Barry Truax
Department of Communication and Centre for the Arts
Simon Fraser University
Burnaby, British Columbia, Canada V5A 1S6

Real-Time Granular Synthesis with a Digital Signal Processor

Introduction

Following earlier models by Gabor (1947) and Xenakis (1971), Curtis Roads (1978, 1985) has proposed granular synthesis as a unique method of achieving complex sounds by the generation of high densities of small *grains* on the order of magnitude of 10-20 msec duration. He has also used the technique in his composition *prototype* (1975) and as a component of several other tape pieces. The complexity of calculation involved has in the past necessitated a non-real-time approach involving a general-purpose computer music system. As a result, few composers have worked with the technique or heard the range of sounds it produces.

Current digital signal processing (DSP) hardware offers the potential for real-time implementation of this technique by dividing the responsibility for calculation between the DSP and various levels of controllers. This type of implementation can be regarded as an instance of real-time composition, and therefore it is suggestive of a trend towards systems that combine the complexity associated with studio composition with the spontaneity of live performance.

This article describes a real-time implementation of granular synthesis and signal processing, and its use in my work *Riverrun* (1986). This piece has recently appeared on compact disk (Truax 1987b) and is probably the first to be realized entirely with real-time granular synthesis.

A Granular Synthesis Implementation

Two problems that must be solved for the effective use of granular synthesis are generating the large amount of data required to specify the sound, since typically 1000–2000 grains/second can be involved, and designing the control variables required to give the musician a powerful means to link the lower-level data to macro-level compositional strategies and gestures.

Although powerful score editors or algorithmic compositional programs can satisfy the latter need, the resultant score files are usually so large that they are impractical to handle. On the other hand, real-time grain generation does not appear to be possible on most digital synthesizers, particularly those that are controlled by the relatively low bandwidth (Musical Instrument Digital Interface)(MIDI) signals.

The real-time implementation of granular synthesis described here uses the microprogrammable DMX-1000 Digital Signal Processor (Wallraff 1979), one of the earliest DSPs. Although this implementation involves the DMX controlled by a Digital Equipment Corporation PDP Micro 11 (LSI-11/23+), itself a relatively expensive installation, I believe that the program architecture could be transferred to one of the newer and less expensive DSP boards controlled by a microprocessor. However, to achieve sufficient control-level bandwidth, it will probably be necessary to use two control levels, such as an onboard 68000-based controller to schedule the grains as events and a standard microprocessor to respond to user-initiated gestures.

The GSX and GSAMX Programs

The real-time programs, GSX and GSAMX, currently implement three instruments for granular synthesis, each of which provides a different model of unit grain as the basis of the synthesis (Truax, 1986, 1987a). These three models are:

- A simple oscillator with specifiable frequency, waveform, and duration
- A simple frequency modulation oscillator pair with specifiable *c:m* ratio, carrier frequency, duration, and maximum modulation index

Computer Music Journal, Vol. 12, No. 2, Summer 1988,

Sampled sound with specifiable duration and offset time

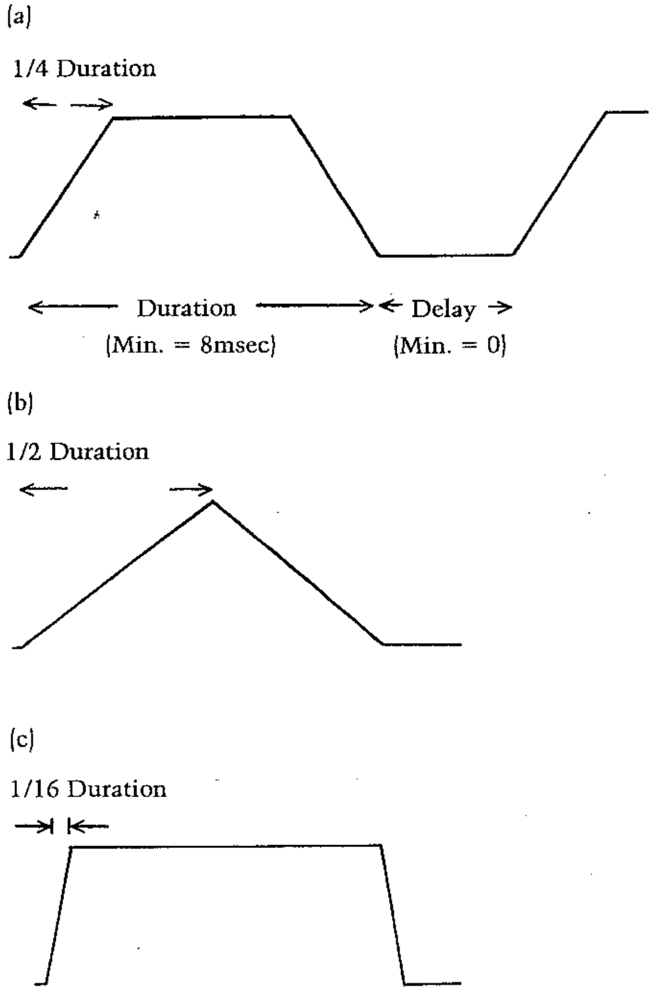
We refer to these instruments as the additive synthesis (AS) model, the frequency modulation (FM) model (Chowning 1973) and the sampling (SAM) model respectively.

In all synthesis models, each grain has a three-part linear envelope whose attack and decay portions both default to one-quarter of the grain duration (Fig. 1). This envelope is a simplified version of the one proposed by Roads, which has a Gaussian-shaped attack and decay. Other proportions of attack and decay from 1/2 to 1/16 the grain duration are specifiable. In the FM case, the same envelope controls the amplitude of the carrier and modulating frequencies. The symmetry of the envelope makes granular sound textures palindromic; that is, sound generated with the AS and FM models can be played backwards on tape with no timbral change.

The synthesis instrument is a bank of envelope generators controlling the basis synthesis unit, and each generator can be thought of as a "voice," even if its output cannot be perceived as a separate string of events. The number of voices determines the maximum vertical density of sound. Twenty simultaneous voices of the additive synthesis model, eight voices of the FM model, and twenty of the sampled sound model are possible with the DMX-1000 as a result of the number of instructions required for the calculation. Half of these voices are assigned to the left output channel and the other half to the right; therefore, all output from the instrument is in stereo. Given the complexity of the granular sound textures and the possibility of cancellation, the output is much richer when synthesized in this way.

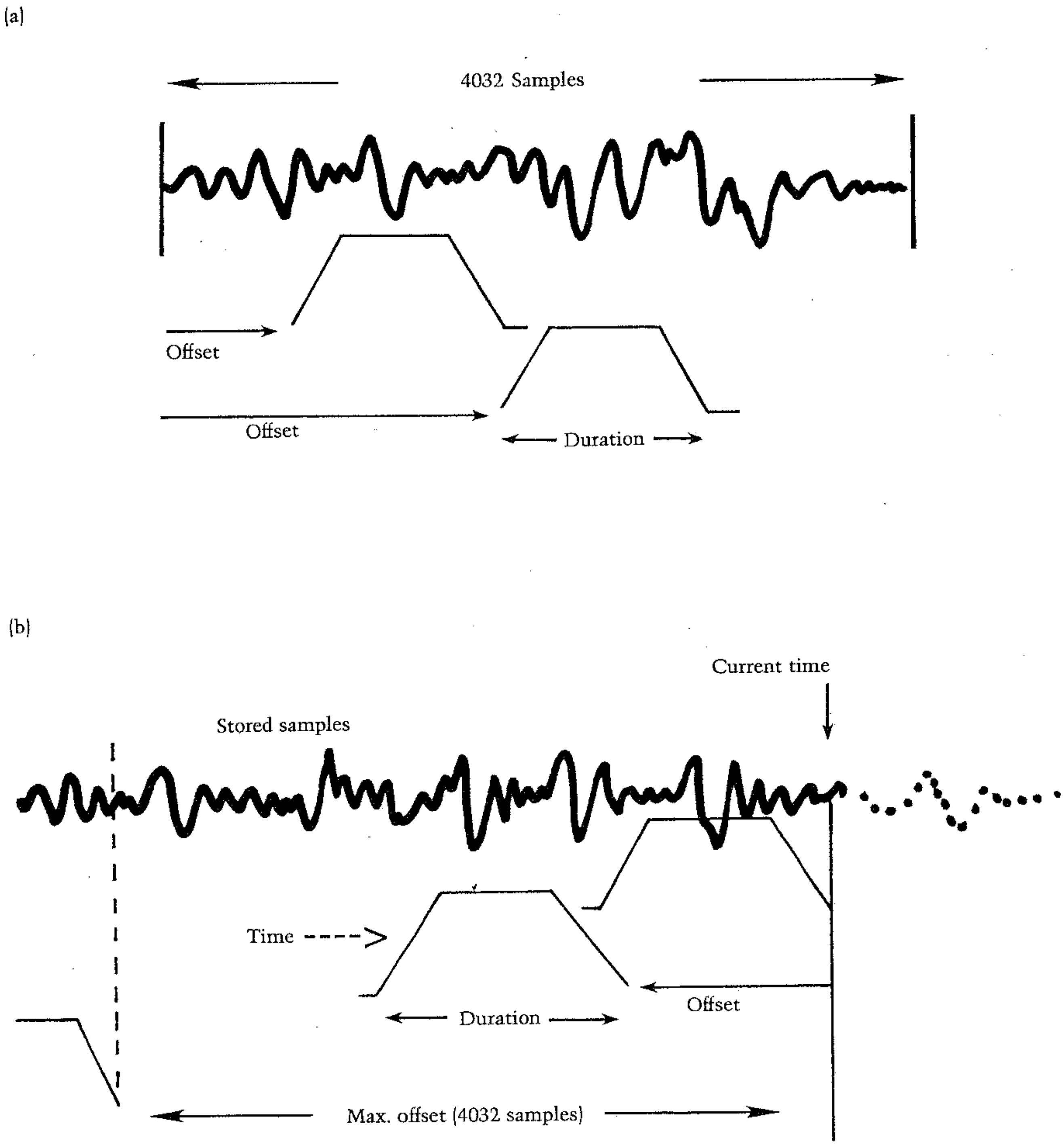
The synthesis instrument is controlled by a scheduler program on the host Micro 11 (Truax 1984). The scheduler initiates each grain by setting the attack ramp and terminates it by setting the decay ramp under clock interrupts every 1 msec. This "foreground" program level also imposes a variable delay time (in milliseconds) until the next grain is started, with a minimum delay time of zero. The shorter the grain, the higher the overall density of *grains per second* (*gps*). The minimum grain duration that can be effectively controlled in real-time by the Micro 11 is 8 msec or 125 gps per voice; hence sound densities can range up to 2500 gps for the 20-voice AS and SAM models, and up to 1000 gps for the 8-voice FM version.

Fig. 1. Grain envelopes. (a) Attack and decay is 1/4th of grain duration. (b) Attack and decay is 1/2 of grain duration. (c) Attack and decay is 1/16th of grain duration.



With the sampling model, two versions exist for *granulating* sampled sound. In the first, a fixed and rather short source sample is used, namely 4032 samples or around 150–170 msec of sound, because of the limitation of 4 Kwords of onboard memory in the DMX (Fig. 2). The duration of grains used in granular synthesis is typically less than this limit so the effect of the fixed sample size is to limit the variety of simultaneous "windows" that can be accessed from the sound material. The second version involves real-time granulation of continuous sound

Fig. 2. Granulating sampled sounds. (a) Granulating a fixed (stored) sample segment in the memory of the DMX-1000. (b) Granulating a continuous stream of input samples.



with the 4 Kword memory acting as a short delay-line or time window that is tapped to furnish the various grains.

Because each grain has an attack and decay, there is no possibility of clicks or transients produced by starting at an arbitrary point in the sound sample. Moreover, when the grains are unsynchronized (i.e., of variable duration) or when each grain starts at a different point within the sound sample, very complex textures can result from even a very simple sound source.

Control Variables

Four *control variables* available to the user determine how successive grain parameters are calculated:

1. Center frequency and frequency range (AS and FM only)
2. Offset number of samples from the start and offset range (SAM only)
3. Average grain duration and duration range
4. Delay time between grains

In certain cases it is desirable to have a variable delay time with a fixed grain duration (for instance, when the delay time is very long and one is working with discrete rhythmic events). The user can switch from variable duration to variable delay time during sound synthesis. Similarly, the user can change from a continuous frequency band to a fixed or harmonic spacing of frequencies. This spacing is equal to the center frequency for each voice, with specific frequencies chosen from within the current frequency range.

In addition to the basic control variables, each model has the following additional variables that are specific to it:

AS:
  Number of voices with each of three waveforms
  Total number of voices sounding (maximum = 20)
FM:
  Average modulation index and index range
  Total number of voices sounding (maximum = 8)
SAM:
  Speed of output, which acts as a pitch/time transposition
  Number of voices sounding at transposed sample rates
  Total number of voices sounding (maximum = 20)

The background level of the scheduler has two functions: to service control requests from the user and to calculate new random values for the parameters when the range of their variation is nonzero. The "randomizable" parameters are frequency, grain duration or delay time, offset number of samples (in the case of sampled sound) and modulation index (in the FM case). For instance, if the center frequency is 300 Hz and the frequency range is 20 Hz, grains are calculated with frequencies randomly chosen from the range of 290–310 Hz, distributed uniformly. Any calculated frequency that is negative is changed into the corresponding positive frequency, and grain durations less than 8 msec are adjusted to this minimum value. Recalculation of new random events occurs constantly whenever a foreground activity or user controls do not demand the processor's attention. Therefore, all sounds have some degree of granular texture because of constant random variation in the grain specification—hence the lively and sometimes natural-sounding result.

No variation in grain duration (i.e., duration range equals zero) produces an amplitude-modulated signal, whereas even a small range of variation results in a stochastic texture. As predicted by Roads (1985), amplitude modulation (AM) results because each grain is immediately followed by another of the same duration; therefore the grain duration becomes the period of the modulating wave and the grain envelope its waveform. For instance, the minimum duration of 8 msec is identical to a modulating wave of 125 Hz. Sidebands are thus produced around the center frequency, and if that frequency is harmonically related to 125 Hz, an enriched harmonic spectrum results.

Even in the case of no variation in grain duration and frequency, the resulting sound is still not steady because each of the voices in the instrument cannot be exactly synchronized at the micro level.

That is, by the time the last voice is initiated, even during the same clock interrupt, the others will have already started. They will therefore be in a different phase relation to the current voice, thus producing a variable output amplitude because of cancellation and reinforcement.

The delay time between grains can also be used in conjunction with the AM effect since it changes the overall periodicity of the resulting sound. The minimum delay time is 0, and as it increases to a significant fraction of the grain duration, various modulation effects are heard. Longer delay times result in a lessening of the sense of texture (since fewer grains/second are being heard), until with very long delay times, the grain can be heard as a separate event.

In all synthesis models, voices can be turned on and off, thereby allowing changes in vertical density. In the AS model, the number of voices sounding with each of three waveforms can also be controlled. In the sampled sound version, two other control variables are available. The first is a simple speed control that causes the synthesizer to run more slowly, similar to a variable speed tape recorder. With the fixed sample version (Fig. 2), a second option allows a certain number of voices in the instrument to output samples two times faster than normal (by skipping alternate samples) or two times slower than normal (by repeating every sample). Duration of the grain is not affected by these different sample rates unless the end of the sample is reached. As a result, part of the sound texture can sound an octave above or below the rest of the material. With the continuous sampling version, this option is replaced by a feedback control to recirculate samples through the delay line.

Psychoacoustic Variables

Each of the control variables cited previously have a psychoacoustic correlate that may be more suggestive as a basis for compositional organization than the numerical values of each variable (Table 1).

These parameters result in three psychoacoustic axes, the first of which is the familiar pitch-noise continuum. A narrow frequency range allows the auditory system to ascribe a pitch to the resultant sound, whereas when the frequency range exceeds about 10% of the center frequency, pitch sensation gives way to an impression of narrowband noise and, with larger values, to broadband noise.

Secondly, the average duration of the grains, assuming low values of the duration range and delay parameters, determines the overall density of events. If the grain duration is much less than the threshold of 50 msec, the result is a fusion of the grains into a continuous texture. Grains with durations much longer than 50 msec tend to be perceived as separate events, depending on whether they occur over a wide or narrow frequency range. Around 50 msec there is a continuum between audio rate fusion and discrete events—a kind of "pulling apart" of the component grains.

The third axis is the modulation phenomenon described previously as occurring when the grains are synchronized (i.e., when they occur with little variation in duration). The frequency of the modulation is the inverse of the grain duration plus delay time (e.g., a 15 msec grain duration plus 5 msec delay produces a 50-Hz modulation). The modulation phenomenon is particularly strong when the center frequency of the grains corresponds to the modulation frequency or one of its harmonics. The waveform of the modulation is determined by the shape of the grain envelope with more sharply defined envelopes producing a greater number of sidebands. When the grain consists of an FM event, additional relationships are possible between the *c:m* ratio of the FM event and the sidebands of the AM effect (Truax 1977b).

The delay time, as noted above, can contribute to the density parameter. It lessens the overall density as its value increases until eventually discrete events can be observed, or else it can affect the modulation phenomenon, at least for small values. In the current implementation, durations and delay times are quantized to millisecond units; hence all possible modulation frequencies are subharmonics of 1 KHz. For instance, an 8-msec grain results in a modulating frequency of 125 Hz. A small change in the delay parameter changes this frequency to other submultiples of 1 KHz. A finer time resolution is required to achieve other AM frequencies.

Table 1. Granular synthesis control variables and their psychoacoustic correlates

Control Variable	*Psychoacoustic Correlate*
Center frequency	Average pitch or pitch range
Frequency range	Bandwidth (pitch → noise continuum)
Average duration	Density (audio rate fusion → discrete events)
Duration range	Modulation (periodic AM → random modulation)
Delay	Secondary density and modulation control

With the sampled sound model, granulation produces the effects of density and modulation as it does with the fixed waveform and FM models. However, instead of bandwidth and pitch, there is a relationship between the duration of the grain and the acoustic characteristics of the source sound. For instance, the perception of pitch and timbre depends on the duration of a sound. With isolated sounds, the onset of pitch is around 13 msec (Olson 1967) and timbre emerges fully after 40–50 msec. The duration of very short grains may in fact be less than a period of the sampled sound and produce a broadband result! The 50 msec threshold is a rough dividing line between the possible psychoacoustic effects. Around and above that threshold, the sampled sound's timbral qualities dominate, and below it, audio rate fusion of the sampled fragments dominates.

Compositional Control Strategies

Given the enormous amount of data involved in specifying thousands of events per second, powerful control strategies are required to make this synthesis technique effective for the composer. The current implementation has developed a hierarchy of control levels (Fig. 3). At the lowest level are the control parameters already described, which can be altered by various keystroke commands. Groups of these control parameters are called *presets* and these in turn can be referenced by a higher level score.

A second control route is via *ramps*, that is, patterns of change in the parameters at a specific rate. These ramps can also be predetermined and stored in a ramp file. Ramps can be combined with presets in any manner.

The third compositional strategy involves *tendency masks*. Although they are translated into the equivalent of a combination of presets and ramps, they appear as graphic control shapes to the composer and hence suggest a different compositional approach.

In general, all of these compositional strategies are compatible with each other and can be implemented in any combination. The reason behind this flexibility is a type of programming strategy that may be termed parallel processing. All of the various program levels operate simultaneously (via interrupt programming in addition to the external DMX-1000 program) and have access to a set of data that represents the current state of the synthesis process. When grains are to be initiated, their values are drawn from this constantly changing description. New values assigned to the control variables by the user are written into the current data representation. At all other times, a background operation uses the control variables as a basis to calculate new values for the grain parameters (whether they are ever executed or not). Tests on the DMX implementation show that with a density of about 700 gps (grain duration = 15 msec), there are about as many new grains calculated as are executed; at higher densities grains may be repeated, but there is no audible effect of this redundancy.

This type of parallel processing suggests possible directions for real-time composing, similar to work by Chadabe (1984) and others. The key is to abandon linear modes of compositional thinking, which result in deterministic output (e.g., score or sequencer driven), and to substitute process-oriented

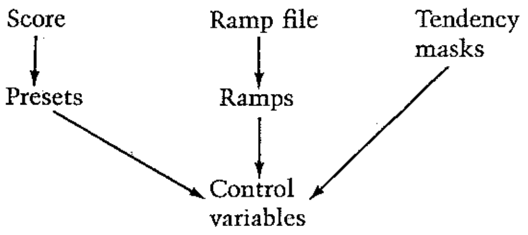


Fig. 3. Hierarchy of compositional controls for real-time granular synthesis.

multitask strategies for real-time execution. To harness the ultimate power of this approach, significant knowledge bases (Laske 1988) will be needed to provide intelligent means for organizing the complexity of the output. Strategies drawn from automated compositional approaches (e.g., Koenig, Hiller, et al.) should be reevaluated in terms of their applicability to real-time implementation. The power of current DSP chips and host processors to organize complex control systems will enable and should encourage this direction to be followed.

The current implementation, while just a beginning, has already gone some distance in establishing an appropriate hierarchy of levels of compositional control for granular synthesis, ranging from the control variables at the grain level, through to groups of such variables (presets), rates of change (ramps) of the control variables, and macro-level tendency masks and scores to determine large-scale forms. The following is a brief summary of the general characteristics of this control hierarchy.

Real-Time User Controls

Several modes of real-time control are available to the musician during synthesis:

A new value can be typed in for any parameter.
A single parameter can be changed by a specified increment.
A group of "synchronized" variables can be changed by a specified increment where the synchronization can represent a direct or inverse variation of each variable.
All parameters or only the "synchronized" parameters can be reset to any of a group of stored (preset) values.
Ascending, descending, or random ramps can be initiated on all of the "synchronized" variables according to a time or ramp value that represents the rate at which a specified increment is to be added to or subtracted from each variable. The ramp value itself can be one of the synchronized variables, thus allowing acceleration and deceleration of the ramp. Ramps can also be scaled to proceed at different rates, ranging between factors of 1 through 10.
Overall amplitude level can be set, and a global attack and decay initiated at a predetermined rate.

It should be noted that all these control possibilities can be activated by single alphanumeric keyboard strokes during the synthesis. All controls are compatible with each other and therefore can be executed in conjunction with them. For instance, individual parameter changes and resets can be executed during ramps; likewise, presets can be stored during a ramp. Similarly, variables can be synchronized, removed from synchronization, and the nature of their correlation (direct, inverse) switched during ramps. At any point, the user can type in a new value for any parameter (including the ramp time).

A line of control parameters is displayed on the CRT as shown in Table 2.

In the example in Table 2, INC is an increment value to be added to or subtracted from any variable; FREQ is the center frequency (in Hz); FRQ. RNG is the frequency range around the center frequency; DUR'N is the average duration (in msec); DUR.RNG is the range of durations; DELAY is the delay time between grains (in msec); RAMP is the time (in msec) before the INCrement value is added to a variable; NO.VOI.W.F.#2 is the number of voices in the synthesis instrument with the second waveform loaded; NO.VOI.W.F.#3 is the same for the third waveform; and TOTAL NO.VOI. is the number of voices sounding in the instrument. The AMP indication is the global amplitude value, which only appears when requested.

In the FM version, the waveform variables are replaced by average modulation index and modulation index range. These variables can be changed and ramped like any others except that the incre-

Table 2. Line of control variables

AMP	*INC*	*FREQ*	*FRQ.RNG*	*DUR'N*	*DUR.RNG*	*DELAY*	*RAMP*	*NO.VOI. W.F.#2*	*NO.VOI. W.F.#3*	*TOTAL NO.VOI.*
	1	100	20	20	10	1	1000	0	0	20

ment is limited to 1 (because of the relatively large size of the modulation index as a parameter). The total number of voices sounding can be changed incrementally, including their reduction to zero. Since there are 20 voices in the AS version, all those not assigned to waveforms 2 and 3 are given waveform 1 by default. In the FM version, the carrier and modulating waveforms as well as the *c:m* ratio are specified before synthesis begins. However, the user can change the *c:m* ratio during synthesis.

Ramps

Variables that are singled out for "synchronization" are indicated with a + or − sign appearing before the number. The former indicates that with the up arrow, ascending or random ramps, the INC value will be added to the parameter. The latter indicates that the INC value will be subtracted from the parameter, provided it does not go below permissible limits. With the descending ramp or down arrow, + variables are decremented, and − variables incremented. In general, these signs indicate direct or inverse variation during a parameter change, and they can be used in any combination. In the case of the random ramp, only a fraction of the INC value (from 0 to INC-1) will be added or subtracted according to the + or − sign. Because the synchronized parameters can change by imperceptibly small amounts (or not at all) with the random ramp, its use generally produces smoother changes than the other types of ramp.

Ramp increments can be scaled at rates from 1 to 10. For instance, if the INC value is 2 and the rate is 5, the parameter is incremented by 10 with + synchronization or decremented by 10 (assuming the result is permissible) with the − synchronization. This option is particularly useful with frequency parameters, since they often need larger amounts of change than do the duration and delay values. The ramp rate (if not equal to 1) is printed immediately to the right of the number. A typical control line might appear as in Table 3.

The specification shown in Table 3 means that the FREQuency and DURation RaNGe values are incremented with an up arrow or ascending ramp, and the RAMP value is decremented by the same action. However, the rate of change of the frequency value is five times greater than that of the ramp value and two times greater than that of the duration range. Therefore, after a single positive increment, the new frequency value is 105, the duration range 11, and the ramp value 998. Since the INC value can also be changed during an ascending, descending, or random ramp, quite complex changes can be easily implemented.

Any ramp (ascending, descending, random) can be made *cyclic*. This means that when the parameter reaches the minimum value, the direction of the synchronization (+,−) is reversed to the opposite mode; likewise when the maximum value is reached the synchronization is reversed again. Should the current parameter value be outside those limits, a reversal of synchronization brings it back within them.

Envelope Shape

As shown in Fig. 1, the envelope shape is controlled by a single number (ranging from 2 to 16), which indicates the fraction of the total grain duration devoted to the attack and decay portions. In terms of amplitude modulation, this fraction controls the waveshape of the modulation, going from a triangle

Table 3. Line of synchronized control variables

AMP	*INC*	*FREQ*	*FRQ.RNG*	*DUR'N*	*DUR.RNG*	*DELAY*	*RAMP*	*NO.VOI. W.F.#2*	*NO.VOI. W.F.#3*	*TOTAL NO.VOI.*
	1	+100 5	20	20	+10	1	−1000 2	0	0	20

wave (attack = 1/2 duration) to approximately a rectangular wave (attack = 1/16 duration). Hence it is an important control variable. With sampled sound as a source, the envelope prevents transients from occurring when an arbitrary group of samples is chosen. The envelope of each grain can be thought of as a fade-in and fade-out; therefore, changing the envelope shape determines the smoothness or abruptness of this effect.

Presets and Objects

Keyboard characters that are not reserved for special meanings are used to indicate a set of current variables and to store or retrieve them as a preset. With the AS model, these presets include the six variables (frequency, frequency range, duration, duration range, delay, and ramp value). The sampled sound version substitutes offset and offset range for frequency. With FM, two additional variables, namely the modulation index and index range, are added for a total of eight values.

Storage of presets can occur during a ramp when values are caught "on the fly;" they can later be retrieved and edited once the ramp is stopped. The user can store such presets in a diskfile and retrieve them with no interruption in synthesis.

Each preset can be thought of as a *sound object*, similar to the timbral definition of a sound object elsewhere in the author's POD and PODX system (Truax 1977a, 1985). A score editor has been created which can combine up to 160 presets into a single file, along with data about how these objects are to be scored in time. The score parameters are:

Entry delay (in centiseconds)
Object number
Maximum amplitude
Optional duration (in milliseconds)
Optional frequency (in hertz)

The duration and frequency values are optional because they can override the corresponding duration and center frequency values in the object. Therefore each score can be performed in four different ways—with or without replaced duration or frequency values. In addition, each score can be performed with different waveforms, different envelope shapes for the grains, and in the FM case, different *c:m* ratios. During the performance of such a score, ramps can also be used at the same time, either manually activated or from a ramp file. The rule is that any variable singled out for synchronization is no longer controlled by the score, but rather by the user's commands (e.g., typed in values, ramps, etc.). Therefore, the score replaces the live performance aspect of playing the instrument by calling back presets, but it still allows considerable real-time modification during synthesis.

Ramp Files

To facilitate composition with ramps, an editor allows ramps to be specified, stored in a file, recalled and implemented during synthesis. Up to 18 sequential ramps can be specified in this way. Each ramp includes:

Ramp type (ascending, descending, random)
Ramp scaling (1–10) for each synchronized parameter
Synchronization (+, −, or none)
Cycle switch
End condition (elapsed time or specific parameter value, whichever comes first)

When both end conditions are specified, the one reached first results in the next ramp (if any) being implemented. The user can also store a list of the minimum and maximum values for each parameter during a cyclic ramp. These values can be edited and put into effect whenever a cyclic ramp is specified.

Compositional Control with Tendency Masks (GRMSKX)

Most of the control variables that have already been described are based on the average value of a variable and the range around it from which a random choice is to be made. These two variables are exactly those that describe a tendency mask. However, a tendency mask is expressed as an area within which values can be chosen. It is one of the simplest ways to specify time-dependent selection of a variable. The width of the mask at any point determines the range of choices available, thereby providing a continuum between deterministic and stochastic choices. The average value, although an imaginary line through the middle of the mask, is usually perceptible, particularly when it changes over time.

Although the control offered by tendency masks and the average-plus-range method is very similar, there is a striking difference between how the composer formulates the desired effect when using each method. With presets or ramps controlling average and range values, one is more aware of these numerical values and less aware of the minimum and maximum values they allow. The tendency mask, being inherently a visual control method, presents a visual image of the control shape based on the limiting values within which choices are made. The tendency mask suggests gestures, whereas the pair of changing numerical parameters suggests ongoing processes.

The GRMSKX program allows the user to formulate compositional strategies for each of the control variables as a set of tendency masks (when average and range variables are available), or as an envelope (when a single variable is involved). A graphic overlay of the masks and envelopes shows their synchronization (Fig. 4). With the three synthesis models, this set of controls includes:

Tendency Masks:
  Frequency (AS and FM) or offset (SAM)
  Duration
  Modulation index (FM only)
Envelopes:
  Amplitude
  Delay time

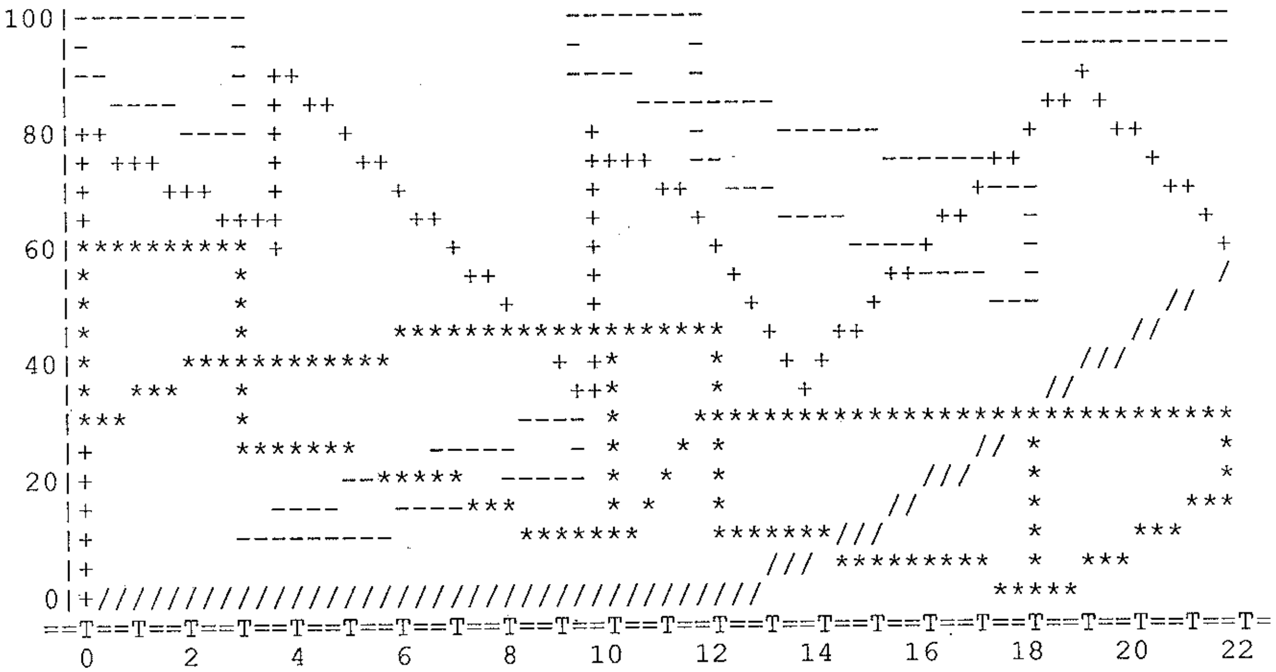
Up to three tendency masks and two envelopes (with a maximum of 10 segments each) can be specified and stored in a file. Alternatively, the masks can be generated as interpolations between presets; that is, each segment of a set of masks and the delay-time envelope can be derived as starting with the values of a certain preset and ending with another preset. The difference between the two methods is that with interpolation, all segments of each mask and envelope have the same duration, whereas with the direct specification method, no synchronization between masks is required.

One of the main advantages of the use of tendency masks is that ramps on any parameter are no longer tied to those on another parameter. Although ramps in GSX and GSAMX can proceed at different rates of increase or decrease (1–10), they are synchronized in terms of the speed at which such updates are made. With GRMSKX, such ramps are completely independent. They are calculated with their own rate of increase and the current values are reported to the user via the screen values once every second, even if changes are being made much more quickly. The user can distort the predetermined mask shape by changing the INC value.

The use of amplitude envelopes is a major addition to the control of this variable with granular synthesis textures. The other programs allow only manually activated amplitude ramps, whereas a more complex amplitude curve can be followed with the amplitude envelope in GRMSKX.

Masks can also be executed repetitively, thus providing a great deal of variation within even a simple set of masks when their respective durations are not equal. For instance, a 10-sec pattern of amplitude control can repeat while a 60-sec pattern

Fig. 4. Screen image (on a 24-line terminal) of tendency masks for granular synthesis control. Shown are frequency mask (*) duration mask (−), and envelopes for amplitude (+) and delay time (/).



of frequency change and 20-sec pattern of duration control is occurring.

Musical Applications

The complexity and dynamic quality of granular synthesis sound makes it an attractive alternative to synthesis models based on fixed waveforms and envelopes. Moreover, the basic unit or "quantum" of the grain is a potentially more flexible alternative to the sine wave as a building block for sound synthesis (Gabor 1947). It is also a very flexible means of manipulating sound samples, particularly because the envelope of the grain avoids transient clicks when extracting and combining sample segments. When granular synthesis is used to produce continuous textures, it has no resemblance to instrumental and other note-based music; instead, its sound world is more closely related to analog electroacoustic music, but with greater precision of control. In certain cases, the acoustic result resembles environmental sounds in terms of their inner complexity and statistical texture. However it is used, granular synthesis is clearly situated in a different psychoacoustic domain than that occupied by most computer music; it creates a unique sound world and suggests new approaches to the way music made with it is formed.

The psychoacoustic domain of high density events has recently been described in terms of *streaming* (McAdams and Bregman 1979) where events can be perceived as isolated, grouped into streams, or fused together, depending on their frequency range and temporal density. More recently, John MacKay (1984) has described increasing densities of events as creating "a spectrum of impressions ranging from the simple 'sequence' of tones to that of a 'flow,' a 'swarm,' a fused 'textural band,' and finally a 'massed sonority' depicting the different degree of density-determined solidity and consistency of the texture." In terms of increasing bandwidth, he describes (p. 171) "a spectrum of impressions of stratification ranging from a noise band to bandwidths with very prominent upper and lower edges but no clearly perceivable tonal identities in the middle, to bandwidths with very prominent lower edges and mildly prominent upper edges and some fleetingly identifiable tonal content in the middle of the bandwidth."

In my own work, granular synthesis is an extension of what I have called the *stochastic texture* (Truax 1982, 1984), as realized in *Arras* (1980), where the superimposition of many similar and spectrally related subevents produces a clearly defined and controllable macro-level texture. The presence of any particular frequency component at the micro-level, however, can only be statistically determined. The difference with the present work is that much shorter events and higher densities are generated, such that one passes the audio-rate threshold (around 20 Hz) at the micro-event level. However, the approach to structuring the sound and the music remains the same, namely a hierarchic organization of levels.

All sounds in *Riverrun* (1986) were generated with real-time granular synthesis, up to a maximum density of 2375 gps. However, in many cases, lesser densities were also used since the progression from isolated sounds or a rapid sequence of events to a fused texture is an interesting feature of the synthesis method. All layers forming the work were multi-tracked with four simultaneous stereo versions, and up to four of these eight-track source tapes were later mixed. Considerable use was made of ramps applied to the synthesis variables; that is, certain parameters were made to change over time at a specific rate, sometimes with several parameters simultaneously ramped at different rates. Therefore, all sound in the piece is in a constant state of flux, much like environmental sound generally and water sound in particular, whether through the use of ramps or because of the random variation of the thousands of component grains heard in each sound.

The fundamental paradox of granular synthesis—that the enormously rich and powerful textures it produces result from its being based on the most "trivial" grains of sound—suggested a metaphoric relation to the river whose power is based on the accumulation of countless "powerless" droplets of water. The opening section of the work portrays that accumulation as individual "droplets" of sound multiply gradually into a powerful broadband texture. The dynamic variation found in the use of ramps allows the piece to create a sound environment in which stasis and flux, solidity and movement coexist in a dynamic balance similar to a river, which is always moving yet seemingly permanent. The piece, I find, also captures some of the awe one feels in the presence of the overpowering force of such a body of water, whether in a perturbed or calm state, and as such it seems to create a different mode of listening than does conventional instrumental or electroacoustic music.

Future Directions

Further development of the granular synthesis technique, apart from its implementation on microprocessors, will probably involve other methods of organizing the granular events or using other sound materials as a basis. Other forms of parametric control are also possible. For instance, the current implementation does not include a maximum amplitude parameter for each grain, only a global amplitude control. Another control model might place the grains in different frequency regions, in which case amplitude control for each voice of the instrument would be desirable. The notion of *critical bandwidth* (Petersen and Boll 1983) could be used to construct the frequency range of each voice. Spatial control of the sound may also be desirable (Truax 1983).

Other statistical distributions might prove useful in the use of this technique (Xenakis 1971), and certain analogies might be discovered to recent work in self-organizing or recursive systems (Wolfram 1983; Poundstone 1985). Indeed, mathematical models involving the behavior of large masses of cellular units would seem to be applicable to this technique. As processors become more powerful and can be used in parallel as well as hierarchically, complex control systems implementing many levels of decision making will allow real-time composition to achieve a level of sophistication that we can now only suspect.

References

Chadabe, J. 1984. "Interactive Composing: An Overview." *Computer Music Journal* 8(1): 22–27.

Chowning, J. 1973. "The Synthesis of Complex Audio Spectra By Means of Frequency Modulation." *Journal of the Audio Engineering Society* 21(7):526–534. Reprinted in C. Roads and J. Strawn, eds. 1985. *Foundations of Computer Music*. Cambridge, Massachusetts: MIT Press, pp. 6–29.

Gabor, D. 1947. "Acoustical Quanta and the Theory of Hearing." *Nature* 159(4044):591–594.

Laske, O. 1988. "Introduction to Cognitive Musicology." *Computer Music Journal* 12(1):43–57.

MacKay, J. 1984. "On the Perception of Density and Stratification in Granular Sonic Textures: An Exploratory Study." *Interface* 13:171–186.

McAdams, S., and A. Bregman. 1979. "Hearing Musical Streams." *Computer Music Journal* 3(4):26–41. Reprinted in C. Roads and J. Strawn, eds. 1985. *Foundations of Computer Music*. Cambridge, Massachusetts: MIT Press, pp. 658–698.

Olson, H. 1967. *Music, Physics, and Engineering*. New York: Dover, p. 250.

Petersen, T., and S. Boll. 1983. "Critical Band Analysis-Synthesis." *IEEE Transactions on Acoustics, Speech, and Signal Processing* 656–663.

Poundstone, W. 1985. *The Recursive Universe*. New York: William Morrow.

Roads, C. 1978. "Automated Granular Synthesis of Sound." *Computer Music Journal* 2(2):61–62.

Roads, C. 1985. "Granular Synthesis of Sound." In C. Roads and J. Strawn, eds. 1985. *Foundations of Computer Music*. Cambridge, Massachusetts: MIT Press. pp. 145–159.

Truax, B. 1977a. "The POD System of Interactive Composition Programs." *Computer Music Journal* 1(3):30–39.

Truax, B. 1977b. "Organizational Techniques for C:M Ratios in Frequency Modulation." *Computer Music Journal* 1(4):39–45. Reprinted in C. Roads and J. Strawn, eds. 1985. *Foundations of Computer Music*. Cambridge, Massachusetts: MIT Press, pp. 68–82.

Truax, B. 1982. "Timbral Construction in *Arras* as a Stochastic Process." *Computer Music Journal* 6(3):72–77.

Truax, B. 1983. "The Compositional Organization of Timbre in a Binaural Space." In A. Schindler and A. Brinkman, eds. *Proceedings of the 1983 International Computer Music Conference*. San Francisco: Computer Music Association.

Truax, B. 1984. "Models of Interactive Composition with the DMX-1000 Digital Signal Processor." In W. Buxton, ed. *Proceedings of the 1984 International Computer Music Conference*. San Francisco: Computer Music Association.

Truax, B. 1985. "The PODX System: Interactive Compositional Software for the DMX-1000." *Computer Music Journal* 9(1):29–38.

Truax, B. 1986. "Real-Time Granular Synthesis with the DMX-1000." In P. Berg, ed. *Proceedings of the 1986 International Computer Music Conference*. San Francisco: Computer Music Association.

Truax, B. 1987a. "Real-Time Granulation of Sampled Sound with the DMX-1000." In J. Beauchamp, ed. *Proceedings of the 1987 International Computer Music Conference*. San Francisco: Computer Music Association.

Truax, B. 1987b. *Digital Soundscapes*. Cambridge Street Records, CSR-CD 8701, 4346 Cambridge St., Burnaby, B.C., Canada V5C 1H4. Note: this compact disk includes *Riverrun*.

Wallraff, D. 1979. "The DMX-1000 Signal Processing Computer," *Computer Music Journal* 3(4):44n49. Revised and updated version in C. Roads and J. Strawn, eds. 1985. *Foundations of Computer Music*. Cambridge, Massachusetts: MIT Press, pp. 225–243.

Wolfram, S. 1983. "Statistical Mechanics of Cellular Automata." *Reviews of Modern Physics* 55(3):601–642.

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