Synthesizing a choir in real-time using Pitch Synchronous Overlap Add (PSOLA)

Norbert Schnell, Geoffroy Peeters, Serge Lemouton, Philippe Manoury, Xavier Rodet
{schnell,peeters,lemouton,manoury,rod}@ircam.fr
IRCAM - CENTRE GEORGES-POMPIDOU
1, pl. Igor Stravinsky, F-75004 Paris, France
http://www.ircam.fr

ABSTRACT

The paper presents a method to synthesize a choir in real-time and its application in the framework of an opera production. It intentionally integrates artistic considerations with research and engineering matters, thus giving a complete picture of a concrete collaboration in the context of the creation of electronic music.

The synthesis of the ”virtual choir” is implemented for the jMax real-time sound processing system using the Pitch Synchronous Overlap Add (PSOLA) technique. The synthesis algorithm derives multiple voices of a same group from a single recording of a real choir singer. The first stage of the analysis segments harmonic, non harmonic and transient parts of the signal. The second stage places PSOLA markers in the harmonic parts by a novel two-steps algorithm.

The synthesis algorithm allows various transformations of the analysed sound of a single voice by the introduction of stochastic as well as deterministic variations. It is controlled by an extended set of parameters and results in a wide range of different timbres and textures in addition to those of a realistic choir sound.

The last section of the paper is dedicated to the application of the algorithm in the context of the composition and its integration into the rest of the environment of the opera production. It describes the experiments with the recordings of a choir and the work in the production studio using the jMax environment. Finally a set of commented examples is associated with the paper, which will be presented during the paper session.

1 INTRODUCTION

The opera “K...” and the concept of the virtual choir

Since spring 1998 Philippe Manoury is working on the composition of the opera “K...” based on Franz Kafka’s novel “Der Prozess” which will have its premiere in march 2001 at the Opera Bastille in Paris. The work has an important electro-acoustic part, which is entirely implemented in *jMax* [Déchelle et al., 1998] [Déchelle et al., 1999a] and realized at IRCAM with the musical assistance of Serge Lemouton.

For several scenes of this Opera (such as the trial) Manoury has expressed the need for choral voices evoking the notion of crowd. This led to the concept of a *virtual choir*.

The goal was to create an algorithm which is able to realistically reproduce the sound of a choir, permitting sounds unusual or impossible for a real choir. It was decided to evaluate several technical possibilities. Although there is a lot of research on synthesis methods for a single voice [Sundberg, 1987] [Ternströn, 1989], the domain of vocal ensemble synthesis is not much explored .

After some unsatisfying trials to obtain a choir sound with various techniques such as granular synthesis, modified additive synthesis or various chorus effects it was found that the only way to obtain the realistic notion of a choir would be by superposition of multiple well enough distinguishable solo voices.

This assumption leads to the following two questions:

1. How to efficiently synthesize a single voice allowing a wide range of transformations?
2. Which individual variations should be attributed to each voice in order to obtain a chorus effect when superposing them?

The answer to the first question was found in the *PSOLA* technique described in the first part of this paper. The second part part of the paper explains the real-time algorithm implemented for the synthesis of a group of voices proposing an answer to the second question. The paper concludes with the experiments made during the research on the virtual choir and its integration into the opera.

2 PSOLA

PSOLA (**P**itch **S**ynchronous **O**ver**L**ap-**A**dd [Charpentier, 1988] [Moulines and Charpentier, 1990]) is a method based on the decomposition of a signal into a series of elementary waveforms in such a way that each waveform represents one of the successive pitch periods of the signal and the sum (overlap-add) of them reconstitues the signal.

PSOLA works directly on the signal waveform without any sort of model and therefore does not lose any detail of the signal. But in opposition to usual sampling, PSOLA allows independent control of pitch, duration and formants of the signal.

One of the main advantages of the PSOLA method is the preservation of the spectral envelope (formant positions) when pitch shifting is used. High-quality transformations of signals can be obtained by time manipulation only, therefore with very low computational cost. For a simultaneous modification of pitch and spectral envelope, a **F**requency **S**hifting (*FS*-PSOLA [Peeters and Rodet, 1999]) method has been proposed.

PSOLA is very popular for speech transformation because of the properties of the speech signal. Indeed, PSOLA requires the signal to be harmonic and well-suited for a decomposition into elementary waveforms by windowing, which means that the signal energy must be concentrated around one instant inside each period.

The PSOLA method can be understood as

- granular synthesis in which each "grain" corresponds to one pitch period
- synthesis based on a source/filter model like *CHANT* [d'Alessandro and Rodet, 1989]: the elementary waveforms can be considered as an approximation of the *CHANT Formant Waveforms* but without explicit estimation of source and filter parameters

G. Peeters has developed a PSOLA analysis and synthesis package described in the following.

2.1 Time/Frequency signal characterization

By its definition, the PSOLA method allows only modification of the periodic parts of the signal. It is therefore important to estimate which parts of the signal are periodic, which are non-periodic and which are transient. In the case of the voice, the periodic part of the signal is produced by the vibration of the vocal chords and is called "voiced".

At each time instant t, a "voicing" coefficient $v(t)$ is estimated. This coefficient is obtained by use of the "Phase Derived Sinusoidality measure" from *SINOLA* [Peeters and Rodet, 1999]. For each time/frequency region, the instantaneous frequency is compared to the frequency measured from spectrum peaks. If they match, the time/frequency region is said to be "sinusoidal". If for a specific time most regions of the spectrum are sinusoidal, this time frame is said to be "voiced" and is therefore processed by the PSOLA algorithm.

2.2 PSOLA analysis

PSOLA analysis consists of decomposing a signal $s(t)$ into a series of elementary waveforms $s_i(t)$. This decomposition is obtained by applying analysis windows $h(t)$ centered on times m_i:

$$s_i(t) = h(t - \mathrm{m}_i)s(t) \tag{1}$$

The m_i, called "markers", are positioned [Peeters, 1998]

- pitch-synchronously, i.e. the difference $\mathrm{m}_i - \mathrm{m}_{i-1}$ is close to the local fundamental period [Kortekaas, 1997],
- close to the local maxima of the signal energy. This last condition is required in order to avoid deterioration of the waveform due to the windowing.

After estimating the signal pitch period $\mathrm{T0}(t)$ and the signal energy function $e(t)$, the markers m_i are positionned using the following two-step algorithm.

Step 1: Estimation of the local maxima of the energy function

Because PSOLA markers m_i must be close to the local maxima of the energy function, the first step is the estimation of these maxima.

Let us define a vector of pitch instants $\Theta_l = [\theta_{l,0}, \theta_{l,1}, ..., \theta_{l,i}, ...]$ such that $\theta_{l,i} - \theta_{l,i-1} = \mathrm{T0}_{i-1}$ (see Figure 1). Around each instant $\theta_{l,i}$ let us define an interval $I_{l,i} = \left[\theta_{l,i} - \frac{\mathrm{T0}_{i-1}}{\alpha}, \theta_{l,i} + \frac{\mathrm{T0}_i}{\alpha}\right]$, where α controls the extent of the interval. Inside each interval $I_{l,i}$, the maximum of the energy is estimated and noted $t_{l,i}$. For each vector Θ_l, i.e. for each choice of starting time $\theta_{l,0}$, the sum of the values of the energy function at the times $t_{l,i}$, $\sigma_l = \sum_i e(t_{l,i})$, is computed. Finally the selected maxima τ_i are those of the vector Θ_l which maximize σ_l: $\tau_i = t_{l',i}$ with $l' = \arg\max_L \sigma_l$.

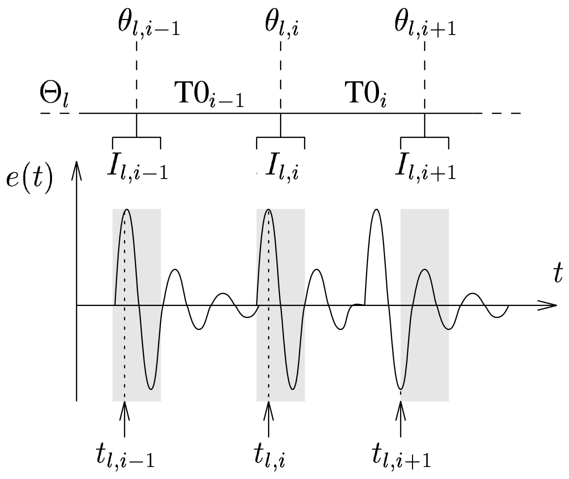

Figure 1: Estimation of the local maxima of the energy function

Step 2: Optimization of periodicity and energy criterions

Because PSOLA markers m_i must be placed pitch-synchronously and close to the local maxima, the two criteria have to be minimized simultaneously.

A novel least-squares resolution is proposed, as follows:

Let m_i denote the markers we are looking for, τ_i the time locations of the local maxima of the energy function estimated at the previous stage, $\mathrm{T0}_i$ the fundamental period at time τ_i. A least-squares resolution is used in order to minimize the periodicity criterion (distance between two markers close to the fundamental period: $\mathrm{m}_i - \mathrm{m}_{i-1} \simeq \mathrm{T0}_{i-1}$) and energy criterion (markers close to the local maxima of energy: $\mathrm{m}_i \simeq \tau_i$). The quantity to be minimized is $\epsilon = \sum_i((\mathrm{m}_i - \mathrm{m}_{i-1}) - \mathrm{T0}_{i-1})^2 + \beta(\mathrm{m}_i - \tau_i)^2$. β is used to weigh the criteria: $\beta < 1$ favours periodicity while $\beta > 1$ favours energy.

If the vector of markers is $\overline{\mathrm{m}} = [\mathrm{m}_0\ \mathrm{m}_1\ ...\ \mathrm{m}_i\ ...\ \mathrm{m}_{N-1}\ \mathrm{m}_N]^T$, the optimal marker positions are obtained by

$$\overline{\mathrm{m}} = M^{-1} \begin{pmatrix} 0 & -\mathrm{T0}_0 & +\gamma\tau_0 \\ \mathrm{T0}_0 & -\mathrm{T0}_1 & +\beta\tau_1 \\ & \vdots & \\ \mathrm{T0}_{i-1} & -\mathrm{T0}_i & +\beta\tau_i \\ & \vdots & \\ \mathrm{T0}_{N-2} & -\mathrm{T0}_{N-1} & +\beta\tau_{N-1} \\ \mathrm{T0}_{N-1} & 0 & +\gamma\tau_N \end{pmatrix} \tag{2}$$

where M is a tri-diagonal matrix, with main diagonal $[1+\gamma\ 2+\beta\ ...2+\beta\ ...\ 2+\beta\ 1+\gamma]$ and lower and upper diagonal $[-1\ -1\ ...-1\ ...\ -1\ -1]$ where γ is used for specific border weighting.

2.3 PSOLA Synthesis

2.3.1 Voiced parts

For the voiced parts, PSOLA synthesis proceeds by overlap-add of the waveforms $s_i(t)$ re-positionned on time instants $\tilde{\mathrm{m}}_j$ (see Figure 2):

$$\begin{cases} \tilde{s}_j(t) = s_i(t + \mathrm{m}_i) \\ \tilde{s}(t) = \sum\limits_j \tilde{s}_j(t - \tilde{\mathrm{m}}_j) \end{cases} \tag{3}$$

where m_i are the PSOLA markers which are the closest to the current time in the input sound file.

A modification of the pitch of the signal from $\mathrm{T0}(t)$ to $\mathrm{T}(t)$ is obtained by changing the distance between the successive waveforms: $\tilde{\mathrm{m}}_j - \tilde{\mathrm{m}}_{j-1} = \mathrm{T}(t)$. In the usual PSOLA, time stretching/compression is obtained by repeating/skipping waveforms.

However, in case of strong time-stretching, the repetition process produces signal discontinuities. This is the reason why a *TDI*-PSOLA (**T**ime **D**omain **I**nterpolation PSOLA) has been proposed [Peeters, 1998]. TDI-PSOLA proceeds by overlap-add of continuously interpolated waveforms:

$$\begin{cases} \tilde{s}_j(t) = \alpha s_i(t + \mathrm{m}_i) + (1-\alpha) s_{i-1}(t + \mathrm{m}_{i-1}) \\ \alpha = (\hat{\mathrm{m}} - \mathrm{m}_{i-1})/(\mathrm{m}_i - \mathrm{m}_{i-1}) \\ \tilde{s}(t) = \sum_j \tilde{s}_j(t - \tilde{\mathrm{m}}_j) \end{cases} \tag{4}$$

where m_{i-1} and m_i are the PSOLA markers which frames the current time, $\hat{\mathrm{m}}$, in the input sound file.

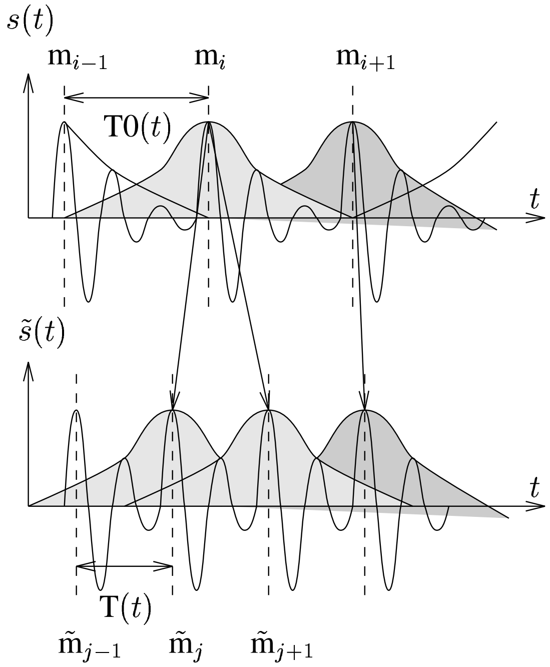

Figure 2: Example of pitch-shifting and time stretching using PSOLA

2.3.2 Unvoiced parts

Unvoiced parts of signals are characterized by a relatively weak long-term correlation (no pitch period) while a short-term correlation is due to the (anti)resonances of the vocal tract.

Special care has to be taken in order to avoid introducing artificial correlations in these parts, which would be perceived as artificial tones (“flanging effect”).

Several methods [Moulines and Charpentier, 1990] [Peeters and Rodet, 1999] has been proposed in order to process the unvoiced part while keeping the low computational-cost advantage of the OLA framework. These methods use various techniques to randomize the phase, in order to reduce the inter-frame correlation.

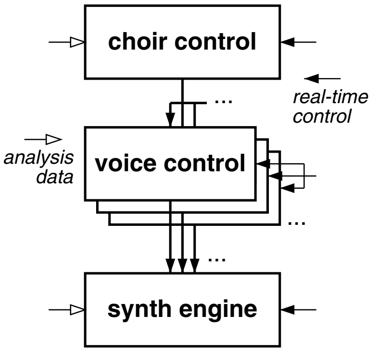

Figure 3: Stages of the voice group synthesis module

3 SYNTHESIZING A GROUP OF VOICES IN REAL-TIME

It was decided to apply a PSOLA resynthesis on recordings of entire phrases of singing solo voices.

In addition to the PSOLA markers determined by the analysis stage two levels of segmentation were manually applied to the recorded phrases:

- pitched notes according to the original score
- segments of musical interest for the process of resynthesis such as phonemes, words and phrases

A synthesis module for *j*Max [Déchelle et al., 1999b] [IRCAM, 2000] was designed, which reads the output of the analysis stage as well as the original sound file and performs the synthesis of a group of individual voices. It was decided to “clone” a whole group of voices from the same sound and analysis data file.

The chosen implementation of the voice group synthesis module shown in figure 3 divides the involved processes into three stages. The first stage determines the parameters, which are common to a group of voices derived from the same analysis data. The parameters are the common pitch and the onset position within an analyzed phrase.

The second stage contains for each voice a process applying individual modulations to the output of the first stage, which causes the voices not to be synchronous and assures that each voice is distinguished from the others. The third stage is a synthesis engine common to all voices performing an optimized construction of the resulting sound from the parameter streams generated by the voice processes of the second stage.

3.1 A PSOLA real-time synthesis algorithm

In the simplest case, the output of the analysis stage is a vector of increasing time values m_i each of them marking the middle of an elementary wave form. For simplicity non-periodic segments are marked using a constant period.

The real-time synthesis algorithm reads a marker file as well as the original sound file. It copies an elementary waveform from a given onset time m_i defined by a marker, applies a windowing function and adds it to the output periodically according to the desired frequency. The fundamental frequency can be either taken from the analysis data as $f0 = \frac{1}{m_{i+1} - m_i}$ or determined as a synthesis parameter of arbitrary value[1].

[1] It is evident that the higher the frequency - or better, the ratio between the orig-

An analysis file can be understood as a pool of available synthesis spectra linearly ordered by their appearance in a recorded phrase[2]. The onset time determines the synthesized spectrum.

In general the onset time and the pitch are independent synthesis parameters so that time-stretching/compression can be easily obtained by moving through the onset times with an arbitrary speed. Modifications of the pitch can be performed simultaneously. The variable increment of the onset time (i.e. speed) represents an interesting synthesis parameter as an alternative to the absolute onset time. The *TDI*-PSOLA (see 2.3.1) interpolation produces a smooth development of timbre for a wide range of onset speeds including extremely slow stretching.

3.2 Resynthesis of unvoiced segments

A first extension of the synthesis algorithm described in the previous section uses the voicing coefficient $v(t)$ output from the analysis stage. The coefficient $v(t)$ indicates whether the sound signal at time t is voiced or unvoiced.

PSOLA synthesis is used for voiced sound segments only. For the synthesis of unvoiced segments a simple granular synthesis algorithm is used [Schnell, 1994]. Grains of constant duration are randomly taken from a limited region around the current onset time. The amount of the onset variation and an overlapping factor are parameters which can be controlled in real-time.

Signal transients are treated in the same way as unvoiced segments.

In order to amplify and attenuate either the voiced or the unvoiced parts, the output of the synthesis stage can be weighted with an amplitude coefficient $c(t)$ calculated from the voicing coefficients by a clipped linear function:

$$c(t) = \begin{cases} 0 & : \frac{v(t)-a}{b-a} \leq 0 \\ 1 & : \frac{v(t)-a}{b-a} \geq 1 \\ \frac{v(t)-a}{b-a} & : else \end{cases} \qquad (5)$$

Giving adequate values for a and b for example the voiced parts can be attenuated or even suppressed so that only the consonants of a phrase are synthesized.

PSOLA synthesis as well as the synthesis of unvoiced segments can be performed by a single granular synthesis engine applying different constraints for either case. Figure 4 shows an overview of the implemented voice resynthesis engine and its control parameters.

The pitch and the onset are computed by a previous synthesis control stage which will be described below.

3.3 Original pitch modulation

Experiments with the implemented synthesis engine for a single voice like other algorithms performing time-stretching on recordings containing vibrato show undesired effects. Blind time-stretching slows down the vibrato frequency and often leads to the perception of an annoying pitch bend in the resulting sound. It is desirable to change the duration of a musical gesture while leaving the vibrato frequency untouched.

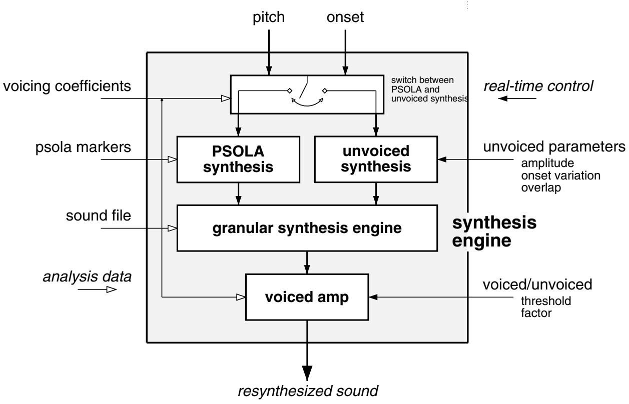

Figure 4: Synthesis engine combining PSOLA and unvoiced synthesis

For the implemented algorithm, the original pitch modulation is removed from the analysis data in two steps:

1. segmentation of the recorded singing voice into notes for voiced segments
2. determination of an averaged (note) frequency $\bar{f}0$ for each segment

An example of the segmentation of a singing voice phrase derived from the voicing coefficient, and the assignment of the note frequency according to the score is shown in figure 5.

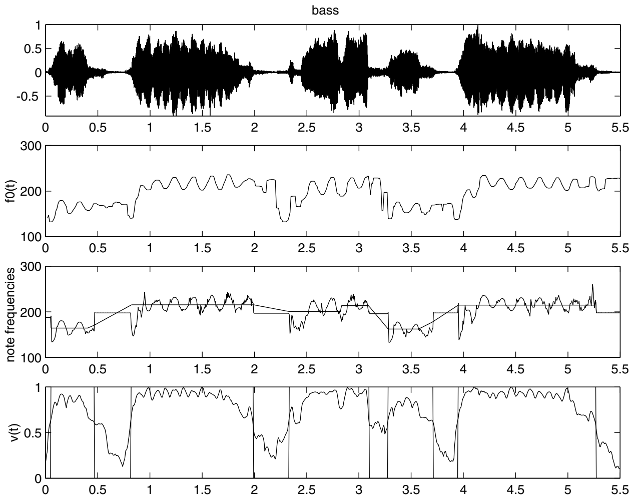

Figure 5: Note segmentation and pitch of a singing voice phrase

The note frequency is integrated into the analysis data by assigning it to each marker within a given segment representing a note. In addition, a modulation coefficient $k(t)$ is stored with each marker which contains the original pitch modulation of a note:

$$k(t) = \frac{f0(t) - \bar{f}0(t)}{\bar{f}0(t)} \qquad (6)$$

inal frequency and the synthesized frequency - the more the elementary waveforms overlap. Since the computation load of a typical synthesis algorithm depends of the number of simultaneously calculated overlapping waveforms, it increases with the synthesized frequency.

[2] Although this is convenient for the resynthesis of entire words and phrases for further applications, it could be interesting to construct differently structured feature spaces from the same analysis data.

The original instantaneous frequency can be recalculated as $f0(t) = \bar{f}0(t)(1 + M \cdot k(t))$. The modulation index M determines the amount of original re-synthesized pitch modulation. This technique allows a preservation of the musical expression contained in the pitch modulation of a note when the absolute original frequency is replaced. For a modulation index of $M = 0$ the modulation is removed and can be replaced by a synthesized modulation independent of the applied time-stretching/compression. With $M > 1$ an exaggerated modulation can be achieved.

3.4 Controlling a group of voices

Figure 6 shows the control stage determining pitch and onset for the synthesis of a single voice as well as for a group of voices.

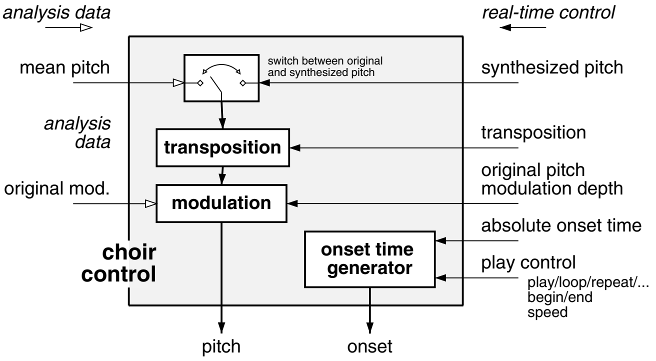

Figure 6: Pitch and onset control for a group of voices

The pitch is input from the analysis data or as real-time control parameter and a transposition (given in cent) is calculated before the original modulation. The onset time is generated by a module, which advances the onset time according to an arbitrary segmentation. A segment is specified by its begin and end time, its reading mode (play forward/backward, loop back and forth, repeat looping forward, ...) and the speed at which the onset time is advancing.

3.5 Individual variations of the voices

A major concern designing the algorithm was the variations of timbre and pitch performed by each voice in order to obtain a realistic impression of a choir by the superposition of multiple voices re-synthesized from the same analysis data.

In intensive experiments comparing synthesized groups of voices with recordings of real choir groups the following variations where found important:

- pitch variations
- timing (onset) variations
- vibrato frequency variations

The pitch and timing variations are mainly corresponding to the individual imprecision of a singer in a choir making that never two singers sing exactly the same pitch and start and end the same note at the same time. The onset variations lead as well to a diversity of the spectrum of the voices at each moment. A synthesized vibrato of an individual frequency can be added to each voice.

It was considered to give individual formant characters to each synthesis voice in order to create additional individuality close to reality. However the experiments have shown that in the context of the accompanying sound and spatialization effects, the additional computation was found to be too costly in comparison with the produced effect[3].

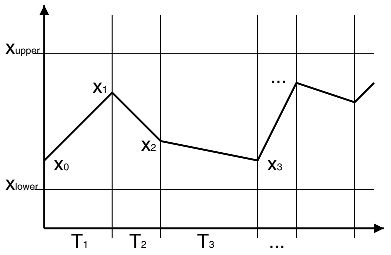

Figure 7: Example of a random break point function

The variations for each voice are performed by *random break point functions (rbpf)*. In the synthesis cycle of the algorithm an *rbpf* computes for each synthesized waveform a new value $x(t)$ on a line segment between two break-points x_i guaranteeing a smooth development of the synthesized sound (see figure 7). A new target value x_i as well as a new interpolation time T_i are randomly chosen inside the boundaries each time a target value x_{i-1} is reached.

The parameters of a general *rbpf* generator are the boundaries for the generated values (x_{lower}/x_{upper}) and for the duration (T_{lower}/T_{upper}) between two successive break-points. As an alternative to its duration as well the slope of a line segment can be randomly chosen taking in this case the minimum and maximum slope as parameters.

Using these generators a constantly changing pitch transposition, onset time and vibrato frequency can be performed. Depending on the chosen parameters this can result either in a realistic chorus effect or, when exaggerating the parameter values, a completely different impression.

A schematic overview of the modulations for each voice acting on the pitch and onset produced by the choir control module is shown in figure 8. The produced pitch and onset parameters are directly fed into the synthesis engine.

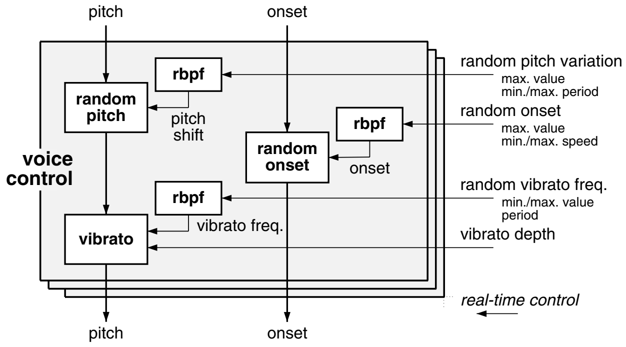

Figure 8: Individual pitch and onset variations performed for each voice

[3] The computation load for a synthesis voice using a simple re-sampling technique in order to modify its formants must be estimated as about three times as costly as a straight forward PSOLA synthesis with the same transposition or overlap ratio.

4 CONSTRUCTING THE VIRTUAL CHOIR

The implementation of the voice group synthesis module was accompanied by intensive experiments in order to adjust the synthesis algorithm and parameter values corresponding to a realistic choral sound.

The sound sources for the PSOLA analysis and further choral sounds for comparative tests were obtained in a special recording session with the choir of the Opera Bastille Paris in the *Espace de Projection* at IRCAM configured for a dry acoustic. The same musical phrases written by Manoury based on a Czech text were sung individually by the four choir sections (soprano, alto, tenor and bass) in unison. For each choir section several takes of 2, 4, 6 and 10 singers as well as a solo singer were recorded.

Various analysis tools have been tested in the research of the choir sound as a phenomenon of the superposition of single voices and their individualities as well as its particularities of the signal level.

Classical signal models (such as those used for the estimation of pitch period or spectral peaks) are difficult to apply in the case of a choir signal. The signal is composed of several sources of slightly shifted frequencies spreading and shifting the lines of the spectrum and preventing usual sinusoidal analysis methods from working properly. The de-synchronization of the signal sources prevents most usual temporal method from working with the mixed signal.

The nature and amount of variation between one singer and another in terms of timbre and intonation[4] have been considered as well as the amount of synchronization between the singers at different points of a phrase and the synchronization of their vibrato. For example it was found that plosive consonants correspond to stronger synchronization points than than vowels.

Only the recordings of solo singers have been analyzed and segmented. The re-synthesized sound of a group of voices by the implemented module was perceptually compared with the original recording of multiple singers singing the same musical phrase. The experiments have shown that about 7 well differentiated synthetic voices gave the same impression as a group of 10 real voices. A pitch variation in the range of 25 cents and a uncertainty of 20 ms for the onset position have been found to give a realistic impression of a choir.

4.1 Segmentation

In addition to the segmentation into elementary waveforms (by the PSOLA markers), voiced and unvoiced segments as well as pitched notes (manually, see 3.3), a fourth level of segmentation was applied to the analysis data. It cuts the musical phrases into segments of musical interest like phonemes, words and entire phrases.

With this segmentation, the recorded phrases can be used as a data base for a wide range of different synthesis processes. The sequence of timbre and pitch of the original phrases can be completely re-composed. In order to reconstitute an entire virtual choir, phrases of different voice groups, based on different analysis files, can be re-synchronized word by word.

Interesting effects can be obtained controlling the synthesis by a function of the voicing coefficients. For example, the voiced segments of the signal can be more stretched than unvoiced segments. Similarly, vowels and consonants can be independently processed and spatialized.

4.2 Spatialization

The realization of the piece ”Vertigo Apocalypsis” by Philippe Schoeller at IRCAM [Nouno, 1999] showed the importance of spatialization for a realistic impression of a choir. In this work multiple solo recorded singers were precisely placed in the acoustic space. For “K...”, each re-synthesized voice or voice section will be processed by IRCAM’s *Spatializateur* [Jot and Warusfel, 1995] allowing the composer to control the spatial placement and extent of the virtual choir.

In the general context of the electro-acoustic orchestration of “K...”, an important role will be given to the Spatializateur taking into account the architectural and acoustic specificities of the opera house.

4.3 Conclusions

The implemented system reveals itself to be very versatile and flexible. The choir impression obtained with it is much more interesting and realistic than any classical chorus effect.

The used synthesis technique produces an excellent audio quality, close to the choir recordings. The quality of transformation achieved with PSOLA is better than the usual techniques based on re-sampling.

The application of an individual vibrato for each synthesis voice after having canceled the recorded vibrato turned out to be extremely effective for the perception of the choral effect.

The efficiency of the algorithm allows polyphony of a large number of voices. The virtual choir is embedded into a rich environment of various synthesis and transformation techniques such as phase-aligned formants synthesis, sampling and classical sound transformations like harmonizing and frequency-shifting. The virtual choir will be constituted of 32 simultaneous synthesis voices grouped into 8 sections.

During the experiments it appeared clearly that vocal vibrato does not affect only the fundamental frequency. It is accompanied by synchronized amplitude and spectral modulations. Canceling the vibrato by smoothing the pitch leaves an effect of unwanted roughness in the resulting sound.

Another limitation of the system appears for the processing of very high soprano notes (above 1000 Hz). For these frequencies the impulse response of the vocal tract extends over more than one signal period and can not be isolated by simple windowing of the time domain signal.

4.4 Future extensions

While the used analysis algorithm performs signal characterization into voiced and unvoiced parts in the time/frequency domain, in the context of “K...” it has only been applied for segmentation in the time domain. Separation into both time and frequency domains would certainly benefit the system, especially for mixed voiced/unvoiced signals (voiced consonants).

In order to produce timbre differences between individual voices, several techniques are currently being evaluated. They rely on an efficient modification of the spectral envelope (i.e. formants) of the vocal signal.

An interesting potential of the paradigm of superposing simple solo voices can be seen in its application to non-vocal sounds. The synthesis of groups of musical instruments could be obtained in the same way as the virtual choir, i.e. deriving the violin section of an orchestra from a single violin recording.

[4] Expressed by Sundberg’s *degree of unison* [Sundberg, 1987].

REFERENCES

[Charpentier, 1988] Charpentier, F. (1988). *Traitement de la parole par Analyse/Synthèse de Fourier application à la synthèse par diphones.* PhD thesis, ENST, Paris, France.

[d'Alessandro and Rodet, 1989] d'Alessandro, C. and Rodet, X. (1989). Synthèse et analyse-synthèse par fonctions d'ondes formantiques. *J. Acoustique*, (2):163–169.

[Déchelle et al., 1998] Déchelle, F., Borghesi, R., Cecco, M. D., Maggi, E., Rovan, B., and Schnell, N. (1998). jMax: A new JAVA-based Editing and Control System for Real-time Musical Applications. In *Proceedings of the International Computer Music Conference*, San Francisco. International Computer Music Association.

[Déchelle et al., 1999a] Déchelle, F., Borghesi, R., Cecco, M. D., Maggi, E., Rovan, B., and Schnell, N. (1999a). jMax: An Environment for Real-Time Musical Applications. *Computer Music Journal*, 23(3):50–58.

[Déchelle et al., 1999b] Déchelle, F., Cecco, M. D., Maggi, E., and Schnell, N. (1999b). jMax Recent Developments. In *Proceedings of the 1999 International Computer Music Conference*, San Francisco. International Computer Music Association.

[IRCAM, 2000] IRCAM (2000). *jMax home page*. IRCAM, http://www.ircam.fr/jmax.

[Jot and Warusfel, 1995] Jot, J.-M. and Warusfel, O. (1995). A real-time spatial sound processor for music and virtual reality applications. In *Proceedings of the International Computer Music Conference*, Banff. International Computer Music Association.

[Kortekaas, 1997] Kortekaas, R. (1997). *Physiological and psychoacoustical correlates of perceiving natural and modified speech.* PhD thesis, TU, Eindhoven, Holland.

[Moulines and Charpentier, 1990] Moulines, E. and Charpentier, F. (1990). Pitch-Synchronous Waveform Processing Techniques for Text-To-Speech Synthesis using Diphones. *Speech Communication*, (9):453–467.

[Nouno, 1999] Nouno, G. (1999). Vertigo apocalypsis. *Internal Report IRCAM*.

[Peeters, 1998] Peeters, G. (1998). Analyse-Synthèse des sons musicaux par la mèthode PSOLA. In *Journées Informatique Musicale*, Agelonde, France.

[Peeters and Rodet, 1999] Peeters, G. and Rodet, X. (1999). Non-Stationary Analysis/Synthesis using Spectrum Peak Shape Distortion, Phase and Reassigned Spectrum. In *ICSPAT*, Orlando, USA.

[Schnell, 1994] Schnell, N. (1994). GRAINY - Granularynthese in Echtzeit. *Beiträge zur Elektronischen Musik*, (4).

[Sundberg, 1987] Sundberg, J. (1987). *The Science of Singing Voice*. University Press, Stocholm.

[Ternströn, 1989] Ternströn, S. (1989). *Acoustical Aspects of Choir Singing*. Royal Institute of Technology, Northern Illinois.