On the creative use of score following and its impact on research

Arshia Cont

ON THE CREATIVE USE OF SCORE FOLLOWING AND ITS IMPACT ON RESEARCH

Arshia Cont
IRCAM CNRS STMS
Musical Representations Team
arshia.cont@ircam.fr

ABSTRACT

Score following research is one of the active disciplines of sound and music computing since almost 30 years that have haunted both algorithmic and computational development in realtime music information retrieval, as well as artistic applications in interactive computer music. This paper explores the creative use of such technologies and brings attention to new scientific paradigms that emerge out of their artistic use. We show how scientific and artistic goals of score following systems might differ and how the second, continuously helps re-think the first. We focus mostly on the musical goals of score following technologies which brings us to an underestimated field of research, despite its obviousness in creative applications, which is that of *synchronous reactive programming* and its realization in *Antescofo*.

1. INTRODUCTION

Score following is traditionally the automatic and realtime alignment of audio streams from musician(s) on the stage into a symbolic music score. In its artistic use, it allows realtime coordination and synchronization of live electronic programs with human performers for mixed interactive computer music pieces. In the scientific literature, it is also employed in off-line mode for alignment of audio to symbolic music scores as a front-end for music information retrieval applications.

The score following literature is one of the research disciplines in sound and music computing with clear impacts on both research literature and artistic applications in computer music. The number of published articles on score following algorithms are constantly increasing every year, and since a few years, more composers of interactive computer music are employing such technologies into their compositions. Since the inception of score following paradigms in the 1980s, the two fronts have been evolving together and giving birth to a handful of interactive softwares and concepts for computer music. With the advent of robust score following techniques with explicit musical considerations both for composer and performers and the recent flow of composers employing such systems (such as [1, 2]), the interaction between artistic use and scientific paradigms of score following is more than apparent.

This paper explores the creative use of score following and its impact on the research. The artistic cases discussed are limited to mixed electronics and instrumental pieces in the computer music repertoire. Specifically, we draw lines between the *scientific* and *artistic* goals of score following in general, and attempt to show how the second, often underestimated in the research literature, gives rise to new scientific paradigms to explore. The scientific paradigm exposed here brings the act of composition, as the authorship of time and interaction, close to *synchronous programming paradigms* in computer science.

The pieces and concepts explored in this paper are taken from the *Antescofo* [1] [1] repertoire, an anticipatory score follower equipped with a synchronous language for realtime computer music composition and performance. *Antescofo* is probably the first score following system featuring a coupled recognition system and synchronous language. This feature of *Antescofo* came rather as a necessity from its artistic use than pure scientific endeavor.

We begin the paper by some background on the creative use of score following. We then clarify the scientific and musical goals of score following in section 3. Particularly we draw on specific architectures used in most known score following paradigms within an artistic context, and draw conclusions on specific research paradigms that should be considered within this context in section 4. We proceed in section 5 by defining the architecture in *Antescofo* that addresses these issues, and define the semantics of our synchronous action language in section 6. We finish by exposing some recent examples employing *Antescofo* in section 7 and demonstrating the discussed points.

2. HISTORICAL BACKGROUND

Score following research was introduced in [3, 4] and initially geared towards *automatic accompaniment* applications in which the computer would synchronously perform and render the accompaniment section with a live performer undertaking the solo part of a given music score. The technical paradigm of score following has passed various stages ever since, evolving from symbolic string-matching techniques, to pitch detection, and probabilistic models. For a historical overview of score following algorithms we

[1] http://repmus.ircam.fr/Antescofo

refer the curious reader to [5] and instead, focus on its artistic employment hereon.

Artistic uses of score following technologies have been made within two trends: *Automatic Accompaniment* and mixed instrumental and electronics pieces.

2.1 Automatic Accompaniment

The goal of automatic accompaniment systems is first to *listen* to the live performer and extract position and tempo parameters with regards to its music score, and second to perform the accompanying parts synchronous to this live performance. The accompaniment part can be either symbolic data rendered into audio via some synthesis techniques (such as in early versions in [3, 4]) or employ realtime phase vocoding techniques on an audio recording in the style of *Music-Minus-One*. Among systems employing the latter is that of Christopher Raphael [6], performing automatic accompaniment on a number of classical music repertoire.

2.2 Mixed Instrumental and Electronics Repertoire

The consensus for interaction between a live music performance and composed electronics dates back to early experiments of Bruno Maderna [2], Karlheinz Stockhausen and Mario Davidovsky among other composers in 1950s, through tape and instrumental pieces. Synchronization between the instrumental music and electronics was assured either by using click-tracks or active listening. Despite the new possibilities that electronics had brought into the musical language, production means of electronics had introduced enough burden in the process of composition *and* performance that this realm remained highly experimental up to the 1980s.

Shortly after the advent of score following technologies, several composers recognized the opportunities that such tools could offer both at the compositional and performative levels of computer music. The most evident application would be naturally in synchronizing a pre-written electronic score to a live performance, extending the applications of score following from automatic accompaniment to a mixed repertoire.

Besides the performative comfort in employing score following technologies, some composers immediately recognized and incorporated the new opportunities that score following would bring in authoring *interactive electronic scores* coupled with realtime capabilities sound generation and transformations. The possibility of creating interactive music systems attracted new artists and researchers, and created one of the most fruitful periods in computer music. Robert Rowe's two volumes [7, 8] demonstrates how such novel paradigms have affected different practices in computer music. Among composers exposed to these new possibilities, Philippe Manoury was one of the earliest composers who integrated interactive music systems into his compositions and as a compositional process both for authoring and live performance. In particular, Manoury's early formalizations of the paradigm in collaboration with Miller Puckette, led to the birth of the *Max* programming environment [3], further developed and integrated by other composers such as Boulez, Lippe, and Settle, and since then widely referred to as the *realtime school* of composition. The most interesting concept brought by Manoury is that of *Virtual Scores*[9] developed hereafter.

2.2.1 Virtual Scores

A virtual score is a musical organization in which we know the nature of the parameters that will be processed but not their exact outcome at runtime since they're expressed as a function of the live performance. A virtual score hence consists of electronic programs with fixed or relative values/outcomes to an outside environment. A realtime electronic process is therefore one that exists in a music score, next to the instrumental transcription, and whose outcome is evaluated during live performance and as a function of the instrumental part's interpretation with all its diversity and richness.

The idea of virtual score is thus to bring in both the performative and compositional aspects of computer music within one compositional framework. A score following technology is then responsible for enabling the communication channels between the computer and the musicians according to a score and by allowing complex musical interactions similar to that of human musicians.

The framework of *virtual scores* is present and at the core of most interactive programming environments in computer music today. Despite its similarity to a traditional framework of composition, it does not limit its practice to traditional norms of music composition and on the contrary it has integrated non-traditional practices of computer music such as interactive composition [10], hyperinstrument composition [11], composed improvisations [12] and more, as employed in Manoury's early realtime pieces among others [13].

It is worthy to note that the realtime school was subject to constructive and interesting criticisms and debates at its very inception in a 1999 issue of *Contemporary Music Review* journal. Of particular interest to our work are that of Risset [14] and Stroppa [15] who underlined the lack of compositional and temporal considerations in existing frameworks at the time.

We would like to emphasize that interactive pieces in this sense, are not dissociable from *automatic accompaniment* paradigms in their architecture. Where automatic accompaniment deals with notes or chords in the accompaniment rendering, virtual scores replace them with processes and realtime electronic programs and transformations.

2.2.2 Score Following in Practice

To motivate further discussions, we attempt to provide the architectural design of a typical realtime mixed piece employing score following technologies. We use the "Introduction" part of the piece *Anthèmes II* composed by Pierre Boulez for violin and live electronics (1997) as demonstrated in figure 1. This music score shows the instrumen-

[2] The first mixed music for tape and instrument appears to be "Musica su due dimensioni" by Bruno Maderna for Flute and Tape (1952).

[3] In fact, Manoury's pieces *Jupiter* and *Pluton* can be considered as the first historical *Max* pieces and patches.

Figure 1. First two bars of *Anthemes II* by Pierre Boulez, for violin and live electronics (1997).

tal (violin) section in parallel to an approximative notation for the realtime electronics accompanying the system. Each system corresponds to a specific electronic process, whether realtime or samplers, by themselves accompanied by spatialization parameters. The sequencing of electronics in this score are either notated as relative to the performance tempo or fixed absolute values if necessary. The circled numbers in the score correspond to *synchronization points* between the instrumental and the electronics scores.

Figure 2 shows a generalized design diagram for the computer music realization of a mixed piece similar to the one in figure 1. This diagram generalizes most mixed electronic pieces employing score followers in the Ircam repertoire or employing *MaxMSP* or *PureData* [4]. It demonstrates the common trend which consists of having separate *instrumental* and *electronics* scores. The instrumental score plus synchronization tags (circled numbers in figure 1) are fed into the score follower which takes care of online alignment. The electronic score in turns is stored as tagged sequential data-structures (commonly referred to as `qlists` in *Max* and *PureData*). The electronic queues store variable/message pairs attached to symbolic time indexes and usually scheduled on a milli-seconds basis. The symbolic time indexes (tags) would then correspond to synchronization pivots in the instrumental score, destined for live synchronization. The modularity of environments such as *Max* or *PureData* allow co-existence of multiple sound processes in a single patch that can be controlled through the sequential electronic score.

The general diagram of figure 2 can be seen as two complementary systems: an *interactive system* consisting of the score follower and the musician, and a *reactive system*

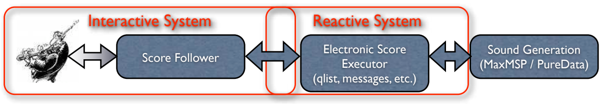

Figure 2. General diagram of a typical interactive mixed music.

consisting of the electronic score and its sequential execution as a reaction to the received tags from the interactive system. The two components are *interactive* and *reactive* due to their implicit nature of time and following [16]. The score follower is an *interactive systems* since it should be considered as part of the physical world (with the musician), and the second *reactive* since it runs on its own implicit clock and in reaction to an external environment.

3. SCIENTIFIC AND MUSICAL GOALS OF SCORE FOLLOWING

The score following literature is constantly increasing every year with new algorithmic contributions. It is important to distinguish between the *scientific goals* and *artistic goals* of such systems. Our argument here is that while the two goals are non-dissociable, they are however distinct and achieving one does not necessarily entail the other, and hence new research paradigms should be explored.

3.1 Scientific Context

Score following, in its scientific context today, deals with correct alignment of audio streams onto a symbolic score and correct extraction of musical parameters of the interpretation in hand. Besides automatic accompaniment systems, score followers and automatic alignment systems are

[4] See Pd Repertory Project: `http://crca.ucsd.edu/~msp/pdrp/latest/files/doc/`

employed as front-ends for many MIR applications such as score-informed editing softwares and source separation. It is clear that in such applications the alignment and extraction precision is of utmost importance and much algorithmic effort has been dedicated to achieve this. Figure 3 shows a simplified diagram of this aspect of score following where the computer is aware of the symbolic score expected to be performed by the live performer.

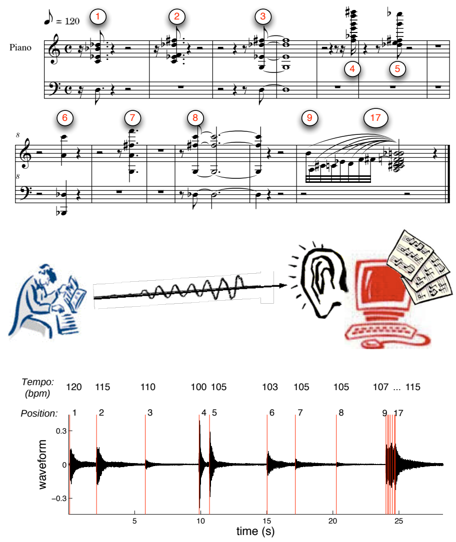

Figure 3. General diagram for an online alignment system.

Most score followers in the literature focus on this aspect of its applications. This is particularly clear by the effort of the community for evaluations of such systems [17, 18].

3.2 Artistic Context

Despite advances in alignment systems, very few systems have explicitly considered the artistic goals of score following systems and their direct design consequence. The artistic goals of score following as an interactive music system, and their discrepancy with its scientific goals can be discussed within two folds:

1. *Realtime Music Performance:* Score following is naturally a tool for realtime performance of mixed instrumental and live electronics. The discrepancy between the scientific and artistic goals in this context has much to do with the idea of *robustness* of the realtime alignment algorithms in score followers, especially in the context of live performance and realtime processing within uncertain environments. An ideal musical performance for an architecture depicted in figure 2 can be naturally achieved if the interactive system in hand has 100% precision in the recognition phase and given any performance. While recent systems demonstrate high precision and performance in realtime, the architecture in figure 2 is probably a bad choice for the musical finality of an interactive music system. To summarize, the musical goal of such interactive systems require that the *musical output* is acted upon expected despite any error from the live performer or the recognition system.

2. *Authoring and Composition of Realtime Music Processes:* The ultimate goal of score following as an interactive music system is naturally to express reactive programs that create the electronics part during the composition phase. This task requires a minimum of *musical expressivity* within the language that describes such interactions and gets naturally close to the idea of *virtual scores* as discussed earlier. Treating the *interactive* and *reactive* phases of an interactive piece separately as depicted in figure 2 most often evades expressing such interactions and most existing systems have found comfort in leaving this very important issue apart.

To demonstrate the above issues, consider again the score in figure 1 as an example: The first issue is clear by differentiating electronic events tagged by ① and ③. Event ① does not necessarily depend on the recognition of the first note in bar 1 (an F) while event ③ can be considered as having a local scope. If the musician or the recognition system misses the first high F in the violin part, it is evident that ① should not be dismissed. This is however not true for ③. If the chord corresponding to ③ is missed, that event can be consequently dismissed in order to maintain a musical coherence in the output. The second issue has much to do with the authoring of the electronic events presented graphically in figure 1. Parallel lines in this score correspond to *concurrent* electronic programs which are expected to output synchronously during live performance with their timing notated relative to the live performer's tempo. Most realtime programming environments however do not neither allow such timing expressivity for programming as time is usually expressed in absolute values, nor explicit concurrency in expressing electronic processes.

4. CONSIDERATIONS FOR CREATIVE USE OF SCORE FOLLOWING

With the above introduction, we draw important requirements for the use of score following as an interactive music system, destined both for *composition* and *performance*:

4.1 Time is Resource

Explicit modeling of *time* is of utmost importance for a score following system both at the recognition phase (for realtime performance) and authoring (composition). However, most score following techniques have focused on the event level (pitch, spectrum observation, etc.) and left temporal models approximate or implicit. At the same time, any music score contains important timing information such as tempo, relative durations and timing hierarchies within elements, that can help both phases of score following use. This issue is of extreme importance when such systems are to be employed in realtime (and thus in absence of future information for decoding), and can significantly enhance recognition and also access to temporal elements for computer music composition. The only practical score following systems which consider explicit time models for both phases are Music Plus One[2] and Antescofo[5] where tempo and event durations are first-class citizens in the systems and are employed both during recognition and accompaniment. In [2], temporal considerations are taken into account as a secondary pass and cascaded to an event recognition HMM based system, requiring offline learning of time parameters for best performance. In *Antescofo* an explicit time model is coupled with an audio recognition system through *Anticipatory Learning*, attempting to reduce complexity of computation and with no requirement for off-line learning. *Antescofo* in particular makes time and tempo variables explicitly available for programming reactive electronics.

4.2 Heterogeneous Models of Time

Any classical piece of music has multiple and heterogeneous models of time, which should be explicitly considered in the conception of any interactive music system. For example, grace notes in classical music are typical of *atemporal* events which exist spatially in a score but do not contribute to the tempo variations as opposed to *temporal* events (regular notes and chords). Glissandis whenever the instrumentation allows are also typical of *continuous time* events as opposed to *discrete time* events in regular notes and chords. Finally, trills and tremolos in the classical repertoire undergo *hierarchical time* structures where the global event itself can occupy a discrete or continuous duration while its internal elements can constitute (for example) atemporal elements.

Presence of heterogeneous times is more than evident in the contemporary music repertoire and more than essential in expressing electronic processes. An electronic process can contain discrete (relative or absolute time) events as well as continuous controls, or in some cases recursive processes. The point here is that such considerations are neither unique to electronic music, nor to any specific style of music. Western music notation has internalized such temporal structures that are in use by all composers and performers while computer music languages are still behind in terms of expressivity of time and their models. We will come back to this issue later in section 5.

4.3 Critical Safetiness

The musical output of an automatic accompaniment or score following system should not solely depend on the recognition system, or even to the live performer at some instances. This is in analogy to human coordination for ensemble performance: A live music performance should be smooth in time, and does not halt in presence of any error in realtime. As discussed in section 3, one of the discrepancies between scientific and artistic goals of a score following system is the issue of live performance and robustness of the recognition in realtime. We showed on a simple example in section 3.2 how a simple specification of electronic processes can save the musical output despite any error from the environment or the interactive recognition system. Interestingly the issue of critical safetiness is the subject of study in most realtime systems [19] and already employed in the industry. These paradigms should also be adopted for score following systems.

4.4 Authoring of Time and Interaction

The most important issue for creative use of score following, is in *how* such systems would bring live interaction as a first-class citizen in the compositional phase and enable an authoring of time and interaction for artists. While common computer music programming environments enable live interaction with musicians, they are particularly poor for authoring of time and interaction for composed music. The lack of explicit authoring tools of this kind has led to a common division between the performative and compositional aspects of computer music [20], criticized thoroughly by several pioneers of computer music [14, 15], and has been the subject of debate in a recent colloquium on the subject between various artistic disciplines [21].

This issue is directly related to domain-specifc computer language design, and in our case for realtime interactive computer music. We believe that this topic should not be treated separately from common technical considerations of score following systems and is directly related to the musical goals of such systems.

5. *ANTESCOFO*'S ARCHITECTURE

Antescofo is the latest incarnation of score following technologies at Ircam since 2008[1]. It is a realtime score following technologies that aims to integrate the points discussed in section 4 within one single environment. To this end, it consists of a state-of-the-art realtime alignment system, capable of aligning complex polyphonic instruments as well as decoding the realtime tempo of live performance. The novelty of the recognition system in *Antescofo* lies in its coupling of a realtime audio and tempo agent, and capability of handling heterogeneous times during the recognition phase. It is an *Anticipatory System* with an attempt to predict event positions in the future in order to aid recognition and undertaking of electronic actions. We leave detailed discussions of the recognition algorithm to [5] and instead focus on its musical aspects with regards to points discussed in the previous section.

Antescofo is destined for interactive mixed instrumental and electronics pieces and aims at bringing both *interactive* and *reactive* components of a typical piece within this repertoire as discussed in section 2.2.2 within one single framework. Figure 4 shows the general diagram of this architecture. Within this architecture, electronic programs are handled within one framework that allows employment of various time scales and coupling of electronic actions to the live tempo if needed. *Antescofo* can be employed as a traditional score follower, in which case realtime positions and tempo of the performance are obtained as the module's direct outputs during performance. The composer can optionally integrate electronic messages inside the instrumental score, in which case, *Antescofo* handles their message-passing to host programs that produce the electronics part. In this sense, *Antescofo* is used both during the compositional phase as an authoring tool for programming synchronous electronic events with regards to the instrumental score, and also in the performance phase by attempting to produce the desired musical output as described in the original score.

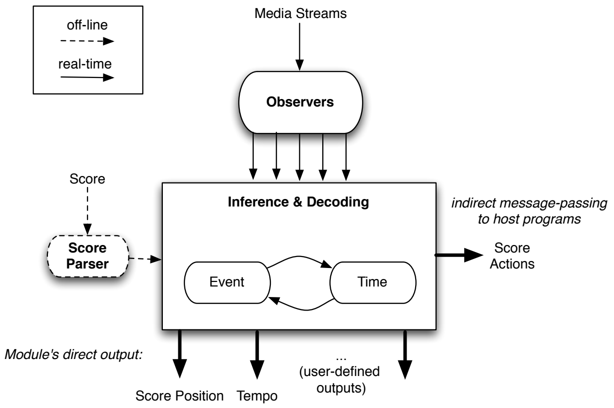

Figure 4. *Antescofo*'s general architecture, comprising of both *interactive* and *reactive* systems of figure 2 .

Antescofo can thus be used as compositional resource during the authorship of both instrumental and electronic score, on top of traditional use of score followers in live performance paradigm. The use of *Antescofo* as an authoring tool is possible because of the coupling of the recognition paradigm (natural to any score follower) with a synchronous realtime language for computer music composition. The synchronous language aspects of *Antescofo* has brought into focus the musical goals of such interactive music systems, which has been underestimated so far as a research paradigm itself. In the following section, we define briefly important aspects of this language with regards to points mentioned in section 4.

6. SEMANTICS OF PERFORMANCE SYNCHRONOUS LANGUAGE IN *ANTESCOFO*

The musical goal of score following has to deal with both the compositional and performative aspects of the piece of music in question. In the compositional phase, it has to be able to describe electronic processes in parallel to and ordered with regards to instrumental scores, and by employing the rich temporal semantics of musical intellect. The performative phase of such systems is responsible for evaluating electronic processes at a given position and tempo and as a reaction to the live performance. In this respect, an electronic score in a mixed interactive piece, is in close analogy to an orchestral or accompaniment score except that simple notes are replaced by programs with heterogeneous notions of time, and whose outcomes are not known in advance but deterministic in a musical context. Technically speaking, it is a reactive program with the ultimate goal of *determinacy* (in the computer science term), correct ordering at runtime, and (musical) critical safetiness. Such paradigms have been widely studied in the computer science literature for *realtime synchronous languages* [16] and widely applied in the industry for realtime critical systems. Our goal in this project is to adopt a musical semantics for such languages, whose application paradigms seem to be closely related to the acts of composition and performance. We will not expose the syntax of the language and leave it to curious users, and instead focus on the constructive semantics that allow an authoring of time and interaction in computer music.

An important consequence of the architecture discussed earlier is the coexistence of the *instrumental score* and *electronic score* within one single score. An *Antescofo* score contains two semantics: one for describing the music score of the human performer, and another for describing electronic events in an *action semantics*. Both semantics are capable of describing multiple scales of time (absolute, pulsed, continuous) heterogeneously within one score. The electronic score is a simple *message-passing coordination language*, where messages are bound to symbols, ordered and grouped as desired to imitate a musical score. These synchronous messages are then scheduled in realtime to be delivered timely to electronic modules. The choice of such semantics is in accordance with the wide practice of interactive music within *MaxMSP* and *PureData* programming environments. The goal of the *action semantics* in *Antescofo* is to provide expressivity for authoring of time and interaction and the synchronous scheduling of events in realtime. The primitives of this semantic consists of:

Discrete Events Message(s) bound to symbols with an optional delay. The delay can be in absolute time or relative to tempo and thus evaluated in runtime and reactive to tempo changes for synchronous output.

Continuous Events Similar to Break-Point Functions (BPF) where output is interpolated between discrete elements and scheduled in relative or absolute time.

Periodic Events A constructive semantic that allows (absolute or relative) periodic discrete or continuous messages, running forever when launched unless *killed* somewhere in the score.

On top of these primitives, the user can employ the following constructive semantics:

Parallel Groups Primitives above, can be optionally *grouped* to construct polyphonic phrases in the electronic score using an optional process name. This semantic is there to bring polyphonic authorship as well as independent but relative timing between groups of electronic phrases. This feature also brings a *temporal scope* for each group within the score during composition which is respected in runtime using the synchronous scheduling relative to tempo and position.

Nested Hierarchies Groups can be nested hierarchically and recursively to allow independent but ordered timings; respected during realtime scheduling.

Macros Evaluated at score load (and non-runtime) in order to provide motivic patterns both in message content and timing for composition of the electronic score.

Dataflow Functionals Mathematical expressions evaluated at runtime, useful to make message content relative to external variables.

Each block of programs in *Antescofo* accept specific attributes. Among such attributes, composers can specify the *scope* of each program dealing with their critical safety in case of performance errors, and also the ability to define independent or varying tempo (accelerando or rubato) relative to the performance tempi.

The development of the above primitives and compositionals have been undertaken incrementally and by observing various uses of interactive systems in composition and performance. *Antescofo* language is currently text-based with graphical support through *NoteAbility Pro* notation software editor [5]. We believe that a thorough and well-defined semantics can give rise to graphical semantics of programming which is already the case for synchronous languages within the avionics industries [22].

7. EXAMPLES

In this section, we aim at representing the compositional aspects of the semantics described above. It goes without saying that a thorough presentation of this system is within a performance situation and live coordination of electronic programs with the live performer within the new score following paradigm. Curious readers can refer to *Antescofo* website [6] for online videos and upcoming performances with the system.

Figure 5 shows an excerpt of the *Antescofo* score of "Otemo" for Vibraphone and live electronics by the composer Vassos Nicolaou [23] in the *NoteAbilityPro* score editor. The top staff is the Vibraphone score, where as the 11 bottom staves show the grouped primitives, similar to polyphonic lines, for the electronic scores. In realtime performance, each group is launched according to the performer's position and during the entire life of each group (shown here as their length) rescheduled according to detected position and tempo to assure synchronicity.

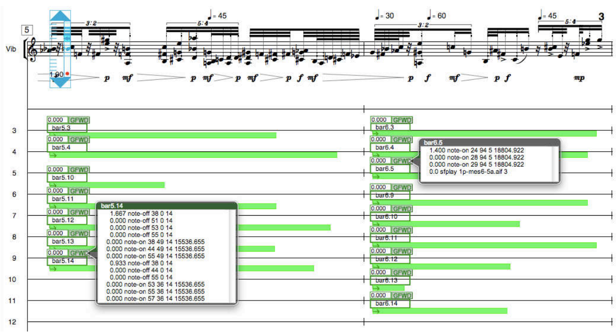

Figure 5. *Antescofo* grouped primitives visualization in NoteAbilityPro, excerpt from Vassos Nicolaou's *Otemo* for Vibraphone and Live electronics (2008)

Figure 6 shows an excerpt graphical representation of the electronic part for "Hist Whist" by composer Marco Stroppa for violin and chamber electronics as written in Antescofo. The timeline of the score is on the x-axis, where as the vertical bar demonstrate individual, parallel and nested processes within each block. This excerpt makes extensive use of nested macros written by the composer that control rhythmic progression of harmonization values and their amplitudes running on the realtime audio from the violin. It consists of 8 main blocks corresponding to 8 generated groups, each having hierarchical periodic primitives with periodic *kills* to imitate a rhythmic progression. Each pair actuates on one harmonizer module (one for transposition values and other on amplitudes), making it a total of four polyphonic realtime processors.

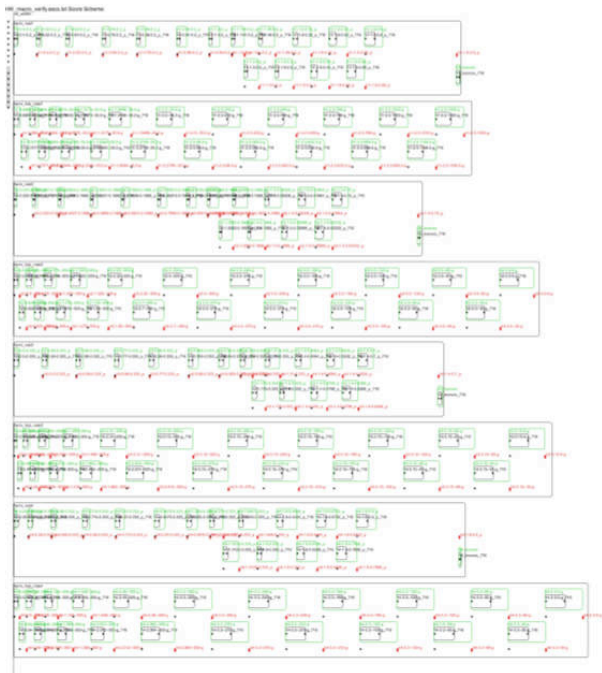

Figure 6. Visualization of electronic processes (excerpt from movement 3) of *Hist Whist* by Marco Stroppa, for violin and chamber electronics (2009).

The *Antescofo* score in figure 6 actually consists of 8 lines calling two macros with different arguments. Each macro recursively calls others defined by the composer to create the above patterns. The final outcome consists of more than 100 ordered and concurrent actions that will be synchronized to the performance in realtime.

The visualization in figure 6 is an experimental feature in *Antescofo* allowing composers to verify visually the contents of the generated scores during composition. It can be seen as hierarchical automata constructing a complex temporal system with resemblance to visual approaches in synchronous languages [24].

8. CONCLUSIONS

Whereas traditional score following paradigms have put emphasis on high precision in alignment and extraction of symbolic parameters from realtime audio, the creative use of such systems infer other goals which have been underestimated as a research paradigm in the literature. In this

[5] http://debussy.music.ubc.ca/NoteAbility/
[6] http://repmus.ircam.fr/antescofo

paper we attempted to show those missing paradigms that deal with authorship of time and interaction, critical safety of realtime electronic scores during performance, heterogeneous representations of time, and explicit models of time, both for composition and performance of interactive computer music.

We showed the close relation between the creative use of score following systems with that of realtime synchronous programming, bridging the gap between compositional and performative aspects of computer music, and bringing the rich expressivity of musical vocabularies into a simple computer language. The emergence of this research paradigm is mostly due to creative uses of score following systems which do not hesitate to rethink our practices and interfaces with computers for making music. We believe that this synergy will create an important momentum between artists and researchers in the years to come and hope that this paper has shed some lights on the importance of this new paradigm in both communities.

9. REFERENCES

[1] A. Cont, "Antescofo: Anticipatory synchronization and control of interactive parameters in computer music," in *Proceedings of International Computer Music Conference (ICMC)*. Belfast, August 2008.

[2] C. Raphael, "Music Plus One: A System for Expressive and Flexible Musical Accompaniment," in *Proceedings of the ICMC*, Havana, Cuba, 2001.

[3] R. B. Dannenberg, "An on-line algorithm for real-time accompaniment," in *Proceedings of the International Computer Music Conference (ICMC)*, 1984, pp. 193–198.

[4] B. Vercoe, "The synthetic performer in the context of live performance," in *Proceedings of the ICMC*, 1984, pp. 199–200.

[5] A. Cont, "A coupled duration-focused architecture for realtime music to score alignment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 6, pp. 974–987, June 2010.

[6] C. Raphael, "The informatics philharmonic," *Commun. ACM*, vol. 54, pp. 87–93, March 2011. [Online]. Available: http://doi.acm.org/10.1145/1897852.1897875

[7] R. Rowe, *Machine Musicianship*. Cambridge, MA, USA: MIT Press, 2004.

[8] ——, *Interactive music systems: machine listening and composing*. Cambridge, MA, USA: MIT Press, 1992.

[9] P. Manoury, *La note et le son*. L'Hamartan, 1990.

[10] J. Chadabe, "Interactive composing: An overview," *Computer Music Journal*, vol. 8, no. 1, pp. 22–27, 1984.

[11] T. Machover and J. Chung, "Hyperinstruments: Musically intelligent and interactive performance and creativity systems," in *International Computer Music Conference (ICMC)*, 1989, pp. 186–190.

[12] X. Chabot, R. Dannenberg, and G. Bloch, "A workstation in live performance: Composed improvisation," in *International Computer Music Conference (ICMC)*, Octobre 1986, pp. 537–540.

[13] M. Puckette and C. Lippe, "Score Following in Practice," in *Proceedings of the ICMC*, 1992, pp. 182–185.

[14] J.-C. Risset, "Composing in real-time?" *Contemporary Music Review*, vol. 18, no. 3, pp. 31–39, 1999.

[15] M. Stroppa, "Live electronics or live music? towards a critique of interaction," *Contemporary Music Review*, vol. 18, no. 3, pp. 41–77, 1999.

[16] N. Halbwachs, *Synchronous Programming of Reactive Systems*. Kluwer Academics, 1993.

[17] A. Cont, D. Schwarz, N. Schnell, and C. Raphael, "Evaluation of real-time audio-to-score alignment," in *International Symposium on Music Information Retrieval (ISMIR)*. Vienna, Austria, October 2007.

[18] ScofoMIREX, "Score following evaluation proposal," webpage, August 2006. [Online]. Available: http://www.music-ir.org/mirex/2006/index.php/Score_Following_Proposal

[19] N. Storey, *Safety critical computer systems*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1996.

[20] M. Puckette, "Using pd as a score language," in *Proc. Int. Computer Music Conf.*, September 2002, pp. 184–187. [Online]. Available: http://www.crca.ucsd.edu/~msp

[21] Ircam, "Colloque international écritures du temps et de l'interaction," in *Agora Festival*. Paris, France.: Ircam-Centre Pompidou, June 2006.

[22] F. Dormoy, "Scade 6: a model based solution for safety critical software development," in *Proceedings of the 4th European Congress on Embedded Real Time Software (ERTS'08)*, 2008, pp. 1–9.

[23] S. Lemouton and V. Nicolaou, "Polyphonic audio score following: The otemo case," Ircam - Centre Pompidou (Composer in Research Project 2009), Tech. Rep., 2009. [Online]. Available: http://articles.ircam.fr/textes/Lemouton09c/

[24] D. Harel, "StateCharts: a Visual Approach to Complex Systems," *Science of Computer Programming*, vol. 8-3, pp. 231–275, 1987.