A Novel Prosodic-Information Synthesizer Based on Recurrent Fuzzy Neural Network for the Chinese TTS System

Chin-Teng Lin, *Senior Member, IEEE*, Rui-Cheng Wu, Jyh-Yeong Chang, and Sheng-Fu Liang

***Abstract*—In this paper, a new technique for the Chinese text-to-speech (TTS) system is proposed. Our major effort focuses on the prosodic information generation. New methodologies for constructing fuzzy rules in a prosodic model simulating human's pronouncing rules are developed. The proposed Recurrent Fuzzy Neural Network (RFNN) is a multilayer recurrent neural network (RNN) which integrates a Self-cOnstructing Neural Fuzzy Inference Network (SONFIN) into a recurrent connectionist structure. The RFNN can be functionally divided into two parts. The first part adopts the SONFIN as a prosodic model to explore the relationship between high-level linguistic features and prosodic information based on fuzzy inference rules. As compared to conventional neural networks, the SONFIN can always construct itself with an economic network size in high learning speed. The second part employs a five-layer network to generate all prosodic parameters by directly using the prosodic fuzzy rules inferred from the first part as well as other important features of syllables. The TTS system combined with the proposed method can behave not only sandhi rules but also the other prosodic phenomena existing in the traditional TTS systems. Moreover, the proposed scheme can even find out some new rules about prosodic phrase structure. The performance of the proposed RFNN-based prosodic model is verified by imbedding it into a Chinese TTS system with a Chinese monosyllable database based on the time-domain pitch synchronous overlap add (TD-PSOLA) method. Our experimental results show that the proposed RFNN can generate proper prosodic parameters including pitch means, pitch shapes, maximum energy levels, syllable duration, and pause duration. Some synthetic sounds are on-line available for demonstration.**

***Index Terms*—Chinese text-to-speech system, fuzzy inference engine, prosodic information, recurrent neural network, sandhi rules, speech synthesizer.**

I. Introduction

TEXT-TO-SPEECH system (TTS) is the automatic conversion of a text into speech that resembles a native speaker of the language reading the text. The potential applications of high-quality TTS systems are numerous, for example, telecommunications services, language education, aid to persons with disabilities, talking books and toys, vocal monitoring, human-machine communication, etc. An ideal text-to-speech synthesizer could mimic the pronunciation style of human beings in order to generate natural, and fluent speech for any input text. As human reading, the TTS system comprises a natural language processing (NLP) module, capable of producing a phonetic transcription of the next text to be read, together with the desired intonation and rhythm, and a digital signal processing (DSP) module, which transforms the symbolic information it receives into natural-sounding speech.

There are three procedures when the text-to-speech conversion is performed. The first is text analysis, or alternatively linguistic analysis. The task is to convert an input text, which is usually represented as a string of characters, into a linguistic representation. This linguistic representation is usually a complex structure that includes information about the word sequence, the part-of-speech (POS) tags [1], tonal properties, prosodic phase information, or any combinations of those information on the grammatical categories or pronunciation of words. The second procedure is to generate suitable prosodic parameters according to the linguistic representation. These prosodic parameters may be the fundamental frequency, duration, energy, and pause. The last procedure is the synthesis of a speech waveform [2] using the desired speech segments and prosodic parameters. A conventional approach used in the last procedure is to record a speech inventory that consists of all the basic units for the target language, and use a prosodic algorithm to modify and concatenate the units to generate output. In the followings, we discuss the major problems in most existing TTS systems.

1) Synthesis Approach is not Complete: Among existing synthetic techniques, the approach based on acoustic parameters can adjust both segmental and supra-segmental features of synthetic units flexibly and can be considered as the most reasonable synthetic technique in theory. However, the parameter-based synthesizer is over-dependent on the developments of parameter extracted methods, and the model of speech production is still unperfect; the intelligibility of synthetic speech does not satisfy the requirement of real applications. Therefore speech synthesis of the intelligibility and naturalness for limited vocabulary are raised. It can be used in some practical fields, such as talking toys. Since only the simple waveform concatenating techniques are used, once the waveforms of concatenating units are determined, they can't be changed afterwards, and the prosody of synthetic speech can't be adjusted according to different context. In this paper, the time-domain pitch-synchronous-overlap-add method [3]–[6], is used as the synthesis algorithm. It not only preserves the main segmental features of original waveforms, but also adjusts the pitch contour, the duration, and the intensity of the waveform of each concatenating unit before concatenation.

Manuscript received January 28, 2002; revised October 9, 2002. This paper was recommended by Associate Editor P. L. Willett.

The authors are with the Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: ctlin@fnn.cn.nctu.edu.tw).

Digital Object Identifier 10.1109/TSMCB.2003.811518

1083-4419/04$20.00 © 2004 IEEE

2) Knowledge of Natural Language is not Enough: Language processing is generally concerned with the attempt to recognize a large pattern (sentence) by decomposing it into small subpattern according to the rules that reduce entropy. But the language processing is a complicated and difficult task, and its practicability is weak. In most TTS systems, the homograph processing is too simple and the language processing is only to convert an input text into a linguistic representation.

3) Prosodic Information is not Complete: In the case of Mandarin, each character is pronounced as a base syllable (word). Only about 1300 phonetically distinguishable syllables comprise the set of all legal combinations of 411 base-syllables and five tones. Each syllable is composed of an optional consonant initial, a vowel final, and five tones. While these tone types have rather clear manifestation in the time-contour of fundamental frequency in the case of isolated syllables, they undergo various variations in continuous speech due to coarticulations. The tone contour of a syllable changes due to the influences of tones of adjacent syllables in a word or phrase. In most current TTS systems, limited rules are adopted to meet the prosodic information, but these rules are not enough for natural speech. The lack of synthesis rules is due to the lack of philological knowledge.

The first TTS system is proposed in 1986 for English. Since then, many other TTS systems have been proposed for various languages. Over the past years, TTS systems usually adopted the rule-based approach to generate prosodic information. In the rule-based methods [7]–[18], input text is analyzed firstly and then parsed based on some predefined lexicon to extract useful linguistic features. These features include word syllables, phonetic structures, syntactical structure, intonation patterns, and semantic interpretations, etc. Although some of them have high performance, it is still difficult to disclose the set of rules for synthesizing high-quality and natural synthetic speech. On the other hand, the effects of mutual interactions among different linguistic features are complex and hard to be analyzed.

A new approach, which uses a statistical model and linear regression methods to learn the rules from a large set of training data, was recently proposed [19]–[23]. These models are trained using large sets of real utterances and can automatically learn the phonological rules from the database and store in the weights of neural network and model parameters. Although some achievements have been reached by using this new approach, it is still far away from the goal of generating proper prosodic information to synthesize natural speech for unlimited texts.

As long as the neural network is extensively adopted in many different applications, especially in signal processing, Chen *et al.* proposed a RNN-based prosodic model used in their TTS system in 1996 [24], [25]. However, this RNN-based prosodic model lacks the explicitness of the hidden pronunciation states which interpret the relation between the prosodic information and the linguistic features of the input text. In other words, the phonological rules of tone modification cannot be explicitly extracted from the RNN-based prosodic model, although they may be implicitly learned and stored in the weights of the RNN. In most current TTS systems, limited rules are adopted to determine the prosodic information, but these rules are not enough for natural speech because of the lack of philological knowledge. This motivates us to focus our major efforts on the study of synthesizing proper prosodic information.

In this paper, a Recurrent Fuzzy Neural Network (RFNN) is developed as the prosodic model for Chinese speech synthesis. The RFNN integrates a Self-cOnstructing Neural Fuzzy Inference Network (SONFIN) [26] into a multilayer recurrent neural network (RNN) [27], and can properly explore the hidden pronunciation states which control the prosody information generation, based on interpreting the linguistic features of the input text properly. The proposed RFNN-based prosodic model mimics the experience of human experts as a speech knowledge-based system where the knowledge is stored as a set of fuzzy IF-THEN rules by forming a rule-based system [28]. Unlike other intelligent systems, the fuzzy inference model can transfer human knowledge or experience as fuzzy rules by implementing the mapping from its input feature to the output space and store the results in its knowledge base with the help of the learning ability of the neural network. When there comes fresh knowledge, the RFNN model can extend the knowledge base automatically without causing any essential structure changes within the knowledge base and expert model. When finding a relationship between speech and text, the RFNN system primarily selects the knowledge most closely related to the situation to be solved from the knowledge base and processes it by using the inference engine. The incorporation of SONFIN and RNN techniques is motivated by this conception and is used to design the complex prosodic model in which analytical technologies and expert knowledge are combined.

There are several advantages of our proposed RFNN-based prosodic model compared with the previously proposed rule-based and neural-network-based methods [29], [30]. First, our proposed approach provides a total solution to the problem of prosodic information generation. Second, the proposed RFNN-based prosodic model can generate accurate prosodic information by automatically inferring the fuzzy rules in SONFIN. Third, the start or the end of the prosodic phrase is not necessary to be defined. Finally, all prosodic parameters are automatically generated by the RFNN, since they are all embedded in the weights of the RFNN.

This paper is organized as follows. Section II is the introduction of general background for Chinese text-to-speech (TTS) system. The architectures of the SONFIN and RFNN are presented in Section III. In Section IV, experimental simulations are presented to evaluate performance of the RFNN-based prosodic synthesizer and a Chinese TTS system is implemented for the subjective listening test. Finally, the conclusions are summarized in Section V.

II. Chinese Text-to-Speech System

A Chinese TTS system basically has several phases such as text analysis, synthetic-units selection, prosodic-information generation, and speech synthesis as shown in Fig. 1.

1) Text Analysis Phase: Text analysis can be generally divided into several stages, such as word segmentation, syntactic parsing, semantic interpretation, etc. Ambiguities would probably occur in each stage, for instance, in looking up a lexicon, in syntactic parsing, and so on. These ambiguities

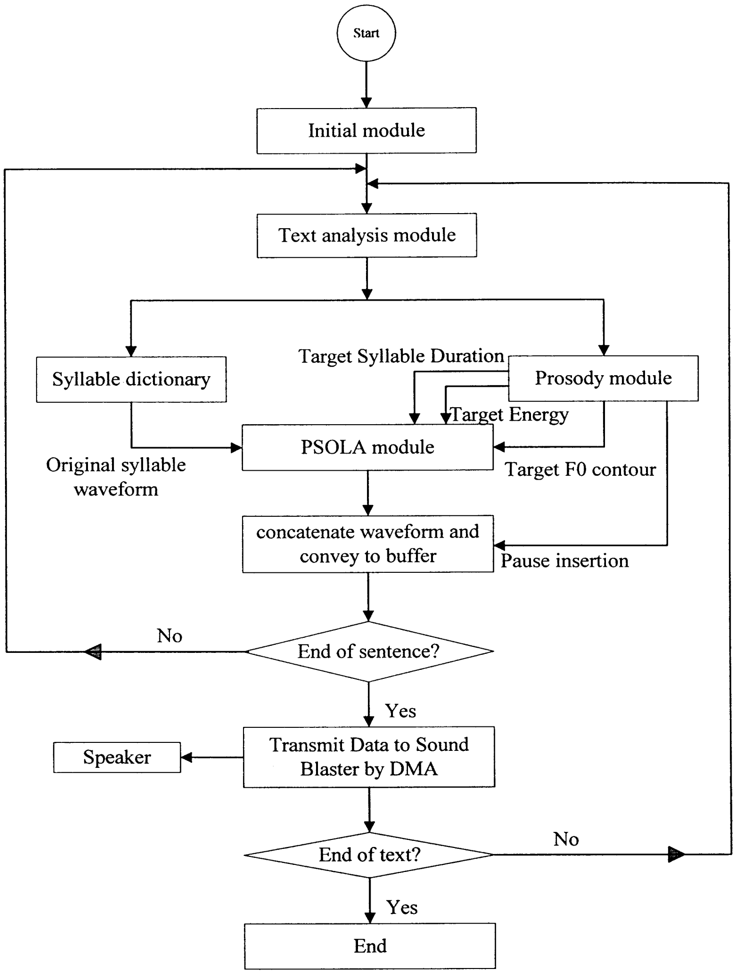


Fig. 1. Flowchart of a Chinese TTS system.

should be resolved in every stage in order to obtain feasible results during the process of Chinese sentences. In Chinese, the word, which is the smallest syntactically meaningful unit, consists of one to several syllables. The phrase is considered as a brief expression, sometimes a single word, but usually two or more words form an expression by themselves, or become a portion of a sentence. A Chinese sentence should be segmented into a sequence of words before syntactic analysis. There are several segmentations when dividing an input sentence into a sequence of words. For example, the sentence "hao3 ren2 cai2 sh4 nan2 de2 de5" (A talent man is rare) can be segmented as "hao3 ren2 cai2 sh4 nan2 de2 de5" or "hao3 ren2 cai2 sh4 nan2 de2 de5." This phenomenon is called as the word segmentation ambiguity.

In a Chinese lexicon, there may be multiple lexicon entries for a word due to multiple syntactic functions. That is, a word may have multiple POSs (part of speech). For example, "gong1 ji2" (attack) may play a verb role in "mei3 guo2 gong1 ji2 i1 la1 ke4" (American attack Iraq) or may play a noun role in "gong1 ji2 sh4 zui4 jia1 fang2 u4" (A attack is the best defense). We call this phenomenon as the lexical ambiguity. These ambiguities should be reduced before further processing of Chinese. Hence, a text analyzer plays an important role in TTS systems, because the segmentations and POSs of the sentence will influence the prosody of speech such as pitch contour, the duration of syllable or pause, and stress.

In this paper, we propose a process to resolve word segmentation ambiguity and lexicon ambiguity based on statistical language model as shown in Fig. 2. The steps for word segmentation and tagging by applying the dynamic programming algorithm consists of i) read the probability file including bi-gram probabilities; ii) read the input sentence; iii) scan the input sentence as well as look up the dictionary and construct a multistage graph; iv) apply the bi-gram Markov model in the multistage graph, and use dynamic programming to find the tagging path with the highest probability; v) output the tagging path with the highest probability; and vi) repeat Step ii to Step vi for all sentences in the input file.

2) Synthetic-Unit Selection Phase: Mandarin is a tone language. Each word is pronounced as a monosyllable according not only to its phonetic sign but also to its tonality. There are only five basic tones in Mandarin speech, namely Tone1, (high-level tone, with symbol "—"), Tone2 (mid-rising tone " / "), Tone3 (mid-falling-rising tone "√"), Tone4 (high-falling tone "\"), Tone5 (Neutral tone "·"). The information of the tonality of a word mainly appears on its pitch contour [31] so that we have only five basic shapes of pitch contour for Tone1 to Tone4 as shown in Fig. 3. The pitch contour of Tone5 is highly con-

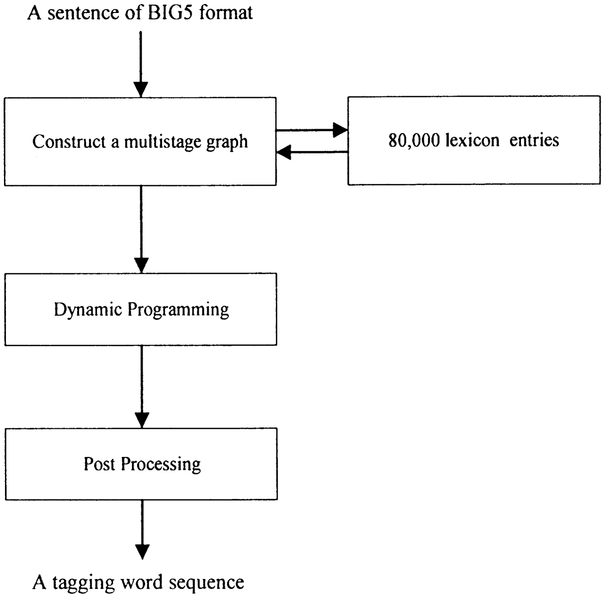


Fig. 2. Block diagram of the text analysis module.

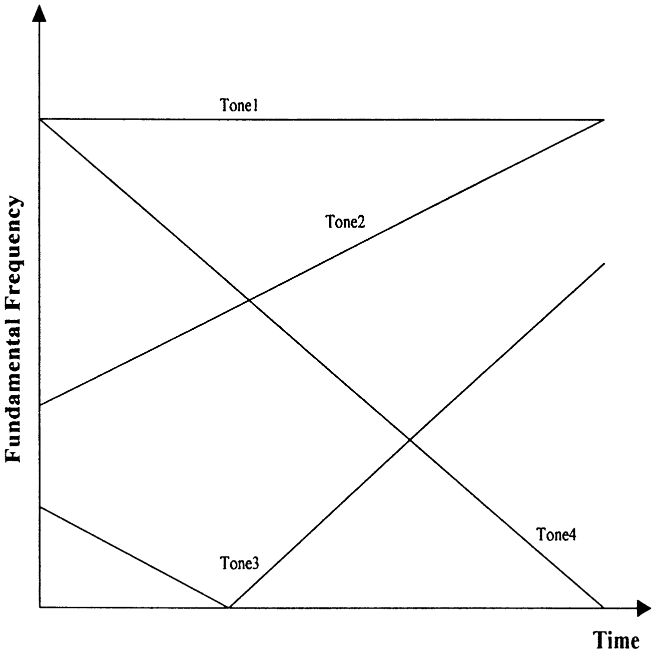


Fig. 3. Standard pitch contour of four tones.

text dependent and it is always pronounced short and light. After the processing of text analysis, the linguistic representation includes information on word pronunciation. In our system, the unit selection phase tends to choose the monosyllable from the speech corpus containing 411[1] monosyllables and to concatenate the synthesized speech.

3) Prosodic-Information Generation Phase: There are two main factors affecting the prosody information in Mandarin speech. One is low-level linguistic feature (word pronunciation mechanism), such as word phonetic structures, which is defined as the pronunciation of the syllable composed of a consonant, a vowel, and five phonetic signs. The other one is high-level linguistic feature (syntactic pronunciation mechanism), such as a syntactic boundary [32]. The term "prosody" refers to certain properties of the speech signal such as audible changes in pitch, loudness, and syllable length [33], [34]. The set of prosodic features also includes aspects related to speech timing such as rhythm (mostly determined by the timing of stressed syllables) and speech rate. Synthesis system has such a prosodic model in the sense that fundamental frequency, duration, and stress must be assigned in the production of speech.

A simple system for Mandarin might assign to each syllable a tone shape selected by the lexical tone, and assign constant duration and constant stress to each phone or just specifies the fundamental frequency (F0) contours of Mandarin lexicon tones. It was realized that even a few simple tonal rules can improve the smoothness of the synthesized speech.

The prosodic model in early Mandarin synthesis systems did not go much beyond this simple description. Later systems have fairly sophisticated models predicting variations of tone and duration suitable for the context. Modeling of expressive reading styles is even harder, especially since no TTS system can claim to truly understand what it is reading. Thus, important information such as when to emphasize a word and how much emphasis to put in, can't generally be reliably predicted. Only a few papers on Mandarin synthesis system report on duration and stress models [14], [15].

Among these relative few systems reported on duration modeling, several different approaches are taken. Some use handcrafted duration rules, while some derive duration and stress values from a labeled speech database, such as the researchers in Bell Labs and the Department of Communication Engineering of Chiao-Tung University (NCTUCE), Taiwan [24], [25]. The Bell-Lab model is a parametric mathematical model and the parameters are trained based on duration and stress values in the database. The NCTUCE system is based on neural networks. The results of these two systems are better than those of simple rule-based systems.

Many systems implemented the rules, including the neutral tone rules, tone sandhi rules,[2] the half Tone3 rules, and the half Tone4 rules. Some systems also include tone rules that are intended to capture the tendency for pitch drop as the sentence proceeds. For example, a Tone1 following a Tone3 is replaced by a variant of tone1 with lower pitch. The tone of "zong3" is Tone3 but it becomes Tone2 in "li3 zong3 tong3" (President Lee). Some systems are with the help of a full acoustic intonation model, which examines prosodic data at a higher level and accounts for F0 curves with a limited number of parameters, such that it is possible to perform a wide range of prosodic effects by changing the parameters [35].

4) Speech Synthesis Phase: Three modern approaches, called articulatory synthesis, formant synthesis, and concatenative synthesis, have a long history. The main difference between these three basic synthesis methods is the way in which the sets of transfer functions for an utterance are computed.

[1] The total number of phonologically allowed syllables in Mandarin speech is only about 1300, and there are only 411 monosyllables regardless of tones.

[2] The sandhi rule is defined as tones vary based on their context. These changes, and the rules with which they are associated, are called tone sandhi. The canonical sandhi of Mandarin relates primarily to Tone3.

TABLE I
COMPARISON OF TTS SYSTEMS [37]

Inst. or author	Synth. Meth.	Unit	Text Anal.	Prosody	date
Suen	VOTRAX	phoneme	no	5 tones	1976
Lee et al.	VOTRAX	phoneme	no	5 tones	1983
Huang et al.	LPC	pseudo-demi-syllable	no	4 tones	1983
Zhou et al.		pseudo-demi-syllable	no	7 tones	1984
Lin & Luo	LPC	toned phoneme	no	no	1985
Taiwan U.	LPC	syllable	no	rules	1985
KTH	formant		no	rules	1986
Bell Labs	LPC	diphone	no	rules	1987
Taiwan U.	formant	syllable	no	rules	1987
Tsinghua(Beijing)	LPC	demisyllable	no	4 tones	1987
Acad. of Soc. Sci.	formant	syllable	no	stat. model	1988
Qin & Hu	LPC	pseudo-demi-syllable	no	4 tones	1988
U.C London	formant	pseudo-demi-syllable	no	rules	1989
Matsushita	formant	pseudo-demi-syllable	no	model	1989
Telecom. Labs	LPC	syllable	yes	rules/stat.model	1989
Tsing Hua U.	hybrid	demisyllable	no	rules/stat.model	1991
Telecom. Labs	LPC	toned syllable	yes	stat.model	1991
Chiao Tung U.	PSOLA	syllable		stat.model	1992
Chiao Tung U.	LPC	syllable		stat.model	1992
Hong Kong U.	LPC	syllable	no	rules	1992
Xu et al.		word	yes	rules	1993
Bell Labs	LPC	diphone	yes	rules/stat.model	1994
Inst. Acoustics	formant		yes	rules	1994
Apple	Apple	diphone	yes	rules	1994
Tsinghua	PSOLA	toned syllable	yes	rules	1995
Inst. Acoustics	PSOLA	toned syllable	yes	rules	1995
Cheng Kung U.	CELP	toned syllable	yes	stat. model	1995

Articulatory synthesis attempts to model the articulators—the tongue, the lips, and so forth. But the difficulty is how to mimic human articulator motion and compute acoustic properties from vocal tract shapes.

Formant synthesis bypasses the difficulty of articulator synthesis by using a set of carefully designed rules to compute spectral properties and transfer function directly from the linguistic representation. The early well-known formant system is designed by Klatt in 1980 [36]. Designing a set of rules to do this is certainly easier than constructing an articulator model, but it is still a difficult task.

The simplest approach is the concatenative synthesis, since it takes real recorded speech, cuts it into segments, and concatenate these segments back together during synthesis. Some systems, such as the Bell-Lab system, use recorded speech coded by linear predictive coding (LPC), and the other systems perform the synthesis directly in the time domain (e.g., PSOLA) by storing and concatenating waveform segments. In general, the speech quality of these systems that use the time-domain method is higher than that of LPC-based systems. According to Table I, we can find that most Mandarin systems are concatenative, and time-domain (PSOLA) as well as frequency (LPC) domain coding schemes are equally used. Syllables are mostly used as the basic units among these systems.

III. RFNN-Based Prosodic Information Synthesizer

In this section, a recurrent fuzzy neural network (RFNN) architecture shown in Fig. 4 is proposed to perform the prosodic model. The prosody mechanism interprets the linguistic features including low-level lexical features such as the tone of a syllable as well as word phonetic structures, and high-level features such as a syntactic boundary. According to the above definition of the high-level and low-level linguistic features, we can divide the prosodic model into two parts. The upper part of RFNN is a self-constructing neural fuzzy inference network (SONFIN), which takes some high-level linguistic features in the sentence as its inputs. The lower part of RFNN is a multilayer recurrent neural network (MLRNN) which takes some additional low-level linguistic features as its inputs. The prosodic phrase structure of Chinese speech can be automatically obtained by training the RFNN through the use of a large set of real speech and the associated texts. The detailed descriptions of the SONFIN and MLRNN are given in Sections III-A and III-B, respectively.

A. Self-cOnstructing Neural Fuzzy Inference Network (SONFIN)

In our previous work [26], a neural fuzzy network architecture, called the self-constructing neural fuzzy inference network (SONFIN) as shown in Fig. 5 was proposed. The SONFIN is a general connectionist model of a fuzzy logic system, which can find its optimal structure and parameters automatically. Both the structure and parameter identification schemes are done simultaneously during on-line learning without any assignment of fuzzy rules in advance. The SONFIN can solve the dilemma between the number of rules and the number of consequent terms. The number of generated rules and membership functions can be small even it is applied to model a sophisticated system. The SONFIN can always construct itself with an economic network size, and the learning speed as well as the modeling ability is well appreciated. Comparing with other neural

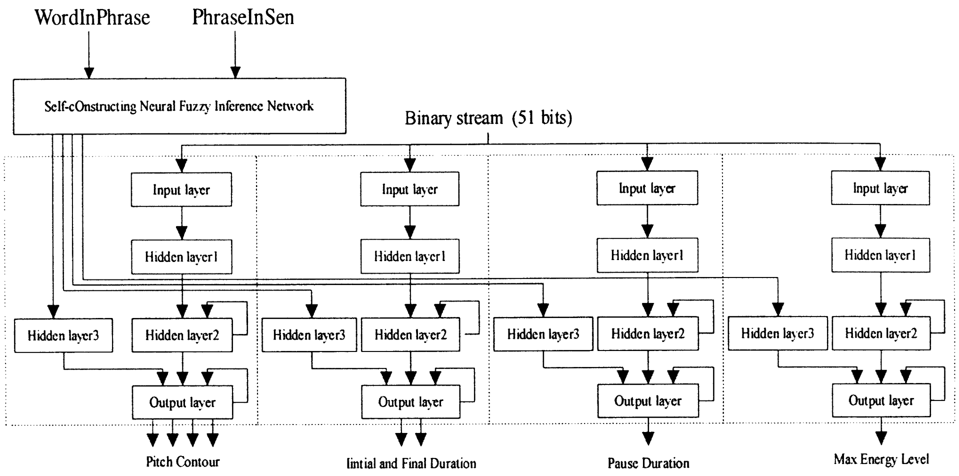


Fig. 4. Block diagram of the proposed RFNN-based prosodic model.

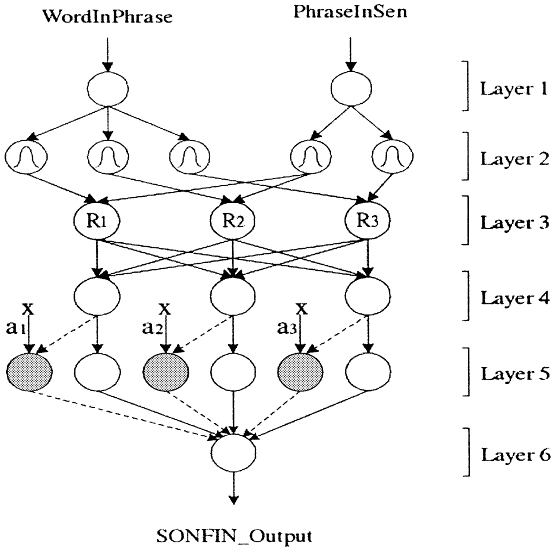


Fig. 5. Network structure of SONFIN in the proposed RFNN-based prosodic model.

networks [38]–[43], in different areas including control, communication, and signal processing, the on-line learning capability of the SONFIN has been demonstrated.

Fig. 5 shows the 6-layer structure of the SONFIN that realizes a fuzzy model of the following form:

$$\text{Rule}\, i\text{: IF}\, x_1 \text{ is } A_{i1} \text{ and } \cdots \text{ and } x_n \text{ is } A_{in},$$
$$\text{THEN } y \text{ is } s_{0i} + a_{ji}x_j + \ldots \quad (1)$$

where A_{ij} is a fuzzy set, s_{0i} is singleton of an output, and a_{ji} is a consequent parameter. It is noted that only the significant ones are used in the SONFIN; i.e., some a_{ij}s in the above fuzzy rules are zero. The following describes the functions of the nodes in each of the six layers of the SONFIN.

1) In Layer 1, each node corresponds to one input variable and only transmits input value to the next layer directly.
2) In Layer 2, each node corresponds to one linguistic label (small, large, etc.) of the input variables in Layer 1. In other words, the membership value which specifies to what degree an input value belongs to a fuzzy set is calculated in Layer 2.
3) In Layer 3, a node represents one fuzzy logic rule and performs precondition matching of the rule.
4) In Layer 4, the number of nodes is equal to that in Layer 3, and the result (firing strength) calculated in Layer 3 is normalized in this layer.
5) Layer 5 is called the consequent layer. Two types of nodes are used in this layer, and they are denoted as blank and shaded circles in Fig. 5, respectively. The node denoted by a blank circle (blank node) is the essential node representing a fuzzy set of the output variable. The shaded node is added only when necessary. One of the inputs fed to a shaded node is the output delivered from Layer 4, and the other possible inputs (terms) are the selected significant input variables from Layer 1. Combining these two types of nodes in Layer 5, the whole function of this layer performs as the linear equation on the THEN part of the fuzzy logic rule shown in (1).
6) In Layer 6, each node corresponds to one output variable. The node integrates all the actions performed by Layer 5 and acts as a defuzzifier to produce the final inferred output.

Firstly, we develop a novel on-line input space partitioning method, which is an aligned clustering-based approach by projecting the generated cluster onto each dimension of the input space to form a projected one-dimensional membership function for each input variable, and represent a cluster by the product of the projected membership functions. Basically, it aligns the clusters formed in the input space, so it reduces not only the number of rules but also the number of membership functions under a prespecified accuracy requirement.

This method creates only the significant membership functions on the universe of discourse of each input variable by using a fuzzy measure algorithm. It can thus generate necessary fuzzy rules from numerical data dynamically based upon orthogonal least square (OLS) method. The input membership functions are all tunable; a rule is considered to be necessary and is generated when it has a low overlapping degree with others.

After partitioning, two types of learning, structure and parameter learning, are used concurrently for constructing the SONFIN. The structure learning includes both the precondition and consequent structure identification of a fuzzy if-then rule. Here the precondition structure identification corresponds to the input-space partitioning and can be formulated as a combinational optimization problem with the following two objectives: to minimize the number of rules generated and to minimize the number of fuzzy sets on the universe of discourse of each input variable.

As to the consequent structure identification, the main task is to decide when to generate a new membership function for the output variable and which significant terms (input variables) should be added to the consequent part (a linear equation) when necessary. For the parameter learning based upon supervised learning algorithms, the parameters of the linear equations in the consequent parts are adjusted by either LMS or RLS algorithms and the parameters in the precondition part are adjusted by the backpropagation algorithm to minimize a given cost function.

After structure learning, the following parameter learning is performed on the whole network no matter whether the nodes (links) are newly added or are existent originally. The idea of backpropagation [44] is used for this supervised learning. Considering the single-output case for clarity, our goal is to minimize the error function

$$E = \tfrac{1}{2}\left(y(t) - y^d(t)\right)^2 \tag{2}$$

where $y^d(t)$ is the desired output, and $y(t)$ is the current output.

For each training data set, starting at the input nodes, a forward pass is used to compute the activity levels of all the nodes in the network to obtain the current output $y(t)$. Then starting at the output nodes, a backward pass is used to compute $\partial E/\partial w$ for all the hidden nodes layer by layer. Assuming that w is the adjustable parameter in a node (e.g., a_{ji}, m_{ij}, and σ_{ij} in the SONFIN), the general update rule used is

$$\Delta w \propto -\partial E/\partial w, \tag{3}$$

$$w(t+1) = w(t) + \eta(-\partial E/\partial w) \tag{4}$$

where η is the learning rate.

The SONFIN can be used for normal operation at any time during the learning process without repeated training on the input-output patterns when on-line operation is performed. There are no rules (i.e., no nodes in the network except the input-output nodes) in the SONFIN initially. They are created dynamically as learning proceeds upon receiving on-line incoming training data by performing the following learning processes simultaneously: 1) input/output space partitioning; 2) construction of fuzzy rules; 3) optimal consequent structure identification; 4) parameter identification. In the above, learning process 1), 2), and 3) belong to the structure learning phase and 4) belongs to the parameter learning phase. The details of these learning processes can be found in [26].

The TTS system with the proposed RFNN-based prosodic model converts text to speech sentence by sentence. After text analysis, a sentence is usually decomposed into several subsentences, and then these subsentences are decomposed into breathing groups, and finally these breathing groups are decomposed into words or phrases, where the word means a Chinese monosyllable word and the phrase means a Chinese syntactic word (may be composed of two or more monosyllable words). According to the acoustic study, the position of the word of one sentence affects the intonation of the word in a tonic language, and the position of the syllable of one word indicates the stress level of the word and reflects the rhythm [24], [45], [46]. Therefore, the position of the word in a phrase (*WordInPhrase*) and the position of the phrase in a sentence (*PhraseInSen*) are selected to be the two inputs of SONFIN in order to learn these prosodic phrase structure rules. Both the input variables *WordInPhrase* and *PhraseInSen* are restricted to the interval $[0, 1]$.

The sentence "lao3 wang2 jiong3 de5 hen4 bu4 de2 zuan1 dao4 di4 dong4 li3 qu4" is used as an example to illustrate how to calculate these two variables. After parsing, the sentence is divided into 9 phrases; they are "lao3 wang2," "jiong3," "de5," "hen4 bu4 de2," "zuan1," "dao4," "di4 dong4," "li3," and "qu4." The *PhraseInSen* of the phrase "di4 dong4" is calculated as $7/9 = 0.77$, and *WordInPhrase* of the word "di4" is $1/2 = 0.5$. In our method, the maximum word length in a phrase (*WordInPhrase*) is ten after parsing in the text analysis.

The effects of these prosodic phrase structure rules can be easily illustrated by graphics since the number of the input dimensions is two. After training, the membership functions in layer 2 of these two inputs *WordInPhrase* and *PhraseInSen* are shown in Fig. 6. The total amount of the generated output clusters is 7. The final assignment distribution of fuzzy rules is shown in Fig. 7. These fuzzy rules can be described by the following IF-THEN fuzzy rules:

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } A \text{ and } (\mathit{PhraseInSen}) \text{ is } X\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C1\end{aligned} \tag{5}$$

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } B \text{ and } (\mathit{PhraseInSen}) \text{ is } Y\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C2\end{aligned} \tag{6}$$

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } B \text{ and } (\mathit{PhraseInSen}) \text{ is } Z\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C3\end{aligned} \tag{7}$$

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } C \text{ and } (\mathit{PhraseInSen}) \text{ is } X\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C4\end{aligned} \tag{8}$$

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } C \text{ and } (\mathit{PhraseInSen}) \text{ is } Z\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C5\end{aligned} \tag{9}$$

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } D \text{ and } (\mathit{PhraseInSen}) \text{ is } Y\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C6\end{aligned} \tag{10}$$

$$\begin{aligned}&\text{IF } (\mathit{WordInPhrase}) \text{ is } D \text{ and } (\mathit{PhraseInSen}) \text{ is } Z\\ &\quad \text{THEN } (\mathit{cluster}) \text{ is } C7\end{aligned} \tag{11}$$

where $A{\sim}D$ means the position of the input variable $\mathit{WordInPhrase}$ relative to the corresponding phrases, $X{\sim}Z$ means the position of the input variable $\mathit{PhraseInSen}$ relative to the corresponding sentences. Fig. 8 shows the output distribution of the two-input SONFIN. Seven clusters ($C1{\sim}C7$) are generated by SONFIN after training. That is, the strength of the intonation of a word is classified into 7 groups representing 7 different strengthen degrees, respectively, and the prosody information of each group (C_i) corresponds to a fuzzy singleton

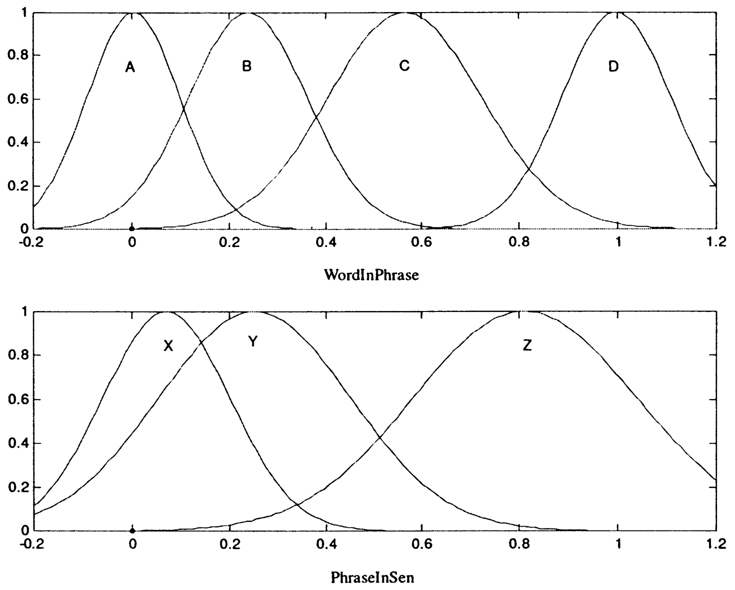


Fig. 6. Distribution of the learned membership functions in the two-input SONFIN.

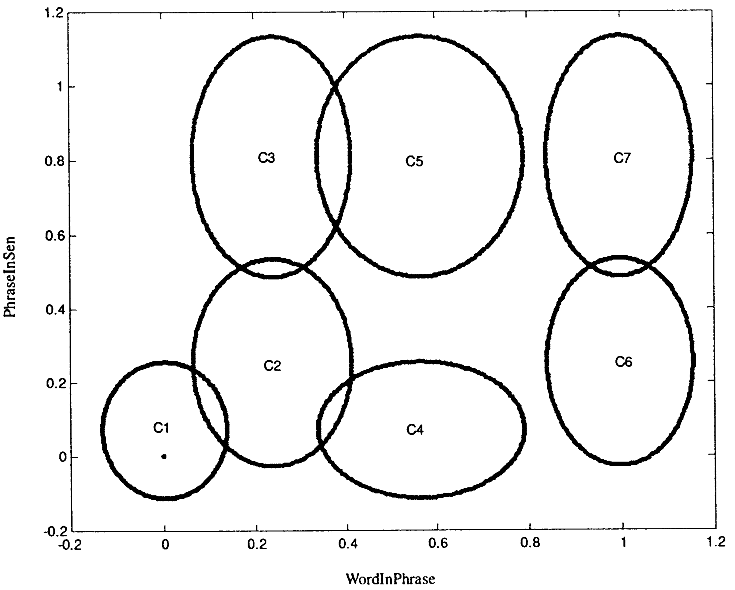


Fig. 7. Final assignment of rules in the two-input SONFIN.

output (O_i) in the consequent part of fuzzy inference system. The fuzzy singleton output O_i is a crisp value that is learned and generated without assigning any thresholds before/after training. A crisp output value of fuzzy system is usually defuzzified by taking the centroid or center of gravity according to fuzzy inference theory. In this paper, we use center of gravity method for defuzzification of SONFIN as shown in (12) at the bottom of the page. Then these outputs were fed to the RFNN for advanced learning.

B. Multilayer Recurrent Neural Network

The architecture of the multilayer recurrent neural network (RNN) in the RFNN is shown in Fig. 9. Some important parameters in the architecture of this RNN are as follows.

$$\text{Output} = \frac{\sum_i (\text{fuzzy output}_i)(\text{fuzzy singleton position of output fuzzy set})}{\sum_i (\text{fuzzy output}_i)} \tag{12}$$

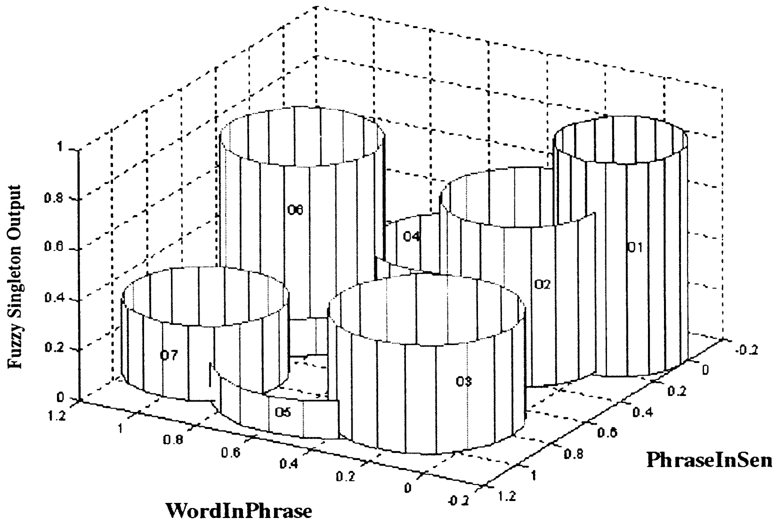


Fig. 8. Output function of the two-input SONFIN.

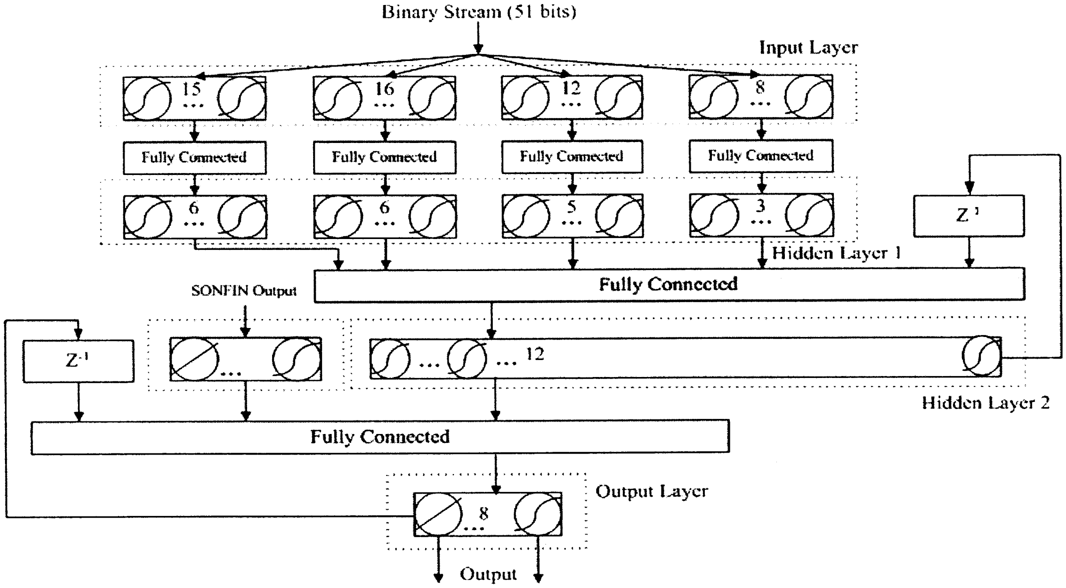


Fig. 9. Architecture of the multilayer recurrent neural network in the proposed RFNN-based prosodic model.

1) The number of nodes in the input layer is 51.
2) The number of nodes in hidden layer 1 is totally 20, which are divided into four groups with 6, 6, 5, and 3 nodes, respectively, as shown in Fig. 9.
3) The number of nodes in hidden layer 2 is 12.
4) The number of nodes in the output layer is 8, where one for the maximum energy level, one for pause duration, 2 for syllable duration, and 4 for pitch contour.
5) The activation functions of nodes in all layers are sigmoidal functions except the input layer.
6) The activation functions of nodes in the output layer are bisigmoidal functions.
7) The nodes in the hidden and output layers are feedback to themselves at the next time step.

The RFNN can be trained by the back-propagation through time (BPTT) algorithm with a large set of utterances of real speech. The BPTT algorithm can be regarded as an extension of the standard back-propagation algorithm. It can be derived by unfolding the temporal operation of the network into a multilayer feedforward network. The number of layers in the unfolded structure grows by one at each time step. The detailed description of the BPTT algorithm can be found in [44], [47]–[50].

The input of the RFNN is the linguistic symbols extracted from the database through the use of a text analysis model which contains a 80 000-lexicon database and a Markov probability model trained by a 2 300 000-word database with bigram probability and lexical probability [51], [52]. The linguistic input symbols are listed in Table II. Each linguistic input symbol is encoded into 51 binary digits arranged as {Current Tone} {Previous Tone} {Last Tone} {Current Vowel} {Current Consonant} {Last Consonant} {Current Punctuation}. We first encode each input symbol according to Table II.

TABLE II
REPRESENTATION OF LINGUISTIC INPUT SYMBOLS FOR (a) TONE, (b) VOWEL, (c) CONSONANT, AND (d) PUNCTUATION

Values	Tone Symbols
1	Tone 1
2	Tone 2
3	Tone 3
4	Tone 4
5	Neutral

(a)

Values	Vowel Symbols	Values	Vowel Symbols
1	a,ia,ua	9	en,in,un
2	o,e,uo	10	ang,iang,uang
3	e,ie	11	eng,ing,ong,iong
4	ai,uai	12	i
5	ei,ui	13	u
6	ao,iao	14	yu
7	ou,iu	15	er
8	an,ian,uan	16	Others

(b)

Values	Consonant Symbols
1	m,n,l,r,Others
2	h,x,sh
3	b,d,g
4	j,zh,z
5	p,t,k
6	q,ch,c,f,s

(c)

Values	Punctuation Symbols
0	Not a punctuation
1	,
2	.
3	:
4	!
5	[,], (,), Others
6	;
7	‘
8	?

(d)

The phonemes of each input symbol are then grouped by using Table III according to their properties relative to the time-domain waveform, frequency characteristics, manner of articulation, place of articulation, type of excitation, and so on. After encoding, binary codewords can be obtained by using the bit allocation table listed in Table IV. For example, the codeword of “yi2” in “jiao4 yi2 ke4 rou4 si1 mian4” is “01000/00010/00010/0000000000010000/100000/000010/00000000.”

The desired outputs of the RFNN are the manually extracted prosodic parameters corresponding to each of linguistic input symbols including pitch contour, energy level, syllable duration, and pause duration. The pitch contour of each syllable can be represented by a smooth curve formed through orthonormal polynomial expansion with coefficients up to the third order [53]. The pitch contour is further divided into pitch mean and pitch shape, where the pitch mean is defined as zero-order coefficient and pitch shape is defined as the other three coefficients. The syllable duration is also divided into initial duration and final duration that denote the consonant and the vowel durations of a syllable, respectively.

IV. EXPERIMENTAL RESULTS

Performance of the proposed RFNN prosody model can be examined through the simulation of the TTS system. The flowchart of the system is shown in Fig. 1. The system starts from the text analysis module. After that, the corresponding parameters are fed to RFNN prosodic module. Once obtaining the prosodic information, the speech synthesis module uses the time-domain pitch synchronous overlap add (TD-PSOLA) method [4]–[6], [45] to synthesize the syllable waveform based on a waveform dictionary containing 408 monosyllables. Finally, the TTS system concatenates the synthetic waveforms and inserts pause duration in proper position between two continuous syllables, and then delivers the concatenated synthetic waveforms to the playback device.

The database used in this experiment contains 35 242 Chinese syllables provided by the Telecommunication Laboratories, NCTU, Taiwan. The database is divided into two independent parts: 28 191 syllables for training and 7051 syllables for testing. The texts in the database are all news selected from a large news corpus to cover a variety of subjects including business (12.5%), medicine (12%), social events (12%), sports (10.5%), literature (9%), computers (8%), food and nutrition (8%), movies (6.5%), family life (6.5%), tours (6%), politics (2.5%), traffic and transportation (2.5%), etc. A male speaker generated all utterances. They were all spoken naturally at a speed of 3.5 to 4.5 syllables/s.

The learning process of the proposed TTS system consists of two individual parts. At beginning, the SONFIN is trained alone to learn the fuzzy rules and structures based on supervised learning with the ⟨input, output⟩ pair of two inputs and one stress level output. Then the resulting parameters are fixed and SONFIN is integrated to the RFNN to perform the prosody model. The off-line training process of RFNN converged approximately 40 training epochs using greatest decent algorithm with the mean square error functions. It took about 12 h run on our Acer ultra-station workstation. The on-line operation of the whole TTS requires: 1) Pentium 75 MHz, 2) Windows 95/98 or NT 4.0, 3) Hard disk with a minimum of 20 MB of storage, 4) 16 MB of RAM, and 5) SoundBlaster 16 or compatible sound card.

The experimental results of the trained RFNN-based prosodic model are firstly evaluated in Section IV-A. Two checks on the tone concatenation prosodic rules and fuzzy inference rules for prosodic phase structure learned by the proposed RFNN are given in Sections IV-B and IV-C, respectively. Finally, we provide a subjective listening test to verify the naturalness and fluency of the synthetic speech on our web sites [59], [60].

A. Evaluation Results of the Trained RFNN-Based Prosodic Model

The average root-mean-square errors (RMSEs) per frame (frame length: 22 ms) of the synthesis prosodic parameters by using the RFNN-based prosodic model are listed in Table V, where “Training” means that the training database is used, and “Testing” means that the testing database is used. We also list the RMSEs obtained in [24] for comparison. In order to further verify the performance of our proposed prosodic model, each

TABLE III
CLASSIFICATION OF CHINESE PHONEMES FOR (a) CONSONANT AND (b) VOWEL

(a) Consonant

Diction	Clear	Aspiration	Impediment					
			Lip	Tooth	Gum	Cacuminal	Jaw	Softpalate
Explosive	Yes	No	b		d			g
		Yes	p		t			k
Nasal	No		m		n			
Fricative	Yes		f	s		sh	x	h
	No					r		
Semivowel	No				l			
Affricative	Yes	No		zi		zh	j	
		Yes		c		ch	q	

(b) Vowel

Diction	Single Vowel	Bivowel	Vowel with nasal
Open Mouth	a,o,e	ai,ei	an,en
	er,e	ao	ang,eng
Ordered Tooth	i	ia,ie	ian,in
		iao,iu	iang,ing
		io	
Close Mouth	u	ua,uo	uan,un
		uai,ui	uang,ong
Pout Mouth	yu	ue	yuan,yun
			iong

TABLE IV
AN ENCODING SCHEME FOR INPUT LINGUISTIC SYMBOLS

Linguistic Symbols	Bit location	Number of Bits
Current Tone	0∼4	5
Previous Tone	5∼9	5
Last Tone	10∼14	5
Current Vowel	15∼30	16
Current Consonant	31∼36	6
Last Consonant	37∼42	6
Current Punctuation	43∼50	8

TABLE V
AVERAGE RMSEs PER FRAME OF PROSODIC INFORMATION GENERATED BY THE RFNN-BASED PROSODIC MODEL WITH A TWO-INPUT SONFIN AND THE RNN-BASED MODEL [24]. (FRAME LENGTH: 22 ms)

	RFNN		RNN	
	"Training"	"Testing"	"Training"	"Testing"
Pitch Contour	0.9ms	1.09ms	0.84ms	1.09ms
Energy Level	4.25dB	4.39dB	3.39dB	4.17dB
Initial Duration	20.21ms	20.86ms	17.20ms	18.5ms
Final Duration	35.03ms	37.17ms	33.30ms	36.7ms
Pause Duration	43.52ms	44.33ms	23.70ms	54.5ms

prosodic parameter generated by the RFNN-based prosodic model is discussed below.

The average RMSEs of the synthetic pitch contour for the training and testing databases are 0.86 ms and 1.06 ms per frame, respectively. Since the influence of local linguistic features on the pitch contour of a syllable in Chinese speech is greater than that of global linguistic features, global linguistic features and local linguistic features of a syllable should be considered separately. This implies that the pitch mean and the pitch shape of pitch contour belonging to global linguistic features or local linguistic features should be separately considered. Some experimental results of the pitch shapes and pitch means in the testing database are shown in Figs. 10 and 11, respectively. These two figures show that the trajectories of the synthetic pitch shape and pitch mean are quite close to their original counterparts of most syllables.

The RMSEs of the synthetic energy levels are 3.96 dB and 4.09 dB for the training and testing databases, respectively. The synthetic energy levels are very similar to the original counterparts of most syllables as shown in Fig. 12(a). Because the initial duration is very relevant to the final duration, they are simultaneously trained by using the same input text.

The RMSEs of the synthetic initial and final durations are (19.81 ms, 20.26 ms), and (34.38 ms, 36.30 ms) corresponding to the training and testing databases, respectively. Fig. 12(b) and (c) display the synthetic initial and final durations of syllables. These two figures show that the trajectories of the synthetic syllable durations are also very close to their original counterparts of most syllables.

However, the synthetic result of pause duration is not as good as that of the previous parameters, since the RMSEs of the pause duration are 42.22 ms and 44.79 ms for the training and testing databases, respectively. The reason is that the pause duration in the training database varies greatly in different speaking conditions. For example, the pause duration between the end of one sentence and the start of the next sentence is very long, whereas the pause duration between two successive syllables in a sentence is rather short. Fig. 12(d) shows the trajectory of the synthetic pause duration. In addition, the RFNN-based prosodic model also cannot track the trajectory of the pause duration very well when the pause duration varies too quickly as shown in Fig. 12(d). This situation often happens at the end of a sentence. That is, the trajectory of the synthetic pause duration between the end of a sentence and the start of the next sentence cannot be tracked very well by the RFNN-based prosodic model. However, the pause duration between the end of a sentence and the start of the next sentence does not influence intonation very much. In other words, the synthetic speech with

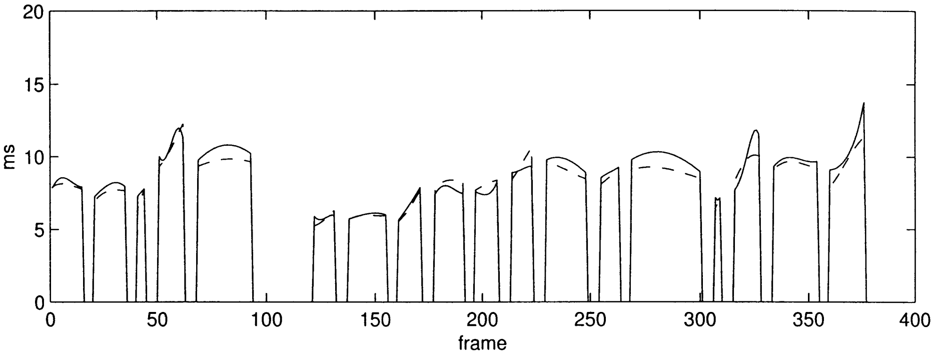


Fig. 10. Simulation results of pitch shapes in the "Testing" database, where solid and dashed lines correspond to the original and synthetic pitch contours, respectively.

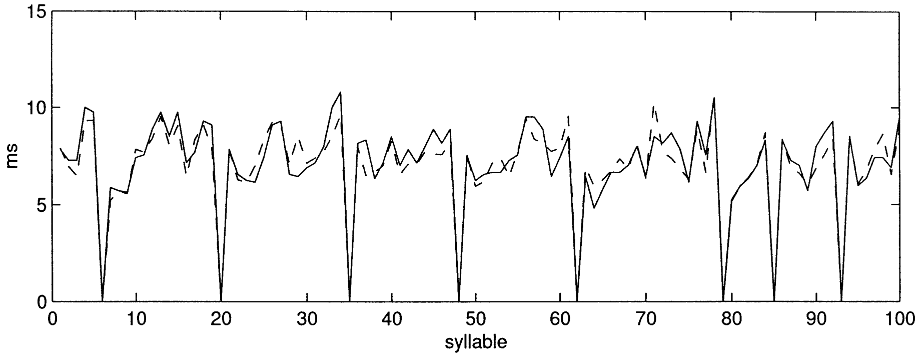


Fig. 11. Simulation results of pitch means in the "Testing" database, where solid and dashed lines correspond to the original and synthetic pitch means, respectively.

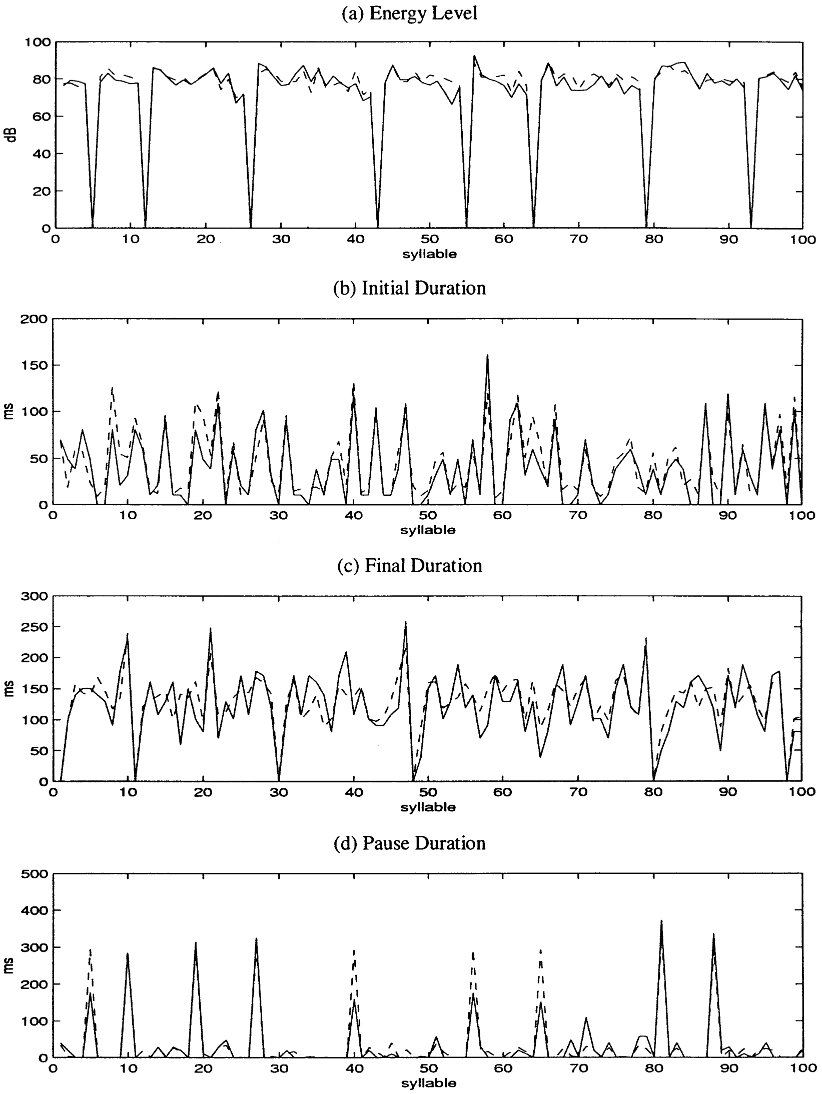
large mismatched pause duration does not cause terrible listening effects.

B. Check of the Learned Tone Concatenation Prosodic Rules

In our experiment, the shapes of the synthetic pitch contours for bisyllabic words with two "Tone 3" tonalities look like the standard patterns of "Tone 2" and "Tone 3," respectively. This result shows the famous sandhi rule of changing a "Tone 3" to a "Tone 2" when "Tone 3" is followed by a "Tone 3" [10]. The followings show the sandhi rules that are correctly learned by the RFNN. Fig. 13 shows three cases of the pitch contours for (a) "jiu3 dian3" (Nine o'clock), (b) "zong3 tong3" (President), and (c) "hao3 ji3 ba3 xiao3 yu3 san3" (Several small umbrellas). The pitch contour of "jiu3 dian3" is not a "Tone 3 + Tone 3" pattern any more, but is a "Tone 2 + Tone 3" pattern as shown in Fig. 13(a). This shows that the pitch contour of "Tone 3 + Tone 3" pattern is influenced by its adjacent words. The pattern "zong3 tong3" (President) also presents the same sandhi rule as shown in Fig. 13(b). The sandhi rule is not recursively applied to the sentence "hao3 ji3 ba3 xiao3 yu3 san3" (Several small umbrellas), where all morphemes are of "Tone 3," as shown in Fig. 13(c). This is because syntactic boundaries within a sentence act like barriers. That is, the sandhi rule is applied to morphemes within syntactic categories only when the preceding syntactic category consists of only one monosyllable word.

The above experimental results confirm that the sandhi rule for "Tone 3" has been automatically learned and memorized in the RFNN-based prosodic model. Five other tone concatenation rules [10]–[13], [54] of pitch contour modification for Chinese speech learned by the RFNN-based prosodic model are also examined as follows.

1) $4 \rightarrow 4'/\_\_4$: When a "Tone 4" precedes another "Tone 4" without any pause between them, the first Tone 4 will be modified such that the slope of the pitch contour will be decreased by an order of about 20%. An example is shown in Fig. 14(a), where the two syllables "xing4 yun4" (Lucky) both are Tone 4, and the difference in the slopes between the two contours is quite clear.
2) $3 \rightarrow 3'/4\_\_$: When a "Tone 3" follows a "Tone 4," the "Tone 3" will be modified such that the entire pitch contour slightly shifts downward to make a continuous contour connecting the preceding syllable. An example is shown in Fig. 14(b), where the two syllables "hao4 ma3" (Number) have a continuous pitch contour which is caused by a shift of the pitch contour of the second syllable.
3) $1 \rightarrow 1'/\{3,\ 4\}\_\_$: When a "Tone 1" follows a "Tone 3" or "Tone 4," the pitch level of the "Tone 1" should be decreased by an order of about 30%. An example is shown in Fig. 14(c), where the first and third syllables "wu1" (House) and "fong1" (Wind) both are "Tone 1," but their pitch levels are different. The reason is that the third syllable "fong1" following a "Tone 4" syllable "yi4" (Wing) causes a slight decrease in pitch level of the following "Tone 1."



(a) Energy Level

(b) Initial Duration

(c) Final Duration

(d) Pause Duration

Fig. 12. Simulation results of energy levels in the "Testing" database: (a) energy level, (b) initial duration, (c) final duration, and (d) pause duration, where solid and dashed lines correspond to the original and synthetic signal, respectively.

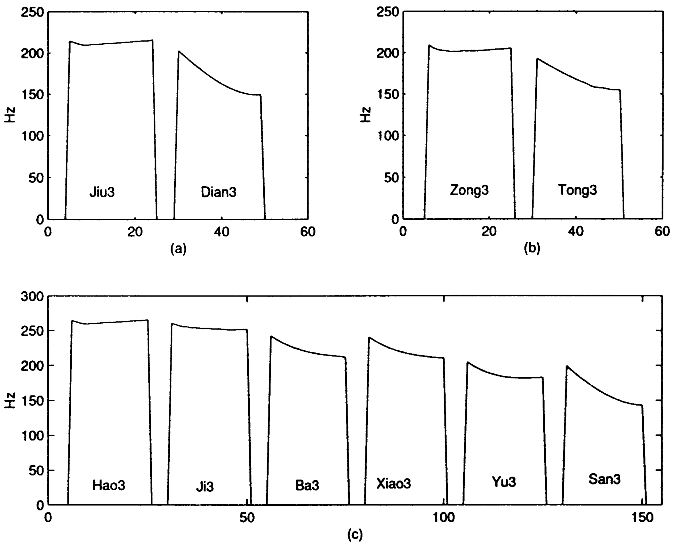


Fig. 13. Simulation results of pitch contours for (a) Jiu-3 Dian-3, (b) Zong-3 Tong-3, and (c) Hao-3 Ji-4 Ba-3 Xiao-3 Yu-3 San-3.

4) $1 \rightarrow 1'/1'\_$: When a "Tone 1" follows another "Tone 1," any modification made on the first syllable will be naturally repeated for the second one. An example is shown in Fig. 14(d). The pitch level of the last two syllables (both of them are "Tone 1"), is lower than that of the first syllable (also "Tone 1"), since the last two syllables "san1" (Three) and "yi1" (One) follow "er4" (Two) which is "Tone 4," so "yi1" (One) is shifted according to the previous rule 3) and "yi1" is then modified accordingly.

C. Check of the Learned Fuzzy Rules for Prosodic Phrase Structure

In order to verify the performance of the learned fuzzy rules, three well-known rules of the prosodic phrase structure, which are explored and collected by the linguists due to the habits in Chinese society [55]–[58] are used for testing. These three rules are listed below.

Prosodic-Phase-Rule (a): For bisyllabic words, the stress usually falls on the second syllable.

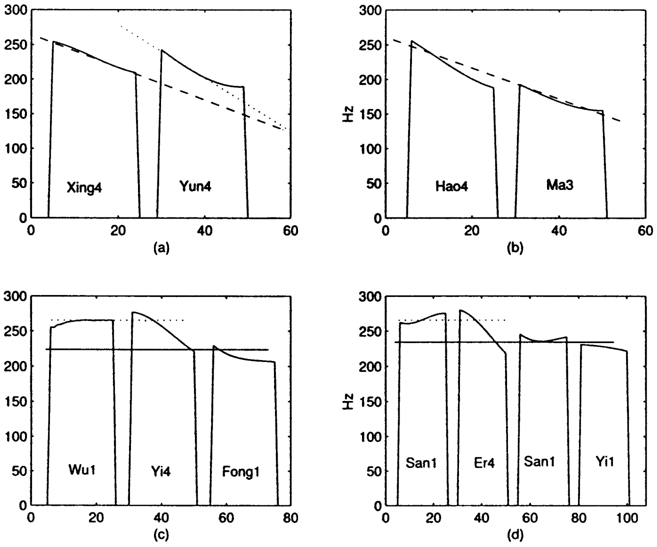


Fig. 14. Simulation results of pitch concatenation rules for (a) Xing-4 Yun-4, (b) Hao-4 Ma-3, (c) Wu-1 Yi-4 Fong-1, and (d) San-1 Er-4 San-1 Yi-1.

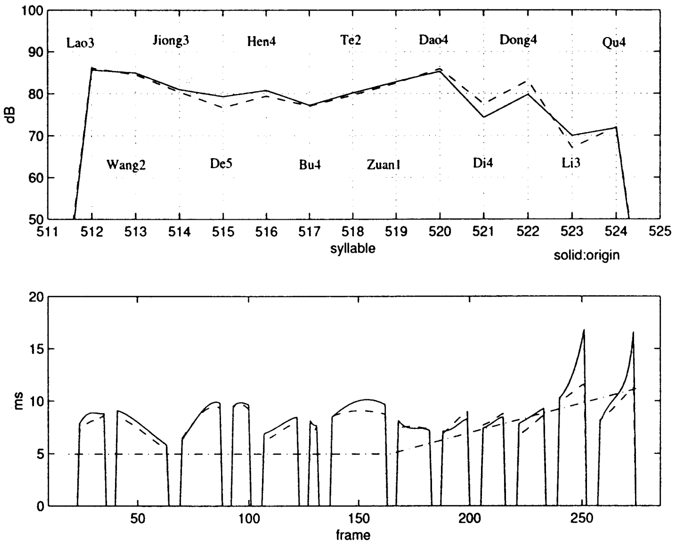


Fig. 15. The energy levels and pitch-period contours of the sentence, "lao3 wang2 jiong3 de5 hen4 bu4 de2 zuan1 dao4 di4 dong4 li3 qu4," where the energy levels and pitch-period contours are plotted in the upper half and lower half figures, respectively. The solid and dashed lines correspond to the original and synthetic energy levels and pitch-period contours, respectively.

Prosodic-Phrase-Rule (b) For trisyllabic or polysyllabic words, the primary and secondary stresses usually fall on the last and first syllable, respectively.

Prosodic-Phrase-Rule (c): The energy level and the pitch frequency at the start of the sentence are larger than those at the end of the sentence. In other words, the tendencies for the energy level and the pitch-frequency mean of one entire sentence goes downward.

The average RMSEs per frame of the prosodic parameters generated by the RFNN-based prosodic model with a two-input SONFIN are listed in Table V.

For example, one sentence, "lao3 wang2 jiong3 de5 hen4 bu4 de2 zuan1 dao4 di4 dong4 li3 qu4," is used to realize this verification. The energy level and pitch-period contour of this sentence are plotted in Fig. 15 and used to illustrate the following testing results.

1) "di4 dong4"

The learned fuzzy rules of the bisyllables in the trained SONFIN are listed in the following:

"di4": $(\mathit{WordInPhrase})$ is C and $(\mathit{PhraseInSen})$ is Z THEN $(\mathit{cluster})$ is $C5$

"dong4": $(\mathit{WordInPhrase})$ is D and $(\mathit{PhraseInSen})$ is Z THEN $(\mathit{cluster})$ is $C7$.

According to Fig. 8, the output value of "dong4"-rule $(C7)$ is obviously larger than that of "di4"-rule $(C5)$. This means that the stress (energy) of the bisyllabic word "di4 dong4" lies on the second syllable "dong4" as shown in Fig. 15. This is the verification of prosodic-phrase rule (a).

2) "hen4 bu4 de2"

The learned fuzzy rules of the trisyllabic word in the trained SONFIN are listed in the following:

"hen4": $(\mathit{WordInPhrase})$ is B and $(\mathit{PhraseInSen})$ is Y THEN $(\mathit{cluster})$ is $C2$

"bu4": $(\mathit{WordInPhrase})$ is C and $(\mathit{PhraseInSen})$ is X THEN $(\mathit{cluster})$ is $C4$

"de2": $(\mathit{WordInPhrase})$ is D and $(\mathit{PhraseInSen})$ is Y THEN $(\mathit{cluster})$ is $C6$.

The output values of these rules corresponding to "hen4," "bu4," and "de2" syllables are $O2$, $O4$, and $O6$ in Fig. 8, respectively. The relation of the three output values obviously are $O6 > O2 > O4$. This relation reveals that the stress of the trisyllabic word is on the last syllable "de2" and the secondary stress falls on the first syllable "hen4," as illustrated in Fig. 15. This is the verification of prosodic-phrase rule (b).

3) "lao3 qu4"

The learned fuzzy rules of the two syllables, "lao3" and "qu4," at the start and the end of the sentence in the trained SONFIN are listed in the following:

"lao3": $(\mathit{WordInPhrase})$ is D and $(\mathit{PhraseInSen})$ is Y THEN $(\mathit{cluster})$ is $C6$

"qu4": $(\mathit{WordInPhrase})$ is D and $(\mathit{PhraseInSen})$ is Z THEN $(\mathit{cluster})$ is $C7$.

According to Fig. 8, the output value of "lao3"-rule is obviously greater than that of "qu4"-rule. This reveals that the energy level and the pitch frequency of one sentence at the start of the sentence are larger than those at the end of the sentence, as illustrated in Fig. 15. This is the verification of prosodic-phrase rule (c).

D. A Subjective Listening Test

In this section, we provide two ways for a subjective listening test to verify the performance of the proposed RFNN-based prosodic model by implementing a Chinese TTS system. Because the experience of "naturalness" and "fluency" is subjective and is hard to be defined, the performance of the Chinese TTS system is tried to be evaluated as follows. Five Chinese articles randomly selected from the internet are prepared to do the subjective listening test. These articles are firstly translated into the synthetic speech by using the Chinese TTS system. Fifty native Chinese-speakers living in Taiwan are randomly selected to subjectively score the speech quality of the synthetic speech generated by the TTS system. The evaluation includes intelligibility and naturalness of the synthetic speech. The intelligibility evaluation consists of the clarity of syllables and words. The naturalness evaluation only considers the naturalness of sentences. In our informal listening test, all synthetic speech of the five Chinese articles sounds natural and intelligible. This confirms that the proposed RFNN-based prosodic model can improve the intelligibility and naturalness of the synthetic speech in Chinese TTS systems, since intelligibility and naturalness of the synthetic speech are mainly influenced by the prosodic model in TTS systems. In addition, some synthetic-speech examples compared with other Chinese TTS systems are put in our web sites such that readers can listen to them and judge the performance of our system [59], [60].

V. Conclusions

A novel prosodic-information synthesizer based on RFNN for Chinese TTS is proposed in this paper. The RFNN-based prosodic model integrates a multilayer recurrent neural network (RNN) and a Self-cOnstructing Neural Fuzzy Inference Network (SONFIN) into a recurrent connectionist structure in order to explore the relationship between the linguistic features and prosodic information. The experimental results confirm that the RFNN-based prosodic model performs considerably well. The advantages of combining SONFIN and RNN to construct a connectionist fuzzy neural network include the structured knowledge representation, approximate reasoning, parallel fuzzy inference, and self-learning. Besides, this RFNN can generate fuzzy IF-THEN rules to describe the relationship between the linguistic features and prosodic information.

Our experimental results also shows that most synthesis prosodic parameters generated by the proposed RFNN-based prosodic model matched with their original counterparts well. These prosodic parameters include pitch contour, energy level, initial duration, final duration, and pause duration. In addition, the experimental results confirm that the proposed RFNN-based prosodic model can learn the tone concatenation prosodic rules and fuzzy inference rules for prosodic phrase structure. Besides, a subjective listening test indicates that the proposed RFNN-based prosodic model can be used to improve the intelligibility and naturalness of the synthetic speech in Chinese TTS systems.

Acknowledgment

The authors would like to thank Mr. Cheng-Hsiung Tsai and Shean-Yih Lin for their help with developing the Chinese TTS system.

References

[1] Chinese Knowledge Information Processing Group, “The analysis of Chinese part-of-speech,” Institute of Information Science, Academia Sinica, Tech. Rep. 93-06, 1993.

[2] F. C. Chou, C. Y. Tseng, K. J. Chen, and L. S. Lee, “A Chinese text-to-speech system based on part-of-speech analysis, prosodic modeling, and nonuniform units,” in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 1997, pp. 923–926.

[3] E. Moulines and F. Charpentier, “Pitch synchronous waveform processing techniques for text-to-speech synthesis using diphones,” *Speech Commun.*, vol. 9, pp. 453–4677, Dec. 1990.

[4] ——, “Pitch synchronous waveform processing techniques for text-to-speech synthesis using diphones,” *Speech Commun.*, vol. 9, pp. 453–467, Dec. 1990.

[5] C. Hamon, E. Moulines, and F. Charpenter, “A diphone system based on time-domain prosodic modifications of speech,” in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 1989, pp. 238–241. S5.7.

[6] T. Dutoit, *An Introduction to Text-to-Speech Synthesis*. Norwell, MA: Kluwer, 1997.

[7] J. Allen, S. Hunnicutt, R. Carlson, and B. Granstrom, “MITalk-79: The MIT text-to-speech system,” *Acoust. Soc. Amer. Suppl.*, 1979. 165, s130.

[8] D. H. Klatt, “The Klattalk text-to-speech system,” in *Proc. ICASSP*, 1982, pp. 1589–1592.

[9] J. Zhang, “Acoustic parameters and phonological rules of a text-to-speech system for Chinese,” in *Proc. ICASSP*, Tokyo, Japan, Apr. 1986, pp. 2023–2026.

[10] L. S. Lee, C. Y. Tseng, and M. O. Young, “The synthesis rules in a Chinese text-to-speech system,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1309–1320, May 1989.

[11] Y. Sagisaka, “On the prediction of global F0 shape for Japanese text-to-speech,” in *Proc. ICASSP*, May 1990, pp. 325–328.

[12] N. C. Chan and C. Chan, “Prosodic rules for connected Chinese synthesis,” *J. Inf. Sci. Eng.*, vol. 8, no. 2, pp. 261–281, Jun. 1992.

[13] L. S. Lee, C. Y. Tseng, and C. J. Hsieh, “Improved tone concatenation rules in a formant-based Chinese text-to-speech system,” *IEEE Trans. Speech Audio Processing*, vol. 1, no. 3, pp. 287–294, 1993.

[14] K. Bartkova and C. Sorin, “A model of segmental duration for speech synthesis in French,” *Speech Commun.*, vol. 6, pp. 45–260, 1987.

[15] W. N. Campbell and D. Isard, “Segment durations in a syllable frame,” *J. Phonetics*, no. 19, pp. 37–47, 1991.

[16] H. Fujisaki, “The role of quantitative modeling in the study of intonation,” in *Proc. Int. Symp. Japanese Prosody*, 1992, pp. 163–174.

[17] J. ’t Hart, “F0 Stylization in speech: Straight lines versus parabolas,” *J. Acoustical Soc. Amer.*, vol. 90, no. 6, pp. 3368–3370, 1991.

[18] J. Pierrehumbert, “Synthesizing intonation,” *J. Acoust. Soc. Amer.*, vol. 70, no. 4, pp. 985–995, 1981.

[19] Y. C. Chang, Y. F. Lee, B. E. Shia, and H. C. Wang, “Statistical models for the Chinese text-to-speech system,” in *Proc. Eurospeech*, 1991, pp. 227–240.

[20] S. H. Hwang and S. H. Chen, “A prosodic model of Chinese speech and its application to pitch level generation for text-to-speech,” in *Proc. Int. Conf., Acoustics, Speech, and Signal Processing*, 1995, pp. 616–619.

[21] M. D. Riley, “Tree-based modeling for speech synthesis,” in *Talking Machines: Theories, Models, and Designs*. Amsterdam, The Netherlands: North Holland, 1992, pp. 265–273.

[22] C. Traber, “F0 generation with a database of natural F0 patterns and with a neural network,” in *Talking Machines: Theories, Models, and Designs*. Amsterdam, The Netherlands: North Holland, 1992, pp. 287–304.

[23] W. Black and A. J. Hunt, “Generating F0 contours from ToBI labels using linear regression,” in *Proc. ICSLP*, vol. 3, Philadelphia, PA, 1996, pp. 1385–1388.

[24] S. H. Chen, S. H. Hwang, and Y. R. Wang, “An RNN-based prosodic information synthesizer for Chinese text-to-speech,” *IEEE Trans. Speech Audio Processing*, vol. 6, May 1998.

[25] ——, “A Chinese text-to-speech system,” in *Proc. ICSLP*, 1996.

[26] C. F. Juang and C. T. Lin, “An on-line self-constructing neural fuzzy inference network and its application,” *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 12–32, Feb. 1998.

[27] C. L. Giles, G. M. Kuhn, and R. J. Williams, “Dynamic recurrent neural networks: Theory and applications,” *IEEE Trans. Neural Networks*, vol. 5, pp. 135–156, Mar. 1994.

[28] C. T. Sun, “Rule-based structure identification in an adaptive-network-based fuzzy inference,” *IEEE Trans. Fuzzy Syst.*, vol. 2, no. 1, pp. 64–73, Feb. 1994.

[29] M. S. Scordilis and J. N. Gowdy, "Neural network based generation of fundamental frequency contours," in *Proc. ICASSP*, 1989, pp. 219–222.

[30] Y. Sagisaka, "On the prediction of global F0 shape for Japanese text-to-speech," in *Proc. ICASSP*, May 1990, pp. 325–328.

[31] L. S. Lee, C. Y. Tseng, J. Huang, and K. J. Chen, "Digital synthesis of Mandarin speech using its special characteristics," *J. Chinese Inst. Eng.*, vol. 6, pp. 107–115, Mar. 1983.

[32] M. Ostendorf and N. Veilleux, "A hierarchical stochastic model for automatic prediction of prosodic boundary location," *Comput. Linguist.*, vol. 20, no. 1, pp. 27–54, June 1994.

[33] D. Crystal, *A Dictionary of Linguistic and Phonetics*. New York: Blackwell, 1997.

[34] M. Nespor and I. Vogel, *Prosodic Phonol.*, 1986.

[35] H. Fujisaki, "Realization of linguistic information in the voice fundamental frequency contour of the spoken Japanese," in *Proc. ICASSP*, vol. 1, May 1988, pp. 663–666.

[36] D. H. Klatt, "Software for a cascade/parallel formant synthesizer," *J. Acoust. Soc. Amer.*, vol. 67, no. 3, pp. 820–857, Mar. 1980.

[37] C. Shih and R. Sproat, "Issues in text-to-speech conversion for Mandarin," *Comput. Linguist. Chinese Language Process.*, vol. 1, no. 1, pp. 37–86, Aug. 1996.

[38] C. J. Lin and C. T. Lin, "An ART-based fuzzy adaptive learning control network," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 477–496, 1997.

[39] L. X. Wang, *Adaptive Fuzzy Systems and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[40] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, pp. 1414–1427, 1992.

[41] H. B. D. Sorensen, "A cepstral noise reduction multilayer neural network," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1991, pp. 933–936.

[42] C. W. Omlin, K. Thornber, and C. L. Giles, "Fuzzy finite-state automata can be deterministically encoded into recurrent neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 6, no. 1, pp. 76–89, 1998.

[43] L.-N. Teow and K.-K. Loe, "Effective learning in recurrent max-min neural networks," *Neural Networks*, vol. 11, pp. 535–547, 1998.

[44] P.-J. Werbos, "Generalization of backpropagation with application to a recurrent gas market model," *Neural Networks*, vol. 1, pp. 338–356, 1988.

[45] M. Chu and S. N. Lu, "A text-to-speech system with high intelligibility and naturalness for Chinese," *Chinese J. Acoust.*, vol. 15, no. 1, pp. 81–90, 1996.

[46] M. Chu, S. H. Lu, H. Y. Si, L. He, and D. H. Guan, "The control of juncture and prosody in Chinese TTS system," in *Proc ICSP'96*, 1996, pp. 725–728.

[47] D. E. Rumelbert, G. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing*. Cambridge, MA: MIT Press, 1987, vol. 1.

[48] R. P. Lippmann, "An introduction to computation with neural nets," *IEEE ASSP Mag.*, pp. 4–22, 1987.

[49] S. Haykin, *Neural Networks—A Comprehensive Foundation*. Englewood Cliffs, NJ: Prentice Hall, 1994.

[50] S. F. Liang, A. W. Su, and C. T. Lin, "Model-based synthesis of plucked string instruments by using a class of scattering recurrent networks," *IEEE Trans. Neural Networks*, vol. 11, no. 1, pp. 171–185, 2000.

[51] S. M. Karz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 400–401, Mar. 1987.

[52] S. Martin, J. Liermann, and H. Ney, "Algorithms for bigram and trigram word clustering," *Proc. EUROSPEECH*, pp. 1253–1256, Sept. 1995.

[53] S. H. Chen and Y. R. Wang, "Vector quantization of pitch information in Chinese speech," *IEEE Trans. Commun.*, vol. 38, pp. 1317–1320, Sept. 1990.

[54] C. Shih and R. Sproat, "Issues in text-to-speech conversion for Chinese," *Comput. Linguist. Chinese Language Processing*, vol. 1, no. 1, pp. 37–86, Aug. 1996.

[55] C. L. Shih, "The prosodic domain of tone Sandhi in Chinese," Ph.D. dissertation, Univ. California, San Diego, 1986.

[56] Z. S. Zhang, "Tone and tone Sandhi in Chinese," Ph.D. dissertation, Ohio State Univ., Columbus, 1988.

[57] Z. Lu, "Research on acoustic cues of bisyllables in Mandarin," *Learn. Chinese*, vol. 6, pp. 41–48, 1984.

[58] M. Y. Chen, *Tone Sandhi: Patterns Across Chinese Dialects*. Cambridge, MA: Cambridge Univ. Press, 2000.

[59] [Online]. Available: http://www.cc.nctu.edu.tw/~u8612812/index.htm.

[60] [Online]. Available: http://falcon.cn.nctu.edu.tw/~www/sections.php?op=viewarticle&artid=89.

Chin-Teng Lin (S'88–M'91–SM'99) received the B.S. degree in control engineering from the National Chiao-Tung University, Hsinchu, Taiwan, R.O.C., in 1986 and the M.S.E.E. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1989 and 1992, respectively.

Since August 1992, he has been with the College of Electrical Engineering and Computer Science, National Chiao-Tung University, Hsinchu, where he is currently a Professor and Chairman of Electrical and Control Engineering Department. He served as the Deputy Dean of the Research and Development Office of the National Chiao-Tung University from 1998 to 2000. His current research interests are fuzzy systems, neural networks, intelligent control, human–machine interface, image processing, pattern recognition, video and audio (speech) processing, and intelligent transportation system (ITS). He is the co-author of *Neural Fuzzy System—A Neuro-Fuzzy Synergism to Intelligent System* (Englewood Cliffs, NJ: Prentice-Hall), and the author of *Neural Fuzzy Control Systems With Structure and Parameter Learning* (Singapore: World Scientific). He has published over 70 journal papers in the areas of soft computing, neural networks, and fuzzy systems, including about 51 IEEE TRANSACTIONS papers.

Dr. Lin is a member of Tau Beta Pi and Eta Kappa Nu. He is also a member of the IEEE Computer Society, the IEEE Robotics and Automation Society, and the IEEE System, Man, Cybernetics Society. He has been the Executive Council Member of Chinese Automation Association since 1998. He is the Society President of Chinese Fuzzy Systems Association T since 2002. He is the Chairman of IEEE Robotics and Automation Society, Taipei Chapter since 2000, and the association editor of IEEE TRANSACTIONS ON SYSTEMS, MAN, CYBERNETICS since 2001. He won the Outstanding Research Award granted by National Science Council (NSC), Taiwan, from 1997 to 2001, the Outstanding Electrical Engineering Professor Award granted by the Chinese Institute of Electrical Engineering (CIEE) in 1997, and the Outstanding Engineering Professor Award granted by the Chinese Institute of Engineering (CIE) in 2000. He was also elected to be one of the 38th Ten Outstanding Young Persons in Taiwan, R.O.C., (2000). He currently serves as the associate editor of the IEEE TRANSACTIONS ON SYSTEMS, MAN, CYBERNETICS, PART B, IEEE TRANSACTIONS ON FUZZY SYSTEMS, and the *Journal of Automatica*.

Rui-Cheng Wu received the B.S. degree in nuclear engineering from National Tsing-Hua University, Taiwan, R.O.C., in 1995, and M.S. degree in control engineering from National Chiao-Tung University, Taiwan, in 1997. He is currently pursuing the Ph.D. degree in electrical and control engineering at the National Chiao-Tung University, Taiwan, R.O.C.

His current research interests are audio signal processing, speech recognition/enhancement, fuzzy control, neural networks, and linear control.

Jyh-Yeong Chang received the B.S. degree in control engineering in 1976 and the M.S. degree in electronic engineering in 1980, both from National Chiao Tung University, Taiwan, R.O.C. From 1976 to 1978 and 1980 to 1982, he was a Research Fellow at Chung Shan Institute of Science and Technology (CSIST), Taiwan. He received the Ph.D. degree in electrical engineering from North Carolina State University, Raleigh, in 1987.

Since 1987, he has been an Associate Professor in the Department of Electrical and Control Engineering at National Chiao Tung University. His research interests include fuzzy sets and systems, image processing, pattern recognition, and neural network applications.

Sheng-Fu Liang was born in Tainan, Taiwan, R.O.C., in 1971. He received the B.S. and M.S. degrees in control engineering from the National Chiao-Tung University (NCTU), Taiwan, in 1994 and 1996, respectively. He received the Ph.D. degree in electrical and control engineering from NCTU in 2000.

Currently, he is a Research Assistant Professor in electrical and control engineering at NCTU. His research activities include music synthesis, neural networks, audio processing, and image processing.