

The development of GiST, a Granular Synthesis Toolkit Based on an Extension of the FOF Generator

Gerhard Eckel and Manuel Rocha Iturbide

IRCAM
Paris, France

Barbara Becker
GMD
St. Augustin, Germany

Abstract: The Granular Synthesis Toolkit (GiST) comprises a set of Max/FTS external objects running on the IRCAM Signal Processing Workstation (ISPW). These objects can be used to build a large variety of granular synthesis and sound granulation applications. Unlike other approaches of implementing granular synthesis on the ISPW, the GiST allows for precise temporal control as it is needed for high-quality results. Besides the musical motivations of the project and its technological context, the development of GiST was driven by a set of design guidelines stemming from empirical and theoretical investigations concerning the use of computer tools in music composition in general.

Introduction

The development of GiST has been driven and conditioned by three main factors: a particular compositional interest in the various types of granular synthesis and their possible articulation, a set of hypotheses stemming from empirical and theoretical investigations concerning the use of technological tools in the compositional process, and the technical constraints imposed by the computer platform chosen to realize the granular synthesis toolkit. Throughout this text we will present each of these three main aspects individually and with some detail while trying to highlight the underlying links between them.

1. Compositional Motivations

The use of synthesis techniques referred to as granular synthesis or sound granulation has a long and rich history in computer music

(for a survey see: Roads, C.). Despite the mathematical complexity of some of the granular models used today, granular synthesis is nevertheless attractive to composers because of its conceptual simplicity: small fragments of sounds are superimposed to construct more complex sound material. All complexity of a particular granular synthesis application lies in the way this basic concept is applied and how a chosen synthesis setup can be controlled by the composer. Depending on the time scale the control operates on, it will allow for musical structuring of sound on the level of the temporal or rhythmical (Xenakis, I.) or on the level of spectral, timbral, or harmonic organisation (Risset, J.-C.). The fact that the same structures used on different time scales do affect different domains of musical perception has always been a rich source of compositional imagination (Stockhausen, K.). In the field of sound synthesis, it is certainly granular synthesis which is best suited to compositionally exploit this fundamental principle of perception. And it was this very dual nature of perception with respect to time domain and frequency domain structures which gave first rise to the idea of granular synthesis (Gabor, D.).

In the context of this project the main motivation of using granular synthesis was its capacity to allow for a natural and rich link between temporal and spectral organisation. We sought for a synthesis method allowing for the composition of sound on both the micro and the macro time level. The fact that in the case of granular synthesis the same underlying model or representation can be used to operate on these two aspects appeared particularly appealing to us. Furthermore, we consider the genericity of control achievable with a granular approach an important source of unexpected and surprising results that may stimulate the compositional process. This unpredictability, which is due to the complexity inherent to granular synthesis is sometimes regarded as a defect. We, on the contrary, consider it a richness which we want to explore in this project. But in order to avoid drifting into uncontrollable chaos, a well defined environment had to be built that allows for an intuitive exploration of the yet unstructured field of new possibilities.

When trying to articulate spectral and temporal organisation with a granular approach, one well known synthesis technique, although

not always thought of as a granular technique, had to be taken into account in this project: CHANT (Rodet, X.). The formant-wave-function (FOF) approach, which is the constitutive element of the CHANT technique, was developed to allow for an efficient production and control of sounds with formantic structure (like the speaking or singing human voice). Since a FOF can be regarded as a special granular generator, the basic idea of this project was, as already suggested by others before (Clarke, M.), to extend the FOF technique in order to link it to other, more traditional granular synthesis approaches. Considerable practical experience and a good understanding of both the FOF technique and statistically controlled granular synthesis was considered a good starting point to reach our goal of merging and articulating the two.

In order to interactively explore the domains of interest spanned between extreme cases such as periodic triggering of grains as in the case of CHANT and stochastic triggering (with probabilistic control of the other grain parameters), a real-time environment was sought for. This rather pragmatic approach seemed the only possibility to cope - in compositional practice - with the complexity introduced by our extension of the FOF generator.

2. Theoretical and Empirical Background

The question of how tools for composers should look like and which development strategy was best suited to develop them has been subject of empirical and theoretical inquiry prior to this project (Becker, B. & Eckel, G.). For the development of GiST three hypotheses about the nature of tools for composers built the background for the concrete development work. These three hypotheses are: 1) The concepts imbedded in tools have to be made transparent; 2) Tools should be conceived as toolboxes to allow for maximum flexibility; 3) The composers have to be integrated into the development process.

2.1 Transparency

The fact that every tool structures and influences the work carried out with it is of special importance in the case of tools used for artistic work. The confrontation of compositional ideas with the concepts imbedded in technological tools like sound synthesis systems has usually an important influence on the compositional

work itself. Unfortunately most of these concepts remain implicit in currently available sound synthesis tools. This may be due to insufficient awareness of the composers about the consequences of such implicit influence on their work. But usually composers are aware of the dangers that may result from using tools they do not master. It is rather that the tools are made such that the imbedded concepts remain implicit and no effort is made by their developers to make them transparent. It is thus essential to conceive tools for artists such that the concepts they convey can easily be discovered. Especially limitations due to technological constraints as they can be found in all kinds of computer tools have to be explicated in detail in order to prevent composers from being misled. It is rather the rule than the exception that composers spend far too much time with trying to understand why something does not work the way they expect although their approach is conceptually sound. Much frustration results from such forced detours typical for the work with many sound synthesis environments. Once the tool's implicit limitation causing a particular problem is found, much time is wasted in circumventing the obstacle. It has been (cynically?) argued that this kind of resistance of tools may stimulate the creative work because it can lead to unpredictable and surprising results. Although that may be true in rare cases we believe that there is enough unpredictability introduced to artistic work by the idiosyncrasies of perception that there is no need to add some more stemming from the proper nature of the devices stimulating perception.

2.2 Flexibility

Openness and flexibility are generally regarded as the most important characteristics of tools for artists. This is because the very nature of the creative process makes it hard to predict how artists will use tools in a particular situation. As recent examples show convincingly (Puckette, M.), the quest for openness is best responded to by modular systems that define only the basic tool building blocks and allow (and oblige) composers themselves to assemble their tools corresponding to their concrete needs. Such approach minimizes also the effect that specialized tools tend to side-track a user who wants to employ them to solve problems slightly different than they were designed for. The idea of a modular system or toolbox allows the final design of the tool to

respond to the concrete problem. The resulting implication of the composer in the tool building process responds to yet another problem frequently observed when asking artists about their needs: Most of the time they are unable to describe with sufficient precision what they need or they really do (and can) not know exactly before starting to compose. It can also be observed that the few composers who develop their own systems from scratch usually end up building large and flexible toolboxes (e.g. Essl, K.), which they use to construct the tools needed for a particular composition. But since these composers are a very small minority, the development of the basic tool building blocks usually has to be undertaken by software engineers.

2.3 Participative Design

In order to guarantee that the modules of a toolbox are well adapted to a problem domain (e.g. granular synthesis) close collaboration between software engineers and composers is essential. In our project we adopted an approach known as evolutionary participative software design. During an intense period consisting of many rapid prototyping and testing cycles, engineers and composers try to elaborate together a specification of the basic modules of a toolbox tailored for a specific problem domain. The resulting specification is precise enough in order to be turned into a solid and efficient implementation by software engineers. The final implementation is validated by the composers and compared to the prototypes which serve as reference implementations. This approach, which has already proven its validity in other projects (Eckel, G. & González-Arroyo, R.), is supported at IRCAM by a working mode proposed to composers allowing them to take active part in the software development process (*compositeurs en recherche*).

3. Implementation

The granular synthesis toolkit (GiST) is a small set of generic synthesis and control modules implemented as Max/FTS (Puckette, M.) external objects on the ISPW (Lindemann, E.). The GiST was developed with the goal of supporting a large variety of synthesis applications including the CHANT synthesis technique (Rodet, X.) and other granular synthesis or sound granulation techniques used to explore the inner complexity of sound (Truax, B.). The

development of the GiST owns much to the experiences made with the development of the Foo synthesis system (Eckel, G. & González-Arroyo, R.): Besides providing a reference implementation of the granular modules, Foo was used as a prototyping environment for the GiST modules.

The basic difficulty of GiST's implementation on the ISPW was the precise temporal control usually required by granular synthesizers. In many cases, as for example with the FOF generator used in the CHANT synthesizer, the grains have to be triggered with a temporal precision of much less than one sample period. This is needed in order to be able to produce clean periodic sounds with sufficient frequency resolution even when the fundamental frequency is relatively high. However, in the case of CHANT, which is normally used to produce periodic signals, it is not necessary to give the user access to this temporal information and therefore it can be treated inside the object. Yet, direct control over the trigger dates is needed when using other - non-periodic - modes of triggering as it is the case with the different types of stochastic control employed in other granular synthesis applications. Thus GiST's essence is to allow for precise temporal control.

3.1 Time-Tagged Triggers

Since the scheduling mechanism of the current version of Max/FTS permits control messages to be exchanged between objects only every 64 samples (= 1 tick), this mode of control could not be used directly. Another than the standard protocol of triggering by BANG messages was defined and implemented in all the GiST modules: Time-Tagged Triggers (T3). In Max/FTS, a T3 is nothing else than a message containing one floating-point number which specifies the delay in ms after which, counting from the current tick, the trigger should go off. This simple protocol could be implemented on the level of the external objects only, without requiring any changes on the system level of FTS. And since a T3 is simply a floating-point number, it can be treated with the standard Max objects easily.

Example 1 shows the use of GiST's **tsig~** object to produce a unit impulse 0.5 ms after the current tick. Assuming that the current

value of the **tsig~** is zero, the output signal will be set to one after 0.5 ms. One sample (i.e. $\sim 23\mu\text{s}$ at a sample rate of 44.1 kHz) later it will be set back to zero. The **tsig~** object has two inlets: The right one receives the value to which the output signal will jump after the delay specified by the T3 arriving at the left inlet. Like many other Max objects, the left inlet accepts also lists containing a T3 and the output value. The list notation is used in example 1.

Example 1

Besides objects that accept T3 messages GiST also comprises objects that generate them. The simplest one is shown in example 2: **tphasor~**. This object accepts at its left inlet a signal specifying the frequency (in Hz) of the series of T3 messages to be generated and output to the objects outlet. Example 2 shows a patch fragment producing a pulse train with a frequency of 882 Hz (chosen such that the period amounts to precisely 50 samples at a sample rate of 44.1 kHz)

Example 2

The two examples illustrate how the T3 messages allow for a precision of temporal control otherwise impossible to achieve with Max/FTS. Thus, the T3 messages are the basis for the design and implementation of the granular synthesis modules presented below.

Prior attempts to realize granular synthesis applications with Max/FTS on the ISPW (e.g. Lippe, C.) were severely hindered by the lack of precise temporal control. For the average user of Max/FTS the limitation of temporal precision is by no means obvious and thus very often a source of problems. Following the quest for transparency, this problem is treated by GiST explicitly by proposing a clearly defined solution (T3 messages), which is also valid for other than granular synthesis applications.

3.2 FOG, the Extended FOF Generator

The central module of the GiST is the FOG generator. This generator is an extension of the formant-wave-function (FOF) generator, which was first implemented on the ISPW in 1994. A FOF produces eventually overlapping fragments of exponentially decaying sinusoids. These fragments are shaped by a special amplitude envelope which consists of an attack and a decay part with cosine shape and a flat sustain part. The duration of the attack portion of the FOF envelope is called *TEX* (*temps d'excitation*), the starting time of the decay portion is known as *DEBATT* (*début d'atténuation*) and its duration is named *ATTEN* (*atténuation*). In addition to these parameters the signal produced by a FOF is usually specified by four main parameters: fundamental frequency, amplitude, central frequency and bandwidth (the latter two describing the formant). Usually FOFs are triggered periodically in order to produce harmonic sounds with a more or less pronounced formant. In CHANT, sets of parallel synchronous FOFs are used to synthesize sounds with several formants.

Because of its time-domain nature, the FOF technique is sometimes referred to as a granular synthesis technique. This characteristic led to the definition of the FOG generator, which replaces the sinusoid used in the FOF by an arbitrary sound sample. Consequently, the center frequency parameter of the FOF is replaced by a transposition factor for the sample. The FOG generator accepts yet another parameter, which specifies a begin time for the reading in the sound sample. Since the FOG generator can have several outputs, the individual amplitudes of each output can be specified in form of a list of values.

In order to liberate the user from building multi-voice synthesis patches the standard way using the Max/FTS voice allocation object **loco** and voice banks, GiST's FOF and FOG objects can handle several overlapping grains (voices) at the time. An internal scheduling mechanism takes care about dispatching the computing resources. As a consequence of this the user only needs to specify a maximum number of simultaneous grains desired, which can currently reach up to 17 per ISPW processor for a mono FOG at a sampling rate of 44.1 kHz (each ISPW card has 2 processors, a workstation can have up to 3 cards). Requests to produce more than this maximum number of grains at the time are not taken into

account and a warning is signalled.

The FOG can be used to produce the same output than the FOF if a sine wave is used as sound sample. The center frequency of the formant is then equal to the frequency of the sine wave times the transposition factor. Example 3 shows such a case. A periodic signal with a fundamental frequency of 50 Hz and with a 100 Hz wide formant centered around 500 Hz is produced.

Example 3

Using other samples than the sine wave in example 3 will produce more complex output signals whose nature may sometimes be hard to predict especially if the sample is already a complex signal. Simpler cases are easier to predict, such as a mixture of 3 sinusoids for example, which will produce a harmonic signal with three formants with center frequencies corresponding to the frequencies of the sinusoids.

3.3 Controlling the FOG

The initial motivation of merging traditional granular synthesis approaches with CHANT's formant-wave-function technique led to the definition of the FOG generator. In its hybrid form, the FOG combines the characteristics of the FOF and a resampling generator. Thus it can be used for CHANT-type formantic synthesis, normal sampling applications, standard granular synthesis, and sound granulation approaches. Possible articulations between these techniques, as they are searched for in this project, will find their expression through the particular parametrization of the FOG generator discussed below.

Rarely found in real-time sound granulation systems is the possibility to cleanly transpose the input sample (i.e. by using interpolation techniques other than the linear one to obtain good resampling quality). Transposition is especially useful when creating stochastic clouds of grains, in which case slight detuning of the same grain may considerably enrich the result. Another particularity, the way to specify the amplitude envelope by the

FOF-type parameters `TEX`, `DEBATT`, and `ATTEN` provides for a wide range of possible types of envelopes. Furthermore, the bandwidth parameter allows to superimpose an exponential decay on the amplitude envelope, which permits an easy simulation of resonance-like phenomena with non-resonant samples.

Besides the control of the grain parameters, which can change from one grain to the other, refined control over the temporal structure of the trigger messages is essential for our project. Using T3 messages allows to explore the range between periodic and aperiodic triggering. The development and experimentation with control patches realizing the various ways of passing from more periodic to more stochastic triggering are under way at the moment and we plan to show first results at the conference.

Besides the objects introduced so far (`tsig~`, `tphaser~`, and `fog~`) the GiST comprises other specialized T3-type control objects needed for the construction of control patches. These are `tmetro`, `tdelay`, and `ttimer`, the T3-type counterparts of the standards Max/FTS objects `metro`, `delay`, and `timer`. Furthermore GiST contains some more experimental objects which are developed along with current experimentation on the control level (i.e. in close collaboration with the composer Manuel Rocha Iturbide, who is currently exploring the possibilities of the GiST). There is for example the `tlinenv~` object which can be used to produce envelopes defined by linearly interpolated break-points with T3 precision.

Conclusion

The development of the granular synthesis toolkit GiST was motivated by the unique capacity of granular synthesis to allow for a unified control over the temporal and spectral organisation of sound. By applying a set of design guidelines stemming from empirical and theoretical investigations concerning the use of computer tools in music composition we developed a transparent toolkit for high-quality real-time granular synthesis applications on the ISPW. Our development approach, which favored an evolutionary participative software design strategy, was carried out in the context of IRCAM's *compositeur en recherche* facility.

References

Roads, C., "Asynchronous Granular Synthesis of Sound." In: G. De Poli, A. Picialli, and C. Roads, eds. *Representations of Musical Signals*. Cambridge: The MIT Press, 1991.

Stockhausen, K., "Wie die Zeit vergeht ..." In: Herbert Eimert, ed. *die Reihe*, 3, Vienna: Univeral Edition, 1957.

Xenakis, I., *Formalized Music*, Bloomington: Indiana University Press, 1971.

Gabor, D., "Acoustical quanta and the theory of hearing." *Nature* 159 (4044), 1947.

Risset, J.-C., "Timbre et synthèse des sons." In: J.B. Barrière, ed. *Le timbre, métaphore pour la composition*. Paris: Christian Bourgois Éditeur / IRCAM, 1991.

Rodet, X., "Time-Domain Formant-Wave-Function Synthesis." *Computer Music Journal* 8(3):9-14, 1984.

Clarke, M., "FOF Synthesis on the Atari ST." *Composers Desktop Project Conference at Keele University, England*. York, England: Composers Desktop Project, 1988.

Clarke, M., "VOCEL. New implementations of the FOF synthesis method." In: Ch. Lischka and J. Fritsch, eds. *Proceedings of the 1988 International Computer Music Conference*. San Francisco: International Computer Music Association, 1988.

Becker, B. & Eckel, G., "The Use of Technology in Contemporary Music." In: *Proceedings of the 5th International Symposium on Electronic Art, Helsinki*: http://www.uiah.fi/bookshop/isea_proc/nextgen (Internet), 1994.

Puckette, M., "Combining Event and Signal Processing in the Max Graphical Programming Environment." *Computer Music Journal* 15(3):68-77, 1991.

Lindemann, E., Starkier, F. & Dechelle, F., "The IRCAM Musical Workstation: Hardware Overview and Signal Processing Features." In: S. Arnold and G. Hair, eds. *Proceedings of the 1990 International Computer Music Conference*. San Francisco: International Computer Music Association, 1990.

Essl, K., "Lexikon-Sonate. An Interactive Real-time Composition for Computer-Controlled Piano." *Proceedings of the 2nd Brazilian Symposium on Computer Music*, Canela, 1995.

Eckel, G. & González-Arroyo, R., "Musically Salient Control Abstractions for Sound Synthesis." In: S. Brandorff, ed. *Proceedings of the 1994 International Computer Music Conference*. San Francisco: International Computer Music Association, 1994.

Truax, B., "Discovering Inner Complexity: Time-Shifting and Transposition with a Real-time Granulation Technique." *Computer Music Journal* 18(2), 1994.

Lippe, C., "A Musical Application of Real-time Granular Sampling Using the IRCAM Signal Processing Workstation." In: T. Taguti, ed. *Proceedings of the 1994 International Computer Music Conference*. San Francisco: International Computer Music Association, 1994.