\arg\min_{\{A_k\}} \sum_k \sum_n \sum_t r_{t,k,n}|y_t^n - A_k\hat{x}_t^n|^2. \tag{9}$$

This can be further decomposed into K independent linear optimization problems,

$$\arg\min_{A_k} \sum_n \sum_t r_{t,k,n}|y_t^n - A_k x_t^{\hat{n}}|^2. \tag{10}$$

The optimal matrix for Eq. 10 is given by $A_k^{\#} = \mathbb{Y}X_k^\top(X_k X_K^\top)^{-1}$. These calculations are closely related to those discussed in our previous work on MPLR [8].

2.2.2. *Maximum likelihood estimation*

Although least square estimation is simple and has a closed form solution, it doesn't consider the covariance matrices $\{\Sigma_s\}$ in Eq. 2. In the section, we make use of maximum likelihood (ML) estimation to overcome this problem. For linear regression, LSE and ML estimations lead to the same estimations. However, as we will see shortly this is not the case for our problem. Formally, ML estimation is defined as,

$$\begin{aligned}&\max_{B_k, b_k, \Sigma_k} \prod_n \prod_t p(y_t^n|X_n)\\ =&\prod_n \prod_t \sum_{k=1}^K p(s_t^n = k|X_n)N(y_t^n|B_k x_t^n + b_k, \Sigma_k).\end{aligned} \tag{11}$$

Then log likelihood function is given by,

$$\begin{aligned}&\mathcal{L}(\{B_k, b_k, \Sigma_k\})\\ =&\sum_n \sum_t \log\left(\sum_{k=1}^K p(s_t^n = k|X_n)N(y_t^n|B_k x_t^n + b_k, \Sigma_k)\right).\end{aligned} \tag{12}$$

For convenience, we introduce parameters $\gamma_{t,k,n}$ and $\beta_{t,k,n}$ as follows $\gamma_{t,k,n} = N(y_t^n|B_k x_t^n + b_k, \Sigma_k)$ and $\beta_{t,k,n} = \frac{\gamma_{t,k,n} r_{t,k,n}}{\sum_j \gamma_{t,j,n} r_{t,j,n}}$.

To maximize Eq. 11, we calculate the derivatives of log likelihood $\mathcal{L}$ as,

$$\frac{\partial\mathcal{L}}{\partial b_k} = \sum_n \sum_t \beta_{t,k,n}(\Sigma_k)^{-1}(y_t^n - B_k x_t^n - b_k) = 0, \tag{13}$$

$$\frac{\partial\mathcal{L}}{\partial B_k} = \sum_n \sum_t \beta_{t,k,n}(\Sigma_k)^{-1}(y_t^n - B_k x_t^n - b_k)x_t^{n\top} = 0, \tag{14}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial \Sigma_k} =& \sum_n \sum_t \frac{1}{2}\beta_{t,k,n}\{(\Sigma_k)^{-1}(y_t^n - B_k x_t^n - b_k)\\ &(y_t^n - A_k x_t^n - b_k)^\top(\Sigma_k)^{-1} - (\Sigma_k)^{-1}\} = 0.\end{aligned} \tag{15}$$

The above formulas don't have closed form solutions, since $\{\beta_{t,k,n}\}$ include the parameters $\{B_k, b_k, \Sigma_k\}$. Then we develop the following EM algorithm for parameter estimation.

Algorithm 1 EM algorithm for ML estimation

```
1: Initialize transformation parameters $\{B_k, b_k, \Sigma_k\}$.
2: E-step: Calculate hidden parameters $\{\gamma_{t,k,n}\}$ and $\{\beta_{t,k,n}\}$.
3: M-step: Estimate the following parameters.
       $N_k = \sum_n \sum_t \beta_{t,k,n},$   (16)
       $\bar{x}_k = \frac{1}{N_k} \sum_n \sum_t \beta_{t,k,n} x_t^n,$   (17)
       $\bar{y}_k = \frac{1}{N_k} \sum_n \sum_t \beta_{t,k,n} y_t^n,$   (18)
       $\Sigma_k^{xx} = \frac{1}{N_k} \sum_n \sum_t \beta_{t,k,n}(x_t^n - \bar{x}_k)(x_t^n - \bar{x}_k)^\top,$   (19)
       $\Sigma_k^{yx} = \frac{1}{N_k} \sum_n \sum_t \beta_{t,k,n}(y_t^n - \bar{y}_k)(x_t^n - \bar{x}_k)^\top,$   (20)
       $\Sigma_k^{yy} = \frac{1}{N_k} \sum_n \sum_t \beta_{t,k,n}(y_t^n - \bar{y}_k)(y_t^n - \bar{y}_k)^\top.$   (21)
   Update transformation parameters as
       $B_k^* = \Sigma_k^{yx}(\Sigma_k^{xx})^{-1},$   (22)
       $b_k^* = \bar{y}_k - B_k^*\bar{x}_k,$   (23)
       $\Sigma_k^* = \Sigma_k^{yy} - \Sigma_k^{yx}(\Sigma_k^{xx})^{-1}(\Sigma_k^{yx})^\top.$   (24)
4: Evaluate the log likelihood $L(\{B_k, b_k, \Sigma_k\})$.
5: Terminate the procedure when convergence, otherwise go to step 2.
```

3. EXPERIMENTS

We carried out experiments to evaluate the proposed two HMM-based voice conversion methods. We made use of the ATR-503 phoneme balanced sentences in the experiments. The data set used contains 503 utterances from a male speaker and another 503 utterances from a female speaker with the same linguistic contents. The sampling frequency is 16k Hz. For each utterance, we calculated its 24-D cepstrum sequence. We converted the female voice to the male voice. For conversion, the training utterances of the source speaker and the target speaker are aligned by dynamic time warping. In all the following experiments, we use ergodic HMM for density calculation of source sequence. The cepstrum distortion [1] between the target cepstrum vector $[y_t^1, ..., y_t^{24}]$ and the converted cepstrum vector $[y_c^1, ..., y_c^{24}]$ is defined by, CD[dB] $= \frac{10}{\ln 10}\sqrt{2\sum_d(y_t^d - y_c^d)^2}$. And we use the average cepstrum distortion as an objective evaluation measure.

3.1. Comparison of LSE and MLE

We make comparisons between the two parameter estimation methods, least square estimation (LSE) and maximum likelihood estimation (MLE). We changed the number of training utterances and the number of states. In both experiment, the testing set includes 50 new utterances. The results are summarized in Table 1. As one can see, LSE and MLE have very similar performance, but the cepstrum distortion of LSE is a bit smaller than that of MLE. This is because LSE directly optimizes squared errors. Note that as MLE requires EM iterations, MLE is much more computationally expensive than

Table 1. Average cepstrum distortions [dB] of LSE and MLE. (N is the number of training utterances. M is the number of states.)

Method	N ($M = 10$)					
	10	20	30	50	100	200
LSE	5.030	4.881	4.832	4.784	4.758	4.735
MLE	5.037	4.884	4.836	4.786	4.759	4.736
Method	M ($N = 150$)					
	5	7	9	15	30	50
LSE	4.780	4.766	4.760	4.751	4.741	4.745
MLE	4.781	4.767	4.761	4.754	4.742	4.746

LSE.

3.2. Experiment 2

In this experiment, we made comparison between the proposed HMM-based mapping method with LS estimation and the previous HMM-based mapping method [4]. We conducted two experiments. In the first experiment, we fixed the states of HMM as 5 and changed the number of training utterances. In the second experiment, we changed the number of states of HMM while 150 training utterances were used for training. The test set includes 50 different utterances. The results are shown in Fig. 1. As one can see that the proposed method always outperforms the previous HMM-based conversion method. We can also find that the difference between the two methods enlarges as state number increases. This is because as state number increases, the forced alignment of the previous method leads to more segments and thus more boundaries with spectral jumps, which affects its performance. We also conducted experiments to make comparison with GMM-based mapping. The cepstral distortions of both methods are similar. The reason may be that we only have limited data of source speaker to train ergodic HMM. In ergodic HMM, the transition probabilities for all the state pairs should be calculated. We cannot estimate reliable transition probabilities from the limited data. The experimental results are still limited here. We will examine the proposed method with bigger database and larger numbers of states in the future.

4. CONCLUSIONS AND DISCUSSIONS

This paper studies a HMM-based sequence-to-frame mapping method for voice conversion. We derive a novel HMM-based mapping function with statistical analysis, and develop two methods to estimate transformation parameters of the mapping function, one is least square estimation (LSE) and the other is maximum likelihood estimation (MLE). The former can be reduced to a linear optimal problem, and has its closed form solution. For the later, we develop an EM-based algorithm to calculate the optimal parameters. Compared with the previous GMM-based voice conversion techniques, the use of HMM allows to account for contextual information in speech signals. Compared to the previous HMM-based voice conversion method, our method use a soft mapping function to avoid spectral jumps at state boundaries. We carried out experiments to compare LSE and MLE. The results show that both methods have very similar performance. We also conducted a comparative experiment with the previous HMM-based mapping method [4]. The results indicate that our method has a better performance in terms of cepstrum distortion. One limitation of the current method comes from the assumption made in Eq. 4, which states that when the state of a source frame is given, the converted frame only depends on the source frame and its state. This may not be strictly true in practice. Finally, it is noted that experimental results are only limited. Subjective test should be conducted to assess the proposed methods.

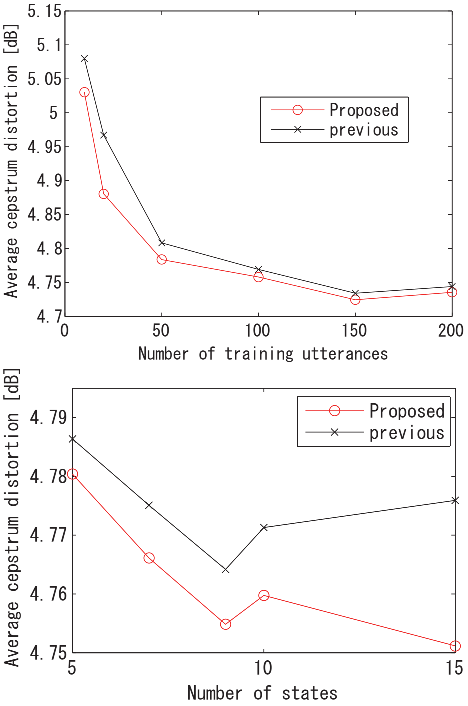

Fig. 1. Comparison of the proposed method and the previous HMM based mapping method.

5. REFERENCES

[1] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. on SAP,*, vol. 6, no. 2, pp. 131–142, 1998.

[2] A. Kain and MW Macon, "Spectral voice conversion for text-to-speech synthesis," *Proc. ICASSP*, pp. 285–288, 1998.

[3] E.K. Kim, S. Lee, and Y.H. Oh, "Hidden Markov model based voice conversion using dynamic characteristics of speaker," *Proc. 5th European Conference on Speech Communication and Technology*, pp. 2519–2522, 1997.

[4] H. Duxans, A. Bonafonte, A. Kain, and J. Van Santen, "Including dynamic and phonetic information in voice conversion systems," *Proc. of the ICSLP*, pp. 1193–1196, 2004.

[5] C.H. Wu, C.C. Hsia, T.H. Liu, and J.F. Wang, "Voice conversion using duration-embedded bi-HMMs for expressive speech synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1109–1116, 2006.

[6] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," *Proc. ICASSP,*, pp. 655–658, 1988.

[7] T. Toda, A.W. Black, and K. Tokuda, "Voice Conversion Based on Maximum-Likelihood Estimation of Spectral Parameter Trajectory," *IEEE Trans. on ASLP*, vol. 15, no. 8, pp. 2222–2235, 2007.

[8] Y. Qiao and N. Minematsu, "Mixture of probabilistic linear regressions: a unified view of GMM-based mapping techniques," *Proc. ICASSP*, pp. 3913–3916, 2009.

[9] LR Rabiner, "A tutorial on hidden Markov models and selected applications inspeech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.