A Minimax Search Algorithm for Robust Continuous Speech Recognition

Hui Jiang, *Member, IEEE,* Keikichi Hirose, *Member, IEEE,* and Qiang Huo, *Member, IEEE*

***Abstract*—In this paper, we propose a novel implementation of a minimax decision rule for continuous density hidden Markov-model-based robust speech recognition. By combining the idea of the minimax decision rule with a normal Viterbi search, we derive a recursive minimax search algorithm, where the minimax decision rule is repetitively applied to determine the partial paths during the search procedure. Because of its intrinsic nature of a recursive search, the proposed method can be easily extended to perform continuous speech recognition. Experimental results on Japanese isolated digits and TIDIGITS, where the mismatch between training and testing conditions is caused by additive white Gaussian noise, show the viability and efficiency of the proposed minimax search algorithm.**

***Index Terms*—Minimax rule, plug-in-MAP rule, robust decision rule, robust speech recognition.**

I. Introduction

IT IS now well known that the mismatches between training and testing conditions will considerably degrade the performance of an automatic speech recognition (ASR) system. How to maintain the recognizer's performance under various mismatches has recently become one of the hottest topics in the area of robust speech recognition. The so-called "compensation/adaptation" approaches [6], which aim at reducing the involved mismatches as much as possible, have formed the mainstream of the current robust speech recognition technology. However, in the past few years, based on robustness theory, some works have been performed to modify the basic decision rule used by the speech recognition system. Instead of directly compensating for the underlying mismatches, the decision rule of the ASR system is designed to be inherently robust to the possible unknown mismatches. This scheme becomes a potential approach for robust ASR because no *rigid* assumptions about the sources and mechanisms of the mismatches have to be made. Two sets of robust decision rule, namely *minimax* decision rule [2], [5], [8] and *Bayesian predictive classification* (BPC) rule [2], [4], [9] have been studied for ASR. In [8], Merhav and Lee first mentioned the minimax rule in speech recognition community and proposed an implementation for isolated word recognition task. In [2], a so-called *modified minimax* method was proposed to perform minimax rule under a Bayesian framework. In both of these existing minimax implementations, instead of dynamically searching a desired answer in a structural network representation of all possible hypotheses, decisions are made only from a list of finite candidates. This makes them difficult to be extended to perform continuous speech recognition (CSR) except in an N-Best rescoring mode.

In this paper, we combine the idea of the minimax rule with a normal Viterbi search to derive a minimax recursive search algorithm for continuous density hidden Markov model (CDHMM)-based speech recognition. The proposed implementation can be outlined as follows:

- for every time instant, the least favorable model parameters in the minimax rule are estimated based on each active partial path via only one iteration; then
- score of the partial path can be recomputed accordingly by using the estimated least favorable parameters, based on these recomputed scores;
- all the active partial paths are propagated in the network in a similar way as in the normal Viterbi search.

Because of its intrinsic nature of a recursive search, the approach can be easily extended to perform CSR. A series of experiments are performed on the recognition of isolated digits and TI connected digit strings (TIDIGITS), where the mismatch between training and testing conditions is caused by additive white Gaussian noise (AWGN). The experimental results show that

- for the isolated digit recognition task, in comparison with the standard Plug-in-MAP method, all three minimax algorithms are able to improve the robustness considerably, while the proposed algorithm performs the best;
- for connected digit task (TIDIGITS), the proposed minimax search algorithm also achieves a much better performance than that of the conventional Viterbi search algorithm;
- increased computational overhead in the minimax search is generally affordable at least in some small vocabulary tasks.

The remainder of the paper is organized as follows. The minimax rule is defined and derived in Section II based on statistical decision theory and robustness theory. In Section III, we briefly review two existing implementations of the minimax rule for speech recognition in the literature. The proposed minimax search algorithm is described in detail in Section IV, followed by a report of the related experiments and results in Section V.

Manuscript received November 17, 1998; revised March 31, 2000. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Richard C. Rose.

H. Jiang was with the Department of Information and Communication Engineering, University of Tokyo, Tokyo 113-0033, Japan. He is now with the Dialogue Systems Research Department, Bell Laboratories, Lucent Technologies, Murray Hill, NJ 07974 USA (e-mail: hui@research.bell-labs.com).

K. Hirose is with the Departments of Frontier Informatics and Information and Communication Engineering, University of Tokyo, Tokyo 113-0033, Japan (e-mail: hirose@gavo.t.u-tokyo.ac.jp).

Q. Huo is with the Department of Computer Science and Information Systems, The University of Hong Kong, Hong Kong (e-mail: qhuo@csis.hku.hk).

Publisher Item Identifier S 1063-6676(00)09261-0.

Finally, we conclude the paper with some discussions in Section VI.

II. Minimax Rule for Robust Speech Recognition

In a typical speech recognition problem, our task is to classify a speech observation (usually feature vector sequence extracted from speech signal) X into one of a fixed number of classes. For convenience, each class is referred to as a *word* W hereafter. Depending on the problem of interest, a word W may be of any linguistic unit, e.g., a phoneme, a syllable, a word, a phrase, a sentence, etc. Let Ω denote the set of all words to be classified, i.e., $W \in \Omega$. Let *feature space* (or *observation space*) χ denote the set of all possible speech observation X, i.e., $X \in \chi$. In a decision problem, we always have a finite set, called *decision space* $\mathcal{D}$, which consists of all possible decisions to be made. For speech recognition, simply $\mathcal{D} = \{d_W | W \in \Omega\}$, where d_W means that the word W is chosen as the final recognition result. Obviously, a speech recognition problem consists in constructing a decision rule which may be defined as a mapping function from χ to $\mathcal{D}$

$$d = d(X) \quad \text{where} \quad X \in \chi,\, d \in \mathcal{D}. \tag{1}$$

Any mapping $\chi \to \mathcal{D}$ defines a decision rule. Hence there exist an infinite set of decision rules, denoted by $\mathcal{D}^*$, for the same problem. Not all of them are of equal value in practice though. They may be compared by many characteristics, e.g., classification accuracy in speech recognition. In a more general setting, we usually assign a *loss function* $w(d(X), X, W)$ to a decision rule $d(\cdot)$, where $w(d(X), X, W)$ denotes the loss involved in making decision $d(X)$ when the observation X actually comes from W. In a statistical paradigm, a word W and an observation X are viewed as a jointly distributed random pair (W, X), whose joint distribution is denoted as $p(W, X)$. The *classification risk* for a decision rule $d(\cdot)$ is then defined as an expected value of the loss function

$$r(d(\cdot)) = \mathbf{E}_{W,X}[w(d(X), X, W)] \tag{2}$$

where $\mathbf{E}_{W,X}[\,\cdot\,]$ denotes mathematical expectation with respect to the joint distribution of W and X.

In speech recognition, what we are interested in is the recognition accuracy. Consequently, a so-called (0-1)-loss w is often used:

$$w(d(X), X, W) = \delta(d(X) - W) = \begin{cases} 0 & d(X) = W \\ 1 & d(X) \neq W \end{cases} \tag{3}$$

where $\delta(\cdot)$ is the indicator function. In this case, the loss is 0 for each correct decision and 1 for each wrong decision. If $p(W, X)$ is exactly known, an optimal decision rule $d^*(\cdot)$ can then be defined as the one to minimize the above classification risk

$$\begin{aligned} d^*(\cdot) &= \arg \min_{d(\cdot)\in\mathcal{D}^*} \mathbf{E}_{W,X}[\delta(d(X) - W)] \\ &= \arg \min_{d(\cdot)\in\mathcal{D}^*} \sum_{W\in\Omega} \int \delta(d(X) - W) \cdot p(W, X)\, \mathrm{d}X \\ &= \arg \max_{W} p(W|X). \end{aligned} \tag{4}$$

This decision rule is known as the optimal maximum *a posteriori* (MAP) decision rule [2], [10]. However, in practice, we have no means to determine $p(W, X)$ exactly. Therefore, the above optimal rule will never be achievable. A simple heuristic solution is first to assume a parametric form for $p(W, X)$ and then to estimate its parameters from training data using a parameter estimation technique. Then, we plug in the estimators to the optimal rule to obtain a *plug-in MAP* rule [2], [9], [10]. In the plug-in MAP rule, the estimators are treated as the true values. The uncertainty with respect to estimation and model assumption is not taken into account in the decision making procedure. Therefore, the plug-in MAP rule is not a robust decision rule.

Alternatively, we have another strategy to construct a decision rule, which can take uncertainty into account in the decision-making as follows: We first assume that the true distribution lies in a neighborhood of the estimated (or hypothetical) one. Such a neighborhood is referred to as ϵ-deviation uncertainty neighborhood in robust statistics literature (e.g., [1]). Then, the decision is made to safeguard some criteria within that neighborhood.

If we define the robustness to mean insensitivity with regard to small deviations from the assumptions, a quantitative measure of robustness might be concerned with the maximum degradation of performance for a possible ϵ-deviation from the assumptions. A robust procedure that minimizes this maximum degradation will thus be called a minimax procedure.

For simplicity, we do not consider the uncertainty of the language model $p(W)$ in this study. Given the conditional distribution $p(X|W)$ and its ϵ-deviation uncertainty neighborhood denoted as $P_\epsilon(W)$, we consider the expected recognition error probability (i.e., classification risk)

$$\begin{aligned} \Pr_e(d(\cdot), p(\cdot|\cdot)) &= \mathbf{E}_{W,X}[\delta(d(X) - W)] \\ &= \int_X \sum_{W\neq d(X)} p(X|W) \cdot p(W)\, \mathrm{d}X. \end{aligned} \tag{5}$$

Let $\Pr_e{}^+(d(\cdot))$ denote the upper bound of the above expected error probability when $p(X|W)$ takes all admissible distributions within $P_\epsilon(W)$, i.e.,

$$\overset{+}{\Pr_e}(d(\cdot)) = \sup_{p(X|W)\in P_\epsilon(W)} \int_X \sum_{W\neq d(X)} p(X|W) \cdot p(W)\, \mathrm{d}X. \tag{6}$$

A decision rule which minimizes the above maximum error probability $\Pr_e{}^+(d(\cdot))$ (with respect to uncertainty $P_\epsilon(W)$) is referred to as *minimax* decision rule [1]

$$\begin{aligned} d_1^*(\cdot) &= \arg \min_{d(\cdot)\in\mathcal{D}^*} \Pr_e{}^+(d(\cdot)) \\ &= \arg \min_{d(\cdot)\in\mathcal{D}^*} \sup_{p(X|W)\in P_\epsilon(W)} \int_X \\ &\quad \cdot \sum_{W\neq d(X)} p(X|W) \cdot p(W)\, \mathrm{d}X. \end{aligned} \tag{7}$$

A minimax rule is in principle geared to protect against the possibly worst case. It serves as a reliable decision strategy when a thorough knowledge about uncertainty is unavailable. However,

as shown in (7), a minimax decision rule usually does not possess a straightforward form to implement, even for some simple distributions. In most of practical applications, some more relaxed minimax rule has to be adopted. One possibility is to use the upper bound of $\Pr_e^+(d(\cdot))$

$$\overset{++}{\Pr_e}(d(\cdot)) = \int_X \sum_{W \neq d(X)} p(W) \cdot \sup_{p(X|W) \in P_c(W)} p(X|W)\, \mathrm{d}X. \quad (8)$$

A decision rule which minimizes the above $\Pr_e{}^{++}(d(\cdot))$ is as follows:

$$d_2^*(X) = \arg\max_W \left[p(W) \cdot \sup_{p(X|W) \in P_c(W)} p(X|W) \right]. \quad (9)$$

This is the so-called *minimax decision rule* which was first studied by Merhav and Lee in [8]. In order to emphasize its difference from the minimax decision rule defined in (7), the above decision rule $d_2^*(\cdot)$ is referred to as *quasiminimax decision rule*.

III. Two Previous Minimax Methods for Robust ASR

Suppose we model each *word* W with an N-state CDHMM with parameter vector $\Lambda = (\pi, A, \theta)$, where π is the initial state distribution, $A = \{a_{ij};\ 1 \le i, j \le N\}$ is the transition matrix, and θ is the parameter vector composed of mixture parameters $\theta_i = \{\omega_{ik}, m_{ik}, r_{ik}\}_{k=1,2,\ldots,K}$ for each state i, where K denotes the number of Gaussian mixtures in each state. The state observation probability density function (p.d.f.) is assumed to be a mixture of multivariate Gaussian p.d.f.s with diagonal precision matrices

$$p(\mathbf{x}|\theta_i) = \sum_{k=1}^{K} \omega_{ik} \mathcal{N}(\mathbf{x}|\theta_{ik}) = \sum_{k=1}^{K} \omega_{ik} \prod_{d=1}^{D} \sqrt{\frac{r_{ikd}}{2\pi}} e^{-(1/2) r_{ikd}(x_d - m_{ikd})^2} \quad (10)$$

where the mixture coefficients ω_{ik}'s satisfy the constraint $\sum_{k=1}^{K} \omega_{ik} = 1$, m_{ikd} is the mean and r_{ikd} is the precision (inverse variance) in dth dimension, and D is the dimension of the feature vector.

As stated above, in practice, the quasiminimax rule is often used for the sake of simplicity. In [8], the ϵ-deviation uncertainty neighborhood $P_\epsilon(W)$ is *parametrically* defined: the true parameters of the CDHMMs are assumed to lie within the neighborhood $\eta(\Lambda)$ of the pretrained models' parameters

$$\eta(\Lambda) = \{\Lambda | \pi_i = \pi_i^*,\ a_{ij} = a_{ij}^*,\ \omega_{ik} = \omega_{ik}^*,\ r_{ik} = r_{ik}^*, |m_{ikd} - m_{ikd}^*| \le C\, d^{-1} \rho^d,\ 1 \le i \le N,\ 1 \le k \le K,\ 1 \le d \le D\} \quad (11)$$

where constants C ($C > 0$) and ρ ($0 \le \rho \le 1$) are used to control respectively the possible mismatch *size* and *shape*, and $\{\pi_i^*, a_{ij}^*, m_{ikd}^*, r_{ik}^*\}$ denote the pre-trained model parameters. Thus, their quasiminimax rule is achieved as

$$\hat{W} = \arg\max_W [p(W) \cdot \max_{\Lambda \in \eta(\Lambda)} p(X|\Lambda, W)] \quad (12)$$

where $\hat{W}$ is the recognition result. In their implementation, in order to approximate the operation $\max_{\Lambda \epsilon \eta(\Lambda)} p(X|\Lambda, W)$ in (12), the following iterative procedure is used.

- Initialize Λ with the values obtained in the training phase.
- In each iteration, for an observation sequence X to be recognized, the optimal path of the corresponding unobserved state sequence s^* and the associated sequence of the unobserved mixture component labels l^* is first decoded using the Viterbi algorithm. Then the model parameter Λ is re-estimated based on s^*, l^* according to maximum likelihood (ML) criterion.
- If the new Λ falls in $\eta(\Lambda)$, it is used to update the old Λ; otherwise, the parameter within $\eta(\Lambda)$ which is closest to the new Λ is chosen.

In this paper, the above Merhav and Lee's implementation of quasiminimax is referred to as minimax1 for convenience.

Besides, another so-called modified minimax rule used in [2] works as follows:

$$\hat{W} = \arg\max_W [p(W) \cdot p(X|\Lambda_{MAP}, W)] \quad (13)$$

where

$$\Lambda_{MAP} = \arg\max_\Lambda p(X|\Lambda, W) \cdot p(\Lambda|\varphi, W) \quad (14)$$

with the prior p.d.f. $p(\Lambda|\varphi, W)$, where φ denotes the *hyperparameters*. This modified minimax method is referred to as minimax2 hereafter. In minimax2, the quasiminimax rule is realized under a Bayesian framework, where the least favorable parameters are acquired by the MAP estimate which is implemented by an iterative EM algorithm [2]. In the following experiments, we adopt one of the prior specification methods described in [3] to choose $p(\Lambda|\varphi, W)$ as the best normal approximation to the constrained uniform distribution within the neighborhood $\eta(\Lambda)$ in (11).

In both minimax1 and minimax2, instead of dynamically searching a desired answer in a structural network representation of all possible hypotheses, decisions are made only from a list of finite candidates. This makes them difficult to be extended to perform continuous speech recognition except in an N-Best rescoring mode.

IV. Minimax Search Algorithm for Robust Continuous Speech Recognition

In order to execute the minimax decision rule in robust continuous speech recognition, we combine the idea of the quasiminimax rule (minimax1) with the normal Viterbi search to derive a recursive minimax search algorithm as outlined in the introduction section. Our implementation of the quasiminimax rule can be represented as

$$\hat{W} = \arg\max_W \left[p(W) \cdot \max_{s,l} \max_{\Lambda \in \eta(\Lambda)} p(X, s, l|\Lambda, W) \right] \quad (15)$$

where s is the unobserved state sequence and l is the associated sequence of the unobserved mixture component labels corresponding to the observation sequence X. This recursive quasi-minimax search method is referred to as minimax3 in this paper.

Given a test utterance $X = (x_1, x_2, \cdots, x_T)$, CDHMM parameter Λ as well as its corresponding uncertainty neighborhood$\eta(\Lambda)$,[1] the recursive search algorithm to *approximately* achieve the minimax3 decision rule in (15) is described as follows.

1) Initialization ($t = 0$)

$$\alpha_0(i) = \pi_i \quad 1 \le i \le N \tag{16}$$
$$\psi_0(i) = 0 \quad 1 \le i \le N \tag{17}$$
$$\phi_0(i) = 0 \quad 1 \le i \le N \tag{18}$$

where $\alpha_t(i)$ denotes the score of the optimal partial path arriving at state i at the time instant t. The corresponding best partial path is represented by a chain of state points started from $\psi_t(i)$ and a chain of mixture component label points started from $\phi_t(i)$.

2) Recursion: for $1 \le t \le T$, $1 \le j \le N$, do

2.1) Path-merging in state j:

$$\alpha_t(j) = \max_{1 \le i \le N} [\alpha_{t-1}(i) \cdot a_{ij}] \tag{19}$$

$$\psi_t(j) = \arg \max_{1 \le i \le N} [\alpha_{t-1}(i) \cdot a_{ij}] \tag{20}$$

$$\phi_t(j) = \arg \max_{1 \le k \le K} \omega_{jk} \cdot \prod_{d=1}^{D} \sqrt{\frac{r_{jkd}}{2\pi}} e^{-(1/2) r_{jkd} (x_{td} - \check{m}_{jkd})^2} \tag{21}$$

where

$$\check{m}_{jkd} = \begin{cases} m_{jkd} - Cd^{-1}\rho^d & \text{if } x_{td} \le m_{jkd} - Cd^{-1}\rho^d \\ m_{jkd} & \text{if } m_{jkd} - Cd^{-1}\rho^d \le x_{td} \\ & \quad \le m_{jkd} + Cd^{-1}\rho^d \\ m_{jkd} + Cd^{-1}\rho^d & \text{if } x_{td} \ge m_{jkd} + Cd^{-1}\rho^d. \end{cases} \tag{22}$$

2.2) For each active partial path, estimate the least favorable parameters Λ^*:

$$\Lambda^* = \arg \max_{\Lambda \in \eta(\Lambda)} p(x_1, \cdots, x_t, s_{\psi_t(i)}, l_{\phi_t(i)} | \Lambda)$$

where $s_{\psi_t(i)}$ and $l_{\phi_t(i)}$ denote respectively the state sequence and the mixture component label sequence corresponding to the active optimal partial path backtracked from the points $\psi_t(i)$ and $\phi_t(i)$. When the neighborhood (11) is adopted, only the mean vectors are adjusted. Thus, all the mean vectors $m_{ik} (1 \le i \le N, 1 \le k \le K)$ of CDHMM are reestimated as follows:

[1]The neighborhood in (11) is still adopted for $\eta(\Lambda)$ here, in which only the uncertainty of mean vectors is taken into account.

if (the mixand m_{ik} is included in the partial path $\{s_{\psi_t(i)}, l_{\phi_t(i)}\}$), **then**

$$\overline{m}_{ikd} = \frac{\sum_{\tau=1}^{t} x_{\tau d} \delta\left(s_{\psi_t(i)}^{(\tau)} - i\right) \delta\left(l_{\phi_t(i)}^{(\tau)} - k\right)}{\sum_{\tau=1}^{t} \delta\left(s_{\psi_t(i)}^{(\tau)} - i\right) \delta\left(l_{\phi_t(i)}^{(\tau)} - k\right)} \quad (1 \le d \le D) \tag{23}$$

else

$$\overline{m}_{ikd} = m_{ikd} \qquad (1 \le d \le D) \tag{24}$$

where $s_{\psi_t(i)}^{(\tau)}$ and $l_{\phi_t(i)}^{(\tau)}$ denote respectively the state and mixture component labels corresponding to the time instant τ in the partial path backtracked from the points $\psi_t(i)$ and $\phi_t(i)$. Here $\delta(\cdot)$ denotes the Kronecher delta function.

Then the least favorable mean vectors are calculated as: (for all $1 \le i \le N$, $1 \le k \le K$ and $1 \le d \le D$)

$$m_{ikd}^* = \begin{cases} m_{ikd} - Cd^{-1}\rho^d & \text{if } \overline{m}_{ikd} \le m_{ikd} - Cd^{-1}\rho^d \\ \overline{m}_{ikd} & \text{if } m_{ikd} - Cd^{-1}\rho^d \le \overline{m}_{ikd} \\ & \quad \le m_{ikd} + Cd^{-1}\rho^d \\ m_{ikd} + Cd^{-1}\rho^d & \text{if } \overline{m}_{ikd} \ge m_{ikd} + Cd^{-1}\rho^d. \end{cases} \tag{25}$$

2.3) Rescore the partial path based on the updated least favorable parameters $\Lambda^* = (\{\pi_i\}, \{a_{ij}\}, \{\omega_{ik}\}, \{m_{ikd}^*\}, \{r_{ikd}\})$:

$$\alpha_t(j) = \pi_{s_{\psi_t(i)}^{(1)}} \cdot \prod_{\tau=1}^{t-1} a_{s_{\psi_t(i)}^{(\tau)} s_{\psi_t(i)}^{(\tau+1)}} \cdot \prod_{\tau=1}^{t} \omega_{s_{\psi_t(i)}^{(\tau)} l_{\phi_t(i)}^{(\tau)}} \cdot \prod_{d=1}^{D} \sqrt{\frac{r_{s_{\psi_t(i)}^{(\tau)} l_{\phi_t(i)}^{(\tau)} d}}{2\pi}} \cdot e^{-(1/2) r_{s_{\psi_t(i)}^{(\tau)} l_{\phi_t(i)}^{(\tau)} d} (x_{\tau d} - m^*_{s_{\psi_t(i)}^{(\tau)} l_{\phi_t(i)}^{(\tau)} d})^2}. \tag{26}$$

3) Termination

$$s_T^* = \arg \max_i \alpha_T(i) \tag{27}$$

4) Path Backtracking

$$s_t^* = \psi_{t+1}(s_{t+1}^*) \qquad t = T-1, T-2, \cdots, 1. \tag{28}$$

The final recognition result $\hat{W}$ can be derived from the optimal path $\{s_t^* | t = 1, 2, \cdots, T\}$.

Because of its intrinsic nature of recursive search, minimax3 can be easily extended to perform continuous speech recognition. In comparison with the normal Viterbi algorithm, minimax3 needs extra efforts to rescore each active partial path during the search process. However, if the size of the network to be examined is moderate, the increased computational cost is generally affordable.

V. Experiments

In order to examine the viability of the proposed minimax3 algorithm, we present a series of experiments where the minimax3

algorithm is compared with other existing methods. Firstly, in an isolated Japanese digit recognition task, minimax3 is compared with the Plug-in-MAP based Viterbi algorithm, minimax1 and minimax2. As a remark, only a Viterbi version of the minimax2 in [2] is implemented here. Next, in another connected word recognition task on TIDIGITS [7], minimax3 is compared with the conventional Viterbi search in terms of both the recognition accuracy and computational complexity. In all the experiments, the mismatch between training and testing conditions is caused by adding, at different SNR (signal-to-noise ratio) levels, computer-generated white Gaussian noise (AWGN) into the test data prior to the pre-processing stage. The AWGN is scaled to a fixed level for all utterances in the test set. The degree of mismatch is measured by SNR level (in terms of dB) of the contaminated speech, which is calculated over the whole testing set as follows:

$$\text{SNR} \triangleq 10\log_{10}\frac{\sum_{i\in\mathcal{S}}\sigma_s(i)}{\sum_{i\in\mathcal{S}}\sigma_n(i)} \tag{29}$$

where $\sigma_s(i)$ denotes the signal variance of the ith speech utterance in test set $\mathcal{S}$, and $\sigma_n(i)$ the variance of noise signal added to the ith utterance. Compared with other methods, the SNR values in this paper which are computed from (29) are relatively low because we summarize signal variances over the whole data set. Moreover, in the following experiments, no knowledge of the related mismatch is explicitly exploited in testing phase. In all of the following experiments, we do not perform cepstral mean normalization in either training or testing phase.

A. Isolated Digit Recognition: ATR-JPD

In order to compare the performance of minimax3 with other two previous quasiminimax methods (minimax1 and minimax2), we first perform a series of comparative experiments on a speaker-independent (SI) recognition task of isolated Japanese digits on the ATR-JPD database, which is selected from ATR Japanese Speech Database and contains isolated utterances of Japanese 0-9 digits from 60 speakers (half male, half female). The database ATR-JPD is recorded in a quiet environment at a sampling rate of 20 kHz with 16 bit quantization accuracy. Each digit is modeled by a left-to-right four-state CDHMM without state skipping and each state has 6 Gaussian mixture components with diagonal covariance matrices. Each feature vector consists of 16 LPC-derived cepstral coefficients, which are not warped to Mel scale. For each digit, in total, we have 56 tokens from 46 speakers for speaker-independent (SI) training, and 24 tokens from other 14 different speakers for SI testing.

In Table I, the averaged recognition accuracy of the minimax3 is compared with that of the standard plug-in-MAP-based Viterbi search algorithm, minimax1, and minimax2 at three SNR levels, 10 (dB), 20 (dB), and 30 (dB). The experimental results clearly show that all three quasiminimax algorithms are able to improve the robustness considerably in comparison with the standard Plug-in-MAP based Viterbi algorithm when the AWGN-caused mismatch exists between the training and testing conditions. We also note that minimax3 significantly outperforms both minimax1 and minimax2 in three examined SNR levels. This can be explained by the fact that the quasiminimax rule is repetitively applied during the recursive minimax3 search, which provides a chance to find a better path than both minimax1 and minimax2 in which the quasiminimax rule is only used to rescore the paths found by the normal Viterbi search. Next, we also see that the performance of minimax1 is better than that of minimax2. This mainly attributes to the prior difference in minimax1 and minimax2. In minimax1, a constrained uniform distribution is chosen while a normal approximation is used in our implementation of minimax2. The heavy tail of the normal distribution obviously degrades the performance 2%–3% in this case. Finally, we also note that, in the specific case here, the performance of all three quasiminimax methods is better than that of Viterbi method even in the matched condition (SNR $= \infty$).

TABLE I
PERFORMANCE (WORD ACCURACY IN PERCENT) COMPARISON OF MINIMAX3 WITH PLUG-IN-MAP, MINIMAX1 AND MINIMAX2 IN ISOLATED JAPANESE DIGIT RECOGNITION TASK WHEN TEST DATA ARE DISTORTED BY ADDITIVE GAUSSIAN WHITE NOISE. [THE NUMBERS IN THE PARENTHESES DENOTE THE OPTIMAL NEIGHBORHOOD PARAMETERS (C, ρ) FOR THE CORRESPONDING METHOD TO ACHIEVE THE SHOWN PERFORMANCE IN EACH CASE]

SNR	Plug-in MAP	minimax1	minimax2	minimax3
∞	98.50	99.58 (1,0.9)	99.58 (1,0.5)	99.58 (1,0.9)
30(dB)	62.08	73.33 (2,0.5)	71.67 (1,0.4)	77.50 (8,0.3)
20(dB)	26.10	57.92 (3,0.4)	53.33 (1,0.6)	61.67 (9,0.2)
10(dB)	5.42	28.33 (7,0.3)	26.25 (5,0.2)	33.33 (8,0.2)

We have to note that in Table I we only give the optimal performance for all three quasiminimax methods when the hyperparameters (C, ρ) are manually adjusted within the range: $C \in [1, 10]$ and $\rho \in [0.1, 0.9]$. As a reference, we also list in Table I the optimal choice of the hyperparameters (C, ρ) for the corresponding methods to achieve the shown performance. Besides the optimal performance in Table I, we also observed in our experiments that minimax1, minimax2 and minimax3 outperform the plug-in-MAP method for a wide range of (C, ρ). However, we have not found a good method yet to automatically adjust (C, ρ) for the optimal performance in all these minimax methods. We also find that the recognition performance of minimax3 method tends to be relatively insensitive to (C, ρ) in a quite wide range, as shown in Table II.

B. Connected Word Recognition: TIDIGITS

In order to examine the feasibility of the minimax3 in terms of its computational complexity in a continuous speech recognition task, we also perform a series of comparative experiments of SI connected digits recognition on TIDIGITS English connected digit-string database[7]. Only the part of adult speech data (111 men, 114 women) is used in the experiments. The feature vector consists of 12 LPC-derived cepstral coefficients, energy, and their delta features, which are also not warped to Mel scale. Because we are using the delta features in this part of experiments, the mean vector m_{ik} consists of static feature in the

TABLE II
RECOGNITION ACCURACY (IN PERCENT) AS A FUNCTION OF NEIGHBORHOOD PARAMETERS C AND ρ OF MINIMAX3 AT SNR = 30 dB

C \ ρ	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90
1.00	62.92	65.42	65.00	67.08	69.17	70.42	72.08	73.33	74.58
2.00	65.00	67.08	70.83	72.08	74.58	73.75	76.67	74.58	68.75
3.00	65.42	70.83	72.50	74.58	75.00	76.25	76.25	66.67	45.42
4.00	67.08	71.25	74.17	75.83	76.67	75.00	74.58	59.58	27.08
5.00	68.33	72.08	75.42	75.83	77.50	75.83	71.25	50.00	19.58
6.00	71.25	73.33	74.17	75.42	74.17	73.75	65.83	41.25	13.33
7.00	72.08	74.58	75.83	75.00	73.33	75.00	62.92	36.25	15.00
8.00	72.08	75.00	77.50	75.42	72.92	74.58	57.92	28.75	13.75
9.00	72.08	75.00	76.25	75.00	74.17	73.33	57.50	30.83	15.42
10.00	72.08	75.42	76.25	72.08	76.67	70.83	57.08	27.08	12.92

TABLE III
PERFORMANCE (IN PERCENT) COMPARISON OF MINIMAX3 WITH PLUG-IN-MAP METHOD ON TIDIGITS CORPUS WHEN TEST DATA ARE DISTORTED BY ADDITIVE GAUSSIAN WHITE NOISE, WHERE **STR** STANDS FOR *STRING CORRECT RATE*, **Wd-C** FOR *WORD CORRECT RATE*, **Wd-A** FOR *WORD ACCURACY*, **WER** FOR *WORD ERROR RATE*, **DEL**, **SUB** AND **INS** FOR *DELETION*, *SUBSTITUTION* AND *INSERTION* ERROR RATES, RESPECTIVELY

SNR		(C, ρ)	Str	Wd-C	Wd-A	WER	Del	Sub	Ins
∞	Plug-in-MAP	-	88.14	98.44	97.34	2.64	0.69	0.87	1.10
	minimax3	(0.1,0.1)	87.64	98.43	97.20	2.80	0.65	0.92	1.22
36.8	Plug-in-MAP	-	17.45	67.37	66.28	33.72	16.22	16.40	1.09
(dB)	minimax3	(2,0.3)	64.78	95.49	90.0	10.0	0.90	3.60	5.49
27.3	Plug-in-MAP	-	0.23	45.25	43.90	56.10	25.10	29.70	1.40
(dB)	minimax3	(2,0.4)	42.03	87.29	79.03	20.97	2.67	10.0	8.26
16.8	Plug-in-MAP	-	0.0	24.89	23.93	76.07	45.40	29.70	0.96
(dB)	minimax3	(5,0.3)	14.47	66.19	57.52	42.48	7.60	26.20	8.70

low dimensions and delta feature in the high dimensions. The uncertainty neighborhood of Λ defined in (11) will be slightly modified as follows:

$$\begin{aligned}\eta(\Lambda) = \{\Lambda \mid \pi_i = \pi_i^*,\ a_{ij} = a_{ij}^*,\ \omega_{ik} = \omega_{ik}^*,\ r_{ik} = r_{ik}^*,\\ |m_{ikd} - m_{ikd}^*| \le C\ d^{-1}\rho^d,\\ |m_{ik(D/2+d)} - m_{ik(D/2+d)}^*| \le C\, d^{-1}\rho^d,\\ 1 \le i \le N,\ 1 \le k \le K,\ 1 \le d \le D/2\} \qquad (30)\end{aligned}$$

where for $1 \le d \le D/2$, m_{ikd}'s correspond to the static feature part while $m_{ik(D/2+d)}$'s correspond to the delta feature part. The SI model for each digit is a ten-state, ten-mixture-per-state CDHMM. These whole digit HMMs are trained on 8623 utterances from adult training data subset of TIDIGITS. The algorithms are evaluated on 8700 utterances from adult test data subset distorted by various levels of computer-generated Gaussian white noises.

The experimental results in Table III show that the minimax3 also performs much better than the conventional Viterbi algorithm in continuous speech recognition task for the three examined SNR levels. In the table, we also give the corresponding values of (C, ρ) which minimax3 used. As far as the computational complexity is concerned, in minimax3, the increased computation mainly lies in 1) estimating the least favorable parameters as in eqs. (23) and (25) and 2) rescoring the partial path in (26). However, in each search step, only a small portion of the individual partial path need to be re-calculated while the most part of it remains unchanged as in (24). In the experiment, we observed that in this small vocabulary task where the recognition network is not very large, the calculation overhead of the minimax3 is affordable. As an example, we list in Table IV the total CPU time used by Viterbi and minimax3 full searches (i.e., the beam width is set to infinity) to recognize in total 300 utterances randomly chosen from the test set of TIDIGITS. The CPU time in Table IV shows that the computational complexity is approximately doubled in the minimax3 search in comparison with the normal Viterbi search. However, we have to note that the result of the quantitative comparison heavily depends on the size of the recognition network used by the search engine. The results in Table IV are obtained by using a recognition network, where all digit models are in parallel, with a silence model allowed preceding and succeeding the digit string.

TABLE IV
TOTAL RECOGNITION TIME (IN SECONDS) COMPARISON OF MINIMAX3 FULL SEARCH WITH PLUG-IN-MAP BASED VITERBI FULL SEARCH ALGORITHM FOR 300 UTTERANCES FROM TIDIGITS (ON A SUN ULTRA-I WORKSTATION)

	Viterbi search	minimax3 search
Total CPU time used (second)	771.94	1538.73

As a remark, in the above experiments on TIDIGITS, we have established the baseline system as simple as possible. The high recognition error rate (2.7% WER) in the baseline system is mainly attributed to the following reasons: we don't include the

delta-delta feature, we use only one model for each digit and do not adopt such as gender-dependent model, we also don't refine the system in many aspects including preprocessing and modeling due to the lack of CPU power, etc.

VI. Summary

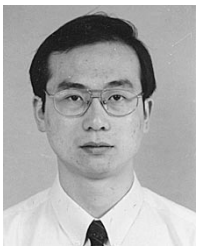
In this paper, we have proposed a novel minimax recursive search algorithm to perform quasiminimax decision rule in continuous speech recognition. In the algorithm, the quasiminimax rule is repetitively applied to determine the optimal partial paths during the search. This provides the minimax search algorithm an opportunity to find a better path for the correct hypothesis than the normal Viterbi search when the mismatch exist between the trained models and the testing speech. From the above experimental results, it is found that given an appropriate uncertainty neighborhood, the robustness of an ASR system can be enhanced by adopting the minimax decision rule. The proposed minimax search algorithm is shown to be effective and efficient for the examined small vocabulary tasks of either isolated words or continuous speech. As future works, we need to develop some methods to automatically determine the hyperparameters $(C,\ \rho)$ of the uncertainty neighborhood. We should also consider other possibility in uncertainty modeling such as the *distribution uncertainty* instead of the current practice of the *model parameter uncertainty*. It will also be interesting to see whether the minimax method can help improve the performance when it is combined with other mismatch compensation approaches.

Acknowledgment

The authors would like to thank Dr. C.-H. Lee at Bell Laboratories for his comments and discussions on this work.

References

[1] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.
[2] Q. Huo, H. Jiang, and C.-H. Lee, "A Bayesian predictive classification approach to robust speech recognition," in *Proc. Int. Conf. Acoustics, Speecg, Signal Processing '97*, Munich, Germany, Apr. 1997, pp. II-1547–1550.
[3] Q. Huo and C.-H. Lee, "A study of prior sensitivity for Bayesian predictive classification based robust speech recognition," in *Proc. Proc. Int. Conf. Acoustics, Speecg, Signal Processing '98*, Seattle, WA, May 1998, pp. II-741–744.
[4] H. Jiang, K. Hirose, and Q. Huo, "Robust speech recognition based on a Bayesian prediction approach," *IEEE Trans. Speech Audio Processing*, vol. 7, pp. 426–440, July 1999.
[5] H. Jiang, "A study on robust decision rules in automatic speech recognition," Ph.D. dissertation, Univ. Tokyo, Tokyo, Japan, June 1998.
[6] C.-H. Lee, "On stochastic feature and model compensation approaches to robust speech recognition," *Speech Commun.*, vol. 25, no. 1–3, pp. 29–47, 1998.
[7] R. G. Leonard, "A database for speaker-independent digit recognition," in *Proc. Proc. Int. Conf. Acoustics, Speecg, Signal Processing '84*, 1984, pp. 42.11.1–4.
[8] N. Merhav and C.-H. Lee, "A minimax classification approach with application to robust speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 1, no. 1, pp. 90–100, 1993.
[9] A. Nadas, "Optimal solution of a training problem in speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 1, pp. 326–329, 1985.
[10] B. D. Ripley, *Pattern Recognition and Neural Networks*. Cambridge, U.K.: Cambridge Univ. Press, 1996.



Hui Jiang (M'00) was born in Kunming, China, in 1970. He received the B.Eng. and M.Eng. degrees from the University of Science and Technology of China (USTC), Hefei, in July 1992 and December 1994, respectively, and the Ph.D. degree from the University of Tokyo, Tokyo, Japan, in September 1998, all in electrical engineering.

From 1992 to 1994, he worked on large vocabulary Chinese speech recognition at USTC. From October 1998 to April 1999, he joined a Japanese national project "Research on Man–Machine Dialogue System through Spoken Language." He worked on large vocabulary continuous speech recognition of Japanese at the University of Tokyo. Since April 1999, he has been with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, where he has been working on telephony speech recognition and speaker verification. His current research interests include all issues related to speech recognition and understanding, especially robust speech recognition, utterance verification, adaptive modeling of speech, dialogue system design, and speaker recognition/verification.

Keikichi Hirose (M'78) received the B.E. degree in electrical engineering in 1972, and the M.E. and Ph.D. degrees in electronic engineering in 1974 and 1977, respectively, from the University of Tokyo, Tokyo, Japan.

In 1977, he joined the Department of Electrical Engineering, University of Tokyo, as a Lecturer. He has been a Professor with the Department of Information and Communication Engineering, Graduate School of Engineering, University of Tokyo, since April 1994. In April 1999, he received a dual appointment as Professor in the University's Graduate School of Frontier Sciences. From March 1987 until January 1988, he was a Visiting Scientist of the Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge. His research interests include prosody, speech synthesis and recognition, and spoken language systems.

Dr. Hirose is a member of the Acoustical Society of America, the International Speech Communication Association, the Institute of Electronics, Information, and Communication Engineers, the Acoustical Society of Japan, the Information Processing Society of Japan, and other professional organizations.

Qiang Huo (M'95) received the B.Eng. degree from University of Science and Technology of China (USTC), Hefei, China, in 1987, the M.Eng. degree from Zhejiang University, Hangzhou, China, in 1989, and the Ph.D. degree from the USTC, in 1994, all in electrical engineering.

From 1986 to 1990, his research work focused on the hardware design and development for real-time digital signal processing, image processing and computer vision, and speech and speaker recognition. From 1991 to 1994, he was with the Department of Computer Science, University of Hong Kong (HKU), where he worked on speech recognition. From 1995 to 1997, he was with ATR Interpreting Telecommunications Research Laboratories, Kyoto, Japan, where he engaged in research in speech recognition. He rejoined the Department of Computer Science and Information Systems, HKU, in 1998 as an Assistant Professor. His current major research interests include speech and speaker recognition, computational model for spoken dialogue processing, Chinese character recognition, biometric authentication, adaptive signal modeling and processing, and general pattern recognition theory.