

Adaptive Prosody Modelling for Improved Synthetic Speech Quality

Moses E. Ekpenyong^{1(✉)}, Udoinyang G. Inyang¹, and EmemObong O. Udoh²

¹ Department of Computer Science, University of Uyo, Uyo, Nigeria
mosesekpenyong@uniuyo.edu.ng, mosesekpenyong@gmail.com,
udoiiinyang@yahoo.com

² Department of Linguistics and Nigerian Languages, University of Uyo, Uyo, Nigeria
ememobongudoh@uniuyo.edu.ng, ememobongudoh@gmail.com

Abstract. Neural networks and fuzzy logic have proven to be efficient when applied individually to a variety of domain-specific problems, but their precision is enhanced when hybridized. This contribution presents a combined framework for improving the accuracy of prosodic models. It adopts the Adaptive Neuro-fuzzy Inference System (ANFIS), to offer self-tuned cognitive-learning capabilities, suitable for predicting the imprecise nature of speech prosody. After initializing the Fuzzy Inference System (FIS) structure, an Ibibio (ISO 693–3: nic; Ethnologue: IBB) speech dataset was trained using the gradient descent and non-negative least squares estimator (LSE) to demonstrate the feasibility of the proposed model. The model was then validated using synthesized speech corpus dataset of fundamental frequency (F0) values of ibibio tones, captured at various contour positions (initial, mid, final) within the corpus. Results obtained showed an insignificant difference between the predicted output and the check dataset with a checking error of 0.0412, and validates our claim that the proposed model is satisfactory and suitable for improving prosody prediction of synthetic speech.

Keywords: ANFIS · Prosody · Speech synthesis · Under-resourced language

1 Introduction

The formulation of prosodic structures (phrase breaks, pitch accents, phrase accents and boundary tones) of utterances remains a major challenge in Text-To-Speech (TTS) synthesis. Hence, the prediction of these elements largely depends on the accuracy and quality of error-prone linguistic procedures such as part of speech tagging, syntax and morphology analysis [1]. In tone languages, tones

M. Ekpenyong—Please note that the LNCS Editorial assumes that all authors have used the western naming convention, with given names preceding surnames. This determines the structure of the names in the running heads and the author index.

(characterized by the variation of speech within syllable) are lexically important as key determinants to speech fluency and therefore constitute the most significant prosodic features in speech synthesis of tone languages [2,3].

The quality and acceptability of synthetic speech is determined by the prosodic well-formedness of the utterances [4]. Well-formedness is a product of various constraints and is classified into four categories namely, metrical, morpho-syntactic, semantic-pragmatic, and alignment. An utterance is prosodically well-formed if the rules that associates the segmental and prosodic tiers are consistent with those governing the formation of prosodic patterns in that language. Thus, a more comprehensive approach is required to account for the constraint hierarchy and effect at the various levels where linguistic and paralinguistic units are processed. This explains why some of the basic principles are violated. Optimality Theory [5] appears to offer some promising solutions in this area, but it is not clear how such a theory is applied in today's TTS synthesis.

The emergence of soft computing (SC) has offered attractive solutions for modelling highly nonlinear or partially defined complex systems and processes. SC techniques are known to cover two major optimization concepts: approximate reasoning and function approximation. Prominent SC techniques include evolutionary computing, fuzzy logic, neural networks and Bayesian statistics. To further improve the quality of synthesized speech, the fuzzy Logic (FL) technique in [6] is combined with the neural network (NN) technique, to obtain an Adaptive Neuro-fuzzy Inference System (ANFIS). The resulting system is then used to train and predict the accuracy of the prosodic features data - mainly the fundamental frequency (F0) of Ibibio tones (i.e., High - H, Low - L, Downstepped -D, Rising - LH, and Falling - HL), extracted at various contour positions (high, mid and low) from original (recorded) and synthesized speech corpora.

2 Tone and Prosody Prediction

One major aspect in TTS synthesis is the successful prediction of tonal events [7], and most predictive models require data labeled with intermediate representations such as Tone Boundary Index (TOBI) symbols. However, this approach is difficult, expensive and error prone [2]. In [8], sentence logarithmic F0 contour is represented as a superposition of tone features on phrase components as in the case of a generation process model - F0 model. The tone components were realized by concatenating their fragments at the tone nuclei predicted by a corpus-based method, while the phrase components were generated by rules under the F0 model framework. Beyond differences in F0 height and contours, tonal contrasts are often accompanied by systematic variations in duration and phonation [9]. A variety of techniques have been explored to improve prosody in tone language synthesis. Hence, with a larger speech corpus from a target speaker, a concatenative approach with unit selection of the F0 contour offers good performance [10,11]. But, this approach greatly suffers for under-resourced languages, given the limited amount of available speech corpus. HMM-based approaches have provided solution to the data sparseness problem experienced

by unit selection systems, and can be exploited to efficiently estimate relatively shallow features close to the text itself. In [2], these features are applied directly as contexts without attempting explicit prediction of intermediate representations. In [4], we arrived at a generic HMM sequence that describes the contextual dependency of the features with prosodic factors defined for tone language synthesis, as,

$$\begin{aligned}
T_{Label} = & \overrightarrow{\theta}_{0,tone(i,1)}^f + \overrightarrow{\theta}_{tone(i,1)} + \dots + \overrightarrow{\theta}_{0,tone(i,n-1)} + \overrightarrow{\theta}_{c(i,n-1),tonepat(i,n-1)}^f \\
& + \overrightarrow{\theta}_{tone(i,n)} + \overrightarrow{\theta}_{0,tone(i,n)} + \overrightarrow{\theta}_{c(i,n),tonepat(i,n)}^f + \overrightarrow{\theta}_{0,tone(i,n+1)}^f + \dots \\
& + \overrightarrow{\theta}_{0,tone(i,N)} + \overleftarrow{\theta}_{C+1,tone(i,N)}^b + \overleftarrow{\theta}_{tpros(i,N)} + \dots + \overleftarrow{\theta}_{tpros(i,n+1)} \\
& + \overleftarrow{\theta}_{c(i,n),tonepat(i,n)}^b + \overleftarrow{\theta}_{pros(i,n)} + \overleftarrow{\theta}_{c(i,n-1),tonepat(i,n-1)}^b \\
& + \overleftarrow{\theta}_{pros(i,n-1)} + \dots + \overleftarrow{\theta}_{pros(i,1)}
\end{aligned} \tag{1}$$

where, $\overrightarrow{\theta}_{0,tone(i,n) \in \{1,2,\dots,n\}}$, represents a vector of current tones of the intended language; $\overleftarrow{\theta}_{pros(i,n)}$, is a vector of current prosody of the language; $tonepat(i,n) \in \{(1,1), (1,2), \dots, (i,n)\}$, describes the tone patterns defined by the tone pair iteration; $t(i,n), t(i,n+1)$; $C(i,n) \in \{0, 1, 2, \dots, C, C+1\}$, describes the co-articulation (effect of sound interaction) at inter-syllable locations between the current syllable, n , and the next syllable, $n+1$; $\overrightarrow{\theta}_{c(i,n),tonepat(i,n)}^f$ and $\overleftarrow{\theta}_{c(i,n),tonepat(i,n)}^b$, are the forward and backward transitions of the tone patterns, respectively, with its implied co-articulation. Eq. 1 is most suitable for modelling the state features of a HMM-based tone language synthesis system and is currently being investigated for completeness.

2.1 Predicting and Evaluating Prosodic Features

Once a prosodic model has been obtained for a system, the prosodic variation with its accompanying prediction scheme from input text can be determined. Early TTS systems relied on hand-crafted rules that predict prosody assignment based on simple part-of-speech (PoS) features or more elaborate syntactic parsing. The major drawback of this approach is extension and maintenance difficulties. Mostly, new rules for prosodic assignments are trailed by unforeseen and undesirable consequences. Corpus-based techniques - the use of relatively huge speech database have since rescued hand-crafted rule systems. They represent annotations of prosodic features and are used as training materials for machine learning algorithms, where decision procedures are derived from automated textual analysis. The automatically derived decisions appear to be limited by the amount of hand-labelled data available for training; but the provision of correct examples in the training corpus must sufficiently outweigh the data that could yield undesirable prediction, else, errors may easily go unnoticed. The challenges here extend beyond those involved in the derivation of prosodic patterning from grammatical information, since general text additionally requires semantic/pragmatic background information on emphasis and contrast, for instance.

But, with some degree of explicit control over prosodic variation, the naturalness of TTS systems could be improved. This control may be accomplished by providing precise user-specific markup capabilities. Evaluating TTS systems in general is extremely challenging. Today, most synthesis systems are of very high quality. Although subjective judgment ratings are mostly used to evaluate prosodic assignments, this subject (prosody assignment) remains a major research question.

3 Our Approach

3.1 The ANFIS Architecture

A block diagram showing the ANFIS process flow is presented in Fig. 1, with the fuzzifier, defuzzifier, rule base and fuzzy inference system as components. Fuzzifier converts the crisp inputs into linguistic variables (low, mid and high) using membership functions while, defuzzifier performs a scale mapping, and converts the range of values of output variables into the corresponding universes of discourse (UoD), thus finally producing a crisp output from an inferred fuzzy control action. The rule base consists of a number of fuzzy IF-THEN rules that guides the inference engine in its reasoning. The fuzzy inference engine forms the kernel of ANFIS. It has the capability of simulating human decision-making processes based on fuzzy concepts, and inferring fuzzy control actions by employing fuzzy implication with the rules of inference in the fuzzy rule base. The most common types of fuzzy inference methods are Mamdani and Sugeno methods [12]. The difference between these two methods lies in the consequent parameter of the fuzzy rules. This paper adopts the Mamdani inference mechanism for the evaluation and extraction of rules and production of the fuzzy output. The reason for using Mamdani is that it is intuitive and has widespread acceptance. In addition, it is well suited to human input. The ANFIS inference engine is a five layered architecture [13], and the rule base consists of rules of the form:

$$\text{IF } (x_j \text{ is } A_i^r) \text{ and } (y_j \text{ is } A_i^r) \text{ THEN } z \text{ is } C_i^r \quad (2)$$

where, r is the rule-number, x and y are input variables, z is the output variable. A_i^r , are the linguistic terms, characterized by the appropriate membership

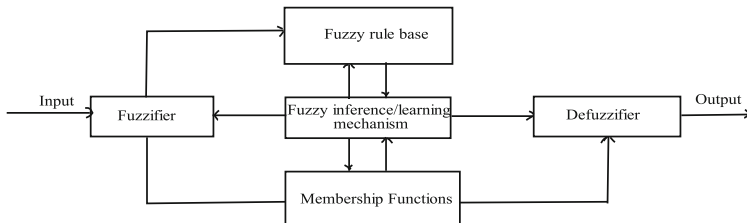


Fig. 1. A generic ANFIS Block diagram

function, μ_{A_n} . ANFIS uses a combination of gradient descent and least square estimator (LSE) depending on the application, with two sets of parameters: a set of premise and a set of consequent parameters. The process of parameter update is achieved using a forward and backward pass learning algorithm. The forward pass (FP) learning computes the neuron outputs, layer after layer, and identifies the consequent parameters by the LSE, leading to the final (single) output. The backward pass (BP) propagates error signals and updates the antecedent parameters according to a chain rule. Each layer of ANFIS consists of nodes described by the node function.

Layer 1 is the input fuzzification layer, where each node in this layer generates fuzzy membership grades for the inputs, and is given by:

$$\begin{aligned} O_i^1 &= \mu_{A_i}(x_i) & i &= 1, 2, \dots, n \\ O_j^1 &= \mu_{A_j}(y_j) & j &= 1, 2, \dots, n \\ O_k^1 &= \mu_{A_k}(L_k) & k &= 1, 2, \dots, n \end{aligned} \quad (3)$$

The general form of the triangular MF is defined as [13]:

$$\mu(x) = \begin{cases} 1 & \text{if } x = b \\ \frac{x-a}{b-a} & \text{if } a \leq x < b \\ \frac{c-x}{c-b} & \text{if } b \leq x < c \\ 0 & \text{if } c = x \end{cases} \quad (4)$$

or

$$\mu_A = \max \left(\min \left(\frac{x-a}{b-a}, \frac{c-x}{c-b} \right), 0 \right) \quad (5)$$

where, a and c , are parameters governing triangular MF; b is the value for which $\mu(x) = 1$, and is given as, $b = \frac{a+c}{2}$.

Layer 2, is the rule evaluation node, and uses either the disjunction or conjunction operator (AND or OR) to determine the firing strengths. This is evaluated using the max (Eq. (6)) or min (Eq. (7)) operator, respectively:

$$\mu_A B(x) = \max \mu_A(x), \mu_B(x) \quad (6)$$

$$\mu_A B(x) = \min \mu_A(x), \mu_B(x) \quad (7)$$

The firing strengths, O_i^2 , are the products of the corresponding membership degrees obtained from layer 1, and is given as:

$$O_i^2 = w_i = \mu_{A_n}(x_i) \mu_{B_n}(y_j) \mu_{D_n}(L_k) \quad (8)$$

Layer 3 is the normalization layer and computes the ratio of each rule firing strength to the sum of all rules firing strength. The output, \bar{w}_i , is defined in Eq. (9). The defuzzification layer (layer 4), consists of consequent nodes for calculating the contribution of each rule to the overall output and is given in Eq. (10). The overall output of the ANFIS model is finally obtained by summing (aggregating) all incoming signals, by layer 5. In this paper, the centroid method as depicted in Eq. (11) is used for this purpose.

$$O_i^3 = w_i = \frac{w_i}{\sum_i w_i} \quad (9)$$

$$O_i^4 = w_i f_i \quad (10)$$

$$O_i^5 = M = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (11)$$

3.2 Neuro-fuzzy System Model

A hybridized approach (a fusion of least-square and back propagation gradient descent methods) [14], is adopted in this paper for training and validating the input dataset. This approach consists of forward and backward passes. In the forward pass, each node's output proceeds until the fourth layer when the consequent parameters are identified by the least squares method. During the backward pass, the premise parameters are updated by gradient descent as the error signal re-propagates backwards. In Fig. 2, the proposed ANFIS-based model architecture is presented, illustrating the contribution of inputs to the various rules. The inputs are crisp (non-fuzzy) numbers limited to a specific range.

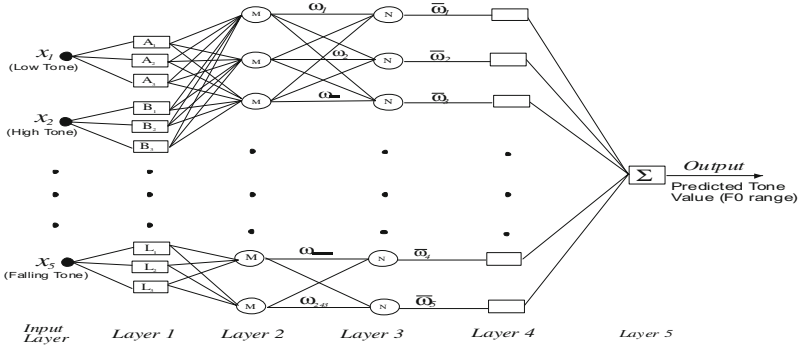
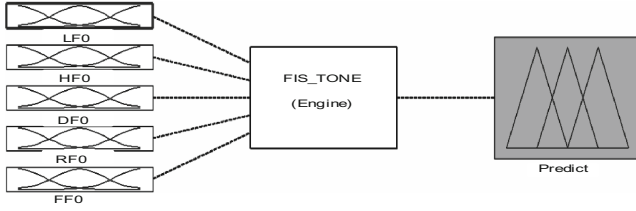


Fig. 2. Proposed ANFIS model

All the rules (a set of IF-THEN statements) are evaluated in parallel - from a set of decomposed linguistic terms (or membership functions) describing the various tones of the language, using fuzzy reasoning. The results of the rules are finally merged and distilled (defuzzified) using the membership functions. The membership functions are used to map the non-fuzzy input values to fuzzy linguistic terms and vice versa. They are used to quantify the membership terms, which mappings finally yield a crisp (non-fuzzy) output (number). Five linguistic variables were identified as input to the fuzzy inference system (FIS). These variables enumerate the tones (including the phonemic variations) of Ibibio,

**Fig. 3.** FIS tone system

i.e., L, H, D, R, F tones. Figure 3 shows a MatLab interface implementing the FIS component of the ANFIS model.

Input Membership Functions: Three linguistic terms were defined over the Universe of Discourse (UoD) for each input variable. The linguistic terms are F0 values extracted from the speech contour described by: $F0(t) = \{initial, mid, final\}$, where, t denotes the linguistic variables.

Eqs. (12), (13), (14), (15) and (16) describe the membership functions of the respective linguistic variables. They represent experimental values annotated using the Praat annotation software:

$$\mu_L(F0) = \begin{cases} 80 \leq F0 \leq 150, & initial \\ 100 \leq F0 \leq 140, & mid \\ 55 \leq F0 \leq 90, & final \end{cases} \quad (12)$$

$$\mu_H(F0) = \begin{cases} 90 \leq F0 \leq 170, & initial \\ 145 \leq F0 \leq 190, & mid \\ 80 \leq F0 \leq 120, & final \end{cases} \quad (13)$$

$$\mu_D(F0) = \begin{cases} 140 \leq F0 \leq 190, & initial \\ 120 \leq F0 \leq 150, & mid \\ 80 \leq F0 \leq 130, & final \end{cases} \quad (14)$$

$$\mu_R(F0) = \begin{cases} 135 \leq F0 \leq 180, & initial \\ 120 \leq F0 \leq 170, & mid \\ 80 \leq F0 \leq 130, & final \end{cases} \quad (15)$$

$$\mu_F(F0) = \begin{cases} 100 \leq F0 \leq 150, & initial \\ 115 \leq F0 \leq 160, & mid \\ 80 \leq F0 \leq 130, & final \end{cases} \quad (16)$$

Output Membership Function: The output membership function was defined by assignment, following a careful analysis and observation of the speech data

by domain experts. The output membership function is viewed as a continuum with each output element spreading across a spectrum area (selection) of the continuum.

ANFIS Engine: As earlier mentioned, the Mamdani-type fuzzy inference mechanism is used to formulate the mapping from a given input to an output using fuzzy logic. This mapping provides the basis on which decisions could be made or patterns discerned. The inference process includes the following: block building, structuring, firing, implication and aggregation of rules. The number of rules is determined by the complexity of the associated fuzzy system. Though we have established $3^5=243$ rules for evaluating the tone contour patterns of the speech corpus, not all the rules fired. Snippets of the extracted F0 data used for training the ANFIS system and coded representations (1-initial, 2-mid,3-final) for building the respective rules, are shown in Tables 1 and 2, respectively.

Table 1. F0s of Ibibio tones, randomly selected for training

S/no	F0 (L)	F0 (H)	F0 (DH)	F0 (LH)	F0 (HL)	Predict
1	104	124	154	146	127	1
2	128	81	115	169	108	2
3	103	141	98	165	101	2
4	136	175	128	168	112	2
5	140	180	172	174	83	1
6	130	156	80	151	127	2
7	112	160	117	179	138	2
8	105	146	156	146	144	1
9	122	94	147	175	119	2
:	:	:	:	:	:	:
241	101	119	120	122	141	2
242	95	160	129	137	123	2
243	110	117	121	127	113	2

Details of the interface implementation of the fuzzy membership functions, rules and consequences can be found in [6].

Different implication operators fit different aggregation operators (e.g. union and intersection). Whereas the union operator uses the Mamdani and Larsen operators, the intersection uses the Lukasiewicz operator [15]. The Mamdani operator is applied in this paper. After inference, the overall result is a fuzzy value and should be defuzzified to obtain a final crisp output. There are different algorithms for defuzzification namely, Centre of Gravity (CoG) or Centroid Average (CA), Maximum Centre Average (MCA), Mean of Maximum (MoM), Smallest of Maximum (SoM) and Largest of Maximum (LoM). As earlier mentioned, the CoG algorithm (Centroid) as defined in Eq. (11) is used in this paper.

Table 2. Coded representation of Table 1 used for building the rules

Rule	F0 (L)	F0 (H)	F0 (DH)	F0 (LH)	F0 (HL)	Predict
1	1	1	1	1	1	1
2	2	3	3	1	3	2
3	2	1	3	1	3	2
4	2	2	2	1	3	2
5	2	2	1	1	3	1
6	2	2	3	1	2	2
7	2	2	3	1	1	2
8	2	1	1	1	1	1
9	2	3	2	1	2	2
:	:	:	:	:	:	:
241	1	3	2	2	1	2
242	1	2	2	2	2	2
243	1	3	2	2	3	2

4 Experiment and Results

4.1 FL Model Validation

To validate the feasibility of the proposed ANFIS model, we annotated and extracted, using Praat - a speech processing and annotation software, F0 values of Ibibio tones at various contour positions (initial, mid and final) from both recorded and synthesised speech corpus. Figure 4 shows a sample annotation of a synthesised Ibibio speech. The sample size used for this experiment were long utterances containing the various tones of the language selected from a set of 1140 sentences used for HMM-based Ibibio synthesis experiment [16]. An objective evaluation of the annotations revealed that falling (F) tones were wrongly perceived as either downstepped (D) or high (H) tones, mostly on the ɔ (O – SAMPA equivalent) sound, which indicated a possibility of phoneme/tone confusion. The evaluation of phoneme and tone confusions for synthesised voices used for this experiment has been investigated in [17]. Using the extracted parameters, the degree of certainty (crisp output) of the FIS was simulated for the purpose of comparing the original and synthesised annotations. Tables 3 and 4 present the input (average F0) values at different contour positions for the various tones of Ibibio, and the simulated crisp output for original and synthesized voices, respectively. We observed from these tables that the degree of certainty of the original speech was higher, compared to the synthesised speech. This result implies that tone patterns of the original voices are well predicted by the FL system.

Generally, predictions at the final positions in both cases were poor. The reason for this may not be unconnected with the fact that rising (R) and falling (F)

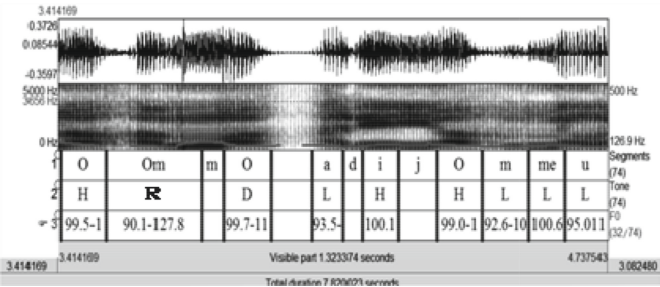


Fig. 4. Sample annotation of a synthesised male speaker

Table 3. Input F0s and crisp output for original male speaker

S/N	Position	Input (average F0)					Crisp output
		L	H	D	R	F	
1	Initial	98	130	165	158	125	0.693
2	Mid	120	168	135	145	138	0.664
3	Final	78	100	105	100	105	0.301

Table 4. Input F0s and crisp output for synthesised male speaker

S/N	Position	Input (average F0)					Crisp output
		L	H	D	R	F	
1	Initial	186	192	144	115	150	0.500
2	Mid	112	146	139	121	126	0.647
3	Final	85	98	87	88	97	0.250

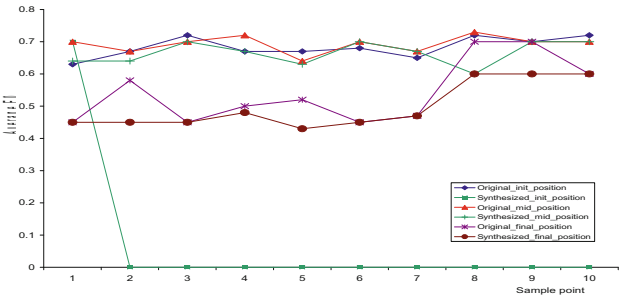


Fig. 5. Graph showing implication and aggregation of prosody rules

tones most rarely occur at the final positions in a well-formed Ibibio utterance/sentence. Also, the resultant F0 averages used for the prediction at these positions were gathered from a range of (tone) values appearing few distances away from the end of the sentence(s). Figure 5 shows plots of rules predictions at the various contour positions for the original and synthesized voices. In Fig. 5, we observe that for original voices, most of the tone rules at the initial and mid positions fired with average F0 predictions of 0.683 and 0.693, respectively; while tone rules at the final position experienced poor firing - i.e. gave a low average F0 prediction of 0.542. For synthesized voices, most tone rules at the mid position fired, compared to rules at the initial and final positions, which yielded poor predictions of 0.07 and 0.498, respectively. The FIS results therefore call for an investigation into the poor synthesis of tones at the initial and final positions in a given utterance. In the next section, we re-train the synthesis data using our ANFIS model to improve on the current results.

4.2 Model Training and Checking

A simulated structure of the proposed ANFIS model, generated in MatLab is presented in Fig. 6.

As shown in Fig. 6, the proposed model is five layered, with five inputs, each with 3 input membership terms. The rule base comprises 243 rules. The properties of the ANFIS model are as listed in Table 5.

ANFIS model training was concluded at the 2nd epoch with training and testing errors of 0.0545 and 2.276, respectively. The graph of the testing and checking of the ANFIS model is presented in Fig. 7. In Fig. 7, the ANFIS output is mapped against the checking dataset. We observed that there is an insignificant difference between the predicted output (*) and the check dataset (+) with a checking error of 0.0412. Hence the proposed solution is satisfactory and suitable for improving prosody prediction of synthetic speech.

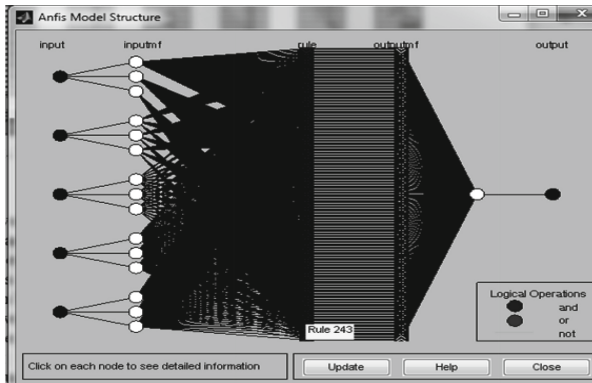
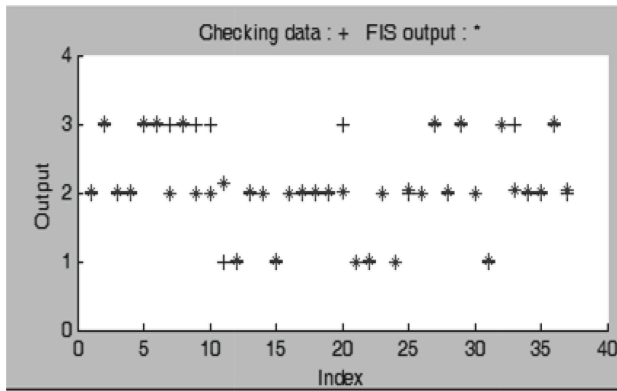


Fig. 6. Simulated ANFIS structure

Table 5. Properties of ANFIS model for prosody prediction

S/No	Parameter	Number
1	Nodes	524
2	Linear Parameters	1458
3	Nonlinear parameters	45
4	Training data pairs	170
6	Checking data pairs	37
7	Testing data pairs	37
8	Fuzzy rules	243

**Fig. 7.** Plots for checking and training data set

5 Conclusion and Future Work

The production of quality (natural and intelligible) synthetic speech depends, in part, on the correctness of the language's prosody. Prosody modelling is useful for associating the variations of prosodic features with changes in structure, meaning and context of spoken languages. These features to a great extent, contribute to enhancing the perceived quality of speech. This paper has presented an adaptive fuzzy Inference system for modelling the prosody of synthetic speech. The proposed model is suitable for the precise prediction of F0 contour patterns in human and synthetic speech. In the future, we shall explore the use of genetic algorithm in determining optimal parameters of the weights and structure of the current approaches and investigate the effectiveness of the design, in a bid to provide a more efficient solution to the prosody problem presented by tone language systems.

References

1. Xydas, G., Spiliotopoulos, D., Kouroupetroglou, G.: Modeling prosodic structures in linguistically enriched environments. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2004. LNCS (LNAI), vol. 3206, pp. 521–528. Springer, Heidelberg (2004)
2. Ekpenyong, M., Urua, E.-A., Watts, O., King, S., Yamagishi, J.: Statistical parametric speech synthesis for Ibibio. *Speech Commun.* **56**, 243–251 (2014)
3. Ekpenyong, M., Udoh, E.O., Udosen, E., Urua, E.-A.: Improved syllable-based text to speech synthesis for tone language systems. In: Vetulani, Z., Mariani, J. (eds.) LTC 2011. LNCS, vol. 8387, pp. 3–15. Springer, Heidelberg (2014)
4. Di Cristo, A., Di Cristo, P., Campione, E., Veronis, J.: A prosodic model for text-to-speech synthesis in French. In: Botinis, A. (ed.) *Intonation: Analysis Modelling and Technology*, pp. 321–355. Kluwer, Amsterdam (2000)
5. Prince, A., Smolensky, P.: *Optimality Theory: Constraints Interaction in Generative Grammar*. Wiley-Blackwell Publishers, New Jersey (2004)
6. Ekpenyong, M., Udoh, E.O.: Intelligent prosody modelling: a framework for tone language synthesis. In: Vetulani, Z., Uszkoreit, H. (eds.) 6th Language and Technology Conference (LTC), Poznan, Poland, Fundacja Uniwersytetu im. A. Mickiewicza, pp. 279–283 (2013)
7. Zervas, P., Xydas, G., Fakotakis, N., Kokkinakis, G., Kouroupetroglou, G.: Evaluation of corpus based tone prediction in mismatched environments for Greek TtS synthesis. In: *Proceedings of 8th International Conference on Spoken Language Processing (INTERSPEECH - ICSLP)*, Jeju, Korea, pp. 761–764 (2004)
8. Sun, Q., Hirose, K., Minematsu, N.: Improved prediction of tone components for F0 contour generation of Mandarin speech based on the tone nucleus model. In: *Proceedings of Speech Prosody Special Interest Group (SProSIG) Conference*, Campinas, pp. 1–4 (2008)
9. Faytak, M., Yu, A.C.L.: A typological study of the interaction between level tones, duration. In: *Proceedings of 17th ICPhS Conference*, Hong Kong, pp. 659–662 (2011)
10. Raux, A., Black, A.W.A.: A unit selection approach to F0 modeling and its application to emphasis. In: *Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 700–705 (2003)
11. Li, Y., Lee, T., Qian, Y.: F0 Analysis and modeling for cantonese text-to-speech. In: *Speech Prosody Conference*, Nara, Japan, pp. 169–180 (2004)
12. Nayak, P.C., Sudheer, K.P., Rangan, D.M., Ramasastri, K.S.: A neuro-fuzzy computing technique for modelling hydrological time series. *J. Hydrol.* **291**(2004), 52–66 (2004)
13. Inyang, U.G., Akinyokun, O.C.: A hybrid knowledge discovery system for oil spillage risks pattern classification. *Artif. Intell. Res.* **3**(4), 77–86 (2014)
14. Mayilvaganan, M.K., Naidu, K.B.: Comparison of membership functions in adaptive network-based fuzzy inference system (ANFIS) for the prediction of ground-water level of a watershed. *J. Comput. Appl. Res. Dev.* **1**(1), 35–42 (2011)
15. Iancu, I.: A Mamdani type fuzzy logic controller. In: Dadios, E.P. (ed.) *Fuzzy Logic - Controls Concepts, Theories and Application*, pp. 325–350. InTech Publishers, Vienna (2012)
16. Ekpenyong, M.E.: *Speech Synthesis for Tone Language Systems*. Ph.D. thesis, Uyo, in Supervision Collaboration with CSTR, Edinburgh (2013)
17. Ekpenyong, M., Udoh, E.O.: Tone modelling in Ibibio speech synthesis. *Int. J. Speech Technol.* **17**(2), 145–159 (2014)

Medical Computer Vision: Algorithms for Big Data
International Workshop, MCV 2015, Held in Conjunction
with MICCAI 2015, Munich, Germany, October 9, 2015,
Revised Selected Papers
Menze, B.; Langs, G.; Montillo, A.; Kelm, B.M.; Müller, H.;
Zhang, S.; Cai, W.; Metaxas, D. (Eds.)
2016, XV, 182 p. 70 illus., Softcover
ISBN: 978-3-319-42015-8