Playing with Virtual Musicians: the Continuator in Practice

François Pachet
Sony Computer Science Laboratories-Paris
Email: pachet@csl.sony.fr

Introduction

Playing with musicians, as if they were there, but without their physical presence, is an old and strong fantasy. Indeed, wouldn't it be nice if you could play with your favorite – and inaccessible - Jazz musician? If you could play with him, or, rather, like him, but still retaining your own musical intentions? Why not then play with a virtual orchestra of musicians? Going further (or closer), what about playing with virtual copies of yourself?

The Continuator project conducted at Sony CSL addresses these issues, by developing techniques that capture efficiently stylistic information, and by designing musical instruments that make these techniques easily useable, and seamlessly integrated in real world music playing contexts. The current Continuator prototype is able both to learn quickly arbitrary musical styles in real time, and to generate music consistent with these styles, while remaining intimately controllable. We describe here the main technical issues at stake, and the new playing modes that are made possible by the resulting system.

The Continuator: an Instrument that Learns Musical Styles

The technical issue of learning automatically and in an agnostic manner a musical "style" has long been addressed by researchers in the community of artificial intelligence and information theory. Shannon introduced in his seminal paper in 1948 the concept of information based on probability of occurrence of certain messages. This notion was quickly used to model musical styles, for instance by (Brooks et al., 1957). These experiments showed that it was possible to create pieces of music that would *sound like* given styles, by simply computing and exploiting probabilities of note transitions. More generally the notion of Markov chains was applied to music in a number of ways. One of the most spectacular applications of Markov chains to music is probably (Cope, 1996), although his musical productions are not entirely produced automatically. A good survey of state-of-the-art of Markov based techniques for music can be found in (Triviño-Rodriguez et al. 2001), including in particular variable-length Markov models, which capture more finely stylistic information.

In our context, we want to learn and imitate musical styles in a faithful and efficient manner, and make the resulting mechanism useable as an actual music instrument. This raised a number of technical issues.

First, the handling of contextual information. Markov-based models, by construction, always generate music which is consistent with a style, itself defined by recurring patterns found in the learnt corpus. Music, especially improvised music, is of course made up of such repetitive ingredients, as Shannon noticed, but also of unexpected events or material. For instance, in a Jazz trio, the pianist may suddenly play chord changes which are not exactly the ones prescribed by the score. Randomness may of course be introduced easily, but Markov models are by definition insensitive such changes in the "external" world. We have introduced a bias in the Markov generation to allow the system to generate music streams which are both stylistically consistent (Markovian), and as close as possible to an arbitrary external source of information (Interative).
For instance, the system may produce musical continuations of the guitarist in a given style, and at the same time try to match the harmony determined by the pianist. It is precisely this novel feature which allows to turn our musical *automaton* into an actual interactive *instrument*.

Second, the management of *imprecision*. Musical phrases in practice are far from "perfect". Academic works in musical Markov chains consider learning corpuses as sets of exact strings. In real music, musicians do not always play musical phrases exactly the same way. There are musical variations, as well as errors. These are not explicitly taken into account by Markov models. If we stick to such a theoretical Markov model, the system takes a lot of time to learn patterns, because these patterns are not played exactly the same way in the learnt corpus. Additionally, the system will introduce stylistic breaks in the generated streams. We have introduced a scheme that allows the system to learn simultaneously several Markov models with various degrees of precision. When the most precise model does not match an input phrase, the system looks for a continuation in a less refined model. For instance the most refined model will take into consideration the exact pitch, duration and velocity of each note making up the phrase. A less refined model will consider only pitch regions (i.e. C0 to E0 are the same item, then F0 to A0 are another item, etc.). This scheme makes it possible in practice to match input sentences that were never learnt before, but which are close to previously learnt sequences. Using this scheme, the system can produce continuations which are optimally consistent, and avoid stylistic breaks typical of earlier musical Markov models.

Lastly, *efficiency*. The style learning and music generation mechanisms should be efficient enough so that the music produced by the system is seamlessly integrated in the playing mode of the musician. In the standard continuation mode, the system produces a continuation of the phrase played by the musician. The real time constraint is that the continuation be produced in less time than the average inter onset time of the input notes. To take a concrete example in Jazz, we analyzed phrases played by John McLaughlin (see e.g. http://www.musicindustries.com/axon/archives/john.htm), said to be one of the fastest Jazz guitarists, and found a minimum inter onset time of about 60 milliseconds. This figure gives an approximate constraint for the computation time of our system: it should be able to learn and produce sequences in less than 30 milliseconds. We do give the details of the implementation of our learning module here, but it was shown to learn and produce continuations for large corpuses of phrases in less than 5 milliseconds with a Java prototype running on a Pentium III laptop. Consequently, the musician can play with the system in a seamless way: the continuation produced may be tuned to be virtually indistinguishable from the human input phrase.

There are other important technical issues to ensure that the learning and production mechanisms fit with the constraints of music (improvised music in particular), such as the management of polyphony and the segmentation of input phrases. These issues are not discussed here for reasons of space, and are addressed in (Pachet, 2002).

New playing modes

In the basic mode, the Continuator is connected in input and output to a synthesizer. A musician plays musical phrases, and the Continuator continues these phrases. However, the Continuator can also be used in many other configurations, some of them actually unexpected. We list here the ones we have started to investigate and found most interesting during our experiments.

The basic mode: Autarcy

In this mode, one musician is connected to the main input of the system. The system starts from scratch (no initial memory), and learns progressively the musical style of the musician. The system is set for instance to produce continuations of increasing size. At the beginning, only a few notes are generated, as continuations or answers to input phrases. The more the system learns, the more accurate the style representation is and the more notes the system plays. An interesting musical effect is that the musician gradually shifts from a “teaching”, active behavior, to a listening, passive mode: the system initially continues the phrases played by the musician, but eventually it is the musician who continues the phrases played by the system.

Virtual Duo

This mode is the same than Autarcy, except that the musician initially loads a pre-recorded memory. This memory is taken from a library of memories, built by simply letting various musicians play freely with the system. This allows a musician to play, virtually, with any body whose memory has been recorded previously.

Contextual Continuation

In this mode, one musician is connected to the main input of the system. Another musician (e.g. a pianist) is connected to the contextual input. The continuations produced by the system are built from the input phrases of the main user, but follow dynamically the contextual information provided by the second musician. This mode enhances the interaction between the two musicians, as the second input has clear, and parameterized, effect on the overall generated music.

Playing twice with oneself

This provocatively labeled mode consists in two phases. First a musician plays harmonically rich music to the system: chords, chord sequences of all kinds. In a second phase, the system produces an infinite stream from the learnt chord sequences. The musician then plays on top of this harmonic material a solo improvisation, which is fed in the harmonic context input of the system. The chord sequence played by the system tries then to "follow" the improvisation. This mode creates a striking impression to the musician (and the audience), as what happens in effect is that the musician virtually follows himself.

Swapping mode

In this mode, several musicians are connected, each one to a different version of the system. The respective memories are swapped. For instance, the guitarist will use the memory of the pianist. The result is that the continuations of the guitarist will use the patterns and style of pianist, and vice versa. This is a new collaborative music-playing mode with lots of potential, and which is still to be experimented further.

Experiments

Experiments with these various modes have been performed both in the laboratory and during live concerts. Musician and composer Gÿorgy Kurtag Jr. has played extensively with the system and performed in *festival d'Uzeste* (August 2001) and festival *Sons d'hiver* (January 2002). Jazz pianists Alan Silva and Bernard Lubat have played with the system and provided valuable feedbacks. An interactive musical composition will be premiered at the Vienna 2002 *Festwochen* as well as for the 2002 *Budapest festival*. Other experiments with Jazz musicians are being conducted with the goal of enriching our library of memories and collecting feedbacks on practical issues in using the system.

Aha Effects

In all cases the reactions of musicians playing with the system were extremely positive. The most striking effect which was noticed systematically on all musicians experimenting with the system can be described as a *Aha* effect, triggered by the sudden realization that the system is starting to play exactly in the same style as oneself. These brief moments of sudden illumination are illustrated in the Figure 1 and 2, with photographs taken during experiments with the system.

Figure 1. Alan Silva playing with Continuator (Jan. 2002).

Figure 2. Bernard Lubat playing with the Continuator (Feb. 2002).

Subjective impressions

The musicians who played with the system (Bernard Lubat, Gyorgy Kurtag Jr., Alan Silva) all expressed a strong subjective impression that is hard to define precisely, but which can be sketched with the following recurring expressions:

Effect of playing with the system (Gyorgy Kurtag):
"The system is a kind of amplifying mirror", "it manages the past for me", "It relieves me of my core, repetitive tasks, and allows me to take care of high level musical tasks, such as organizing superstructure in my musical discourse"

Triggers new ideas (Bernard Lubat):
"The system shows me all the potential ideas I could have developed, but that would take me years to actually develop. It is years ahead of me, yet everything it plays is, unquestionably, me."

Creating new forms of musical improvisation (Bernard Lubat):
"Because the system plays things which are at the border of the humanly possible, especially with the long but catchy melodic phrases played with an incredible tempo, the very notion of virtuosity is challenged. Virtuosity is becoming an actual musical object that can be created, manipulated in simple ways."

Relation to one's own learning (Alan Silva):
"The system is doing what it took me years to learn, in particular through Schillinger's book. That you can do much more with simple musical material (e.g. a few notes) than what the scale-based approach tells you. It is a kind of materialization of Schilinger and Sloniminsky's vision "

In the audience, the reactions ranged from amazement, astonishment, and, strangely, the desire to play with the system. This type of reactions pushed us to organize a concert in the near future in which, at some point, people in the audience can participate, by playing with a keyboard, and having the Continuator continue the phrases in the style of the performing musician.

Lessons learned

Several important lessons have been learnt from these experiments, in particular during live concerts. We list here the main ones.

Finding good default values for parameters is not always possible

The importance of default values is well known: first time users must not be bothered by setting values to parameters they do not yet understand, otherwise they may be quickly discouraged. This is especially true for musicians, who are often not used to manipulating complex software.
For instance, an important parameter is the length of musical phrase generated by the system. This length is specified as a multiple of the length of the input phrase. Initially, the default value was 1, which meant that the system would play the same number of notes than the input phrase. After fumbling around with other default values, the systematic indifference in reaction to my question (do you want the system to play a lot, or only a little?) pushed me to set automatically increasing values for this parameter: the system plays more and more notes as the session goes. This default behaviour seems to fit the need of the first sessions (both Lubat and Silva did not really catch, seemed indifferent). In the longer run, setting this parameter to the right value, during the performance, is typically of the responsibility of the “conductor” (see below).

Tweaking parameters: a new musical instrument

Initially, we had the goal of trying to minimize as much as possible the interaction between the musician and the numerous parameters of the Continuator. This goal progressively appeared as a bad idea. It appeared (during a concert) that parameter tweaking was indeed a determining factor in the success of the overall concert: deciding the length of continuations, the inter connections between musicians, the playing modes, the various musical modes (e.g. type of rhythm, tempo and amplitude responses) required constant attention, listening, and the ability to quickly decide and change these parameters in reaction to the music played. A new role emerged, and we realized that parameter tweaking corresponded to a musical instrument as such. The lesson is that when parameter tuning/hiding becomes very difficult, it is maybe because there is a hidden agent that has not yet been fully identified. In our case, this agent could be coined as a “controlling” musician” though parameter tweaking requires the same musical abilities than actually improvising: ability to listen to others, to participate in the structuring of an improvised piece, and of course the ability to master the system through its interface, seen here as an actual musical instrument.

The difference between continuation and answer is problematic

Most musicians tend to see initially the system as a question/answer system. The difference between a continuation and an answer is real, but indeed not easy to explain: when the system produces a continuation of a phrase played by the musician, the continuation is the second part of the *same* musical phrase. Conversely, a musical *answer* is a musical phrase in its own right, with a beginning and an end. I progressively gave up the idea of convincing musicians that they were playing with a continuation instead of a question / answer system when I realized that they would naturally accept this idea after having spent some time with the system: some fundamental aspects of the system should not be talked about in the initial stages of the interaction…

Application to education

Finally, the system has applications for musical training. Indeed, preliminary experiments show that children are very sensitive to the imitative power of the system. Even when children are not musically trained, they seem to develop instinctively personal playing modes. These modes may sound primitive, but careful studies show that they are in fact rather differentiated. For instance, a child can hammer repeatedly one single note with one finger, or play chords with all his fingers, stick to the central region of the keyboard, or explore various regions, with notes, arpeggios, chords, etc. The experiments conducted show that the use of the system in an imitative mode (autarcy) pushes the child to explore new playing modes. Systematic experiments are being performed in schools to further validate this hypothesis, and others, in the context of musical education.

Figure 3. A 3 year old child discovers new playing modes with the Continuator.

Conclusion

We have described the issues at stake for putting into practice a musical interactive system with a style learning facility. Examples of the system (audio and video extracts) can be found at our web site: http://www.cs.sony.fr/~pachet/Continuator. Many lessons learned from these real words experiments are related to the issues of setting default values for the numerous parameters of the system. One important and unexpected conclusion is also that the complexity of the parameterization somehow hides a new musical role. Instead of trying to assign automatically consistent values to parameters and simplify the system's interface, we now investigate better designs for a fully-fledged interface that would turn the system into a real control instrument. This role corresponds in fact to the transposition of the conductor in the world of digital collaborative music making.

References

Brooks, Hopkins, Neumann & Wright. "An experiment in musical composition." *IRE Transactions on Electronic Computers*, 6(1), 1957.

Cope, David. Experiments in Musical Intelligence. Madison, WI: A-R Editions, 1996.

Pachet, F. "Interacting with Style: the Continuator", submitted to *International Computer Music Conference*, 2002.

C. E. Shannon, ``A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379-423 and 623-656, July and October, 1948.

J. L. Triviño-Rodriguez; R. Morales-Bueno, Using Multiattribute Prediction Suffix Graphs to Predict and Generate Music, Computer Music Journal 25 (3) pp. 62-79, 2001.

Schillinger, J. *Schillinger System of Musical Composition*, Da Capo Press, 1978.

Slonimsky, N. *Thesaurus of scales and melodic patterns*, Music Sales Corp, 1997 (reprint).