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

The Common Prosody Platform (CPP) — where Theories of Prosody can be Directly Compared

Santitham Prom-on [1,2], *Yi Xu* [2], *Wentao Gu* [3], *Amalia Arvaniti* [4], *Hosung Nam* [5], *D. H. Whalen* [6,7]

[1] Department of Computer Engineering, King Mongkut's University of Technology Thonburi, Thailand.
[2] Department of Speech, Hearing and Phonetic Sciences, University College London, UK.
[3] Department of Linguistic Science and Technology, Nanjing Normal University, China
[4] School of European Culture and Languages, University of Kent, UK
[5] Department of English Language and Literature, Korea University, Korea
[6] Haskins Laboratories, USA
[7] City University of New York, USA

santitham@cpe.kmutt.ac.th, yi.xu@ucl.ac.uk, wtgu@njnu.edu.cn, a.arvaniti@kent.ac.uk, nam@haskins.yale.edu, whalen@haskins.yale.edu

Abstract

This paper introduces the Common Prosody Platform (CPP), a computational platform that implements major theories and models of prosody. CPP aims at a) adapting theory-specific assumptions into computational algorithms that can generate surface prosodic forms, and b) making all the models trainable through global optimization based on automatic analysis-by-synthesis learning. CPP allows examination of prosody in much finer detail than has been previously done and provides a means for speech scientists to directly compare theories and their models. So far, four theories have been included in the platform, the Command-Response model, the Autosegmental-Metrical theory, the Task Dynamic model, and the Parallel Encoding and Target Approximation model. Preliminary tests show that all the implemented models can achieve good local contour fitting with low errors and high correlations.

Index Terms: speech prosody, theory comparison, software package, parametric modeling

1. Introduction

Prosody research has seen significant development in recent decades, and numerous theories and computational models have been proposed. However, many fundamental issues remain unresolved and some are still under heated debate. A key reason for the current stalemate is the lack of means to compare the theories and models directly. While many of them offer reasonably good explanations of various prosodic phenomena, and some can even generate fairly good fit to real speech data [1-3], it has been difficult to compare them directly under similar conditions, due to the lack of suitable means. So far only a few serious attempts have been made to our knowledge, e.g., [4-6]. To accelerate progress in prosody research, we have been developing a *Common Prosody Platform* (CPP) to host a set of trainable computational models, each implementing a major theory of prosody. Each model will consist of A) a computational realization of the basic assumptions of the corresponding theory, with the capacity to generate surface prosodic forms (initially, F_0 and duration) that can be imposed onto real or synthetic speech, and B) a set of learning algorithms for automatic analysis-by-synthesis, allowing the models to be trained on any speech corpora marked up with theory-specific categories. CPP will therefore facilitate theory evaluation by enabling them to make predictions in terms of fine-detailed surface prosody that can be compared to natural prosody both numerically and perceptually.

The current version of CPP has included four models: 1) the Command-Response (CR) model [1], 2) the Autosegmental-Metrical (AM) theory [7-8], 3) the Task Dynamic (TD) model [3], and 4) the Parallel Encoding and Target Approximation (PENTA) model [2]. In terms of learning algorithms, at the present stage, all four models have been implemented with local curve fitting capabilities, while only PENTA has been implemented with full-fledged global optimization algorithms in PENTAtrainer2 [9]. So this paper will focus mainly on the local fitting capabilities of all models, the software package that provides a means of comparing all models simultaneously and the results of testing them on Mandarin and English test corpora.

2. Model Descriptions

At their bases, prosodic models are built upon their suppositions on how surface prosodic patterns are linked to underlying representations. CPP, in this version, focuses on the comparison of how these basic assumptions lead to different ways of simulating F_0 contours given specific durations. This section provides brief explanations of the implemented models.

2.1. Command-Response (CR) Model

The CR model, also known as Fujisaki's model, represents F_0 contours in the logarithmic scale as the sum of three layers of data, including (a) a baseline $\ln F_b$, (b) phrase components and (c) accent/tone components. Phrase and accent/tone components are generated from second-order critically-damped linear systems in responses to phrase and accent/tone commands, respectively. Phrase components represent the slow phrase-level F_0 variations while the accent/tone components represent faster F_0 variations. The model is represented by the following equations [11]:

doi: 10.21437/SpeechProsody.2016-1

$$\ln f_0(t) = \ln F_b + \sum_{i=1}^{I} A_{pi} G_p(t - T_{0i}) + \sum_{j=1}^{J} A_{tj} \left\{ G_t(t - T_{1j}) - G_t(t - T_{2j}) \right\} \quad (1)$$

$$G_p(t) = \begin{cases} \alpha^2 t e^{-\alpha t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (2)$$

$$G_t(t) = \begin{cases} \min\left[1 - (1 + \beta t) e^{-\beta t}, \gamma\right], & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (3)$$

where $G_P(t)$ represents the impulse response function of the phrase control mechanism, $G_t(t)$ represents the step response function of the accent/tone control mechanism and F_b is the baseline F_0. The parameters A_{pi} and T_{0i} denote the magnitude and time of the i-th phrase command, while A_{tj}, T_{1j} and T_{2j} denote the amplitude, onset time and offset time of the j-th accent/tone command, respectively. The constants α, β and γ have default values of 3.0 Hz, 20.0 Hz, and 0.9; cf. [1,11].

2.2. Autosegmental-Metrical (AM) Theory

AM differs from the other three models in using point- rather than interval-based annotations, based on the idea that tonal targets are specified in terms of pitch height and their relation to the segmental string, and inter-target contours result from linear or sagging interpolations [7-8]. The inter-target connections have been implemented with linear and parabolic interpolations in an early version of AMtrainer [4] based on [8]. To further improve model fitting, we have applied a linear least square method that estimates coefficients of a quadratic equation for each inter-targets interval. This makes use of all data points in the interval rather than only three points in parabolic interpolation, with the same base quadratic equation. Thus, the AM F_0 model in the current version of CPP is expressed as

$$f_0(t) = c_1 t^2 + c_2 t + c_3 \quad (4)$$

where c_1, c_2, and c_3 are estimated using the linear least square method, which substitutes data points into the equation and solves for coefficients using a pseudoinverse operation.

2.3. Task Dynamic (TD) Model

The TD model represents articulatory gestures in speech as coordination of multiple articulators to accomplish a linguistic task [14]. TD has been implemented as a MATLAB application called Task Dynamic Application (TADA) for simulating inter-articulator speech coordination [10]. Based on the current form of TD as a stable second-order critically damped system, we have simplified it so that its transfer function is

$$H(s) = \frac{1}{(1 + \tau s)^2} \quad (5)$$

The time domain version of TD is quite similar to PENTA but has only static gesture targets and at the second-order:

$$f_0(t) = x + (c_1 + c_2 t) e^{-t/\tau} \quad (6)$$

where x denotes the F_0 gestural target and τ denotes the time constant of the system. c_1 and c_2 are solved from the initial conditions, including F_0 level and velocity. At the end of each interval, these dynamic states are also transferred to the next interval as initial conditions, similar to PENTA.

2.4. Target Approximation (TA) in PENTA Model

The basic idea of PENTA is that surface F_0 results from syllable-synchronized sequential target approximation, whereby each target is an underlying linear trajectory specified by multiple communicative functions [13]. PENTA has been implemented to perform both local and global optimization methods [2, 9]. The detailed implementation of PENTA with global optimization is given in [9]. Target approximation (TA) in PENTA is mathematically realized as a third-order critically damped linear system driven by pitch targets, as shown in:

$$f_0(t) = (mt + b) + (c_1 + c_2 t + c_3 t^2) e^{-\lambda t} \quad (7)$$

Here the first parenthesis is the pitch target while the second is the natural response of the system to the target. m and b indicate slope and height of the target, respectively. This means that the target can be static or dynamic depending on the slope, and can be higher, lower, or at a similar level to the referenced F_0 baseline depending on the target height. λ is an empirically derived rate of target approximation, which indicates how fast F_0 approaches the target. Coefficient c_1, c_2, and c_3 are determined by solving the initial value problems given the initial F_0 level, velocity and acceleration directly obtained from the data. Thus, at the end of each syllable, the final F_0 dynamic states are transferred to become the initial condition of the next syllable. This guarantees the continuity of F_0 contour up to the third order.

3. Learning Model Parameters

As is already seen, the models implemented in CPP have different parameters and are based on different assumptions. To compare them, we need to have parameter learning procedures that work similarly for all models. For local optimization, the goal is to find parameters that give the lowest error between synthetic and natural F_0 contours. The learning is done successively from left to right for TA, TD and CR to search for optimal parameters, although only the first two are strictly interval-based models while CR is a hierarchical model. The search is done by exhaustively exploring the parameter space of each interval for a parameter set that generates the lowest error. For CR, a phrase component is estimated first, and then for each interval, tone components are estimated. For AM, which is a point-based model, the linear least square method is similar in principle to others in terms of data fitting.

3.1. Evaluation Metrics

Learning results can be compared across models by examining evaluation metrics. These metrics consist of root mean square error (RMSE), which indicates the general distance between the synthetic and natural F_0 contours, and correlation, which indicates shape similarity between synthetic and natural F_0 contours. Users can also view the graphical comparison of the resulting F_0 contours using the CPP software.

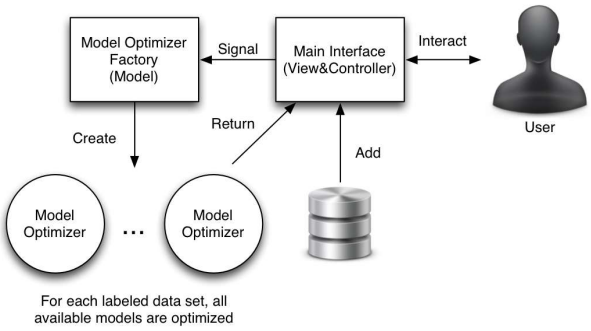

Figure 1. System diagram of CPP.

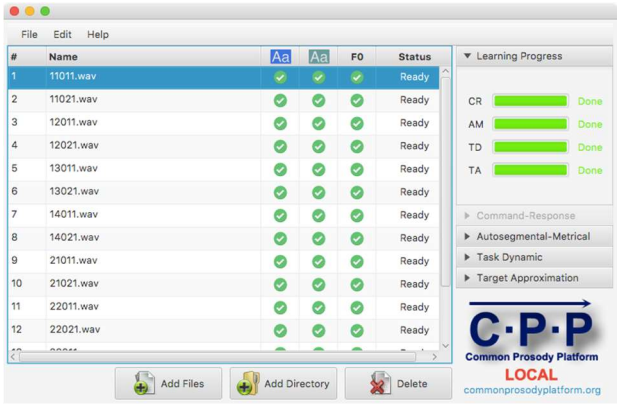

Figure 2. CPP main interface.

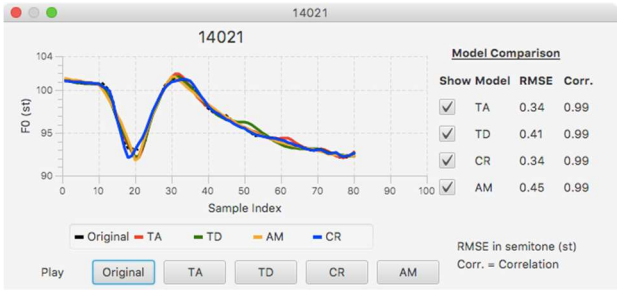

Figure 3. CPP resynthesis comparison window.

3.2. Software Architecture and Design

As a software system, CPP performs analysis-by-synthesis of all models at the same time once the data are added to the program. The software design is based on the Model-View-Controller (MVC) architecture and multithreading programming to utilize modern multicore computers in optimizing multiple models at the same time. In MVC, the user interfaces are implemented as views while the controllers control events associate with the interfaces. The models in MVC refer to the logics of the program, which in this case are the model learning operations.

Figure 1 is a system diagram showing the components of CPP. The program consists of (1) the main interface, which controls what happens when a user interacts with the program and, when speech data and their corresponding annotation labels are added to the program, validates them, (2) the model optimizer factory, which receives the signal from the main interface indicating that the user has added files to the program and spawns the model optimizers, and lastly (3) the model optimizer, which is an instance that optimizes a specific model based on the training algorithm of that model and returns the result to the main interface.

Figure 2 shows the main user interface of CPP, which consists of (1) a data table displaying the list of speech data added to the program and whether either interval-based or point-based annotation files were found as shown in the two "Aa" columns, (2) a button interface for adding data to the program and (3) a tab panel on the right showing the optimization results. Users can view the optimization progress of specific file in the "Learning Progress" panel. Once the optimization is done, users can use the model panel to selectively display the comparisons between the resynthesized F_0 contour of the model and the natural one.

After the completion of optimization, users can view the comparison of resynthesized F_0 contours of all models together by double clicking the data row. This will bring up a dialog box displaying both graphical and numerical results. Figure 3 shows results for the file "14021.wav". The line chart shows an F_0 contour comparison of all models. The button panel in the bottom area allows the user to listen to the sounds resynthesized with model-generated F_0 contours, using the PSOLA algorithm. The right panel displays the numerical results (RMSE and correlation). Displays from selected models can be suppressed by deselecting the radio button on the "Show" column. Also the result will be disabled if no annotation files are available.

4. Testing

4.1. Test Dataset

To test the CPP program, we took subsets of two corpora used previously in the development of PENTAtrainer2 [9]. The first corpus was collected for a study of tone, focus and sentence modality in Mandarin Chinese and the second one was collected for a study of stress, focus and sentence modality in American English [12]. For each prosodic condition, a sample was selected as a representative of the condition. Only data from one speaker was used in each corpus. Because the purpose of the present study is to compare the performance of various prosodic theories in modeling F_0 contours, only a limited number of sentences were included in the test corpora.

The Mandarin corpus consists of 16 eight-syllable utterances with varying tone of the third syllable — High (H), Rising (R), Low (L) or Falling (F), focus location (second or third syllable), and sentence modality (statement or question). The English corpus consists of 24 utterances with 8-10 syllables per utterance. There are three sets of sentences, in each of which the final syllable of the last word was either stressed or unstressed. Each sentence was said as either a statement or a question, and with focus on either the middle or the final target word. Further detail about both corpora can be found in [9].

4.2. Synthesis Accuracy

Tables 1 and 2 show the optimization results in terms of error and correlation of all utterances in both the Mandarin and English corpora. In general, all models can fit the data well with low errors and high correlations. Statistical analysis,

showed that there were generally no significant differences in model performance in terms of both RMSE (Mandarin: $F(3,32) = 0.779$, $p = 0.514$) and correlation (Mandarin: $F(3,32) = 1.536$, $p = 0.224$; English: $F(3,56) = 2.661$, $p = 0.057$), except for the RMSE of English corpus ($F(3,56) = 3.090$, $p = 0.034$). Nevertheless, it should be noted that because only syllable boundaries were marked in the interval-based annotations, TD and CR, which use only level targets, were at a disadvantage here. Dividing syllables into smaller intervals may increase performance for these models.

Table 1. Mean RMSE and correlation for the Mandarin corpus.

Modality†	Tone	RMSE (st)				Correlation			
		CR	AM	TD	TA	CR	AM	TD	TA
Statement	H	0.45	0.66	0.49	0.49	0.99	0.99	0.99	0.99
	R	0.53	0.53	0.46	0.49	0.99	0.99	0.99	0.99
	L	0.88	0.62	1.08	0.73	0.97	0.98	0.94	0.98
	F	0.69	1.12	0.79	0.72	0.99	0.97	0.99	0.99
Question	H	0.54	0.42	0.53	0.44	0.99	0.99	0.99	0.99
	R	0.44	0.51	0.55	0.43	0.99	0.98	0.98	0.99
	L	0.63	0.40	0.87	0.39	0.97	0.99	0.92	0.99
	F	0.62	0.49	0.50	0.45	0.98	0.99	0.99	0.99

Table 2. Mean RMSE and correlation for the English corpus.

Modality	Focus	RMSE (st)				Correlation			
		CR	AM	TD	TA	CR	AM	TD	TA
Statement	Medial	0.88	1.16	0.94	0.62	0.98	0.96	0.98	0.99
	Final	0.58	0.57	0.74	0.33	0.93	0.94	0.90	0.98
Question	Medial	0.83	0.40	0.51	0.31	0.97	0.99	0.99	1.00
	Final	0.38	0.57	0.64	0.31	0.98	0.95	0.92	0.99

5. Discussions

As shown above, the current development of CPP has reached a stage where all the implemented theories and models can be numerically and graphically compared at their basic levels in terms of detailed F_0 contours. So far, all the four implemented models can attain very high accuracy when performing local fitting on individual sentences. This is achieved, however, at the cost of having to make various adjustments to the models to simplify them. For TD, a major simplification is to fix the temporal alignment of the onset and offset of the tonal gesture (TD), so that they coincide with the onset and offset of the syllable. This simplification is a deviation from the flexible timing assumed in the model. TD assumes that articulatory gestures involved in the production of a syllable are overlapped with each other by various degrees, and the amount of overlap is empirically determined [14,15]. In an application of TD on Mandarin tones, the tone gestures are flexibly aligned to consonants and vowels, and the exact alignment is empirically derived based on the timing of F_0 turning points. Also in the current implementation of TD stiffness represented by τ is treated as a free parameter to be optimized, which deviates from the previous practice of allowing only discrete levels of stiffness [14,16], except when it is used for controlling duration for boundary marking [17].

In CR the onset and offset of accent/tone commands are both free parameters to be determined through analysis by synthesis in the model fitting [18]. In [1,11], however, it is found that in both Mandarin and Cantonese, the onset and offset of tone commands are closely aligned to rhyme boundaries. The CPP implementation of CR thus takes the new evidence into consideration. Timing parameters, though not fixed, are estimated around syllable boundaries.

Unlike TD and CR, PENTA assumes that target approximation is fully synchronized with the syllable, with no flexible timing. This is a theoretical assumption only, however, as the alignment of timing in PENTAtrainer can be adjusted by users. In a small-scale modeling test in which flexible tone-syllable alignment is implemented, it was found that RMSE increased and correlation decreased relative to the non-flexible alignment condition [19]. Furthermore, the alignment derived from model fitting with flexible timing was very similar to the syllable-synchronized timing, which is consistent with the finding of [1,11] mentioned above.

The flexible alignment tested in [19] was only partially free, as there was no overlap of target approximation intervals and no allowance of gaps between intervals, as assumed in both TD and CR. The implementation of fully flexible timing will be the task of further development of CPP. At the current stage, however, both TD and CR have the simplified alignment control similar to TA.

For AM, there is yet no clear consensus on how the target-interpolation process should be algorithmically realized beyond the classic work of Pierrehumbert [8]. What we have done in the current version of CPP is therefore highly tentative. Recent work, e.g., has shown the critical role of target-segment alignment [21]. There is also evidence that interpolation can vary in systematic ways that can affect perception as well as production [20]. These new findings could be incorporated into the next generation of CPP.

Further development of CPP will take on a more challenging aspect of prosodic modeling, i.e., to globally optimize the parameters based not only on data from a specific utterance, but also on phonological/functional annotations of the whole corpus. This step will allow the testing of all models on their ability to achieve a much higher level of generalizability. Also the current CPP has focused only on F_0. Further development of CPP will also include the modeling of duration and other aspects of prosody. For future progress, readers can follow the project updates and download the software at http://www.commonprosodyplatform.org.

6. Conclusions

The development of CCP will help bridge the current gaps between theoretical conceptualization, empirical investigation and computational modeling. The computational nature of the resulting trainable models will also make them readily transferable to applied areas, including speech technology, language teaching and speech communication disorders. The research approach developed in this project may also be extendable to a general paradigm in speech research, namely, theory testing by computational modeling.

7. Acknowledgements

Financial support was provided by the National Science Foundation (to Whalen and Xu), the Newton International Fellowship Alumni Scheme (to Prom-on) and the National Social Science Fund of China 10CYY009 and 13&ZD189 (to Gu).

8. References

[1] H. Fujisaki, C. Wang, S. Ohno and W. Gu, "Analysis and synthesis of fundamental frequency contours of Standard Chinese using the command–response model," *Speech Communication*, vol. 47, pp. 59–70, 2005.

[2] S. Prom-on, B. Thipakorn and Y. Xu, "Modeling tone and intonation in Mandarin and English as a process of target approximation," *Journal of the Acoustical Society of America*, vol. 125, pp. 405–424, 2009.

[3] H. Nam, L. Goldstein, E. Saltzman and D. Byrd, "TADA: An enhanced, portable Task Dynamics model in MATLAB," *Journal of the Acoustical Society of America*, vol. 115, pp. 2430, 2004.

[4] A. Lee, Y. Xu and S. Prom-on, "Modeling Japanese F0 contours using the PENTAtrainers and AMtrainers," in *TAL 2014 – 4th International Symposium on Tonal Aspect of Languages, May 13–16, Nijmegen, The Netherlands*, 2014, pp. 164–167.

[5] S. Raidt, G. Bailly, B. Holm and H. Mixdorff, "Automatic generation of prosody: Comparing two superpositional systems," in *Speech Prosody 2004, March 23–26, Nara, Japan*, 2004, pp. 417–420.

[6] X. Sun, "F0 generation for speech synthesis using a multi-tier approach," in *ICSLP 2002 – 7th International Conference on Spoken Language Processing, September 16–20, Denver, Colorado*, 2002, pp. 2077–2080.

[7] J. Pierrehumbert, *The Phonology and Phonetics of English Intonation*. Ph.D. dissertation, MIT, Cambridge, MA, 1980.

[8] J. Pierrehumbert, "Stylizing intonation," *Journal of the Acoustical Society of America*, vol. 70, pp. 986-995, 1981.

[9] Y. Xu and S. Prom-on, "Toward invariant functional representations of variable surface fundamental frequency contours: Synthesizing speech melody via model-based stochastic learning," *Speech Communication*, vol. 57, pp. 181-208, 2014.

[10] H. Nam, C. Browman, L. Goldstein, M. Proctor, P. Rubin and E. Saltzman, TADA: Task Dynamic Application, http://www.haskins.yale.edu/tada_download Last Access: 4 November 2015.

[11] W. Gu, K. Hirose and H. Fujisaki, "Analysis of Tones in Cantonese Speech Based on the Command-Response Model," *Phonetica*, vol. 64, pp. 29-62, 2007.

[12] S. Prom-on, F. Liu and Y. Xu, "Functional modeling of tone, focus, and sentence type in Mandarin Chinese," *in ICPhS 2011 – 17th International Congress in Phonetic Sciences, August 17-21, Hong Kong*, pp. 1638–1641.

[13] Y. Xu, "Speech melody as articulatorily implemented communicative functions," *Speech Communication,* vol. 46, pp. 220-251, 2005.

[14] E. L. Saltzman and K. G. Munhall, "A dynamical approach to gestural patterning in speech production," *Ecological Psychology*, vol. 1, pp. 333-382, 1989.

[15] J. A. S. Kelso, E. L. Saltzman and B. Tuller, "The dynamical perspective on speech production: data and theory," *Journal of Phonetics*, vol.14, pp. 29-59, 1986.

[16] H. Nam, V. Mitra, M. Tiede, M. Hasegawa-Johnson, C. Espy-Wilson, E. Saltzman and L. Goldstein, "A procedure for estimating gestural scores from speech acoustics," *The Journal of the Acoustical Society of America*, vol. 132, pp. 3980-3989, 2012.

[17] D. Byrd and E. Saltzman, "The elastic phrase: Modeling the dynamics of boundary-adjacent lengthening," *Journal of Phonetics*, vol. 31, pp. 149–180, 2003.

[18] H. Fujisaki, "Dynamic characteristics of voice fundamental frequency in speech and singing," In P. F. MacNeilage (Ed.), *The Production of Speech*, New York: pp. 39-55, Springer, 1983.

[19] Y. Xu and S. Prom-on, "Degrees of freedom in prosody modeling," Speech Prosody in Speech Synthesis — Modeling, Realizing, Converting Prosody for High Quality and Flexible speech Synthesis, K. Hirose and J. Tao, eds., pp. 19-34: Springer, 2015.

[20] J. Barnes, N. Veilleux, A. Brugos and S. Shattuck-Hufnagel, "Tonal Center of Gravity: A global approach to tonal implementation in a level-based intonational phonology," *Laboratory Phonology*, vol. 3, no. 2, pp. 337-383, 2012.

[21] A. Arvaniti, D.R. Ladd and I. Mennen, "Stability of tonal alignment: the case of Greek prenuclear accents," *Journal of Phonetics*, vol. 36, pp. 3-25, 1998.