Toward a unification of algorithmic composition, real-time software synthesis, and live performance interaction

Brian Belet

Clark University, Music Program, Department of Visual and Performing Arts
950 Main Street, Worcester, Massachusetts 01610-1477, USA
e-mail: bbelet@vax.clarku.edu

ABSTRACT

Using the Kyma digital synthesis system a composer is able to unify and combine the previously disparate concepts and processes of algorithmic composition, software synthesis, real-time generation, and live performance interaction. This is demonstrated by the composition *Discourse (GUTs 2a)*, written by the composer in early 1992 for bass trombone and Kyma system.

Using the Smalltalk-80 program *COMP2* (written within the Kyma system by the composer in 1991) all aspects of the Kyma music are generated algorithmically from a set of composer-defined proportions. This includes large-scale structure down to individual event frequencies, durations, and timbre parameters. The trombone music uses the same proportions to determine its large-scale structure while the event parameters are composed manually and intuitively the "old fashioned" way.

There are areas in the work where the two musics proceed independently, and other areas where they may interact with each other. When interaction is a possibility, the process is dynamic and live with the ability to change the composition on the event and gesture level. The Kyma system can record, process, and playback live trombone gestures in real or delayed time. The trombonist can also alter his/her events (thereby regarding the score in these areas as a template more than a rule) in reaction to or anticipation of Kyma events.

The Kyma system represents a new generation of computer music systems with the combined capabilities of hitherto separate systems. With this system, the composer is working to unify these various processes with the hopes of creating a truly flexible, user-friendly, and powerful composing and performing environment.

1. INTRODUCTION

The Kyma digital synthesis system is a real-time DSP-based computer music system that provides maximum flexibility to the composer while maintaining the desired powerful capabilities of algorithmic composition (using the Smalltalk-80 programming language), software synthesis, and live performance interaction. Music in Kyma is represented as Sound objects (both literally and conceptually), and the composer works with these Sound objects to create both large- and small-scale structures (Scaletti 1987, 1989 [1] & [2]). The user interface is "friendly" enough to permit a composer to customize the environment (i.e., to design his/her own Sound classes) without necessarily having to get inside the Smalltalk programming environment (Scaletti 1991).

2. GUTs

This composer has been working on a series of compositions which carry the collective title **GUTs**, a philosophical reference to the continuing search for Grand Unification Theories in astrophysics. The compositions in the **GUTs** series are experiments designed to unify the various compositional parameters and structural concerns of a given work through the use of a restricted set of user-defined values. Once established, these initial values are used to generate all aspects of the composition. Several compositions have been generated by the programs *COMP1* (1985-89) and *COMP2* (1990 to the present), written by the author (Belet 1991).

3. *Discourse (GUTs 2a)*

The composition *Discourse (GUTs 2a)* was composed in 1992 for bass trombone and Kyma system, and was composed for trombonist J. Scott Mousseau. This work simultaneously explores the separate aural worlds of trombone and computer, and their interactions in live performance (which exist both by design and chance). All aspects of the Kyma music, from large-scale temporal structure to small-scale parameters of event time and frequency, are generated algorithmically using proportional recursive stochastic procedures that utilize a user-defined set of ratio proportions. In this work, the composer used ratios that correspond to twelve primary intervals in just intonation. This unifying set is used to create a high level of conceptual, and hopefully aural, unity throughout the composition.

The twelve proportions are first manipulated to generate two nonsynchronous formal structures for the separate trombone and Kyma musics. Points of alignment between the two time-lines are used to generate a larger structure of independence and interaction between the musics. This macro structure is diagrammed in Figure 1. In areas of independence, the two musics proceed according to their own designs without undo concern for the other environment. During areas of interaction, the live performer may deviate from the written score to react to the Kyma sounds (i.e., the performer may improvise within the established style of the composition) and some of the Kyma procedures operate on and with the live trombone sounds.

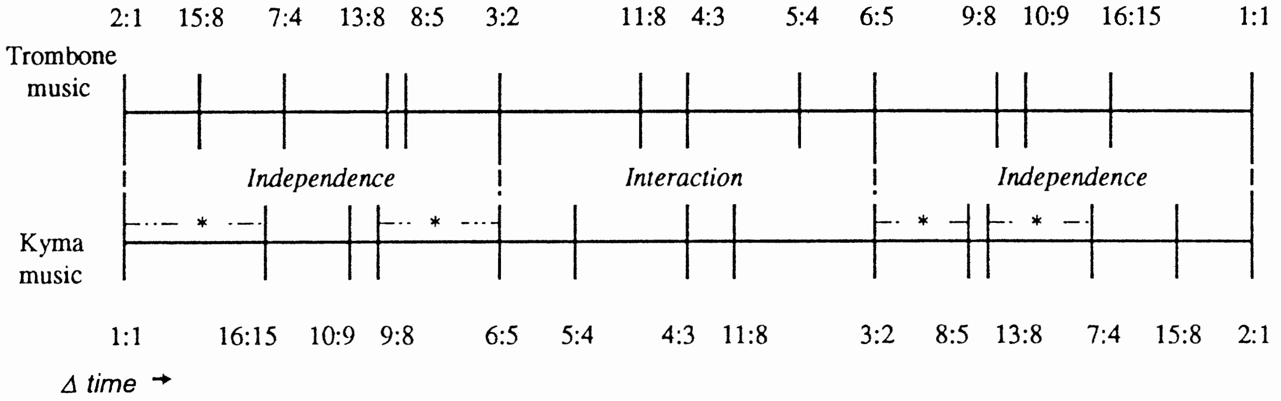

Figure 1. Macro structure for *Discourse* derived from twelve just intonation ratios.

4. COMP2

The overall structure of the Kyma music in *Discourse* is controlled by a summation [Sum] of sections that are temporally displaced [TimeOffset], as diagrammed in Figure 2. This allows earlier sections to overlap successive sections, which could not occur with a strict sequential order [Concatenation].

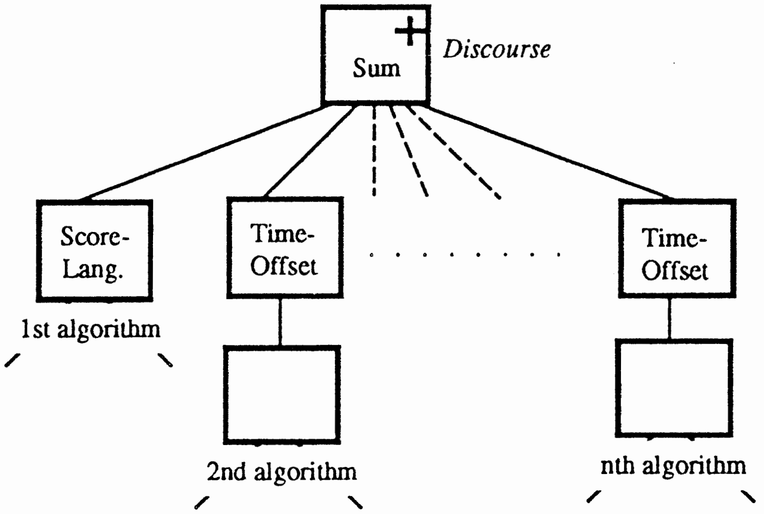

Figure 2. Primary organizational structure for *Discourse*. Each lower Sound is a subSound of the Sound directly above it.

The first algorithm generates events for four separate sections which share the same style or "character" (these sections are identified as "lyrical" and exist within the Independent areas; these sections are marked with "|-- * --|" on Figure 1 above). As a result, the number of sections in the composition do not exactly correspond to the number of algorithms used.

Figure 3 shows a detail of the first algorithm Sound structure from Figure 2. This Sound uses a ScoreLanguage Sound as the top level structure, with subSounds that create AM and FM timbres.

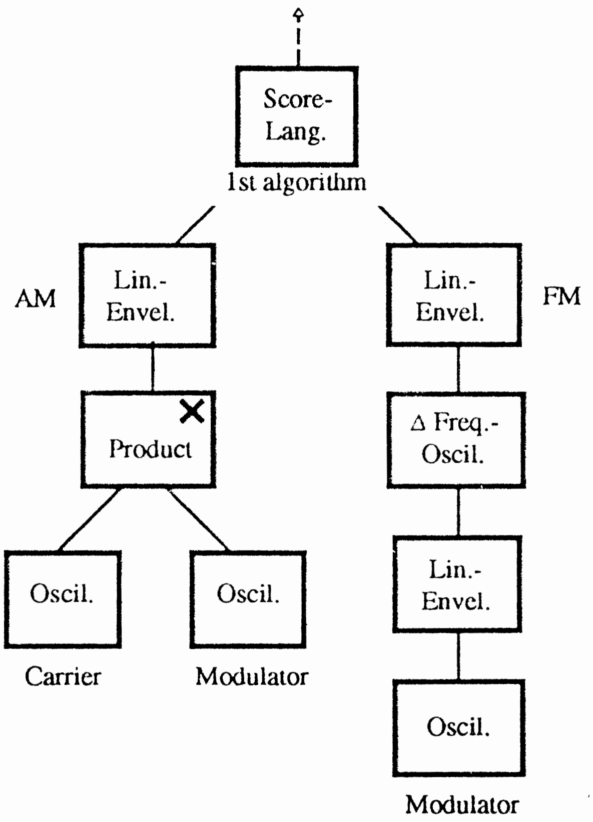

Figure 3. ScoreLanguage Sound structure used to generate four Independent "lyrical" sections in *Discourse*.

Using a random number generator that is focused by the twelve initial ratio proportions, *COMP2* generates the necessary parameter values when the program is compiled [Download]. For these "lyrical" areas the part of *COMP2* within the ScoreLanguage Sound first determines whether or not an event will take place; if an event is to occur it then generates values for carrier duration, frequency curve (ascending or descending melodic interval), frequency, amplitude, sound source (AM or FM), and specific timbral parameters (MI, f_c:f_m, and the envelope attack and decay). Generated frequency, amplitude, and envelope attack and decay values are checked and adjusted, if needed, to not exceed their specific defined boundaries. Using frequency as an example, and assuming that a previous event's frequency value exists as a reference, *COMP2* utilizes the following procedure to generate the next event's frequency value:

```
n <-- random number from 1 to 12 (the number of initial proportions)
newFreq <-- oldFreq * nth proportion (increasing proportion for ascending frequency curve [e.g., 3:2],
        decreasing proportion for descending frequency curve [e.g., 2:3])

  (For ascending frequency curve, check against upper frequency boundary [SR/2])
if newFreq > hiFreq, then adjust newFreq by curved frequency space formula:
adjNewFreq <-- (hiFreq - oldFreq) * (nth decreasing proportion + oldFreq)

  (For descending frequency curve, check against lower frequency boundary [defined as 20 cps])
if newFreq < loFreq, then adjust newFreq by curved frequency space formula:
adjNewFreq <-- oldFreq - [(oldFreq - loFreq) * nth decreasing proportion]
```

Amplitude values are determined in a similar manner, and the remaining parameter values are determined by related procedures.

The composer's intent for the temporal area during which the Kyma music may interact with the trombone music is to permit the Kyma music to reflect upon and immediately process parts of the trombone music, rather than use the live trombone to trigger stored MIDI or other static routines. During this area, the Kyma system uses an ADInput Sound (a microphone connected to the system's ADC) to read in the trombone signal at specific times and then processes the signal through a harmonizer [Harmonizer7]. Using a separate ScoreLanguage Sound another part of *COMP2* assigns on and off times for this procedure, specific ratios used by the harmonizer, and amplitudes for each instance of this process using various manipulations of the original twelve proportions. The four Interactive trombone sections are divided into time segments by means of the twelve proportions: the trombone music's macro structure is proportionally mapped onto each of the four Interactive sections. In the first of these sections, for example (beginning at point "3:2" on Figure 1 above), the microphone is active for the first of these segments, which begins at relative section time 0.0" (actual composition time 4'26.667") and remains active for 6.5" (8:15 corresponds to 6.667% of the section, whose duration is 97.333"). The microphone is inactive for the second segment, then active again for the third segment (which begins at section time 13.9" and lasts for 8.5"), and so on. The resulting commentary process makes intrinsic sense to both the Kyma and trombone musics on a structural level while not necessarily agreeing with the trombone music on the gesture and event level, which adds an aspect of surprise for the trombonist. This middle area of interaction is also where the trombonist is free to improvise while interacting with the Kyma sounds, and so this area can be quite different with each performance.

As a practical time keeping measure, the trombone's second gesture of the composition (beginning at time 8.5"), which is improvised within certain limits, is recorded into temporary DSP memory by the Kyma system for direct and attenuated playback to signal the beginning of later sections for the trombonist.

5. SUMMARY

Each algorithm of the composition (diagrammed in Figure 2 above) contains a different, though related, part of *COMP2*. All parts of the program utilize the same twelve proportions as source material for all generated values, and small-scale event values are generated by means of a random number generator whose results are further constrained by the twelve proportions. Due to the flexible structure of the Kyma software, the actual programming blocks can be located exactly where they are needed within the organizational structure of the composition, which eliminates the need for a separate mega-program. Each part of the program can therefore be optimized for the specific tasks required by a specific part of the composition.

The composition *Discourse {GUTs 2a}* is this composer's first step toward a meaningful unification of the dynamic processes of algorithmic composition, real-time software synthesis, and live performance interaction. The Kyma system serves as a highly flexible and powerful environment with which to continue this research.

6. ACKNOWLEDGEMENTS

This research was supported by grants from the Higgins School of Humanities and the Research Board at Clark University. Additional support was provided by the Music Program, Department of Visual and Performing Arts, Clark University, and by the Tri-College Group for Electronic Music and Related Research [G.E.M.], Worcester, MA. The author wishes to thank Carla Scaletti and Kurt Hebel for their generous advice and friendship during the course of this work.

7. REFERENCES

B. Belet, "Proportional recursive stochastic composition using *COMP2*, a Smalltalk-80 composition program within the Kyma digital synthesis system," *Proceedings of the 1991 International Computer Music Conference*, (Montréal), B. Alphonce and B. Pennycook, eds., pp. 513-16, ICMA, San Francisco, CA, 1991.

C. Scaletti, "Kyma: an object-oriented language for music composition," *Proceedings of the 1987 International Computer Music Conference*, (Urbana, IL), J. Beauchamp, ed., pp. 49-56, [I]CMA, San Francisco, CA, 1987.

C. Scaletti, "The Kyma/Platypus Computer Music Workstation," *Computer Music Journal*, Vol. 13, No. 2, pp. 23-38, Summer 1989 [1].

C. Scaletti, "Composing Sound Objects in Kyma," *Perspectives of New Music*, Vol. 27, No. 1, pp. 42-69, Winter 1989 [2].

C. Scaletti, "Lightweight classes without programming," *Proceedings of the 1991 International Computer Music Conference*, (Montréal), B. Alphonce and B. Pennycook, eds., pp. 505-08, ICMA, San Francisco, CA, 1991.