In a *Learning Zone of One's Own*, Tokoro and Steels Eds, IOS Press, 2004.

6.

On the Design of a Musical Flow Machine

François Pachet

This paper addresses the issue of designing interactive systems that create flow experiences in users. I first describe an interactive musical system called the Continuator, which is able to learn the musical style of users in an agnostic, continuous fashion. I then describe experiments conducted with professional musicians and with 3 to 5-year old children and the Continuator. I show that these interactions are – almost - typical of the Flow phenomenon, as introduced by Csikszentmihalyi. I then focus on the abstraction of the design principles behind the Continuator and propose the notion of *Reflective Interactive System* as a class of applications which trigger Flow experiences. Based on the analysis of the various psychological experiments conducted so far, I identify the issue of flexibility in interaction protocols as a crucial step to enhance the efficiency of Reflective Systems as we envisage them today.

1. When are interactions interesting?

Virtually all things done by a computer program are interactive today. From web sites to word processors, from video games to entertainment robots, users are constantly engaged in various forms of dialogs with computer programs.

In most cases, these dialogs are designed to help users solve precise and well defined tasks. For instance, information retrieval systems are designed to allow users to find quickly information they look for. Many web sites are – at least in theory – often designed to minimize the number of clicks needed to find specific information. Video games propose interactive devices designed so that users can quickly issue commands to move, shoot, or perform various actions in real time, without having to think about their input devices. In all these cases, the design principles put forward consist in satisfying precise criteria: utility, optimisation, conciseness or transparency. Of course there are some exceptions to the utilitarian view of interactivity: artistic installations in the domain of *digital art* (as found in shows such as Ars Electronica) consist often in providing the user with some sort of hopefully novel aesthetic or sensory experience, which does not necessarily correspond to a task or game to play with precise rules to follow. However, this lack of aim in purely artistic experiments makes these installations often hard to understand. Actually, the very idea of an aesthetic experience *as such*, decoupled from any relation to content is itself debatable.

Whatever the subject matter, all interactions at work in man-machine interfaces do not create equally *enjoyable* experiences for users. This very issue of what makes an interactive system appealing, enjoyable or attracting to users is systematically eluded. It is also the subject matter of this chapter: what makes an interaction interesting ?

This bold question is addressed under the form of a particular experiment in music interaction, called the Continuator project, whose goal is precisely to engage users in exciting, appealing interactions. We first introduce the project and the basic mechanisms of the system, and then describe some experiments involving children. Then we show the relations that these experiments may have with the theory of Flow, as introduced by Csikszentmihalyi (1990). In the last section we generalize from the experiments with the Continuator and propose the notion of *Reflective Interactive System* as a class of applications having the same properties than the Continuator. We propose that Interactive Reflective Systems are a possible way to build Flow machines. We conclude on proposing extensions and other examples of this class of system and issues remaining to be solved.

2. The Continuator

The Continuator project stems from major frustrations of the author regarding music interactive systems. The issue of interacting with a computer to make music has long been addressed by many researchers, with many different goals in mind. Today there seems to be two categories of music making systems.

On the first hand, purely interactive systems in which users may trigger various kinds of musical effects. These systems have been experimented for a long time by the pioneers of computer music. For instance, Jean-Claude Risset composed pieces with a Yamaha Disklavier (a piano forte with a Midi input and output) in which the computer played various types of accompaniment based on his input (Risset & Van Duyne, 1996). More recently, the Korg company issued a synthesizer of a new kind, called the Karma, which proposes thousands of such musical effects (Kay, 2000). This notion of musical effect is particularly well adapted to describe the type of interaction at play: when a user presses a key, a chord, or some sort of predefined musical sequence, the system reacts by producing, in turn, a sequence based on the user input. The versatility of the machine makes it possible many effects ranging from arpeggiators to automatic accompaniments in many styles.

On the other hand, a lot of research has been devoted to understanding the notion of musical style, and building systems that produce music in particular styles. On of the first attempt in building an automatic composer is probably Mozart with his Musikalisches

Würfelspiel in 1787 (Chuang, 1995). This composition took the form of a dice game. Each throw would determine a number. This number, associated with the number of the current measure would determine the newt measure, out of a collections of pre-programmed measures composed by Mozart himself. From a table of 176 minuet measures, the device could generate, thanks to two 6-sided dices, 1.3 10^{29} different minuets. The device invented by Mozart was in fact a rudimentary implementation of so-called Markov models, invented much later, but using the same principle: transition tables containing probabilities about transitions between two events. Applying Markov models to analyze and generate music has become, since Mozart, a tradition in itself. Today, many systems have been proposed to analyze automatically corpuses of music material, under their score or Midi form, and to generate new material based on these analysis. The most spectacular of these attempts was probably done by Cope (Cope, 1996) with his EMI system. David Cope's system is able to generate convincing musical pieces in the style of virtually all classical and modern composers, ranging from Jean-Sebastian Bach to Scott Joplin. However, to achieve a reasonable level of accuracy and faithfulness to the style being mimicked, Cope has to enter by hand many information to the system, in particular concerning high-level musical constructs such as forms, beginning, endings, etc. which are not naturally captured by Markov models. However, whatever their level of accuracy music generation system are not interactive: the processes of analysis and generations are separated, with possibly many hand made parameter tweaking inside.

Unfortunately, purely interactive systems are never intelligent, and conversely, intelligent music generators are never interactive. The Continuator is an attempt to combine both worlds: interactivity and intelligent generation of music material in a single environment.

The Continuator System

The Continuator system consists of one MIDI input (typically from a synthesizer) and one MIDI output (typically returning to the same synthesizer). Its operation in the standard mode involves no interface other than the MIDI instrument itself. The user plays musical sequences of any kind, either monophonic, polyphonic, in any playing style. When the phrase is terminated, the Continuator generates a musical phrase in response. This musical phrase has the characteristic of being stylistically similar to the phrases played by the user so far. Technically, it is a continuation of the last input phrase, hence the name of the system.

Although it is difficult to define the notion of *musical style* precisely, we have adopted, like our predecessors in music generation research, a notion of style consisting of the statistical distribution of notes, chords and musical elements in general as well as their ordering. The Continuator, like post of music style replication systems, is based on a Markov model of musical phrases, and the model of the style created by the system retains melodic patterns, harmonic progressions, dynamics and rhythmic patterns of the corpus used for learning. An important consequence of this approach is that the phrases generated by the Continuator are similar but different from the phrases played by the user. The Continuator may therefore be seen as an engine for producing variations of arbitrary musical material.

Some Musical Examples

To illustrate the working of the Continuator, simple musical examples are given below (see Figure 1 and Figure 2). These examples are noted exactly as they are played, i.e. without

rhythmic quantization. They show that the Continuator adapts quickly to arbitrary styles and is able to generate musical material that "sounds like" the user input on a relatively small scale. Issues related to capturing higher-level structure are not discussed here as they are not relevant for our purpose (refer to Pachet 2002a for more details). More sophisticated examples of music created by the Continuator can be found on the web site of the author.
The most important aspect of the Continuator is the fact that the musical material generated always conforms stylistically to the input. Also, the Continuator keeps on learning from whatever input is given. As a consequence, the behavior of the system improves over time: if the user produces phrases which are stylistically consistent, but unique, the Continuator will learn more faithfully and will produce musical phrases that are increasingly accurate, with respect to the musical style of the user.

Figure 1. A simple melody (top staff) is continued by the Continuator in the same style (bottom staff).

Figure 2. A simple chord sequence (top staff) is continued by the Continuator in the same style (bottom staff)

Implementation and Design

There has been considerable research done in the fields of artificial intelligence and information theory regarding the technical issue of learning a musical style automatically in an agnostic manner. Shannon introduced the concept of information based on the probability of occurrence of events in communications (messages) in his seminal 1948 paper (Shannon, 1948). This notion was soon after used to model musical styles, for instance by (Brooks et al., 1957). These early 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 precisely, given a corpus of musical material (typically musical scores or MIDI files), the basic idea was to analyze this corpus to compute transition probabilities between successive notes. New music can then be produced by generating notes using these inferred probability distributions. One of the most spectacular applications of Markov chains for the generation of music is probably (Cope, 1996), although his musical results are not entirely produced automatically. A good survey of state-of-the-art Markov-based techniques for music can be found in (Triviño-Rodriguez et al. 2001), including in particular variable-length Markov models, which capture stylistic information more finely.

The Continuator system is yet another species in the world of musical Markov systems, although with novel features. In our context, we wanted to learn and imitate musical styles in a faithful and efficient manner and make the resulting mechanism useable as an actual musical instrument. This raised a number of technical issues, whose solutions were progressively integrated in the Continuator.
The architecture of the Continuator consists of two modules: an analysis module and a generator module. The analysis module takes as input MIDI sequences played in real time. The system contains three main parts:

1) A phrase end detector, which is able to detect that a musical phrase had "ended". This detection is based on an adaptive temporal threshold mechanism. The threshold is inferred from the analysis of inter onsets intervals in the input sequence. As a result, if the input sequence is slow (or, rather, contains few notes per seconds) then the threshold is increased, otherwise it is decreased. This simple mechanism ensures that the continuation produced will be seamless, temporally.

2) A pattern analyzer. Once detected as completed, these input sequences are sent to a pattern analyzer, which builds up a Markovian model of the sequence. The complete algorithm is described in (Pachet, 2002b), and consists of a left to right parsing of the sequence to build a tree of all possible continuations for all possible prefixes of the sequence. To speed up learning, the system also learns all transpositions of the sequence.

3) A *global property* analyzer. Various global properties of the input sequence are also analyzed, such as: the density (i.e. number of notes per second), the tempo and the meter (location of strong / weak beats), the overall dynamics (i.e. loud or soft), etc. These properties are used to produce a continuation which is musically seamless with the input.

The generator is responsible for producing the continuation of the input sequence. The actual production of the musical material exploits the Markovian graph created by the analysis module (Pachet, 2002b). It essentially consists of producing the continuation on a note-by-note basis. Each note is generated using the Markovian probabilities inferred during the analysis stage. Technically it uses a variable-order Markov generation that optimizes the relevance of each single note continuation by looking for the longest possible subsequence in the graph. Special care has been taken to perform meaningful segmentations of the input phrases for the learning phase. Indeed, real-world input phrases are never composed of perfectly successive notes or chords. In order to "cut" input phrases into chunks, which are then fed to the learning system, a segmentation process is able to detect note or chord transitions and possibly cut across unfinished notes. The module also stores the possible "residual" discrepancy, and restores it at generation phase, so that the material retains the rhythmical "naturalness" of the original style.

This continuation sequence is, however, crude, in the sense that it does not necessarily have the global musical properties of the input sequence. Therefore, a mapping mechanism is applied to transform the brute continuation into a musical phrase that will be played just in time to produce seamlessness. Currently, the properties which are analyzed and mapped are tempo, metrical position, and dynamics. More details can be found in (Pachet, 2002a).

Playing Modes

The basic playing mode of the Continuator is a particular kind of turn-taking between the user and the system determined by three principles:

1) Automatic detection of phrase endings. The Continuator detects phrase endings by using a (dynamic) temporal threshold (typically about 400 milliseconds). When a time lapse exceeds this threshold, the Continuator takes the lead, and produces a musical phrase.

2) The duration of the phrase generated by the Continuator is parameterized, but in most cases the duration is set to be the same as the duration of the last input phrase.

3) Priority given to user. If the user decides to play a phrase while the Continuator is still playing, then the system will stop and return to listening mode (and eventually apply again principle 1).

These principles, in the current implementation, are hard-coded in the system. Moreover, they are set without explicitly telling the users. Experiments with the system has shown that these rules are usually easily learned by the user in an implicit way – the behavior of the system is usually obvious, even for children.

3. Experiments with the Continuator

Experiments with Professional musicians

Apart from the technical issues related to system design and the evaluation of the "stylistic similarity" of the music produced by the system, it soon appeared that there was an important dimension of the project that was more difficult to describe in a standard scientific paper, related to the *subjective impressions* of users playing with the system, or watching it in action.
Indeed, one impressive result of the experiments with professional musicians is that the very use of the system provokes intense subjective impressions and reactions. Rarely but occasionally, these reactions are negative. For instance a composer from Ircam reacted aggressively against the system as soon as he understood the machine would create (compose) music, and would not even try it. His credo was that machines should only be used to do things humans could not do. Since the Continuator only generated music from human inputs, he therefore considered the system uninteresting and did not need further inspection. Another one considered *a priori* that the system was not interesting because it

was not able to capture stylistic information when the user plays only one note: indeed, the Continuator analyses transitions between notes or chords, and a one-note sequence does not contain any transition. The remark was true, but the composer refused to forgive this limitation.

Apart from these two negative and ideological reactions, all the other reactions were very positive. The Continuator captures the attention of the audience beyond the traditional “demo effect” of many computer music presentations. In particular, a systematic *Aha* effect was noticed (Pachet 2002b) for professional users as well as beginners. Aha effects have been introduced in experimental psychology to characterize sudden moments of realization, understanding or inspiration. They are also being studied in neuroscience, where evidence of neuronal activity corresponding to Aha effects has been shown (Mogi, 2003).

Experimentation with the system invariably induces users to reflect on their own musical personality (Pachet, 2002c). Bernard Lubat, a Jazz pianist and drummer, at the forefront of progressive Jazz in Europe played with the system many times. He evoked with great precision (in particular during performances at Uzeste festival and during a concert at Ircam in October 2002, see Figure 3) how the system would speed up his own evolution in improvisation, allowing him virtually to “play ahead of his current thinking”. György Kurtag Jr., a composer and improviser, described the Continuator as a kind of “amplifying mirror”, and his regular use of the system throughout the year 2001 changed his way of improvising and composing music (this collaboration resulted in a composition performed at the Vienna Festwochen 2002 music festival, “The Hollow of the Deep Sea Wave »).

Figure 3. Bernard Lubat during his performance. At some point, he “launches” the Continuator with a characteristic gesture. Later, he makes gestures in a “pretend play” mode while listening to the music produced by the Continuator.

Experiments with Children

When the first author started playing with his daughter and the Continuator (she was 3, see Figure 4), her positive reaction was quite surprising and significant. She started to become interested in playing the keyboard – she would laugh at the system's "answers" and was able to focus her attention longer on musical playing.

Figure 4. A 3 year-old child playing with the Continuator at home.

The idea to push the experiment further with additional children was therefore quite natural. In most schools, music is still taught using outdated methods, in which children are confronted with formalisms before they have experienced the enjoyment of playing or listening to music. With a system like the Continuator, basic playing capabilities might be learned more easily, and earlier than with standard music education practice (piano lessons usually start at the age of 6 at the earliest n most conservatories). Most importantly, the Continuator – or in a general a class of systems able to learn and react - could develop a genuine *desire for music* in children, and consequently prepare them for traditional classical training in a more productive way. This need to bring more fun and interactivity in the classroom has long been advocated by various psychological studies (Webster, 2002; Delalande, 1984). More precisely, our vision of education as a pleasing experience falls within the boundaries of studies concerning *Flow* (Csikszentmihalyi, 1990), as discussed in the last section of this paper.

Preliminary experiments with children took place at a French kindergarten (Ecole Bossuet Notre-Dame in Paris). Later, more systematic experiments, involving in particular crossed experimentations, were conducted under the direction of Anna-Rita Addessi (University of Bologna) to investigate the issues brought to light in the first set of experiments further. A detailed analysis of these sessions from the perspective of experimental psychology is under way (Addessi & Pachet, 2004). Only the key observations are mentioned here.

The goal of these experiments was to test basic hypotheses about the effect of the Continuator on the playing abilities of 3 to 4 year-old children. More precisely, the following hypotheses were made:

1 – *Increase in attention span*: Children can play longer with the Continuator than just a keyboard alone.

2 – *Aha and surprise*: The Continuator produces noticeable Aha effects with children, as is the case with professional musicians (note that children rarely find things surprising, a flying pianoforte would probably not surprise them any more…).
3 – *Autonomy*: The Continuator motivates children to play music. For instance, a child, knowing that the Continuator is there, may express the intention of playing alone more often than with standard musical instruments.
4 – *Exploration and playing modes*: The Continuator pushes children to explore new playing modes. The results of the experiments with a normal piano suggested that children would usually stick to single playing modes, including playing with only one finger, playing clusters with two hands, playing ascending or descending diatonic scales (i.e. white keys), etc.
5 – The Continuator can develop various kinds of *attachment behaviors* in children, similar to what has been observed with the Tamagotchi or Aibo (Kaplan, 2001).

These experiments consisted of several sessions. Each child (3 to 5-year-old) was invited to play with a keyboard (a Korg Karma with piano sounds and no additional effect connected to a pair of amplified loudspeakers). The Continuator was set to a mode that played phrases in the same style as the child, and of approximately the same number of notes as the input phrase. The threshold for triggering the continuation was set to about 500 ms.

The child was left alone with experimenters in a familiar classroom. The protocol consisted in two phases: The child would first be told to play with the keyboard as he/she wanted, with no particular instruction. When the child stopped playing or express significant boredom, he/she would be told that the system would now try to play with him/her. At that point, the Continuator would be turned on. The session stopped when the child stopped playing. Several sessions involved also two or more children. Children were brought in one at a time by their teacher. The sessions were recorded with a video camera. A certain number of interesting points were observed:

- Aha effects were indeed produced and noticeable. Reactions of the children ranged from enhanced attention to surprise when the Continuator started to play initially.
- The Continuator did seem to augment the attention span of most of the children. On two occasions, I had the session had to be terminated because the duration exceeded 40 minutes. There were two exceptions: one child was very tired and played very little, with or without the Continuator. Another child seemed to enjoy playing the keyboard enormously with or without the Continuator.
- The children who engaged in long interactions (one of them played 30 minutes and had to be stopped) also appeared to develop the ability to listen with great attention.
- The interaction mode of the Continuator (stopping when the child plays, and playing when the child stops) induced “turn-taking” behavior from the children, without explicit directions. The emergence of turn taking in such a context is not a trivial phenomenon, as shown for instance by the works on complex dynamic systems by Ikegami (Iizuka and Ikegami, 2002).
- Some children exhibited a wide variety of playing modes. Apart from the classical playing modes (e.g. playing isolated notes, chords, arpeggios) the 1children invented new playing modes such as playing with the sleeves, kissing the keyboard, playing from behind, playing with the palm, etc. Some of them also theatrically accompanied the launching of the Continuator with particular gestures like raising hands at the end of a musical phrase (see Section 4.5.3).

The Continuator as a Flow machine

The Continuator appeared in fine as a machine that promotes musical enjoyment in various forms. The system's ability to maintain children's attention for long periods of time — remarkable for this age group — and in general its ability to attract and hold the attention of users of all ages can be interpreted through the theory of Flow introduced by psychologist Mihaly Csikszentmihalyi (Csikszentmihalyi, 1990). Csikszentmihalyi's notion of Flow describes the so-called optimal experience as a situation in which people obtain an ideal balance between skills and challenges. Two emotional states of mind are particularly stressed in this theory: anxiety, obtained when skills are clearly below the level needed for the challenge, and boredom, when the challenges are too easy for the skill level. In the middle lies Flow. Other states can also be described in terms of balance between skills and challenges (see Figure 5). We can think of the Continuator as a Flow machine in the sense that it produces by definition a response corresponding to the skill level of the user. This approach also allows for the progressive scaffolding of complexity in the interaction, which is not the case for most pedagogical tools designed with a fixed pedagogical goal in mind.

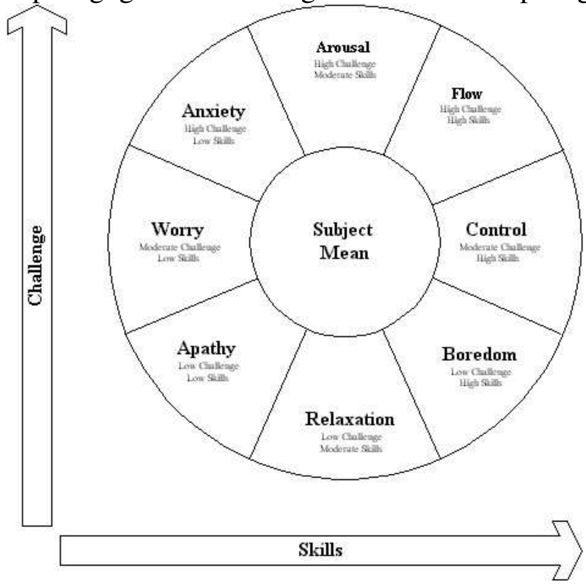

Figure 5. Csikszentmihalyi's Flow diagram describes various emotional states such as boredom or anxiety according to the relation between skills and challenges for a given activity (From Delle Fave, this volume)

More precisely, Csikszentmihalyi describes the state of Flow as consisting of several fundamental traits where the balance between challenges and skills is probably the most important. The other traits can be discussed in light of the experiments conducted:

Focused attention. The experiments show clearly that children are engaged in focused activity both when both playing and listening. The ability to listen and concentrate for several seconds and listen to music is remarkable at this age. As pointed out by Carla Rinaldi (Rinaldi, 2003), listening is probably one of the most important abilities of children to discover the world around them. In her view, children are researchers constantly making up theories about the world and evaluating them. With the Continuator, we observed this phenomenon rather systematically, i.e. children engaging in deep, concentrated listening of the effect of their playing on the system. Figure 7 shows several examples of this phenomenon.

Another typical situation encountered in sessions involving two children was the phenomenon of joint attention. More precisely, one of the children would force the other to stop playing to listen in order to the system. This situation, which we call “aspetta” (the Italian word for “wait”), is illustrated in Figure 8. Of course this behavior was not observed with professional musicians, who so far experimented with the system alone.

Ease of concentration. This is particularly clear given the fact that no instruction is given to the children whatsoever. They play with the system in a self-motivated way, without any external constraints.

Clear-cut feedback. The Continuator produces clear feedback (in fact this is the only thing it does). The interaction in some sense is reduced to the analysis of the feedback produced by the machine.

Control of the situation. Children are in control of the situation most of the time. They understand quickly that they can interrupt the system whenever they want. The limitations in control are due to the difficulties that may arise when interpreting some of the system’s outputs (see example in the next section).

Intrinsic motivation. The most striking result of the experiments is related to the intrinsic motivation of the children, who were not told anything about the rules of the system.

Excitement. Excitement is clearly shown most of the time in particular in the early phases of the sessions. We separate here excitement from surprise in the sense that the surprise effect is most often short in duration, whereas the excitement phase lasted much longer, sometimes for 20 minutes or more. Excitement was observed in most of the cases. Interestingly, the children were excited mostly by what the system was playing, rather than by what they were doing. Figure 6 shows some expressions of this excitement.

Change in the perception of time and speed. A systematic study concerning interaction times is under way, but it is clear already that at least for some of the children time did pass very quickly: some sessions had to be terminated by the experimenters when the time limit was reached.

There is, however, one Flow characteristic that does not apply directly to the Continuator experiments:

Clear goals. No goal was given explicitly to the children except to play until they were bored. Indeed, improvisation is generally not goal-oriented. Similarly, web sailing is usually non goal-oriented but many claim that it can nevertheless create Flow (King, 2003). More importantly, It can be argued that children did create spontaneously goals during their interaction. For instance, several sessions involved children trying to push the system to replicate a particular frantic musical style that they had played some minutes ago.

Figure 6. Various expressions of musical excitement. Excitement is mostly provoked by listening to the system, rather than by actually producing music.

Figure 7. Various gestures showing listening and concentration.

Figure 8. « Aspetta »: when one child forces the other to stop in order to listen to the machine.

Phases to Reach Flow

In order to precisely define the role and importance of the Continuator in developing musical abilities, the phenomena described above deserve careful analysis on an individual basis (e.g. attention span, listening and concentrating, etc.). However, It is also worth studying the "life cycle" of these phenomena, or in other words, the progression of these Flow traits over time. We have observed that they always progress in the same order (see Figure 9).

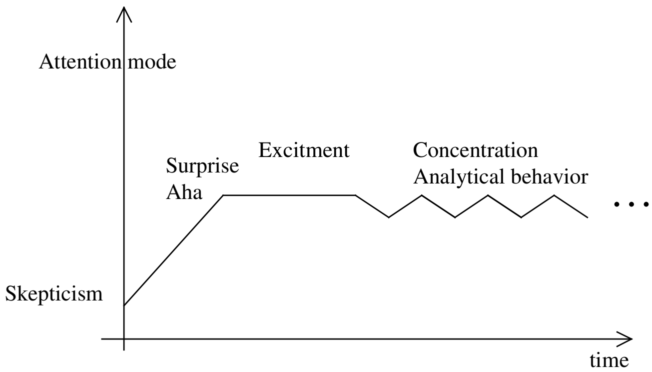

Figure 9. A tentative sketch of the "life cycle" of the interaction mode with the Continuator.

One of the most interesting aspect of this time line is the role of Surprise/Aha: this occurs only once, at the beginning of sessions, or shortly after, and it may be argued that these Aha constitute a sort of *initiatic* experience, acting as a phase transition: once this step is performed, the users change completely their attitude, both towards the system and towards the environment (the experimenters). They become at once involved and self interested.

Observations Concerning Interaction Modes

During the analytical stage (i.e. after the initial periods of surprise and excitement), many other behaviors were also observed with clear indicators. For instance, several children developed spontaneously innovative playing modes. Besides rediscovering standard playing modes such as playing individual notes, arpeggios or chords, they would produce sometimes remarkable arpeggios and clusters, but also new modes such as playing with elbows, turned around with hands in back, kissing the keys with their lips, etc.).

An interesting phase in the interaction occurs when children, after having mastered the basics of the system, somehow abstract the concept of a "musical phrase". This is indicated by a typical gesture demonstrating a pretend "launching" of their own musical phrase (as if it were a golf ball). This gesture determines the end of their musical phrase and also creates the expectation of the system's response. It is remarkably similar to what professionals do themselves (see Figure 3, Bernard Lubat performing at Ircam and Figure 10). This gesture can also be interpreted as a desire to pretend-play, as described e.g. by (Wynants, 2004).

Figure 10. Children raising their hands and virtually « launching » the Continuator, after finishing a musical phrase.

Another phenomenon worth mentioning – and worth studying more in depth – is the children's ability to engage in turn-taking behaviors in a spontaneous way. As mentioned before, the "rule of the game" (in this case a particular kind of master/slave turn-taking) was not explained to the children. However, they learned it, somehow implicitly, extremely quickly, after a few interactions.

This ability to learn the rule of turn making is nontrivial. In particular, a child who knows how it works must also have an understanding of beginnings and endings of musical phrases: of his, as well as the system's. A few cases showed that this skill in ascribing an intentional ending to phrases generated by the system develops very early in the sessions. Figure 11 shows a typical scenario in which a child is confronted with an interesting situation regarding phrase endings. In this case, a particular continuation produced by the system started with the ending of the child's phrase followed by additional notes. This situation, although rare, can occur because the Continuator has no particular notion of beginnings or endings. Figure 12 shows the situation graphically. The phrase played by the child is schematically represented by two consecutive chunks of a few seconds. The last chunk is repeated by the Continuator, and followed by another one.

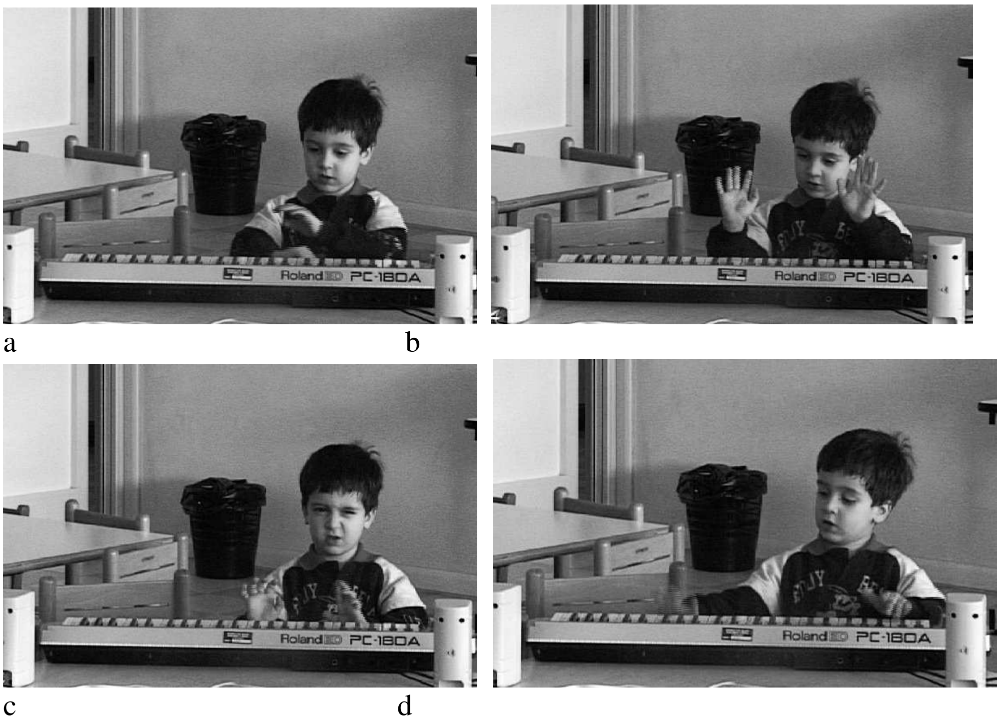

Figure 11. A child clearly showing its interpretation of the Continuator's phrase endings. (a) shows him just finishing a phrase and waiting for the answer. The answer generated by the Continuator turns out to contain the last part of his input phrase followed by some additional notes. At the end of the first part of the continuation (i.e. the repetition of the child's phrase ending) the child gets ready to play again, assuming that the phrase played by the Continuator is finished. It is not (c), and the child shows his misunderstanding with a facial expression. Eventually (d), the Continuator ends his phrase, and the child resumes playing.

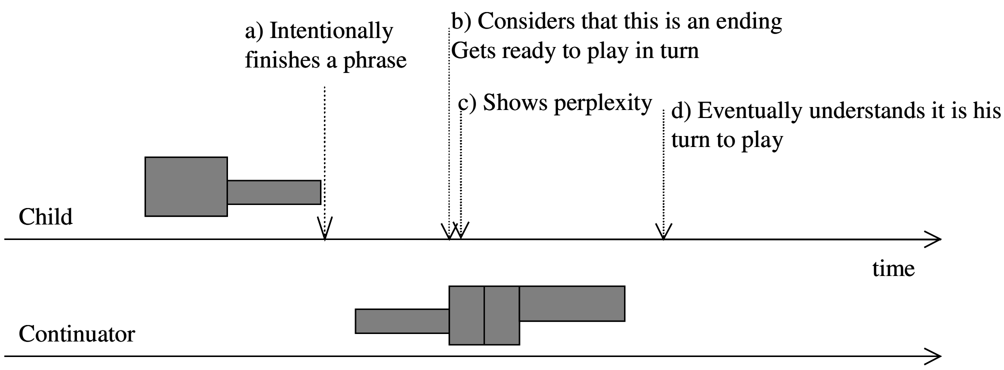

Figure 12. A short scenario demonstrating the complexity of the interpretation of phrase endings. The phrase structure is represented schematically using rectangle of various shapes. The important aspect of the scenario is that the last chunk of the input phrase is the first one of the continuation, thereby creating a false sense of ending for the child.

4. Designing Interactive Systems

The experiments conducted with the Continuator are interesting because they teach us something about the nature of human behavior, and in particular about what human beings find exciting. They also give us precious hints for the design of artificial systems that help to generate flow experiences. I now highlight two key ideas which appear crucial for the success of the Continuator: reflection and emergent interaction protocols.

Reflection in Interactive Systems

A lot of research has been conducted into the properties that make interactive software usable. In the field of music, Sidney Fels (2002) analysed several interactive systems designed for creating novel aesthetic experiences and found that an ideal interactive system requires intimacy. This is caused by the sensation that the system is somehow an extension of the user's body, thus ensuring the efficiency of the user's actions with the instrument.

The Continuator has this property, but the presence of a learning component introduces another dimension of interaction. Because the system is able to learn and imitate the user's musical personality and style, the Continuator acts as a dynamic mirror, and I claim that most of the interesting behaviours observed in the experiments stem from this feature.
This reflective capability brings a number of crucial characteristics:

- *Similarity or Mirroring effect*. What the system produces "looks" like what the user herself is able to produce.
- *Agnosticity*. The system's ability to reproduce the user's personality is learned automatically and agnostically i.e. without human intervention. In our case for instance, no pre-programmed musical information is given to the system whatsoever.
- *Incrementality*. Interactive systems are not designed only for short demos. Because the user is constantly interpreting the output of the system, and altering his playing in response, it is important to consider the longer term behaviour of the system. Incremental learning ensures that the system keeps evolving all the time. Each interaction with the system contributes to changing its future behavior. Incremental learning is a way to endow the system with an organic feel, typical of open, natural systems (as opposed to pre-programmed, closed-world systems). [A preliminary version of the Continuator used an algorithm based on Lempel-Ziv for learning. The Lempel-Ziv algorithm is not complete: several learning steps are needed to ensure that all the information contained in a new sequence is actually learned. As a consequence, the system would be less faithful (would learn more slowly), but this very slowness increased the organic feeling of teaching something, which is less the case in the current version (which learns at once all the information contained in each new sequence): paradoxically, to ensure the incrementality, less complete algorithms may be more efficient than complete ones.]
- *Seamlessness*. The system produces material that is virtually indistinguishable from the user's input. Note that this characteristic does not apply in the case of "classic" hyper-instruments, where the sonic effects are entirely produced through the system, and therefore not resulting in any material directly produced by humans.

One important consequence of reflection is that the "user's center of attention" is not directed towards the *end-product* (the music), but towards the *subject* engaged in the interaction. Engaging in an interaction with such a system is therefore a means of discovering oneself, or at least exploring one's ability in the domain at hand (in our case, musical improvisation). This natural, deep interest in exploring oneself - particularly during the early childhood years of childhood - may explain why the Continuator is so appealing and we can design probably other successful interactive systems using the same principles. In some sense, these systems are an extension of the "second self" (Turkle, 1984). Not only does the machine seem to "think," but it thinks like the user. An interesting consequence of this is the reversal of roles: the student becomes the teacher who teaches the machine about himself.

Emergent Interaction Protocols

The experiments have also shown various limitations of the current design of the Continuator. One limitation concerns the need for another very general principle, namely that interaction protocols need to be flexible and emergent. Interaction protocols specify the rules of the game. They determine how and when the system decides to play. Like in conversations, these rules can vary, and a simple question/answering sequence is by far not the only possible interaction protocol: lectures, smalltalk (in the common sense meaning), exams, group conversations, baby talk, etc. are different situations in which interaction rules differ greatly.

The issue of interaction protocols is closely related to the idea of music as a conversation, put forward for instance by Bill Walker in his ImprovisationBuilder system (Walker, 1997; Walker & Belet, 1999). The ImprovisationBuilder is able to take turns with the player, and also to detect, in the case of collaborative music playing, whose turn it is, using a simple analysis of the various musician's inputs. However, this system as well as all the music interaction systems so far, are based on fixed interaction protocols. The question here is whether it is possible to find mechanisms so that the machine adapts protocols automatically and finds the best protocol suited to the context.

Currently, several interaction protocols were designed and experimentally tested with the Continuator. Here are some of them, by increasing order of complexity (and represented graphically in Figure 13).

- *Turn-taking*. This mode is represented graphically as a perfect succession of turns, with no gap. The Continuator detects phrase endings, then learns and produces a continuation. It stops as soon as the user starts to play a new phrase.
- *Turn taking with delay*. The same as above, except that the Continuator stops only when the user *finishes* a phrases. This produces an interesting overlapping effect in which the user and the Continuator can play at the same time.
- *Single note accompaniment*. The Continuator produces an appropriate chordal accompaniment each time a note is played, and with the same duration. It stops the chord when the key is released.
- *Phrase-based accompaniment*. The same as above except that the chord is produced only at the beginning of a phrase.
- *Collaborative*. In this mode, the Continuator plays an infinite stream of music (based on material previously learned). The user can play simultaneously, and what he/she plays is taken into account by the Continuator, e.g. harmonically. The user's actions are here like high-level control, instead of a question to be answered.

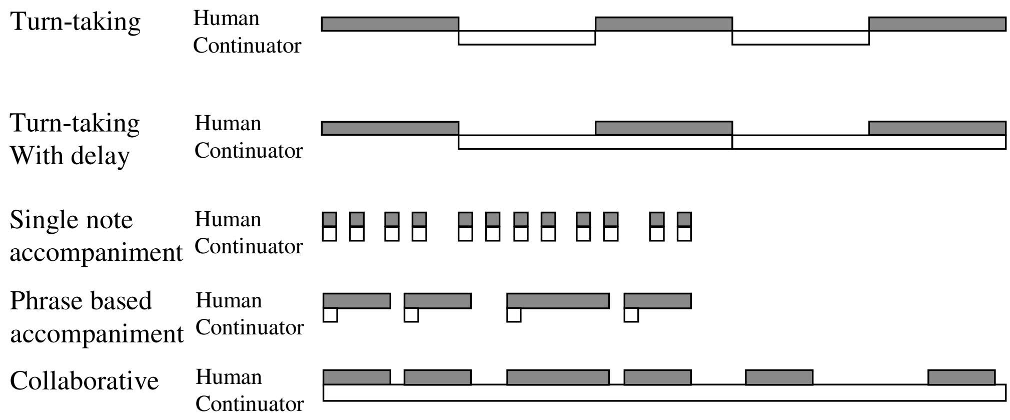

Figure 13. Various interaction protocols with the Continuator.

These various modes are in turn highly parameterised: The phrase length of the continuation in turn taking mode, the rhythm mode, the adaptation or not of the music produced to surface parameters such as dynamics, tempo, etc. In practice, it is easy to see that an infinite number of concrete interaction protocols could be defined, all tailored to a particular situation.

The main problem with these protocols is that they are fixed, therefore sometimes not appropriate. Several observations (with children and professionals) showed that this rigidity is a severe limitation:

- For instance, a child enters the room and remains seated looking at the keyboard. Nothing happens because the Continuator waits for a phrase before it starts playing itself. In this case, an initiative by the system so as to trigger an interaction would be more relevant than just waiting.
- During an interaction, the parameters of a chosen protocol may be unadapted. For instance, we have observed that short answers in early stages of the interaction are preferable, but longer ones are more adapted as the user gets involved in the playing. Conversely sometimes a continuation which is too long can confuse the user (as was the case with one of the children who did not know how to stop the system).
- During a session, different modes may be needed. For instance, while the turn-taking mode appears to be very natural at the beginning, several users (children and professionals) expressed the desire to play at the same time as the system. This requires indeed an explicit change of interaction mode (e.g. with a button), which breaks the fluidity of the interaction, and hence breaks the flow experience.
- Parameters other than the simple musical inputs should also be taken into account to influence the interaction protocol. The child's face in Figure 11 is a clear indication that the interaction was not as expected.

The need for the system to control its own interaction protocol is therefore crucial to enhance its capacity to adapt. More generally, a fully flexible system should be able, like humans, to infer on-the-fly the protocol induced by a particular situation, and follow it, as well as invent itself new protocols: a good music teacher would for instance switch from turn taking to, say, a teaching mode, and back to an accompaniment mode, etc.

To achieve such a control we need to make the system somehow *self-reflective*, i.e. able to manipulate explicitly the interaction protocol at a meta-level: select a protocol, create one, or infer one from the context. Such an endeavour is usually referred to as *reflection* in the world of language design, object-oriented in particular (Demers and Malenfant, 1995).

Building systems with such an architecture is not so simple. Successful attempts at designing self-reflective systems have usually been restricted to so-called structural reflection, i.e. the ability of a program to reason about its data structures. Conversely, systems able to switch between protocols (e.g. for mobile services) are able to do so only within a fixed list of predefined protocols.

To extend flexibility in protocols requires the representation of the intentionality in interactions. The speech act theory proposed by Searle (1969) allows to represent utterances as so-called illocutionary acts, such as stating, requesting, commanding, and so forth. Every speech act consists of an illocutionary Force F applied to a proposition P. (Covington, 1998). This theory has given birth to a number of languages such as KQML (Knowledge Query and Manipulation Language), which are used for specifying various interaction protocols between agents, notably for electronic commerce (Finin et al, 1994). However, speech act theory and KQML are primarily aimed at facilitating knowledge sharing and transfer, in contexts where the set of possible illocutionary forces is known a priori. A language such as KQML may be used to specify each interaction between the machine and the user. But KQML does not solve (or even address) the motivational issue nor the problem of interpreting the “messages”, or inferring optimal protocols.

5. Conclusion

We have introduced the bold question of what makes an interactive system appealing, interesting and attention grabbing, taking example from the experiments performed with the Continuator system, a musical interactive system. We have shown that the Continuator may be seen as a particular example of a Flow machine, i.e. a machine which generates flow-like experiences in users. The reflection principle, which causes a mirroring of user’s behavior after a learning process, appears to be of critical importance.

Current limitations concern primarily the lack of flexibility in interaction protocols. The Continuator proposes many different interaction modes, but is not able to detect automatically the best mode, nor to create new ones on-the-fly for adapting to unexpected situations. This is clearly an importan research line for the future which can build further on a long history of research in computational reflection. Reflection for building Reflective Systems: although it seems a play on words, I believe this is not entirely coincidental.

References

Addessi, A.-R. & Pachet, F. (2004) "Musical style replication in 3/5 years old children: experiments and analysis", submitted to British Journal of Music Education.

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

Chuang, John. (1995). Mozart's Musikalisches Würfelspiel, web site: http://sunsite.univie.ac.at/Mozart/dice/

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

Covington, M.A. (1998) "Speech acts, electronic commerce, and KQML," *Decision Support Systems* volume 22.

Csikszentmihalyi, Mihaly. (1990) Flow. The Psychology of Optimal Experience. Harper and Row, New York.

Cope, David. (1996). Experiments in Musical Intelligence Madison, WI: A-R Editions.
Delalande F. (1984), La musique est un jeu d'enfant, Buchet/Chastel, Paris.

Demers, François-Nicolas and Malenfant, Jacques (1995). Reflection in logic, *functional and object-oriented programming: a short comparative study*. In IJCAI '95 Workshop on Reflection and Metalevel Architectures and their Applications in AI, pages 29--38.

Fels, S., (2000). Intimacy and Embodiment: Implications for Art and Technology**,** Proceedings of the ACM Conference on Multimedia, Marina del Sol, CA, pp. 13-16, Oct.

Finin, Tim Fritzson, Richard McKay, Don McEntire, Robin (1994). KQML as an Agent Communication Language Proceedings of the 3rd International Conference on Information and Knowledge Management (CIKM'94).

Iizuka, Hiroyuki and Ikegami, Takashi. (2002) *Simulating Turn-Taking Behaviours with Coupled Dynamical Recognizers*, The 8th International Conference on the Simulation and Synthesis of Living Systems, University of New South Wales, Sydney, NSW, Australia 9th-13th December.

Ingalls, Daniel (1981). Design Principles Behind Smalltalk, BYTE August 1981 Special Issue on Smalltalk.

Kaplan, Frédéric. (2001). Artificial Attachment: Will a robot ever pass Ainsworth's Strange Situation Test?. in Hashimoto, S., editor, Proceedings of Humanoids 2001: IEEE-RAS International Conference on Humanoid Robots, pages 125-132.

Kay, Stephen. (2000). The Korg Karma music work station, Korg Inc.

King, Andrew (2003). Speed Up Your Site: Web Site Optimization. Chapter 2: Flow in Web Design.

Mogi, ken. (2003). Onceness, in *The Future of Learning*, Tokoro, M. and Steels, L. Eds, IOS Press.

Pachet, Francois. (2002a) The Continuator: Musical Interaction with Style. In ICMA, editor, Proceedings of ICMC, pages 211-218, September 2002. ICMA. best paper award. Extended version in Journal of New Music Research, 31(4), 2003.

Pachet, Francois (2002b) Interacting with a musical learning system: the Continuator. In C. Anagnostopoulou, M. Ferrand, A. Smaill, editor, Music and Artificial Intelligence, Lecture Notes in Artificial Intelligence (vol. 2445), pages 119-132, Springer Verlag. September 2002.

Pachet, François. (2002c) Playing with virtual musicians: the Continuator in practice, IEEE Multimedia, April-June 2002.

Rinaldi, Carla (2003) The Joys of Preschool Learning, in the *Future of Learning: Issues and Prospects*, Tokoro, M. and Steels, L. Eds, IOS Press

Risset, Jean-Claude and Van Duyne, Scott. (1996). Real-time Performance Interaction with a Computer-Controlled Acoustic Piano, CMJ 20(1), pp. 62-75.

Searle, J. R. (1969), *Speech Acts. An Essay in the Philosophy of Language.* Cambridge.

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

Steels, L. (2004) Autotelic Agents: Robots that have fun. In: Hafner, V., F. Iida, R. Pfeifer and L. Steels (eds.) (2004) Embodied AI. Proceedings of the Dagstuhl Seminar. Lecture Notes in Computer Science. Springer Verlag, Berlin.

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

Turkle, S. (1984). The second self: Computers and the human spirit. New York: Simon and Schuster.

Walker, W. (1997) *A computer participant in musical improvisation*. In CHI 97 Electronic Publications.

Walker, W. Belet, B. (1999) Applying ImprovisationBuilder to interactive composition with Midi piano. In ICMC 99, Hong-Kong.

Webster, Peter R. (2002). Computer-Based Technology and Music Teaching and Learning, in R.Cowell-C. Richardson (EDS). The New Handbook of Research on Music Teaching and Learning, Oxford University Press, pp. 416-439.

Wynants, M. (2004). Pretend Play as Learning: Case Studies from the Home. This volume

FRANCOIS PACHET is a civil engineer, with a PhD in artificial intelligence from University of Paris 6. He has been assistant professor at Paris University for 5 years, and now leads a research team at Sony Computer Science Laboratories in Paris. His main research interest is the notion of "Chrono-attention": the ability of a temporal process to attract and sustai attention in time. He investigates what makes temporal forms of entertainment (music, games, movies) interesting, appealing, attention grabbing and sustaining. Francois Pachet has led several projects dealing with particular Facets of chrono-attention from a system design perspective (how to design systems which are chrono-attentive): MusicSpace (how to make interactive music listening attractive through spatialization), Continuator (how to interact and generate interesting musical dialogues and interactions), the Popular Music Browser (how to provoke serendipity and A-ha! effects in music catalogues browsing). His recent interests focus on chrono-attention from a psychological perspective (how does our attention system works), and educational perspective (how to exploit chrono-attention in teaching).