Speech Prosody 2016
31 May - 3 Jun 2106, Boston, USA

Phonetically conditioned prosody transplantation for TTS: 2-stage phone-level unit-selection framework

Mythri Thippareddy[1], *M. G. Khanum Noor Fathima, D. N. Krishna, A. Sricharan, V. Ramasubramanian*[2]

PES Institute of Technology, South Campus, Bangalore, India
[1]mythri.thippareddy@gmail.com, [2]v.ramasubramanian@pes.edu

Abstract

We propose a framework of prosody transplantation for TTS, namely, 2-stage phone-level unit-selection, to transfer the prosody from a 'target' prosody database onto a conventional TTS output unit-sequence. The framework employs 'phonetic conditioning', wherein target prosody-profiles are identified conditioned on their underlying phonetic content over variable length time-scales that tend to be as long as possible. In this 2-stage unit-selection framework, the units determined in a 1st-stage conventional unit-selection are mapped to units in a 2nd-stage prosodic-style database via a phone-level unit-selection, which retrieves units from the 2nd-stage prosody-database with associated prosody (representing the prosodic-style of the 2nd stage prosodic-database) and the selected prosody is further incorporated on to the 1st-stage units. This framework was recently proposed by us with early qualitative results indicating the viability of the approach. In this paper, we elaborate on this approach and characterize the performance of the proposed frameworks using various objective measures using prosodic ground truth, and with respect to the parameters of the system, and show the viability of the proposed approach to realize the target prosody very effectively.

Index Terms: TTS, prosody transplantation, phonetic-conditioning, 2-stage unit-selection

1. Introduction

Text-to-speech synthesis (TTS) systems presently provide speech of such high quality as considered adequate for most applications to date, such as spoken dialog systems, car navigation, mobile applications, talking book etc. While this assumes that such TTS output have adequate naturalness considered acceptable, there still remains a problem in TTS speech quality that is currently considered difficult, not yet solved adequately, and presenting a high degree of challenge - this is the problem of incorporation of prosody in the synthesized speech, as can be derived only from the input text. Prosody effects a broad range of diverse aspects that the synthesized speech can have, such as naturalness at one end, to expressive or emotional speech at the other end, with varying degrees of prosodic effects in between, dictated by the specific task in question where the TTS is functional, such as for example, spoken dialog systems, where certain kinds of interaction with the user may require special prosody (stressed speech of some part of the retrieved information, query clarification, etc.). Apart from such application driven need for appropriate prosody control in TTS, the problem of general expressive prosody, with a certain extent of control on the degree and style of prosody still remains a hard problem with high academic value and importance [1, 2].

The most direct approach to incorporating prosody in a TTS system for a given input text is to simply 'predict' the prosody from the input text. This has attracted considerable attention (see for example a review and recent approaches in [3]) and has solutions ranging from rule-based approaches (that use syntactic and semantic structures and information in the text to arrive at an appropriate prosody prediction) to data-driven machine-learning methods such as CART or neural-network based functional mappings.

In this paper, we examine a framework for prosody transplantation onto speech (synthesized by a 1^{st}-stage concatenative synthesizer) based on selecting the desired prosody from a 'prosody database' conditioned on a phonetic match between the input phonetic units and the units selected from the prosody database. The primary basis of this framework is as below.

It is well known that a conventional unit-selection based concatenative synthesizer produces what is referred to as 'as is' prosody of the 1st stage unit-database, given that the selected units will have a latent associated prosody, which gets synthesized 'as is' when no further prosodic modification is performed [4]. Any such prosodic modification (done as a post-processing) on the synthesized speech again has to be derived from the input text or G2P output, and the onus of realizing the appropriate prosody hence rests on the degree to which such a desired prosody can be predicted and derived from the input text. Most often, it is desired to have a prosody that is different from this 'as is' prosody of the 1st stage unit-selection, such as when a particular prosodic-style needs to be incorporated in to the synthesized speech. Again, predicting such variety of prosody from input text would not be readily feasible, since the desired target prosody may not be predictable from the given text (which will typically yield some latent prosody associated with the text), nor is it feasible to change the 1st stage unit-database to the target prosody so that the 'as is' prosody now yields the target prosodic style, for the simple reason that the 1st stage unit-database needs to be sufficiently large for unit-coverage, and having multiple such large unit-databases - one for each of the target prosody is infeasible and inelegant.

Hence, the framework we propose and examine here takes the approach of using a 2nd stage 'prosodic database' which has speech in the desired prosody, and which is searched by a 2nd-stage unit-selection for appropriate units that match the output of the 1st stage unit-selection, so that, subject to this 'phonetic conditioning' (i.e. matching of the units output by the 1st stage and units selected from the prosodic-database to varying degrees depending on the efficacy of the 2nd stage unit-selection), the prosody associated with the selected units will have the desired target prosody which can then be transplanted on to the units derived from the 1st stage output. In essence, the higher the degree of match (in terms of exact phonetic match)

doi: 10.21437/SpeechProsody.2016-160

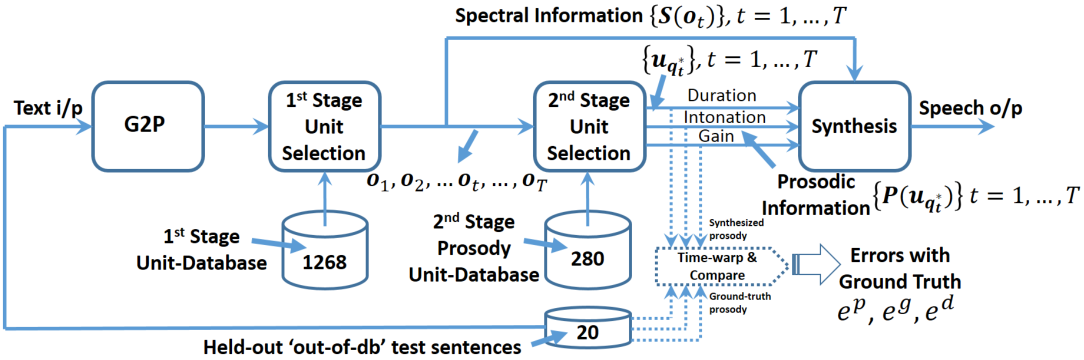

Spectral Information $\{S(o_t)\}, t = 1, \dots, T$
$\{u_{q_t^*}\}, t = 1, \dots, T$
Text i/p
G2P
1st Stage Unit Selection
2nd Stage Unit Selection
Duration
Intonation
Gain
Synthesis
Speech o/p
$o_1, o_2, \dots o_t, \dots, o_T$
Prosodic Information $\{P(u_{q_t^*})\}\ t = 1, \dots, T$
1st Stage Unit-Database
1268
2nd Stage Prosody Unit-Database
280
Synthesized prosody
Time-warp & Compare
Errors with Ground Truth
e^p, e^g, e^d
Ground-truth prosody
20
Held-out 'out-of-db' test sentences

Figure 1: *Schematic of proposed 2-stage unit-selection framework for prosody transplantation*

and longer the contiguity of the match, the prosody associated with the resultant units from the prosodic-database, will be the desired prosody in the sense that 'it is the same text (i.e., phonetic sequence) as the input, but spoken with the prosody-style of the target prosodic-database', thus serving the objective of transplantation on to the input units to realize the target prosody.

A schematic of this framework is shown in Fig. 1. In this framework, the 1st-stage unit-selection is the conventional part of a concatenative TTS (a mixed-Viterbi formalism proposed by us recently [5]). It is driven by the output of the G2P which generates the target unit sequence vectors from the given input text. The 1st-stage unit-selection then selects appropriate units from the 1st-stage unit-database, which can be used to drive a concatenative synthesizer such as LPC, PSOLA or MBROLA. However, the system does not have a synthesizer in the 1st-stage. Instead, the spectral component of the 1st-stage units selected from the 1st-stage unit-database are sent directly to a synthesizer at the 2nd-stage, for a subsequent combination with the prosodic information derived from the 2nd-stage unit-selection. The 2nd-stage unit-database carries the desired prosody and could be spoken by the same or different speaker as the 1st-stage unit-database. For instance, the 2nd-stage unit-database could be in different styles as desired in the synthesized speech of the overall system, such as, news-reading style, story-telling styles, or various emotional styles etc. It is important to note that the 1st-stage unit-database is of a typical size as is required for high quality synthesis, while the 2nd-stage unit-selection could be smaller, being able to adequately provide the prosodic-stylistic aspects of a given unit-sequence.

This is more or less the prime motivation for the prosody transplantation in an earlier work [6] which was further examined and refined by us [7] in the form of incorporating prosody from the 2nd stage 'prosody database' that overrides the 'as is' prosody from the 1st stage synthesis. In Prudon et al. [6], the prosody was obtained by 'selection', i.e., a single stage unit-selection from a specific prosody-database, directly for an input text. The selected prosodic parameters, primarily the duration of units, intonation and gain contours, were combined with the phonetic information from the text (derived via G2P) and synthesized by a diphone synthesis system using MBROLA. By this, the authors completely circumvented syntactic analysis and syntactic-prosody rules, and used the prosodic part of the corpus obtained by such a unit selection.

2. Phone-level unit-selection

Here, we describe the 2nd stage prosody transplantation in Fig. 1 in the form of a 'phone-level' unit-selection whose early principles and results were reported in [7]. Let the unit-sequence output of the 1st-stage unit-selection be the phonetic units $O = (\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_t, \dots, \mathbf{o}_T)$. Each of these could be vectors with components indicating the primary phonetic unit label and any other contextual features that characterize the unit selected from the 1st-stage unit-database, and that is considered relevant for the 2nd-stage prosody-selection. Note that a vector $\mathbf{o}_t$ in this sequence is derived from the 1st-stage unit-database, and not from the input text, and hence can afford to have features derived by signal-processing on the 1st-stage unit-database.

Let the 2nd-stage unit-database be represented as a sequence of N units, $\mathcal{U} = (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_i, \dots, \mathbf{u}_N)$ with durations of the associated acoustic segments as $l_1, l_2, \dots, l_i, \dots, l_N$. For a given input unit-sequence $O = \mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_t, \dots, \mathbf{o}_T$), the 2^{nd} stage 'phone-level' unit-selection finds the optimal unit sequence $U^* = (\mathbf{u}_{q_1^*}, \mathbf{u}_{q_2^*}, \dots, \mathbf{u}_{q_t^*}, \dots, \mathbf{u}_{q_T^*})$ where $\mathbf{u}_{q_t^*} \in \mathcal{U}$ by determining the optimal unit indices $Q^* = (q_1^*, q_2^*, \dots, q_t^*, \dots, q_T^*)$ that minimizes the distortion between O and any U (from the 2nd stage unit-database) as given by,

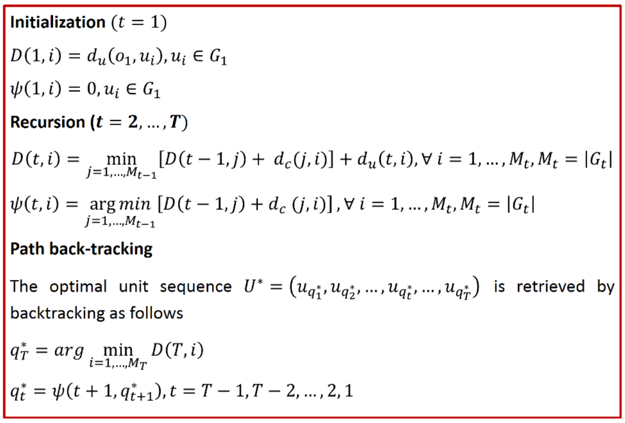
$$Q^* = \arg\min_{Q}(1-\alpha)\sum_{t=1}^{T} D_u(\mathbf{o}_t, \mathbf{u}_{q_t}) + \alpha\sum_{t=2}^{T} D_c(q_{t-1}, q_t) \tag{1}$$

Here, $D_c(q_{t-1}, q_t)$ is the concatenation costs that takes on a value of 0 if $u_{q_{t-1}}$ and u_{q_t} are contiguous in the 2nd-stage unit database, and takes on a value of the Euclidean distortion between the last frame of the acoustic segment annotated as $u_{q_{t-1}}$ and the first frame of the acoustic segment annotated as u_{q_t}, if they are not contiguous.

2.1. Grouped Viterbi unit-selection

In the grouped-Viterbi realization of the above unit-selection, we first form groups $G_1, G_2, \dots, G_t, \dots, G_T$, with group G_t corresponding to input vector $\mathbf{o}_t$, and consisting of all units in the unit-database $\mathcal{U}$ that share some feature components, i.e. for instance, if an input vector $\mathbf{o}_t$'s primary label is a particular phoneme, the group G_t will have all units in $\mathcal{U}$ whose primary label is also the same phoneme, i.e., the group G_t is simply a collection of all phonemic units from the unit-database whose primary labels are same as that of the input vector $\mathbf{o}_t$. The need to define this in a general sense, is to allow scope for a flexible definition of the group by way of allowing various unit definitions such as is outlined in the section on 'Unit definition'.

The 2nd-stage unit-selection performs a grouping-based unit-selection, as outlined in the algorithm shown in Fig. 2. Let i denote the index of the current unit $\mathbf{u}_i$ (in group G_t) being analyzed for the input unit $\mathbf{o}_t$, and let j be the index which spans all the units in group G_{t-1} considered for the previous input unit $\mathbf{o}_{t-1}$. Let the target cost between $\mathbf{o}_t$ and $\mathbf{u}_i$ be denoted by $d_u(t, i)$. Let the concatenation cost defined between two units $\mathbf{u}_j$ and $\mathbf{u}_i$ in the unit database be denoted by $d_c(j, i)$. Let $D(t, i)$ denote the accumulated cost of the best path reaching the co-ordinate (t, i) (i.e. unit $\mathbf{u}_i$ at time t), and let $\psi(t, i)$ record the unit in G_{t-1} at $t - 1$ that is part of the best path reaching (t, i). $d_c(j, i)$ is the concatenation cost that takes on a value of 0 if u_j and u_i are contiguous in the 2nd-stage unit database, and takes on a value of the Euclidean distortion between the last frame of the acoustic segment annotated as u_j and the first frame of the acoustic segment annotated as u_i, if they are not contiguous.



Initialization $(t = 1)$

$D(1, i) = d_u(o_1, u_i), u_i \in G_1$

$\psi(1, i) = 0, u_i \in G_1$

Recursion ($t = 2, \dots, T$)

$D(t, i) = \min_{j=1,\dots,M_{t-1}} [D(t-1, j) + d_c(j, i)] + d_u(t, i), \forall\, i = 1, \dots, M_t, M_t = |G_t|$

$\psi(t, i) = \arg min_{j=1,\dots,M_{t-1}} [D(t-1, j) + d_c\,(j, i)]\,, \forall\, i = 1, \dots, M_t, M_t = |G_t|$

Path back-tracking

The optimal unit sequence $U^* = \left(u_{q_1^*}, u_{q_2^*}, \dots, u_{q_t^*}, \dots, u_{q_T^*}\right)$ is retrieved by backtracking as follows

$q_T^* = arg \min_{i=1,\dots,M_T} D(T, i)$

$q_t^* = \psi(t+1, q_{t+1}^*), t = T-1, T-2, \dots, 2, 1$

Figure 2: *Viterbi algorithm for finding optimal unit-selection from grouped units*

The above algorithm selects optimal units based on equal weights assigned to target cost and concatenation cost. A more general unit-selection uses weighting of these two costs, so as to control the degree of unit-matching against the degree of contiguous units that can be selected reflecting naturally occurring consecutive units in the unit-database, with intrinsically high degree of natural co-articulation. This calls for modification of the recursions in Fig. 2; let α be the weight assigned to the concatenation cost; then the modified equation for $D(t, i)$ is,

$$D(t, i) = \min_{j=1\dots,M_{t-1}} \{D(t-1, j) + \alpha \cdot d_c(j, i)\} + (1-\alpha) \cdot d_u(t, i) \tag{2}$$

The above yields the optimal unit sequence U^* from the 2nd-stage corresponding to the input unit-sequence O from the 1st-stage unit-selection, under the constraints imposed by α that trades unit-cost and concatenation cost. For any given α, the retrieved unit sequence U^* represents the units in the 2nd unit-database that are the best match to the input unit sequence in terms of the feature vectors that define $\mathbf{o}_t$ and $\mathbf{u}_{q_t^*}$ (in addition to the concatenation cost constraints between the chosen units in U^*). Therefore, once U^* is identified, it ensures that the phonetic content of the 1st-stage output is satisfactorily retrieved; given that the 2nd unit-database has a distinct prosodic style, and as shown in [6] and [7], it can be expected that the prosody associated with U^* will be typical of the style of the 2nd-stage unit-database, under the condition that the underlying phonetic units match exactly (which is ensured by the grouping mentioned above) and more importantly, by the degree of contiguity which a choice of α allows (i.e., with $0 \leq \alpha \leq 1$, larger values of α yield units with longer contiguity, in the process, retrieving a long naturally articulated prosody associated with such long sequences of units).

The prosodic transplantation itself is done as follows (and as is shown in Fig. 1): If $S(\mathbf{o}_t)$ denotes the spectral part of $\mathbf{o}_t$ (and hence the phonemic identity of unit $\mathbf{o}_t$) and $P(\mathbf{u}_{q_t^*})$ denotes the prosodic part of $\mathbf{u}_{q_t^*}$, then the synthesizer at the 2nd-stage can transplant the prosodic-style of the 2nd-stage unit-database onto the phonetic information of the input text, by synthesizing speech as a combination of $S(\mathbf{o}_t)$ and $P(\mathbf{u}_{q_t^*})$. For instance, if $S(\mathbf{o}_t)$ represents the speech waveform of unit $\mathbf{o}_t$ and $P(\mathbf{u}_{q_t^*})$ the set of parameters (duration of unit $\mathbf{u}_{q_t^*}$ and its gain and pitch contours), these in turn can be used in a TD-PSOLA or STRAIGHT kind of techniques to perform time-scale and pitch-scale modifications to yield the desired prosody transplantation.

3. Unit definition

We outline here some scenarios that present itself in performing a prosody transplantation as above, particularly with respect to the definition of units. The degree of phonetic conditioning (i.e., match between $\mathbf{o}_t$ and any unit $\mathbf{u}_{qt} \in G_t$) is determined by: a) the granularity of the units - spanning fine- to course-grained representations, such as, i) acoustic representation (e.g. MFCC) at one extreme (as can be derived from the signal, i.e., from the 1st stage output and the 2nd stage unit-database) representing the strongest phonetic-conditioning, ii) fine-category phoneme class (as in the results in the next section) or, iii) coarse-grained, such as various levels of broad-category phoneme classes, b) the contextual span of the unit, in any of the granularity above, such as various extents of left- and right-context information, consequently effecting how the group G_t is populated and reflecting in the unit cost and controlling the degree of conditioning through context-dependency, c) the inclusion of prosodic features into the unit vector definition, such as duration, intensity and pitch with corresponding weighting in the unit-cost, providing selective control on their relative influence on the resultant prosodic-copy and, d) the generalization of the notion of 'phonetic'-conditioning to other types of conditioning, such as long-span linguistic-features (as can be derived from text, as is typically done for various text-to-prosody mapping [8]), structural features [9] or acoustic-features (as can be derived from the signal itself), leading beyond only prosodic 'style' copy, to other prosodic-effects such as narrow-focus word-stress and, in general, expressive speech closely governed by the text content.

4. Experiments and results

We present results on an Indian language 'Kannada' (one of the 22 scheduled languages of India) with the target prosody as 'story-telling' style. In our experimental framework, the 1st stage unit-selection system uses the mixed-Viterbi formalism proposed by us recently [5], and the 2nd stage unit-selection is as described here. The 1st stage unit-selection uses a neutral prosody-style database of 1268 sentences ($\sim$ 4 hrs) and the 2nd stage prosodic-database is of 'story-telling' style with 300 sentences ($\sim$ 45 min). Of these 300, as shown in Fig. 1, 280 sentences are used as the 2nd stage unit-database, and the remaining 20 are used as 'out-of-database' (out-of-db) sentences, so that these provide the ground-truth prosody against which to compare the synthetic prosodic contours, i.e., pitch (p), gain (g) and duration (d) profiles, to yield the respective errors e^p, e^g and e^d computed between time-normalized synthesized and ground-truth prosody contours. 20 sentences from within the 280 sentence prosody-database are used as 'in-database' (in-db) sentences, to be able to calibrate the system, and provide the baseline performance for the out-of-db sentences.

Effect of α: The weighting factor α (trading off unit cost against concatenation cost), controls the degree of match between the primary phonetic identity between the units $\mathbf{o}_t$ and $\mathbf{u}_{q_t^*}$. Values of α close to 0 cause fragmentation of the unit sequence U^*, while values of α close to 1 cause highly contiguous groups of units in U^*. This in turn can be expected to result in perceptually acceptable contiguous and naturally continuous pitch and gain contours (and durations) of the prosodic information from the 2nd-stage unit-database.

To this end, Fig. 3 shows the effect of α for in-db and out-of-db cases on e^p, e^g and e^d (averaged over the 20 sentences in each case). The in-db errors are close to 0 for $\alpha > 0$, as it should be, since the unit-selection selects a single long sequence of the in-db units. For $\alpha = 0$, the contiguous units are not selected, since the concatenation cost (weighted by α) is 0, and the decoding selects units arbitrarily from the respective unit-groups, leading to fragmentation and mismatch of the retrieved acoustic segments. This causes the higher errors for all the three errors at $\alpha = 0$. The out-of-db errors are higher than the in-db errors understandably, and interestingly, the errors show a dip at $\alpha = 0.2$ indicating that an optimal balance has been found between fragmentation ($\alpha = 0$) but localized errors (over small duration units) and long contiguity (higher αs) with an associated error distributed over long unit sequences, if the wrong prosodic-unit has been selected.

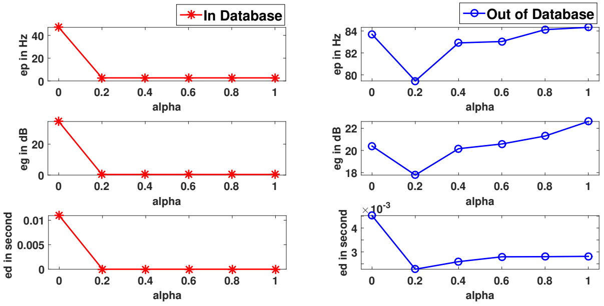


Figure 3: *Effect of α on e^p, e^g and e^d for in-db and out-of-db sentences*

Pitch-contour comparison: Fig. 4 is a panel of 6 plots for one out-of-db sentence, showing the ground-truth and synthesized (and time-warped) pitch contours, for varying $\alpha = 0$ to 1. The decrease in fragmentation for increase in α, but with possible associated longer-spread errors can be noted. In general the pitch contours bear a close match to the ground truth pitch contours, validating the premise that when the degree of phonetic-sequence match is good, the corresponding associated prosody turns out to be a good copy of the prosodic-style in which these sequence of units 'would' have been spoken in the target prosody style, as evidenced by the prosody of the synthesized speech, with respect to the ground truth of the held-out out-of-db sentences.

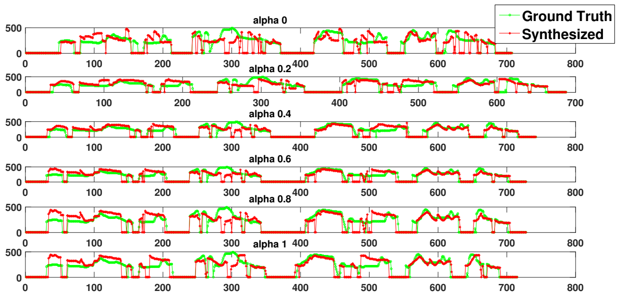


Figure 4: *Pitch contours - synthesized and ground-truth - for various α for one out-of-db sentence*

Effect of prosody-database size: As noted earlier, one of the appealing aspects of the present framework is the use of a small-footprint prosody-database, in the form of annotated prosodic-information only. To bring out the effect of the size of the prosody database, we show in Fig. 5 the pitch, gain and duration errors (averaged over the 20 out-of-database sentences) for different prosody-database sizes ranging from 100 to 280. The significant impact that the database size has on decreasing these errors can be noted. It can also be noted that the footprint of this prosodic-database can indeed be considerably smaller (15-45 min) than that of a conventional unit-database ($\sim$ 4 hrs).

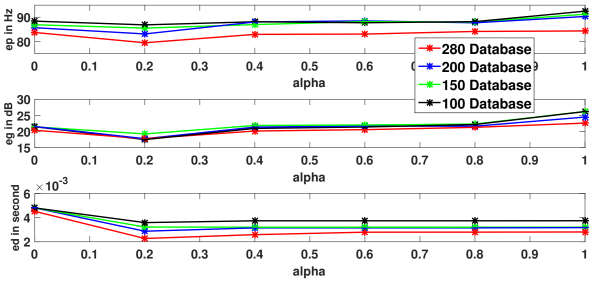


Figure 5: *Pitch, gain and duration errors - e^p, e^g and e^d - for various prosody-database sizes: 100 to 280*

2nd stage prosody override: It was noted earlier that the 1^{st} stage produces speech with an 'as is' prosody of the 1^{st} unit-database and that the proposed 2^{nd} stage unit-selection overrides this 'as is' prosody to the desired target prosody (of the 2^{nd} stage prosodic-database, e.g. 'story-telling' style here). To validate this premise, we show in Fig. 6, the pitch contours of i) one sentence from the out-of-db sentences (ground-truth, green-line), ii) its synthesized version from the 2^{nd} stage (red-line) and iii) 1st stage output of this sentence, i.e. when its text is fed as input to the 1st stage mixed-Viterbi unit-selection synthesis [5] (using a 1^{st}-stage unit-database of 1268 sentences in 'neutral' prosody) (blue-line). It can be noted from this figure that the above said premise is indeed true, given that the 2^{nd}-stage synthesis is i) close to the ground-truth pitch contour (desired prosody) and ii) significantly different from the 1^{st} stage output pitch contour, indicating a clear intonation override. This bears out in listening tests also, with the 2nd stage synthesized output (red-line), i.e., with the proposed prosody transplantation, being clearly in the 'story-telling' style of the prosody database (i.e., very close to the ground truth prosody (green-line)), and distinctly different from the 1^{st} stage prosody (blue-line).

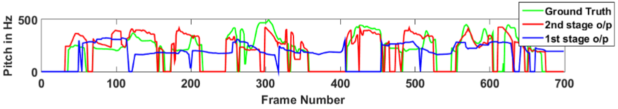


Figure 6: *Pitch contours of one sentence from out-of-db - Ground truth, its 2^{nd} stage o/p and 1^{st} stage o/p*

5. Conclusion

We have proposed a 2-stage unit-selection framework for prosody transplantation in TTS. We showed that the retrieved units from the 2nd-stage unit-database have an associated prosody in the prosodic-style of the 2nd-stage unit-database which can be further transplanted on to the spectral information of the 1st-stage unit-selection. We have presented extensive experimental results with quantified performance using objective measures that validate the viability of the framework.

6. Acknowledgments

The authors thank the Department of Information Technology, Ministry of Communication and Technology, Government of India, for funding the TTS Consortium project, 'Development of Text-to-Speech synthesis for Indian Languages Phase II', Ref. no. 11(7)/2011HCC(TDIL) under which this work is part of 'prosody-control' focus topic for PESIT-BSC in 2015-16.

7. References

[1] M. Bulut, S. Narayanan and L. Johnson. Synthesizing expressive speech overview: challenges, and open questions. Ch. 9, in Text to Speech Synthesis: New Paradigms and Advances, ed. By S. Narayanan, A. Alwan (Pearson Education, Prentice Hall, pp. 197-223, 2005.

[2] E. Eide, R. Bakis, W. Hamza and J. F. Pitrelli. Toward expressive synthetic speech. Ch. 11, in Text to Speech Synthesis: New Paradigms and Advances, ed. By S. Narayanan, A. Alwan (Pearson Education, Prentice Hall, pp. 241-272, 2005.

[3] K. Sreenivasa Rao Predicting Prosody from Text for Text-to-Speech Synthesis. Springer Brief, 2012.

[4] J. van Santen, T. Mishra and E. Klabbers. Prosodic processing. Ch. 23, pp. 471 - 487, in Springer Handbook of Speech Processing, eds. Benesty, Sondhi and Huang, 2008.

[5] Mythri Thippareddy, C. Mahima, S. Adithya, Sunil Rao and V. Ramasubramanian. G2P-free grapheme-to-speech synthesis: UTF-8 based automatic unit-database annotation and mixed Viterbi unit-selection. Proc. O-COCOSDA '15, Shanghai, China, Oct. 2015.

[6] R. Prudon, Christophe DAlessandro and P. B. de Mareuil. Unit selection synthesis of prosody: evaluation using diphone transplantation. Ch. 10, in Text to Speech Synthesis: New Paradigms and Advances, ed. By S. Narayanan, A. Alwan (Pearson Education, Prentice Hall, 2005), pp. 225-239.

[7] Mythri Thippareddy and V. Ramasubramanian. Prosody transplantation using unit-selection: Principles and early results. IEEE CONECCT 15, Bangalore, Jul 2015.

[8] H. Mixdorff. An integrated approach to modeling German prosody. Post-doc Thesis, Universitats verlag, Dresden, Germany, 2002.

[9] K. Raghava Krishnan. Prosodic Analysis of Indian Languages and its Applications to Text to Speech Synthesis. M.S. Thesis, Dept. Electrical Engineering, IIT-Madras, Chennai, India.