INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Optimal Unit Stitching in a Unit Selection Singing Synthesis System

Marius Cotescu

[1]Department of R&D and Linguistics, Acapela Group, Mons, Belgium
marius.cotescu@acapela-group.com

Abstract

Unit Selection based speech synthesis systems are currently the best performing, producing natural sounding speech with minimal CPU load. One of the important reasons behind their success is the amount of recordings that are now commonly used in synthesis applications. However, in the case of singing applications, it is quite hard for a database to cover a large phonetic space due to the relative inefficiency of the recording process. Thus, due to the reduced catalogue of units, singing unit selection systems are more likely to produce spectral discontinuity artefacts. Taking advantage of the quasi stable nature of articulation during singing, we propose a novel unit stitching method. The method was implemented into the system that was used for the "Fill-In the Gap" Singing Synthesis Challenge.
Index Terms: singing synthesis, speech synthesis, unit selection, optimal stitching

1. Introduction

With the constant increase of computing power, and as artists and performers immerse more into technology, singing synthesis applications are becoming more interesting. And they span the wide spectrum between avant-garde music producers and safeguarding traditional singing styles [1]. Current singing synthesis systems focus on applications ranging from pop singing [2], to classical [3], but mainly keep to mainstream singing styles (e.g. pop [2] or children songs [4]). Technologies also vary from unit selection based systems [2, 5] to HMM-based parametric implementations [6], passing through proper performative instruments [7].

As singing is one of the most expressive ways of using one's voice, and when done right can evoke powerful emotions, the current quality of singing synthesis made most artists keep their distance from the technology. The main complai¡nts have been related the strong distortions caused by heavy signal processing. In [2], the problem is addressed by slightly degrading the voice entirely, in order to achieve a constant quality. This choice, however, limits the applications to niche fields, where signal processing is tolerated. Another solution [8] proposes recording different versions of the corpus at different pitch levels. However, this creates competition between musical and phonetic coverage.

In this paper we present a method that takes advantage of the particularities of singing, in order to tackle spectral discontinuities produced by glueing together speech units. More precisely, we rely on the fact that singing is usually performed at a slower rate than speech, creating longer stretches of quasi-stationary sound, especially for vowels. We can exploit these longer stable regions to lengthen the area over which units are blended, spreading the potential spectral discontinuity in time, and making it less audible. The method has the advantage of reducing the requirement for phonetic coverage, and allowing a richer coverage of musical and expressive content. In the current paper we apply the method on two French pop singing corpora (one female and one male voice). As part of the i-Treasures project [1], we have also deployed the system for two traditional singing styles (Sardinian "Canto a Tenore", and Greek Byzantine singing).

The paper is structured in 5 sections. The next section describes the general structure of our singing synthesis system. The third section presents the proposed algorithm for optimal unit stitching, and efficient handling of phoneme duration. In the fourth section we describe the experiments that we run to evaluate the method and we present their results. The final section discusses the results and draws conclusions of the work.

2. Synthesis System

Our entry in the 2016 Singing Synthesis Special Session consists of synthetic singing samples generated by a unit selection based engine. The structure of the system is presented in Figure 1. The input data consists of a MusicXML file with embedded lyrics, which is parsed by the score parser module to extract the syllable – note pairs. The notes are then used to produce the syllable and phoneme durations, as well as for generating the required pitch contour. The syllables are parsed by an NLP system to extract the phonetic transcription that is then used to drive a unit selection engine.

Unlike conventional unit selection systems that concatenate the waveforms directly, we perform the concatenation on vocoded parameters instead. This approach is more practical, as we do have to use a vocoder in order to apply the extreme pitch changes required by singing synthesis. The system uses the STRAIGHT vocoder [9], as it has already proved itself in singing applications [10]. In order to ease the runtime computational load, the corpora are analysed in advance, and the spectral envelope, aperiodicity coefficients and pitch values are stored.

We have used two French singing synthesis corpora, one female, and one male. They were both recorded at a sampling frequency of 48 kHz, and consist of isolated words, sung on a flat note. Eventhough The phonetic content covers the diphones of the French language. The STRAIGHT analysis was performed with a 5 ms frame rate, with the FFT length set to 2048 samples. As the system operates directly with the spectral envelope, we have compacted the spectrogram and aperiodicity coefficients by resampling them to 513 samples per frame, each.

The pitch contour was generated by using a parametric second order system similar to the one described in [11, 4]. It can separately model the damping behaviours of overshoot and preparation, as well as the oscillatory phenomenon needed in vibrato. The parameters were manually set for each voice so that they fit the style of the singer. They were not tailored to the style of the songs, though.

http://dx.doi.org/10.21437/Interspeech.2016-1390

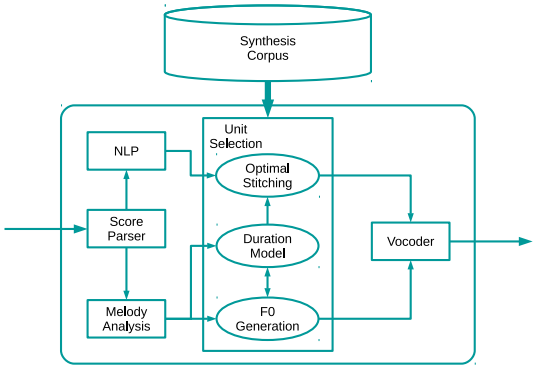


Figure 1: *Structure of the singing synthesis system.*

3. Unit Selection Engine

Current unit selection synthesis algorithms [5, 12] used in singing synthesis rely on checking the spectral mismatch between two units, to ensure that the minimal spectral discontinuity between the selected units is achieved, and then relies on local time-domain or spectral-domain fading to glide from one unit to the next. The method is definitely fast, and has been proved to work in numerous applications. It performs especially well when very large corpora are available. However, in the context of singing, large corpora are hard to obtain. The main limits are due to the singer's repertoire, and the length a recording session can span, without straining the singer's voice.

Controlling the point where the two parts of the phoneme are glued has two advantages. First of all, by varying the length of the overlap, we can control the final length of the resulting phoneme, thus limiting the amount of stretch needed to achieve a required rhythm. Second of all, we can choose an optimal location for the stitching in order to minimize spectral discontinuity. Also, by performing the stitching over several frames, the rate of change is decreased even further.

The current section will present the necessary steps needed to implement the proposed method. First we will define and propose a method for detecting the stationary parts of a segment. Next, we describe the proposed method to obtain an optimal glueing path over the spectral domain. We finally put it all together, and also describe the mechanism used to control the duration of the resulting phonemes.

3.1. Stable Part

One of the particularities of singing is that it presents many sustained, relatively stable sounds. This fact means that the acoustic features, too, hold constant for relatively large sections in the middle of the phonemes. This observation is extremely helpful in the methods we proposed for increasing the quality of the output, but in order to use it, we first need to accurately detect the stable part of each sound in our database.

One intuitive method of measuring the stability of a section of sound is to use dynamic features (e.g. the spectral flux). However, initial tests using this approach have shown that it is rather hard to have a global threshold value. This results in too short stable parts for certain sounds, and too long for others. It becomes even more problematic if the database contains samples of vibrato sounds.

In order to avoid this, and use information that is specific to each phoneme, we turned to the HMMs that we used for segmentation. In this framework, we rely on the state-level segmentation, and consider the most stable part of the signal to be the longest segment that gets assigned to any one particular state. However, this might mean that the first or last states can also be considered stable. To avoid spilling into co-articulation areas around phoneme boundary, we have defined a guard-zone of 40 ms around the beginning and end of any phoneme. The beginning and end of each stable part is provided in the segmentation file, along with phoneme boundaries. The middle point of a phoneme – that is the one that is considered for computing the concatenation cost – is set to be in the middle of the stable part, rather than that of the phoneme.

3.2. Optimal Stitching

Regular unit selection engines assume that the gluing point is fixed, and localised to one frame only. In the case of singing though, because of the long sustained sounds and of the variable overlap, we can assume that the gluing point can be placed anywhere in the overlapping region of the two phonemes' stable parts. Furthermore, we will assume that the two units can be blended over multiple frames. In order to limit smoothing and smearing artefacts introduced by interpolating over a large number of frames, we propose a blending strategy that keeps the transitions local both in time and frequency domains. The algorithm relies on finding the glueing path in the overlapping stable regions that minimizes the spectral discontinuity in the resulting unit.

Let L be the FFT length used to compute the spectrogram of the signal. We will then have $K = L/2 + 1$ samples in the smoothed STRAIGHT amplitude spectrum. In (1) we define the matrix form spectrogram of the N frames long overlapping region of unit i, where n is the frame index, k is the frequency bin index, and $s^i_{n,k}$ are the spectrogram amplitudes of unit i at time n and frequency bin k, expressed in decibels.

$$\mathbf{SP}^i = \{s^i_{n,k}\}, \quad 0 \le n < N, 0 \le k < K \tag{1}$$

The spectral distance over the overlapping region is defined in (2) as the absolute value of the difference between the spectrogram values of the two segments.

$$\mathbf{D} = \{d_{n,k}\} = \left\{\left|s^1_{n,k} - s^2_{n,k}\right|\right\} \tag{2}$$

In (3) we define a glueing path in the overlap space as the list of M points (n^P_k, k) in the time-frequency domain, which satisfies the adjacency criterion $|n_j - n_{j+1}| \le 1$.

$$\mathbf{P}_M = \left\{(n^P_0, 0), \ldots (n^P_k, k), \ldots, (n^P_{M-1}, M-1)\right\} \tag{3}$$

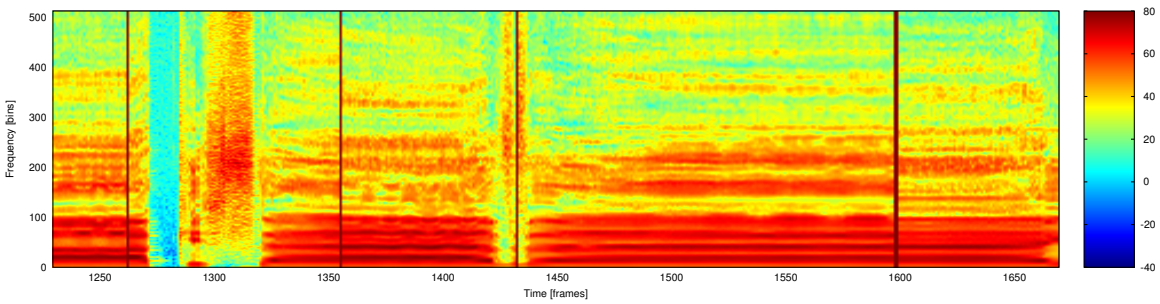
Using the definition of the path given in (3), and that of the distance matrix given in (2), (4) defines the glueing cost of a path $\mathbf{P}$.

$$\mathbf{C_{P_M}} = \sum_{k=0}^{M-1} d_{n^P_k, k} \tag{4}$$

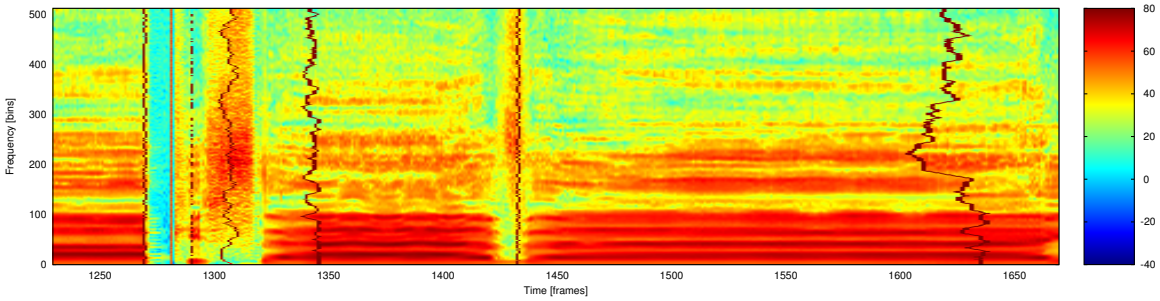
It is easy to see that the cost of a path $\mathbf{P}_M$, can be recursively computed from the cost of the sub-path $\mathbf{P}_{M-1}$, as shown in (5).

$$\mathbf{C_{P_M}} = \mathbf{C_{P_{M-1}}} + d_{n^P_{M-1}, M-1} \tag{5}$$

We define a complete glueing path $\mathbf{P}_K$ (for simplicity $\mathbf{P}$) as being a glueing path that connects all frequency bins, from 0 to K. A complete path can start and end on any frame



(a)



(b)

Figure 2: *Example of local (a) and optimal (b) glueing paths.*

($0 \leq n_0^P < N, 0 \leq n_{K-1}^P < N$). The optimal glueing path for a given overlapping area is given by (6) as being the complete path in the overlapping space with the minimum glueing cost.

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} C_{\mathbf{P}} \tag{6}$$

Considering that all elements of the distance matrix, one can easily see that any sub-path of $\mathbf{P}^*$ is the optimal sub-path between its start and end. Using this observation, one can deploy dynamic programming to recursively search for the optimal path in the overlapping space. At each frequency bin k, the algorithm needs to evaluate the costs to reach every one of the N overlapping frames, using the recursion given by (5), and save the previous point. Once all the costs have been computed for bin $K-1$, the position of the minimum gives the ending frame of the optimal path. The full path can then be retrieved by following the previous optimum point to $k = 0$.

Figure 2b shows an example of optimal glueing paths in several segments. When compared to the results obtained by setting the overlap 1 frame, shown in Figure 2a, several things stand out. First, the transitions from one unit to the other are less abrupt in the optimal path scenario. Also, the optimal path can avoid abrupt transitions and glue around similar structures at different frequencies. Also, one can see that especially for long, sustained sounds, the number of discontinuities per frame is significantly reduced.

3.3. Optimal Overlap and Duration Control

Because the acoustic features of the phoneme hold relatively constant along the stable part of a phoneme, it is safe enough to assume that the concatenation cost will stay relatively constant, wherever we choose to gluing point. This is a very useful tool in controlling the duration of a segment. Otherwise, one should rely on resampling the resulting feature vector. If the resampling factor is too high or too low, there is the risk of creating unnatural artefacts. By moving the gluing point of the first segment closer to its end, and the gluing point of the second closer to its beginning, the duration of the resulting segment is going to be longer. If we move the gluing points in the opposite directions, the resulting duration is going to be shorter.

To obtain the right compromise between the length and the spectral continuity of the resulting unit, we run the optimal stitching algorithm for all the overlap lengths that would yield between 80% and 120% of the desired unit length. We run the optimal stitch and optimal overlap algorithm even if we are glueing segments that were originally adjacent in the database. This is to avoid extreme duration changes, which can artificially increase the spectral flux, thus creating the impression of a discontinuity. If the original duration is close enough to the target, the segments pass through untouched.

In the case where the segments are too short to provide even 80% of the target duration, we use interpolation to fill in the gap between the two units [5]. Thus, we ensure that we have an gradual transition between the two .

4. Experiment and Results

In order to test the efficiency and quality of the proposed unit stitching method, we have generated samples using three different unit stitching strategies. System ”A” serves as the baseline, and uses a standard one frame overlap, at the middle of the phoneme. System ”B” uses interpolation to bridge the gap between the two units that are considerably shorter than the requested length. System ”C” implements the proposed method for stitching overlapping segments, and deploys interpolation where the segments are too short to cover the required duration. The phoneme durations and pitch contour was identical for all

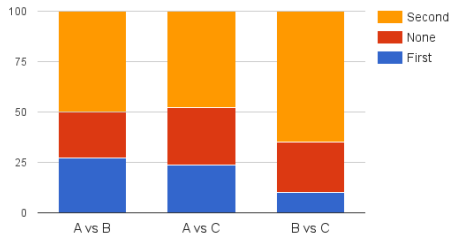


Figure 3: *Preference test results for the female singer (in percent).*

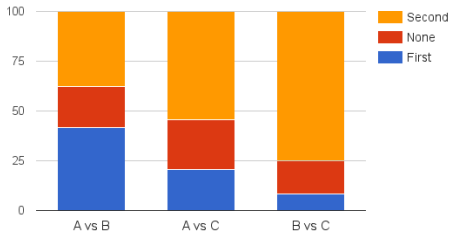


Figure 4: *Preference test results for the male singer (in percent).*

three systems. The test was conducted for the two – male and female, French voices.

In order to evaluate the systems, we have chosen six excerpts from songs whose styles varied from opera, to pop, to French chanson. The three systems were evaluated in a perceptual ABX comparison test. The listeners were presented with a pair of the same song, synthesised using different systems. They were asked to choose the sample they perceived as more natural. The samples and system pairs were randomised for each listener. A total of 12 listeners participated in the listening tests. In addition to the comparison scores, they have also commented on the general quality of the samples, which they found to be at least appropriate. The results of the listening tests for the female and male voices are presented in Figure 3 and Figure 4, respectively.

In the case of the female voice, both systems B, and C are preferred to the baseline system A. System B is considered equal or better than System A in 73% of the answers. System C has a similar performance, and it is considered equal or better in 76% of the answers. The direct comparison between systems B and C shows a more clear picture though, as system C is equal or better in 90% of the cases (with a clear preference in 65% of the answers).

In the case of the male voice, system C is clearly better than both the baseline system A, and the interpolating system B. The clear preference for system C is 54% and 75% when compared to systems A, and B, respectively. There is no significant difference between systems A and B for the male voice, both getting similar preference (42% for system A and 38% for system B). We suspect that system B is producing sustained regions that are too smooth. This, coupled with the low pitch of the male voice leads to a more metallic and buzzy sound.

In addition to the listening tests, we have submitted the two voices to the ”Fill in the Gap” Singing Synthesis Challenge. We have submitted the ”a cappella” versions of the two French versions of the theme songs, using both voices. The samples were synthesized automatically, using the setup used for system C. No manual interventions were made to the parameters.

5. Conclusions

The paper presents a unit selection based singing synthesis system, and a proposed method for optimal unit stitching. We take advantage of the long stationary portions of the sung signal to spread the spectral discontinuity between selected segments. The technique can be deployed both to smooth out the glueing between discrepant segments, as well as meshing together segments that were originally adjacent, but which need shortening of lengthening.

We have compared the proposed method to a baseline system where no overlapping or smoothing is done, and to a system where gaps created by glueing together segments that are shorter than the required duration are filled by interpolation. The results show that the proposed method performs better than the two baselines for both a male and a female voice.

The motivation for the proposed method was two-fold. First we wished to increase the quality of unit-selection based singing synthesis systems. The first motivation was to decrease the bias towards phonetic coverage in singing synthesis corpora. This would allow the same quality to be achieved with a smaller phonetic coverage, allowing more data to be focused on expressive, rhythmic and musical register specific data. Our future work will focus on adding rhythmic coverage in addition to the phonetic content.

6. Acknowledgements

This work is funded by the European Commission via the i-Treasures project (FP7-ICT-2011-9-600676-i-Treasures). The corpora were recorded as part of the ChaNTeR project, supported by the ANR.

7. References

[1] K. Dimitropoulos, S. Manitsaris, F. Tsalakanidou, S. Nikolopoulos, B. Denby, S. A. Kork, L. Crevier-Buchman, C. Pillot-Loiseau, M. Adda-Decker, S. Dupont *et al.*, “Capturing the intangible an introduction to the i-treasures project,” in *Computer Vision Theory and Applications (VISAPP), 2014 International Conference on*, vol. 2. IEEE, 2014, pp. 773–781.

[2] H. Kenmochi and H. Ohshita, “Vocaloid-commercial singing synthesizer based on sample concatenation.” in *INTERSPEECH*, vol. 2007. Citeseer, 2007, pp. 4009–4010.

[3] L. Ardaillon, G. Degottex, and A. Roebel, “A multi-layer f0 model for singing voice synthesis using a b-spline representation with intuitive controls,” in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[4] T. Saitou, M. Goto, M. Unoki, and M. Akagi, “Speech-to-singing synthesis: Converting speaking voices to singing voices by controlling acoustic features unique to singing voices,” in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*. IEEE, 2007, pp. 215–218.

[5] L. Ardaillon, “Synthèse du chant,” Master’s thesis, IRCAM, France, 2013.

[6] K. Oura, A. Mase, T. Yamada, S. Muto, Y. Nankaku, and K. Tokuda, “Recent development of the hmm-based singing voice synthesis system—sinsy,” in *Seventh ISCA Workshop on Speech Synthesis*, 2010.

[7] L. Feugère, C. d'Alessandro, and B. Doval, "Performative voice synthesis for edutainment in acoustic phonetics and singing: A case study using the "cantor digitalis"," in *Intelligent Technologies for Interactive Entertainment*. Springer, 2013, pp. 169–178.

[8] K. Oura, A. Mase, Y. Nankaku, and K. Tokuda, "Pitch adaptive training for hmm-based singing voice synthesis," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5377–5380.

[9] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno, "Tandem-straight: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, f0, and aperiodicity estimation," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 3933–3936.

[10] S. Lee, Z. Wu, M. Dong, X. Tian, and H. Li, "A comparative study of spectral transformation techniques for singing voice synthesis." in *INTERSPEECH*, 2014, pp. 2499–2503.

[11] T. Saitou, M. Unoki, and M. Akagi, "Development of an f0 control model based on f0 dynamic characteristics for singing-voice synthesis," *Speech communication*, vol. 46, no. 3, pp. 405–417, 2005.

[12] J. Bonada, A. Loscos, and H. Kenmochi, "Sample-based singing voice synthesizer by spectral concatenation," in *Proceedings of Stockholm Music Acoustics Conference*, 2003, pp. 1–4.