

Musical Applications using Real-time Frequency Domain Signal Processing

Zack Settel
Lysis : Ircam
1 place Igor Stravinsky
75004 Paris, France
settel@ircam.fr

Cort Lippe
University of Buffalo: Music Dept.
222 Baird Hall: Box 604700
Buffalo, New York 14260-4700
lippe@acsu.buffalo.edu

Abstract

This paper presents real-time musical applications using the IRCAM Signal Processing Workstation which make use of FFT/IFFT-based resynthesis for timbral transformation in a musical context. An intuitive and straightforward user interface, intended for use by musicians, has been developed by the authors in the Max programming environment. Techniques for high quality time-stretching, filtering, cross-synthesis, dynamic range processing, and spectrum shaping are presented along with dynamic control structures that allow for both fine timbral modification and control of complex sound transformations using few parameters.

Key words

convolution, time stretching, cross-synthesis, FFT/IFFT, ISPW, Max, spectral envelope

1. Introduction

The Fast Fourier Transform (FFT) is a powerful general-purpose algorithm widely used in signal analysis. FFTs are useful when the spectral information of a signal is needed, such as in pitch tracking or vocoding algorithms. The FFT can be combined with the Inverse Fast Fourier Transform (IFFT) in order to resynthesize signals based on analyses. This application of the FFT/IFFT is of particular interest in electro-acoustic music because it allows for a high degree of control of a given signal's spectral information (an important aspect of timbre) allowing for flexible, and efficient implementation of signal processing algorithms.

This paper presents real-time musical applications using the IRCAM Signal Processing Workstation (ISPW) [Lindemann, Starkier, and Dechelle 1991] which make use of FFT/IFFT-based resynthesis for timbral transformation in a compositional context. Taking a pragmatic approach, the authors have developed a user interface in the Max programming environment [Puckette, 1988] for the prototyping and development of signal processing applications intended for use by musicians. Development in the Max programming environment [Puckette, 1991] tends to be simple and quite rapid: digital signal processing (DSP) programming in Max requires no compilation; control and DSP objects run on the same processor, and the DSP library provides a wide range of unit generators, including the FFT and IFFT modules. Techniques for filtering, cross-synthesis, noise reduction, and dynamic spectral shaping have been explored, as well as control structures derived from real-time signal analyses via pitch-tracking, envelope following, noise gating, and signal compression and expansion [Lippe & Puckette 1991]. These real-time musical applications offer composers an intuitive approach to timbral transformation in electro-acoustic music, and new possibilities in the domain of live signal processing that promise to be of general interest to musicians.

2. The FFT in Real Time

Traditionally the FFT/IFFT has been widely used outside of real-time for various signal analysis/re-synthesis applications that modify the durations and spectra of pre-recorded sound [Haddad & Parsons 1991]. With the ability to use the FFT/IFFT in real-time, live signal-processing in the spectral domain becomes possible, offering attractive alternatives to standard time-domain signal processing techniques. Some of these alternatives offer a great deal of power, run-time economy, and flexibility, as compared with standard time-domain techniques [Gordon & Strawn 1987]. In addition, the FFT offers both a high degree of precision in the spectral domain, and straightforward means for exploitation of this information. Finally, since real-time use of the FFT has been prohibitive for musicians in the past due to computational limitations of computer music systems, this research offers some relatively new possibilities in the domain of real time.

2.1 Algorithms and basic operations

All of the signal processing applications discussed in this paper modify incoming signals and are based on the same general DSP configuration. The DSP configuration includes the following basic steps: (1) transformation of the input signals into the spectral domain using the FFT, (2) operations on the signals' spectra, (3) and resynthesis of the modified spectra using the IFFT. Operations in the spectral domain (FFT data in the form of rectangular coordinates) include applying functions (often stored in tables), convolution (complex multiplication), addition, taking the square root (used in obtaining an amplitude spectrum), noise gating (of both frequency and amplitude), and compression and expansion. Differences in the choice of spectral domain operations, kinds of input signals used, and signal routing determine the nature of a given application: small changes to the topology of the DSP configuration can result in significant changes to its functionality. Thus, we are able to reuse much of our code in diverse applications. (For example, though functionally dissimilar, high-resolution filtering and subtractive synthesis differ only slightly in terms of their implementation.)

3. Applications

3.1 Complex spectral envelopes and cross synthesis

A static spectral envelope used in a simple filtering application or in subtractive synthesis can be drawn by hand or obtained through signal analysis. In cross synthesis a continuously changing (dynamic) filter can be created by doing an FFT analysis of a signal (signal B) and extracting its spectral envelope, or amplitude spectrum, which describes how another signal (signal A) will be filtered. Thus, the pitch/phase information of signal A and the time varying spectral envelope of signal B are combined to form the output signal. Audio signals produced by standard signal processing modules such as frequency modulation (FM) are of particular interest for cross synthesis because they can produce rich, easily modified, smoothly and nonlinearly varying spectra [Chowning 1973] which can yield complex time varying spectral envelopes. Other standard signal processing techniques, such as amplitude modulation (AM), additive synthesis, or a band-pass filtering offer rich varying spectral information using relatively simple means with few control parameters. One of the advantages of using standard modules is that electronic musicians are familiar with them, and have a certain degree of control and understanding of their spectra.

It should also be noted that interesting transformations can be produced by simply convolving signal A's spectrum with signal B's spectrum (keeping both the phase and amplitude information of signal B as well as signal A). In this case, the phase (frequency) and amplitude information from each signal figures in the output signal.

3.2 Mapping qualities of one signal to another

Musically, we have found that in some cases, the relationship between signal A and signal B can become much more unified if certain parameters of signal A are used to control signal B. In other words, real-time continuous control parameters can be derived from, for instance, signal A, and used to control signal B. For example, the pitch of signal A can be tracked and applied to signal B (an FM pair) to control the two oscillators' frequencies. Envelope following of signal A can yield expressive information which can be used to control the intensity of frequency modulation (FM index) of signal B.

3.3 Frequency dependent spatialization

In the spectral domain, the phases of a given signal's frequency components can be independently rotated in order to change the component's energy distribution in the real and imaginary part of the output signal. Since the real and imaginary parts of the IFFT's output can be assigned to separate output channels, which are in turn connected to different loudspeakers, it is possible to control a given frequency's energy level in each loud-speaker using phase rotation.

3.4 Band-limited energy dependent noise-gating

In the spectral domain, the energy of a given signal's frequency components can be independently modified. Our noise reduction algorithm is based on a technique [Moorer & Berger, 1984] that allows independent amplitude gating threshold levels to be specified for each frequency component in a given signal. With a user-defined transfer function, the energy of a given frequency component can be altered based on its intensity and particular threshold level. This technique, outside of being potentially useful for noise reduction, can be exaggerated in order to create unusual spectral transformations of input signals by emphasizing or masking specific components.

3.5 Band-limited energy dependent noise-gate

In the spectral domain, the energy of a given signal's frequency components can be independently modified. Our noise reduction algorithm is based on a technique [Moorer & Berger, 1984] that allows independent amplitude gating threshold levels to be specified for up to 512 frequency band-limited regions of a given signal. With a user-defined transfer function, the energy in a given frequency range can be altered based on its intensity and a user-specified threshold level. This technique, besides being potentially useful for noise reduction, can be exaggerated in order to create unusual spectral transformations of input signals, resembling extreme chorusing effects. Using non-linear transfer functions, it is possible to modify the relative intensities of the input's frequency components, allowing for example, masked or less important components to be emphasized and brought to the aural foreground.

3.6 Band-limited frequency dependent noise-gate

Similar to the noise-gate described above, this module functions independently of gain; its output depends on the stability of the frequency components of the input signal. Using a technique borrowed from phase-vocoding [Gordon & 1987], time-varying frequency differences of components in a given band-limited region of the spectrum are used to determine the stability of those components. Pitched components in the input signal tend to be stable and can thus be independently boosted or attenuated.

3.7 Improvements to frequency domain-based time expansion/compression techniques

Useful techniques for sound manipulation in the frequency domain are proposed by the phase vocoder [Dolson 1986; Nieberle & Warstat 1992]. One particularly useful application of this technique, popularly referred to as time stretching, allows a sound's length to be modified independently of its pitch (frequency content). However, this application has two particular shortcomings which have been revealed in practice--especially when a sound's duration is lengthened. The first problem is that transient components, normally quite brief, become "unusually long". An additional problem is that noise components tend to become "less agitated" since their rate of change is not retained during time stretching.

These shortcomings can be clearly heard for example, when time stretching speech. A spoken text rapidly loses intelligibility when: A) transients are protracted; consonants are no longer recognizable and lose their functional role as articulators, and B) noise components are "de-agitated"; sibilant components such as "sh", lose their distinctive "windy" noisy quality.

In order to overcome these two problems, we have made the following additions to our phase vocoding algorithm: 1) Selective time stretching, which acts only on non-transient portions of the signal. 2) A resynthesis stage that reproduces the original rate of change (with respect to frequency) of noise components, using statistical approximations.

4. Combining processing modules

The modules described above may be placed in series in order to perform multiple processing operations on a sound. Since most of the cost of processing occurs in time/frequency domain conversion (FFT and IFFT), modules operating in the frequency domain may be ganged together with great efficiency, requiring no additional steps of conversion.

4.1 Dynamic range processing with graphic EQ

The 512 band filter module described above may be easily connected to another module, providing the additional possibility of high resolution filtering.

5. Input scaling and additional control parameters for cross-synthesis

When two input signals of similar intensities and spectral distributions (eg.: two singers' voices) are convolved, the resulting spectral distribution can be quite different. In such a case, strong "low-mid-range" components typically become much louder, while weaker higher frequency components virtually disappear. For example, when convolving a signal by itself, the resulting energy of a given component is its square, as shown below:

$$\text{resulting energy} = A * A$$

By scaling the input or output signal(s), the "original" spectral distribution can be preserved. If in the above example, we take the square-root of the output signal, a given component's energy is now the logarithmic mean of the energy of its input signals, and not the square. In this case we recover the original input signal.

$$\text{resulting energy} = \sqrt{A * A} = A$$

5.1 Including dynamic range processing in cross-synthesis algorithms The algorithms for cross-synthesis discussed above can be expanded to include additional processing units, that allow for scaling and provide more control parameters. A compressor/expander based on the energy dependent noise-gate mentioned earlier is applied to one of the two input signals before convolution. The choice of compression/expansion functions determines the way the energy in the output signal will be scaled; certain choices (such as a square root function) help preserve the spectral distribution of the input signals. Thus, two input signals of similar intensities and spectral distributions will combine to form a spectrum whose components' relative energies remain little changed. By choosing a compression/expansion function that boosts or cuts the energy of weaker components in one of the input signals, the degree of spectral intersection of the two inputs can be specified. The compression/expansion ratio parameter (or intensity of compression/expansion) provides dynamic control over the "degree of spectral

intersection" of the two input signals. This parameter is particularly useful when cross-synthesizing dissimilar sounds such as voice and percussion.

6. Future Directions

The authors are currently working on alternative methods of sampling and granular synthesis that operate the frequency domain, based on real-time phase vocoding [van der Heide, 1994]. At present we are able to modify a sound's spectrum and duration independently, and are working towards being able to perform pitch transposition independently of the spectral envelope (formant structure), thus allowing one to change the pitch of a sound without seriously altering its timbral quality. Additionally, we are exploring techniques for smooth sample looping and cross-fading between sounds.

7. Summary

With the arrival of the real-time FFT/IFFT in flexible, relatively general, and easily programmable DSP/control environments such as Max, non-engineers may begin to explore new possibilities in signal processing. Though our work is still at an initial stage, we have gained some valuable practical experience in manipulating sounds in the spectral domain. Real-time convolution can be quite straightforward and is a powerful tool for transforming sounds. The flexibility with which spectral transformations can be done is appealing. Our DSP configuration is fairly simple, and changes to its topology and parameters can be made quickly. Control signals resulting from detection and tracking of musical parameters offer composers and performers a rich palette of possibilities lending themselves equally well to studio and live performance applications.

Acknowledgments

The authors would like to thank Miller Puckette, Bennett Smith and Stefan Bilbao for their invaluable technical and musical insights.

References

- Chowning, J. 1973. "The Synthesis of Complex Audio Spectra by means of Frequency Modulation" *Journal of the Acoustical Society of America* 21(7), 526-534.
- Dolson, M. 1986. "The phase vocoder: a tutorial", *Computer Music Journal*, 10(4), 14-27
- Gordon, J. and Strawn J. 1987. "An introduction to the phase vocoder", *Proceedings, CCRMA, Department of Music, Stanford University, February 1987.*
- Haddad R, and Parsons, T. 1991. "Digital Signal Processing, Theory, Applications and Hardware", New York: Computer Science Press
- Heide van der, E. 1993. Private communication.
- Lindemann, E., Starkier, M., and Dechelle, F. 1991. "The Architecture of the IRCAM Music Workstation." *Computer Music Journal* 15(3), 41-49.
- Lippe, C. and Puckette, M. 1991. "Musical Performance Using the IRCAM Workstation", *Proceedings of the 1991 International Computer Music Conference.* San Francisco: International Computer Music Association.
- Moorer and Berger, 1984. "Linear-Phase Bandsplitting: Theory and Applications", *Audio Engineering Society* (preprint #2132), New York: 76th AES convention 1984.
- Nieberle, R and Warstat, M 1992. "Implementation of an analysis/synthesis system on a DSP56001 for general purpose sound processing", *Proceedings of the 1992 International Computer Music Conference.* San Francisco: International Computer Music Association.

McNally, G. W., 1984. "Dynamic Range Control of Digital Audio Signals." *Journal of the Audio Engineering Society*, Vol. 32, 1894.

Puckette, M. 1988. "The Patcher." *Proceedings of the 1988 International Computer Music Conference*. San Francisco: International Computer Music Association.

Puckette, M., 1991. "FTS: A Real-time Monitor for Multiprocessor Music Synthesis." *Music Conference*, 420-429, San Francisco: International Computer Music Association.

Smith, Bennett K., 1994 Personal communication.