ADDITIVE MODELING OF ENGLISH F0 CONTOUR FOR SPEECH SYNTHESIS

Shinsuke Sakai

Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
sakai@sls.csail.mit.edu

ABSTRACT

In this paper, we present an approach to fundamental frequency contour modeling of English for speech synthesis, based on a statistical learning technique called *Additive Models* that was successfully applied to the modeling of Japanese F_0 contour previously. In an attempt to model English F_0 contour, we defined a three-layer additive model consisting of an intonational phrase component, a word-level component representing lexical stress types, and a pitch-accent component related to accented syllables. These component functions are estimated simultaneously using a *backfitting* algorithm derived from a regularized least-squares error criterion specified on the model with regard to the training data. The proposed method was trained and tested using the widely used ToBI-labeled speech corpus and promising results were obtained.

1. INTRODUCTION

Corpus-based concatenative approach to speech synthesis has been widely explored in the research community in recent years [1, 2, 3]. Intonation modeling, or generation of fundamental frequency (F_0) contour plays a crucial role in synthesizing natural sounding speech from input text. Target F_0 contour is generated using the features extracted from input text, and it is used either to modify the pitch of selected synthesis units, or in the unit selection where the discrepancies between target F_0 contour and the F_0 values of the synthesis units to be selected are attempted to be made as small as possible in the overall cost minimization through a search in the space of all available synthesis units. There has been a number of efforts in the context of F_0 contour generation for English speech synthesis in the past decade, such as dynamical system [4], linear regression-based approach [5], combination of parametric models with regression trees [6, 7], and the combination of regression trees and kernel smoother [2].

In this paper, we attempt to apply an F_0 modeling frame work that uses statistical learning technique named *Additive Models* [8, 9] to English F_0 modeling. Additive Models are a class of nonlinear regression models, which can be regarded as a generalization of linear models (or multiple linear regression). It and its extension by link function, called *Generalized Additive Modes* [9] have been applied to various statistical modeling practices such as weather forecast [10] and public health research [11], among others.

We previously proposed a framework of F_0 modeling using Additive Models and applied it to Japanese speech [12, 13]. The model basically consisted of the long-term intonational phrase component and the short-term accentual phrase component and we attained a quite encouraging result. After a success in Japanese, we are interested in applying it to English speech.

In the next section, we describe the additive F_0 modeling framework and the specific formulation for the modeling of English intonation. We then describe the experiments using a commonly used Boston University Radio News corpus, followed by a discussion.

2. ADDITIVE F_0 MODELING

We first review the additive F_0 modeling framework briefly, using a two-layer case for simplicity. The basic formulation for the F_0 contour is similar to previous work that models F_0 in a superpositional way, e.g., [14, 15]. In a two-layer additive modeling approach, the F_0 contour, Y, is regarded as the output of a statistical model that combines a fist-layer component, such as intonational phrase, g, and a second-layer component, such as accentual phrase, h:

$$\begin{aligned} Y &= f(I,U,A,V)+\epsilon \\ &= \alpha + g_I(U) + h_A(V) + \epsilon, \end{aligned} \tag{1}$$

where α is a constant, I is a discrete input variable that represents a type of the first layer component such as intonational phrase, and indexes the relevant function g_I. U is a continuous variable representing a time point relative to the starting point of the component of type I. Similarly, discrete variable A designates a type of the second layer component such as accentual phrase, and V represents a time point relative to the starting point of the component of type A. ϵ, is a random error term with zero mean. Figure 1 shows how the three terms form the entire F_0 contour function.

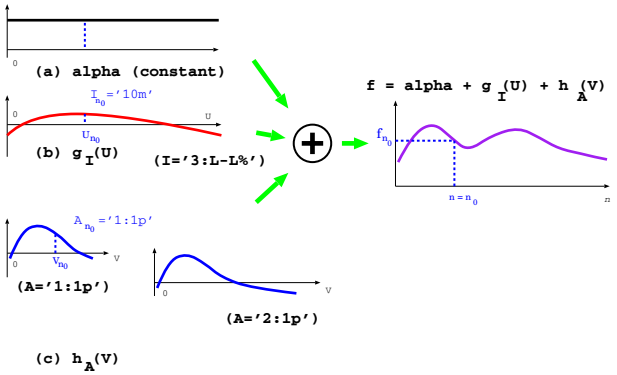

Fig. 1. A schematic diagram of the additive F_0 model $f(I,U,A,V) = \alpha + g_I(U) + h_A(V)$. A constant α and component functions g and h are summed up to form the F_0 contour f.

An advantageous characteristics of the additive model approach, as compared to previous work on superpositional F_0 modeling, is

that we do not have to assume any parameterized functional form. Instead, we assume a smoothness defined in terms of curvature, and use an estimation scheme derived from a least-squares error criterion with a regularization term, or roughness penalty [8, 9]. For the two-layer model, we define the penalized residual sum-of-squares (PRSS) error in the following form:

$$\begin{aligned} PRSS &= RSS + \lambda_g J(g) + \lambda_h J(h) \\ &= \sum_{n=1}^{N} \{y_n - \alpha - g_{i_n}(u_n) - h_{a_n}(v_n)\}^2 + \\ &\quad \lambda_g \sum_{s \in r(I)} \int g_s''(w)^2 dw + \lambda_h \sum_{t \in r(A)} \int h_t''(x)^2 dx, \quad (2) \end{aligned}$$

where $(i_n, u_n, a_n, v_n, y_n)$ $(n = 1, ..., N)$ are a set of training data corresponding to the variables (I, U, A, V, Y), and λ_g, λ_h are fixed smoothing parameters. $r(I)$ and $r(A)$ represents the set of possible values (or *range*) for I and A, respectively. The number of elements in a set will be denoted by vertical bars, e.g., $|r(I)|$ meaning the number of different values for I. The first term in (2) measures the closeness to the data, while the second and third terms penalize the curvatures in the functions, and smoothing parameters λ_g and λ_h establish a tradeoff between them. Large values of λ's yield smoother curves, while smaller values result in more fluctuation.

It can be shown that the minimizer of (2) is an additive cubic spline model, where g_I's and h_A's are *natural cubic splines* in the predictor variables U and V, with *knots*, or break points, at each of the unique values of (i_n, u_n) and (a_n, v_n). We can find the solution for the minimization problem for (2) with a *backfitting* algorithm [8], a simple iterative procedure depicted in Figure 2. To make the solution unique, we assume that $\sum_1^N g_{i_n}(u_n) = \sum_1^N h_{a_n}(v_n) = 0$, therefore α will be the overall mean of y_n $(n = 1, ..., N)$. In the algorithm, we apply a natural cubic-spline smoother matrix, e.g., $\mathcal{S}_i$, to the vector of partial residual, $\{y_{i,l} - \hat{\alpha} - \hat{h}_{a_{i,l}}(v_{i,l})\}_{l=1}^{N_i}$ to obtain a new estimate $\hat{g}_i$. Smoothing of partial residual is done for g_i's and h_a's in turn, using the current estimate of the other component function. The iteration is continued until the estimates $\hat{g}_i$'s and $\hat{h}_a$'s stabilize. Derivation of this backfitting algorithm from the penalized least square criterion is described in detail in [12].

(1) Initialize: $\hat{\alpha} = \frac{1}{N}\sum_{n=1}^{N} y_n$, $\quad \hat{g}_i \equiv 0$, $\hat{h}_a \equiv 0$, $\forall\, i \in r(I)$, $\forall\, a \in r(A)$

(2) Cycle: repeat (2g) and (2h) until the functions $\hat{g}_I$ and $\hat{h}_A$ change less than a prespecified threshold.

(2g) Partition the set of training data $\{(i_n, u_n, a_n, v_n, y_n) \mid n = 1, ..., N\}$, into $|r(I)|$ subsets $\{(i, u_{i,l}, a_{i,l}, v_{i,l}, y_{i,l}) \mid l = 1, ..., N_i\}$ $(i \in r(I))$, so that each training point has the same value of i if in the same subset. Note that $\sum_{i \in r(I)} N_i = N$.

For all $i \in r(I)$,

$$\hat{g}_i \leftarrow \mathcal{S}_i[\{y_{i,l} - \hat{\alpha} - \hat{h}_{a_{i,l}}(v_{i,l})\}_{l=1}^{N_i}].$$

(2h) Repartition the training data $\{(i_n, u_n, a_n, v_n, y_n) \mid n = 1, ..., N\}$ into $|r(A)|$ subsets $\{(i_{a,l}, u_{a,l}, a, v_{a,l}, y_{a,l}) \mid l = 1, ..., N_a\}$ $(a \in r(A))$, so that each training point has the same value of a if in the same subset. As before, $\sum_{a \in r(A)} N_a = N$.

For all $a \in r(A)$,

$$\hat{h}_a \leftarrow \mathcal{S}_a[\{y_{a,l} - \hat{\alpha} - \hat{g}_{i_{a,l}}(u_{a,l})\}_{l=1}^{N_a}].$$

Fig. 2. A backfitting algorithm for the two-layer additive F_0 model.

3. MODELING ENGLISH F_0

In our first attempt at modeling English $F0$, we opted to use the Boston University Radio News Corpus [16] to facilitate the comparison of experimental results with other approaches. This corpus is hand-annotated for prosody in the ToBI labeling framework [17]. We make use of information such as boundary tones, break indices, and pitch accent markers available with this corpus for our three-layer modeling consisting of intonational phrase (or IP)-level components, word-level components, and syllable-sized pitch accent components:

$$f(I, U, A, V, C, T) = \alpha + g_I(U) + h_A(V) + k_C(T) \quad (3)$$

For the IP feature I for indexing the first-layer component function, we use the combination of a ToBI boundary tones and the number of syllables in the intonational phrase. We define an intonational phrase as an interval between phrase boundary tones (which also coincide with a break index of 4). For example, I = (10, L-L%) represents an interval comprising ten syllables that ends with a boundary tone 'L-L%' (a "declarative" contour).

For the feature A that indexes the second-layer, word-level component function, we use the combination of the syllable length of the word and the lexical stress positions in the word. Additionally, 20 single-syllable function words shown in Figure 3 are treated as distinct values for A. For example, $A = (3, 1p3s)$ represents a three-syllable word with primary stress at the first syllable and the secondary stress at the third syllable.

```
a, an, are, as, at, by, for, from, if, in,
is, of, off, on, per, the, to, up, was, with
```

Fig. 3. Function words treated as separate categories

The third-layer component functions represent the effect of pitch accents and each component function spans one syllable interval. We make use of seven pitch accent types, H*, !H*, L*, H+!H*, L*+H, L+!H*, L+H*, and ¡none¿ to represent syllables with no accents. If the syllable is accented, indicator variable C value will designate one of the pitch accent types shown above. A syllable immediately preceding or succeeding an accented syllable within the same word is also assigned a distinct indicator. For example, $C = H*$ represents a syllable with peak accent, and $C = after : L*$ indicates a syllable immediately after a low-accented syllable.

4. EXPERIMENTS AND RESULTS

The model described above was trained and tested using the Boston University Radio News Corpus [16], speaker F2B. The corpus consists of approximately 45 minutes of radio news read aloud by a female speaker of American English. ToBI labels are assigned by

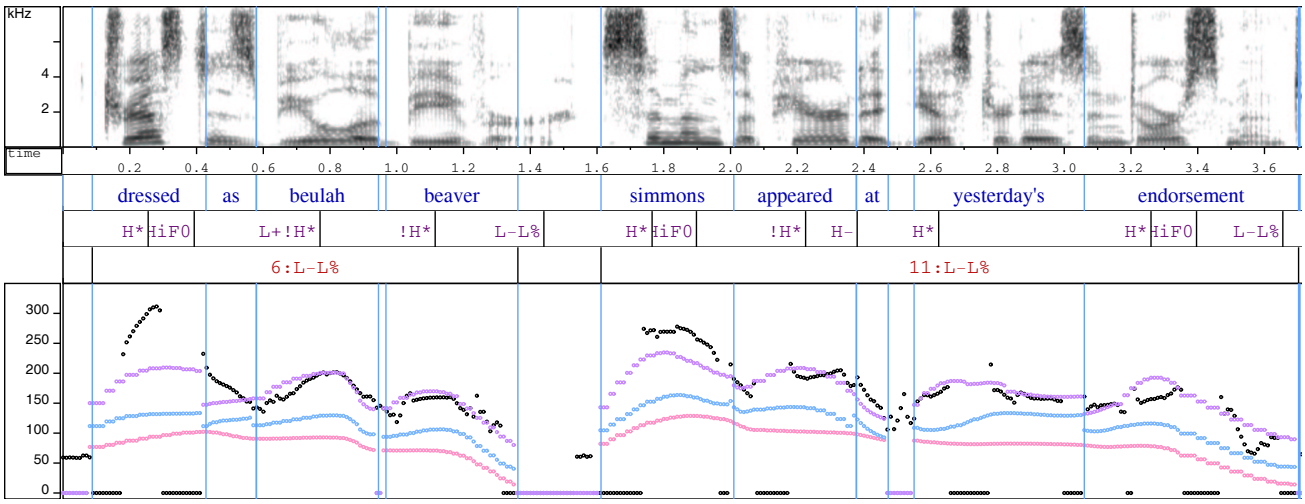

Fig. 4. An example of F_0 contour from the trained model, displayed with the raw F_0 from test data. The black dots are the raw F_0 data extracted from the test part of the corpus, and three light dots represent (1) the intonational phrase component $g_I(U)$, (2) the latter plus the word-level component $h_A(V)$, and (3) the latter with the pitch accent component $k_C(T)$ added furthermore, which is the overall output from the Additive F_0 model, from bottom to top. The constant α is divided to three parts and added to each curve to avoid overlaps to make it easy to see each curve.

hand to the corpus. We transcribed the corpus with syllable and word labels by performing a forced alignment using the MIT SUMMIT recognizer [18] with acoustic models adapted to this corpus. All 122 paragraphs that had full ToBI labels associated with them were divided into 110 paragraphs containing 12,704 syllables for training and the remaining 12 paragraphs with 1,863 syllables for testing. F_0 values were extracted from the corpus every 10ms using the Snack Sound Toolkit, a public domain toolkit developed at KTH [19]. The mean and standard deviation of the F_0 were 166.3 Hz and 47.1 Hz in the training data, and were 166.3 Hz, and 48.4 Hz in the test data.

Model estimation by backfitting was implemented in Scientific Python [20]. We estimated component functions g_I's, h_A's and k_C's in the log frequency, Mel scale, and linear frequency domain from the training data and compared the results. In all cases, the original pitch samples were normalized to have the same number of samples per syllable interval by linearly stretching or shrinking each syllable, before the estimation. About 15 iterations were enough for the convergence of backfitting iteration for this three-layer additive model. As a result, 64 distinct component functions for the first, intonational phrase layer, and 59 distinct component functions for the second word-level layer were obtained. 24 distinct component functions were estimated for the third, pitch accent layer. Figure 4 illustrates an example of F_0 contour from the F_0 model trained in the linear frequency domain, plotted with the raw F_0 data from the test part of the corpus.

As an objective evaluation, we measured the accuracy of F_0 contour production in terms of root mean square error (RMSE) and correlation coefficient (Corr) in the voiced portions of the data, which are widely used to measure the goodness of F_0 models [4, 5, 6, 7].

Table 1 shows the comparative results for the different domains where the additive $F0$ model is trained. As seen in the table, there was not a major difference in RMSE and Corr measures among three measures, although linear frequency and Mel-scale domains gave a little better results than log frequency domain, and linear frequency domain results were slightly better than the Mel scale on the test data.

Table 1. Comparative results in log frequency, Mel scale, and linear frequency domains

	Training		Test	
domain	RMSE	Corr	RMSE	Corr
log frequency	36.55	0.6355	38.19	0.6198
Mel scale	36.21	0.6394	37.62	0.6289
linear frequency	36.24	0.6392	37.59	0.6297

Table 2. Comparison of the results with other approaches

method	preprocessing	RMSE	Corr
proposed method	none	37.6	0.63
Sun [7]	smoothing,etc.	33.1	0.72
Dusterhoff et al. [6]	fit to model	34.3	0.60
Black et al. [5]	smoothing	34.8	0.62
Ross et al. [4]	error removal	34.7	NA

Table 2 summarizes the comparison of the results with various other approaches. Due to the different ways of splitting the corpus into training and test sets, as well as different way of computing the RMSE and Corr measures, we have to be cautious in comparing these results. It can be seen, nevertheless, that the proposed method yields as good correlation coefficient as most of the other approaches except for [7], while RMSE is 3.5 to 5 points worse than others. Considering, however, the fact that the other results applies various kinds of smoothing techniques and corrections to the raw data, to which model output is compared, it is probable that the current results, which compares the model output with the raw F_0 data with no correction or smoothing, is as good as these state-of-the-art results.

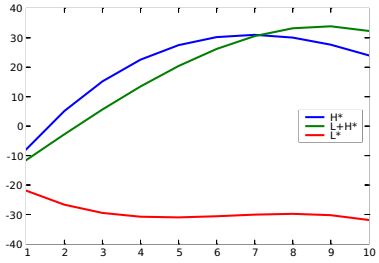

Fig. 5. Pitch accent component functions for accent types H*(peak), L+H*(rising peak), and L*(low). We can see, for example, that there is some 60 Hz difference between L* and H* at the peak.

5. DISCUSSION

The proposed method is advantageous in that it requires no preprocessing to smooth the raw F_0 extracted from the corpus, since the method comprises a "smoother" (cubic smoothing spline) as one of its building blocks. It shares a nice property of non-parametric regression methods, with such technique as regression trees, that it can be easily applied to various different languages as far as effective set of features are supplied to the training algorithm.

It is considered better than regression trees in that model output is already smooth and no further smoothing is required when it is used for F_0 contour prediction. Its decomposition to multiple additive layers is also a nice characteristics in that it suffers less from the data sparseness compared to regression trees, when total combination of the feature value instances grows exponentially with the number of different feature types to be used. Interpretability of its components may be another advantage of the additive F_0 models. For example, by plotting component functions at the pitch accent layer as in Figure 5, we can see the way different pitch accent component functions are learned with an effective difference among each other from the training data, in addition to knowing the effect of introducing this layer into the model just by the RMSE and Corr measures.

On the other hand, since current additive modeling utilizes the continuous component functions, it does not have a direct way to recover if some component functions types are missing from the training data, while regression trees can always yield some answer from the leaf node it reached anyway. However, this may not be a major defect when the developer oneself is in a position to design and collect the speech corpus.

6. CONCLUSION AND FUTURE WORK

In this paper, we have proposed the application of multi-layer additive models approach to English F_0 modeling for speech synthesis, and achieved a promising results. We plan to incorporate the F_0 measures predicted by the model, as one of the target measures to evaluate the goodness of the corpus units, into our next generation speech synthesis system we are currently developing.

7. ACKNOWLEDGMENT

The author would like to thank Jim Glass for the helpful comments, and T.J. Hazen for his help in transcribing the corpus. This research was funded in part by the SLS Affiliate Program.

8. REFERENCES

[1] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. ICASSP '96*, 1996, pp. 373–376.

[2] E. Eide et al., "Recent improvements to the ibm trainable speech synthesis system," in *Proc. ICASSP 2003*, pp. I–708–I–711.

[3] M. Chu, H. Peng, Y. Zhao, Z. Niu, and E. Chang, "Microsoft Mulan – a bilingual TTS system," in *Proc. ICASSP 2003*, pp. I–264–I–267.

[4] K. Ross and M. Ostendorf, "A dynamical system model for generating fundamental frequency for speech synthesis," *IEEE Trans. SAP*, vol. 7, pp. 295–309, 1999.

[5] Alan W. Black and Andrew J. Hunt, "Generating f0 contours from tobi labels using linear regression," in *Proc. ICSLP '96*, pp. 1385–1388.

[6] K. Dusterhoff, A. Black, and P. Taylor, "Using decision trees within the tilt intonation model to predict F0 contours," in *Proc. EUROSPEECH'99*, pp. 1627–1630.

[7] X. Sun, "F0 generation for speech synthesis using a multi-tier approach," in *Proc. ICSLP'02*, pp. 2077–2080.

[8] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2001.

[9] T. Hastie and R. Tibshirani, *Generalized Additive Models*, Chapman and Hall, 1990.

[10] R. Vislocky and J. Fritsch, "Generalized additive models versus linear regression in generating probabilistic mos forecasts of aviation weather parameters," *Weather and Forecasting*, vol. 10, pp. 669–680, 1995.

[11] F. Dominici, A. McDermott, S.L. Zeger, and J.M. Samet, "On the use of generalized additive models in time-series studies of air pollution and health," *Am. J. Epidemiol.*, vol. 56, no. 3, pp. 193–203, 2002.

[12] S. Sakai and J. Glass, "Fundamental frequency modeling for corpus-based speech synthesis based on a statistical learning technique," in *Proc. ASRU 2003*, pp. 712–717.

[13] S. Sakai, "F0 modeling with multi-layer additive modeling based on a statistical learning technique," in *Proc. SSW5*, 2004, pp. 151–154.

[14] H. Fujisaki and K. Hirose, "Analysis of voice fundamental frequency contours for declarative sentences of Japanese," *Journal of the Acoustical Society of Japan(E)*, vol. 5, no. 4, pp. 233–241, 1984.

[15] M. Abe and H. Sato, "Two-stage F0 control model using syllable based F0 units," in *Proc. ICASSP'92*, pp. 53–56.

[16] M. Ostendorf, P. J. Price, and S. Shattuck-Hufnagel, "The boston university radio news corpus," Tech. Rep. ECS-95-001, Boston University, Mar. 1995.

[17] K. Silverman et al., "Tobi: A standard for labeling english prosody," in *Proc. ICSLP '92*, pp. 867–870.

[18] J. Glass, "A probabilistic framework for segment-based speech recognition," *Computer Speech and Language*, vol. 17, pp. 137–152, 2003.

[19] `http://www.speech.kth.se/snack/`.

[20] `http://www.scipy.org/`.