Genetic Improvisation Model

a framework for real-time performance environments

Paul Nemirovsky, Richard Watson

MIT Media Laboratory, 20 Ames St., Cambridge, MA, USA 02139
{pauln, watsonr}@media.mit.edu

This paper presents the current state in an ongoing development of the Genetic Improvisation Model (GIM): a framework for the design of real-time improvisational systems. The aesthetic rationale for the model is presented, followed by a discussion of its general principles. A discussion of the Emonic Environment, a networked system for audiovisual creation built on GIM's principles, follows.

Introduction

Numerous tools for improvisation have appeared over the last few years – increasing numbers of researchers are realizing the value in the improvisational approach to creativity. Improvisational methods have been successfully employed by human performers for thousands of years; yet, until recently, these methods have remained untouched by most engineers and technical artists to whom they would seem relevant. Projects such as Voyager [1], Galapagos [2], ChaOs [3], Swarm Music [4], and others, while not always being explicitly improvisational, explore media-informational spaces in nearly-improvisational ways, allowing the user to modify the exploratory process in the course of exploration. In our work we attempt to expand this research direction, trying to create a playground for building non-idiomatic[1] improvisational environments.

Non-idiomatic improvisation has always been considered a kind of process that doesn't lend itself to modeling; by its nature it seeks to defy any fixed model as a complete description of its behavior. A good improviser is one who is able to simultaneously avoid algorithmic description, always surprising his audience, and appeal to some aesthetic criteria (local or global[2]) in shaping his performance.

In improvisational contexts, it is common practice to describe the characteristics and success of a performance in terms of global aesthetic criteria; one of the most intuitive and widely used has been termed *energy*. Why is it a useful concept? First, energy is a popular analogy with improvising[3] musicians and artists; many of them regard it as a convenient metaphor to think about one's improvisational talents ("Zorn has a great energy"). Second, speaking in terms of energy enables us to think about

1 Not following one fixed aesthetic idiom, such as a particular music style.

2 Here, *global criteria* define what constitutes an overall good improvisation, while *local criteria* are those of an always changing context.

3 From here on *improvisation* & any derivative words refer to the non-idiomatic improvisation.

changes in behavior over time, regarding them in the context of an overall flow rather than as a composition of discrete actions.

Local aesthetic criteria on the other hand are dependent on an endless range of social, cultural, and technical circumstances. No model of improvisation should seek to define these criteria decisively. Models that do end up capturing a specific algorithmic behavior rather than the 'essence' of improvisation: its ability to always generate new criteria and mix the idiomatically unmixable, resulting in a never-ending exploratory process. We therefore strive to design improvisational models that allow both explicit as well as implicit criteria[4], with the latter emergent from the interaction of system agents, both autonomous as well as reactant to the human performer.

The GIM presents a foundation for constructing a vehicle for improvisational exploration. Such vehicle may exercise various degrees of autonomy. Similar to its physical counterpart, a GIM-based system could take the performer 'around the city' all on its own. Our research, however, is directed toward exploring situations of co-improvisation, where the performer and the system are symbiotically contributing to a shared performance. In one conception, the performer's role is to provide a general evaluation of the vehicle's activities and express high-level desires to be carried out by the vehicle. Such a vehicle is neither an autonomous improviser nor a passive assistant. In this model, a lower degree of precise control available to the human performer (in comparison to one who uses traditional instruments) is compensated by a lower responsibility for the output. As the system constantly produces new and modifies old materials and structures, the performer has no assurance as to the output yet is integrally involved in evaluating, transforming, and exchanging it with others. We foresee such co-improvisational approaches becoming increasingly manifest in computational tools for all walks of art making, information exchange, education, and entertainment.

How can the relationship between different elements in a co-improvisational system as well as the relationship between the system and its users be modeled? We found a directed graph (or *network*) representation most useful. In this conception, we do not represent a one-to-one relationship between input and output, but instead model associability of elements within an environment[5]. In this depiction, changing one element or connection affects everything else.

A second question that immediately arises is how to uncover configurations of the network that lead first to interesting connectivity and ultimately to compelling new behavioral modes or trends. Here, we argue that genetic algorithms are uniquely fit to meet this challenge, offering a largely unconstrained ('blind') evaluative process, which, when combined with the mutable non-linear structure of the network, provide a fertile ground for interesting emergent behavior. We discuss these issues further in the first section where we outline the GIM.

The only way to test the assumptions and predictions made in the GIM is by developing a system that implements it. A further question then becomes: what type of evaluation procedures are we to follow in deciding whether the system is taking part in a co-improvisational process or only behaving randomly? To address this question, the second section is dedicated to elaborating a number of guidelines that a system must follow to be considered as GIM-compliant.

---

[4] Meant here as a set of principles observed by a performer in creation / evaluation of aesthetic experience; a style of behavior in matters of artistic beauty and taste.

[5] Here and throughout the paper, *environment* is meant as a workspace for creative activity.

Finally, in the third section, functional guidelines in hand, we discuss our implementation, the Emonic Environment (EE). Here we evaluate just how well we fulfill the principles set forth by the GIM and report on the current development state.

I. GIM Motivations

Motive for the Architecture

It seems to us that the development of particularly interesting computational models of improvisation is hindered largely by adherence to (a) Western ideas of music and art (as a disassociated sphere of "fine art", guided by a set of immovable principles such as tonality, melodic principles, proportion, etc) and (b) the predominant computer science paradigm which concerns itself with predefined operational rules or parameters (what Koza [5] calls the strive for simplicity, convergence, conciseness).

In opposition to the school of thinking that studies channels of human perception and activity in isolation, we believe, drawing our inspiration from Varela [6], Claxton [7], and others that any creative activity is ultimately a social act, tied to the world around and within the improviser. As an implication of that view, creative processes, whatever their output mediums might be, cannot be seen as conceived solely in a particular medium; a musician improvising with sounds employs a much wider non-sonic array of procedural and perceptual memories – images, smells, sounds, etc., tied together and shaped in their perception by the social world of the improviser as well as the external stimuli being perceived.

As a performer improvises, he thinks up new 'things to say'. In order to investigate his thought process, we must ask two important questions: (1) what kind of representation does the improviser employ and (2) how do all the disparate 'things' become unified into one improvisational experience? It has been argued (e.g. in Marcus [8]) that the representations are never distinctly high- or low- level, but rather always a combination. We have adopted this conception as it makes sense and suits our model well. We have named the special combination of levels of representation a *mediated layer*, borrowing from Stafford's [9] concept of a mediating image.[6] Through forming connections in the mediated layer, new meaning is then made.

Creative thoughts rarely come solely in a form of "play a C4, then an E3 flat, then double each note produced with a bass that is always on the distance of a triton from the upper voice". Certainly, such procedural thinking is useful; if we were to never think in precise terms of melody and harmony (or point-of-view and principles of montage in video), we wouldn't be able to produce many sounds or images; our creative drive would never be realized as it would remain solely conceptual.

What such literal thinking lacks, however, is flexibility and abstraction. For in this paradigm, how can we define an algorithm for generating 'cool' sound, exchanging 'sad' melodies, or applying 'fairytale-like' video effects? We see then, that what we need is something that can mediate between the two layers. We must both be able to make manifest an abstract concept or feeling and at the same time be able to react

[6] Stafford believes that there is always a mediating image between two entities of an analogy; the image becomes the space of interaction / negotiation between the two entities (objects).

to the very surface level perceptual happenings, realizing their importance contextually and fitting them into an overall conceptual framework. The two layers are interlinked: the sounds/images that the improviser produces are influenced by what is taking place around him (other performers, the crowd, sounds, colors, etc.); at the same time the surrounding world (and thus his perception of it) is influenced by his action of producing these media artifacts. We refer to these layers as perceptual and structural respectively.

An improviser continuously cycles between the two, evaluating his own performance and attending and responding to the surrounding context. This relationship of feedback brings about a continuous assemblage of the two layers into a mediated one and the resulting creation of one unified experience. This process of controlling the flow of information can be thought of as controlling the energy of improvisation.

From this discussion, it is evident that both the structural and perceptual representations are essential to our understanding of the improvisational processes. Moreover, a dynamic interaction between the two layers appears to be necessary.

Motive for the Methods of Functional Alteration

Too much potential? A case for evolution

From our description above, it is apparent that any system built on the GIM will be rich in both the number and types of elements it contains as well as in the type of interactions between the elements and the layers to which they belong. A possible problem, then, in adopting this sort of model is that it allows for too many potential states or configurations. That is, in setting up a system to model improvisation, we are forced to relax any constraints on dynamics we might tend to put in place in order to manage the size of the parameter[7] space. Consequently, we will potentially find ourselves drowning in a sea of optional configurations from which we must fish out "just the right" one whose product suits our goals and context. If we spent the rest of our lives trying out all the possible combinations of parameters, we would but just graze the surface of that which is available. Must we conclude, then, that a true model of improvisation is incompatible with an implementation that can use it?

GIM vs. TSP

Those familiar with genetic algorithms know of the traveling salesman (optimization) problem (TSP) [10] – an example that illustrates how GAs may overcome painful levels of combinatorial complexity. Though a GIM parameter space is much larger than that of the TSP, our goal criteria are not as exclusive as those deemed by the salesman. Namely, our goal is not to find *the* best configuration, but to find a configuration whose product pleases us. We may safely presume that several such configurations exist. So, although our parameter space is larger, so too is our solution space[8].

Another feature difference of our problem, which goes hand in hand with that of multiple goal states, is the nature of the fitness criteria. We say TSP is transparent

---

[7] Parameters, here, refer to the specifications of elements, element properties and interactions between elements in the system.

[8] A further interesting topic of investigation is just *how* this solution space grows with the parameter space. For lack of room, we postpone this discussion until a later time.

because we know exactly what it takes for one parameter configuration to be better than another. However that is not the case in the creative paradigm addressed by GIM. Instead, we must explore by trial and error. Our fitness criteria are *implicit* and *user specific*, the exact definitions inaccessible to us and to others.

We can furthermore distinguish the two problems by noticing that not only the criteria change between users but also that they change for a given user, even over the course of a single performance.

Genetic Algorithm for exploration

The distinctions we've drawn out above between our problem and that of the TSP point to a very different picture of the job of a GA in our model. What we've sketched here is not a tool for *optimizing* a set of parameters given fixed goals and constraints toward a 'best' solution. Instead we're employing a GA to suggest possible configurations to the user and traverse the parameter landscape based on feedback from that user regarding those suggestions. We cannot say that this movement on the part of the GA is optimization for three reasons: any 'peaks' in the fitness landscape corresponding to our parameter space are (1) plural, (2) fuzzy and potentially plateau-ed, and (3) dynamic.

Instead, we might say, our GA is being used as a tool for *exploring*. The GA's job is to work with the user to find new directions for movement in the parameter space and pick which shall be followed. Furthermore, there is an inherent feedback process involved. As the GA helps the user to traverse the parameter space, the user generates and refines the criteria by which he is judging the suggestions. The whole procedure leads to a richly dynamic, interesting, and unpredictable improvisational trajectory.

Genetic Advantages

We might imagine that we could adopt other search algorithms for use in conjunction with a system built out of the principles of GIM. It is worth mentioning, then what advantages a GA approach might offer over its competitors.

1. GAs are 'blind'. That is, they do not have a plan for what direction they will search in next. A trajectory is mapped solely in virtue of how well local movements along that trajectory bring about positive (desired) changes. This seems to fit better with the idea of improvisation-as-exploration than might an algorithm that tries to 'figure out' the correct direction to move in.

2. GAs are context sensitive. They take into account a continuously changing context while deciding on new directions of movement within a parameter space. This is something that is 'built in' to the algorithm and thus doesn't require any sort of reconfiguration. It is a characteristic that is well suited to an improvisational setting, which, by definition is responsive to context.

3. GAs can be designed to act solely on the basis of emergent behavior. That is, instead of an algorithm that might consider what changes to any particular parameter might do to the system in virtue of itself alone, GAs may be (and usually are) designed to respond to the overall effect. This is important for improvisers who are concerned with how change of a single element may alter interactions between elements in the totality of the performance.

Principles

To define the principles of the GIM, the following questions must be considered: what is improvisation? What aspects of it does GIM try to model? Who would be the consumer of a system built on the GIM – would he be a composer, performer, audience, or some other new type of participant?

1st: Dynamic nature of improvisational structures. Structural representations used in the course of improvisation are incorporated, modified, and purged dynamically to satisfy the improviser's changing goals and attention. The criteria guiding improviser's behavior (in terms of expectation and evaluation procedures he employs) evolve in the course of a performance; an improviser changes what he considers the "right thing to do" based on the combination of these evolving criteria and the stimuli from his environment that only manifest themselves as the performance develops.
2nd: Changing, multiple-leveled focus. An improviser thinks about what he's doing in many different ways. Continuously switching between macro- and micro-level representations, he attends to the very minute (e.g. a particular sound) at one moment, only to switch and think about general development (e.g. a climax) a second later.
3rd: Diversity of types. Improvisation is a result of interrelating multiple perceptual inputs and memories; an improvisation whose 'output' is audio is nevertheless an improvisation that includes visual, tactile, and other formative content. A sound might be inspired by an image, which in turn is inspired by a text or another sound; this free and proactive interaction of types is integral to the improvisational process.
4th: Relevance of context. Following on the above point, the improviser's decision-making is rooted in the totality of his perception of the moment. Thus medium-specific laws of decision-making should be used cautiously in deciding the subsequent output, for the perception of any media is in itself an act shaped by the context. Indeed, improvisation is not formed in a vacuum or in one medium separated from others; it strives to incorporate or reflect the environment in which it is created.
5th: Process, not artifact production, as the goal. An improviser, unlike a Western composer, feature-film cinematographer, or product designer[9], is not concerned with the production of a final artifact, a sonata, a pop-song, a chair, or a movie. While improvisation might be recorded and as such seen as a fixed construct, the true point of improvisation is the process of exploration, contextualizing and interrelating memories and perceptions[10]. An improviser's job is to weave together an array of 'sketches', which gain their relevance (and meaning) only as the improvisation unfolds.
6th: Absence of a plan. Planning does not seem to be the optimal way to think about the process of improvisational creation. Instead, the act of improvisation is better thought of as one of exploration. Another way of putting it is that an improviser is far less concerned with perfectly playing to a specification than he is with breaking new ground and learning from unintended mistakes and successes.

---

9 This of course is not a binary dichotomy; discussion of near-improvisational compositional movements however is beyond the scope of this paper; see for example Nyman's [12] excellent account of the experimental music movement.

10 This memories-perception inside-outside Descartian dichotomy is questionable to say the least; we use it for the reasons of space, not to endorse it. The idea of the rhizome [13] seems to be much better fit for the description of improvisational processes.

7th: Issues of control/responsibility. In an improvisational performance, no fixed contract specifying responsibilities of control (balance of power) exists between the performers; the criteria that define the degree to which each party assumes creative control over different aspects of the ongoing improvisation are set dynamically, according to implicit and explicit negotiations between the performers. Giving up part of the control also frees the improviser from the preoccupation with creating a perfect finalized product. In other words, improvisation implies a lower cost of experimentation, allowing spontaneous exploration of new, 'unproven', ideas.
8th: Continual feedback. Improvisation is not evaluated at one point in time or space. Over the course of a performance, improvisers provide feedback to each other. This feedback ranges from general and vague to particular and precise; what defines its value is the ability of the recipients to learn from it and move in new directions. The learning is not procedural; it cannot be summarized by a symbolic rule. Instead, it can be described as discovering patterns where one didn't see them before.
9th: Meaning-making through exchange. In an improvisational group action, construction of meaning happens through the exchange of elements. In other words, a sound or an image acquires its meaning only through the details of its history of use – where and how it has been employed before. [11] These details determine how it or similar elements are perceived the next time they are encountered.
10th: Audience as participant. From the passive audience of the linear storytelling to the nearly equally passive audience of the multiple-choice "interactive" environments, a strict giver / taker dichotomy has been enforced between the consumer (the audience) and the producer (the performer)[11]. In improvisation however such distinction is obsolete; anyone can co-improvise, so long as the effect of his activity is heard / seen in one way or another by the other performers. Similarly, even when not actively participating in the act of media creation, the audience is not regarded as passive; it is viewed as a part of the improvisational circle.

II. From Principles to Implementation Guidelines

From the above principles may be extracted a set of guidelines that a system based on the GIM should follow. The guidelines enumerated below are directly derived from and continuous with the list above.

1. The structure of the system must be mutable. It must allow for the addition and removal of modular substructures within the system. The system must also have an alteration process that has continual and consistent access to this structure. Furthermore, the system must be able to accommodate a performer's dynamically changing goals and expressive behavior.
2. The system must be accessible on many levels. Performers must have the option of changing very minute details of particular elements in a performance media stream; at the same time, they must be able to have control of persistent abstract patterns.
3. The system must handle a diverse range of media types, both as input and as elements within the system.

11 The few fortunate exceptions from this rule do not do much more than recombine already existent materials without modifying any of their properties.

4. These diverse types of input must be able to 'speak' to one another. That is, the system must have ways to connect various types of media, whether directly through the operators that output the media or through indirect structural relations.
5. The system must have focus not on constructing a polished 'work' but on the process during a performance. This is not to say that an appreciable sequence (that might recur) may not come out of the process, but only that the system must not focus on producing the output as a single whole.
6. Unpredictability must permeate the system; nothing should be 'laid out'. Not only should it be unforeseen where the system will be at the end of a performance but also (to one extent or another) where it will be in the next few seconds. This fits with the idea that improvising involves taking risks.
7. The system should have a configurable 'autonomy level' which determines to what degree it is running independently; specification of what and when the system controls should be open to runtime modification.
8. An ongoing evaluation process from performer to system (and maybe from system to performer) should be in place. Feedback in this process must have the potential to be both specific to particular areas and general, expressing views on the performance of the system as a whole.
9. The system must have the capability to exchange media and system structures with other users (on separate instances of the system), and to employ these shared elements in various ways.
10. Methods of sensing the audience (e.g. using cameras or microphones) should be developed for and within the system.

III. The Emonic Environment

The Emonic Environment (EE) is our attempt at building a system according to the GIM-based guidelines. Using the EE, performers are able to create, modify, and exchange audiovisual content.

The EE is designed with a multi-layered architecture, not unlike an animal nervous system. Interfaces to the system from outside environments are situated in the Input layer, on the bottom. In the middle, the 'brains' of the system exist as a neural network in the Structural layer. Finally, on top, media events are scheduled and processed by a population of media operators known as *emons* which are bonded to one another through a web of connections in the Perceptual layer. This layer is analogous to the motor system of a mobile organism.

This layered design addresses the need for improvisers to access the system on many levels. It affords performers a capability to dynamically refocus their objectives while creating and manipulating media in realtime. The layered design also modularizes the system making it easier to understand from the inside and easier to expand.

Joints between the layers serve to enable flow from outside world influences to inner dynamic structures and finally to media operators which reconnect the system to the outside world. Links within the layers enable richly interactive (largely unpredictable) emergent behavior to arise.

The EE is written in Java and uses various third party components including JSyn for sound synthesis, playback and processing; JMSL for event timing; and custom libraries for audio processing developed in the MIT Media Lab.

Layer Details

The Input layer of the EE consists of tangible and software interfaces that sample information from the outside world (e.g. computer mouse, video camera, gesture controller). The aim of the Input layer is to integrate the world around a performer into the ongoing improvisation. Input received from each interface is used to control functional elements in one or many of the three layers using *transforms*, which map between a particular device's data format and that of a particular element. One category of transform, for instance, maps from a custom gestural controller (called the Emonator) which outputs arrays of 144 short values to a one-dimensional double value. Over time this produces a signal that can be routed to emons on the Perceptual layer to modulate properties such as frequency or amplitude.

The Structural layer is a recurrent neural network, populated with nodes and weighted connections (or *associations*). The network and its elements individually each have a number of properties which can be controlled either explicitly (by the performer) or by a system administered process:

Association: Path, Weight, Time Delay (inner-node stimulus travel time)

Node: Activity, Decay Rate (of activity), Propagation Threshold (above which the node propagates any incoming stimuli), In/Out Stimulus Scalar

Network: Max Simulation Propagations, Auto-Activation Threshold (under which new spontaneous activity will be introduced), Auto Management Features (when operating without any performer input; not yet implemented).

The architecture of a recurrent neural network offers both intricate and largely unpredictable behavioral profiles while at the same time adhering to constraints that generate perceivable patterns of activity.

The Perceptual layer is populated with *emons* – constructs that receive data from other emons or elements in other layers and translate it (if necessary) into directives to modulate the generation, modification and presentation of media that they control. Each emon serves one media function (e.g. sample player or sine-wave generator) with one or more mutable properties (e.g. amplitude). It modifies these properties according to an incoming array of data points (either fresh from another source – emon, input interface, etc. – or 'docked' and reused over a period of time) passed to it directly or through a translation process. Perhaps the most important source that an emon 'listens' to is the collective of nodes on the Structural layer. By attending and responding to these nodes, the layer of emons acquires complex patterns of behavior.

The range of emons employable in the Perceptual layer is diverse. They categories include: audio and video sample playing; audio waveform generating; audio processing (e.g. filtering); video processing; lighting control; and textual generating and playing. Sampled audio and video playing emons have the following properties:

Sampled Audio: Playback, Speed, Direction, Loop Pos. {Start, End}, Amplitude

Sampled Video: Playback, Speed, Direction, RGB channel control, Loop Position {Start, End}, Blend Amount.

These diverse species of emons are able to talk to one another either directly or through a *translator* agent. Emons of a common format (e.g., audio) in most cases may transfer signals directly. (e.g., a sine-wave generator emon can pass a signal directly to an audio sample-playing emon to modulate its amplitude) Otherwise the signals are passed through these translators which make them readable to the receiver.

Elements on all of the different layers as well as all media employed by a performer are sharable. That is, the system is designed to communicate with other instances of the same system, providing means for immediate collaboration as well as co-improvisational performance with other EE performers.

The system is designed to be operated using changeable degrees of autonomy. This variability allows it to be disseminated in a striking range of circumstances. On one end of the spectrum, a musician might employ the system in a performance where he requires full-on immediate influence in the Structural and Perceptual layers. On the other, a visual artist might set up a semi-permanent installation where ongoing network activity in the Structural layer continually and independently modulates room lighting controlled on the Perceptual layer, taking inputs only from ambient sensors.

Evolution Implementation

Evolution engages the EE on all three layers. The process may run singularly, combining the configurations of elements on different layers into one large genetic code; or it may run as distinct processes, allowing different layers in the network to evolve on different time scales.

Evolution modifies the Input layer by controlling just where and how data is mapped between an input interface and another layer. When specified an evolutionary process will run that will modify *transforms* (described above) and destinations of those transforms, shaping the way the outside world affects the system.

Elements in the Structural layer are possibly the prime target of the evolutionary process. Through evolution, the connectivity of the network as well as the properties of individual nodes ripen, developing a complexity as a whole which may never be found manually. Just as the brain of an animal evolves over time to suit its environment, the brain of the EE evolves to suit a performer in a given context.

In the Perceptual layer, the evolutionary process modifies the associated *translator* agents of emons, changing how an emon responds to various forms of input; the connectivity of the web of emons; and its association with the Structural layer's nodes.

We have integrated evolution in our system as three modes of operation. Each mode offers the performer a different way to interact with and change the system and each is appropriate for different goals and contexts in which the system may be used.

1. Browse (offline) In Browse mode, several 'child' networks run in the background. The networks are initialized in corresponding states of activity and each network's output is recorded over a set period of time. Each recording is voted on by the performer, and decisions about breeding, mutating, and killing of the source networks are made based on these votes in a tournament fashion. This mode is designed for a performer who is interested in configuring a network offline in a simple, hassle-free way.

2. Explore (online) For a user who would like to perform with the network, or perhaps just wants a continuous interactive experience with the system, Explore mode offers a realtime option. Here, parameters of a single network are mutated on a consistent periodic basis. Voting is an ongoing process whereby votes are captured and mapped to the appropriate configurations (present at the time of voting) according to user's actions. Direction of mutation is modified continually based on the voting.

3. Navigate (online) Navigate mode offers a more 'hands-off' option for the performer. In this mode, the performer specifies one or more saved network configurations called *magnets* to direct the evolutionary process. The magnets either attract or repel the state of the online network toward or away from that configuration, thus producing interesting phase trajectories in a performance. The mode is made more interactive when the user dynamically (de)emphasizes magnets; or when he combines it with Explore mode, allowing the user to 'drop' magnets as he goes.

Current Status

As of now, emons are strictly dependent on the nodes they listen to (one node per emon, right now). No translator agents have yet been implemented. As such, each emon understands only two types of data: directives from a connected node and the specific media it plays. Data is not yet independent of those emons that use it – while it's possible for emons to exchange media or source nodes, no independent 'data repositories' to which an emon or an agent could connect have been implemented.

Emons currently operate on sampled audio and video, in three forms: (1) prerecorded, (2) as realtime input and (3) streaming (from a URL). The first crop of emons that perform audio synthesis and processing are being developed and should be available over the next couple of months. Emons that operate on text (sampling and generation) are currently being planned. In addition, we are currently developing a bridge between audio emons and DirectX audio plug-ins as well as a built-in sample maker.

Though the functionality of the EE is currently far more limited than the sketch we have given above, the system is able to produce interesting sequences and combinations of sounds and video which enable (through choice of media data and Structural configuration), many various musical and visual patterns. For instance, loading up an array of choral voices yields a satisfying series of choir music arrangements. Combining recordings of a string orchestra with the political orations of Fidel Castro produces an unexpected simultaneous combination of comedy and drama. The environment has been met with good reactions by an initial testbed of non-professional performers. Those working with the EE have been enthusiastic, spending extended amounts of time and generally commenting on how it reminds them of one band or another.

Conclusion

This paper is a progress report. We have focused on theory – showing justification for the creation of the GIM and articulating its main points. Additionally, we have described the actual implementation that we have undertaken. It is our hope and plan to continue exploring improvisational landscapes in the context of evolution, bringing about a change in how we regard creation, modification and dissemination of media in computer environments.

Acknowledgements

The authors thank Glorianna Davenport and Ariadna Quattoni for their ideas and help.

References and Bibliography

1. Lewis, G., *Interacting with latter-day musical automata.* Contemporary Music Review, 1999. **18**(3): p. 99-112.
2. Sims, K., *Galapagos.* 1993. Information online at: http://www.genarts.com/galapagos/
3. Miranda, E. R., *On the Origins and Evolution of Music in Virtual Worlds*, in Bentley, P. and D. Corne, *Creative evolutionary systems.* 2002, San Francisco, CA. San Diego, CA: Morgan Kaufmann ; Academic Press.
4. Blackwell T.M. and Bentley P.J. (2002) “Improvised Music with Swarms”, 2002 Congress on Evolutionary Computation, CEC-2002
5. Koza, J.R., *Genetic programming : on the programming of computers by means of natural selection*. Complex adaptive systems. 1992, Cambridge, Mass.: MIT Press.
6. Varela, F.J., E. Thompson, and E. Rosch, *The embodied mind : cognitive science and human experience.* 1991, Cambridge, Mass.: MIT Press.
7. Claxton, G., *Cognitive psychology : new directions*. International library of psychology. 1980, London ; Boston: Routledge & Kegan Paul.
8. Marcus, G.F., *The algebraic mind : integrating connectionism and cognitive science*. Learning, development, and conceptual change. 2001, Cambridge, Mass.: MIT Press.
9. Stafford, B.M., *Visual analogy : consciousness as the art of connecting.* 1999, Cambridge, Mass.: MIT Press.
10. Traveling Salesman Problem. Information online at http://www.math.princeton.edu/tsp/
11. Mauss, M., The gift; forms and functions of exchange in archaic societies. 1954, Glencoe, Ill.,: Free press. xiv, 130 p.
12. Nyman, M., *Experimental music : Cage and beyond.* 2nd ed. 1999, Cambridge ; New York: Cambridge University Press.
13. Deleuze, G. and F.l. Guattari, *A thousand plateaus : capitalism and schizophrenia*. 1987, Minneapolis: University of Minnesota Press.
14. Minsky, M.L., *The society of mind.* 1st Touchstone ed. Touchstone book. 1988, New York: Simon & Schuster.
15. Bentley et al, in Bentley, P. and D. Corne, *Creative evolutionary systems.* 2002, San Francisco, CA. San Diego, CA: Morgan Kaufmann ; Academic Press.
16. Sloboda, J.A., *Generative processes in music : the psychology of performance, improvisation, and composition.* 2000, Oxford [England] New York: Clarendon Press.
17. Boden, M.A., *The philosophy of artificial life*. Oxford readings in philosophy. 1996, Oxford ; New York: Oxford University Press.
18. Dawkins, R., *The selfish gene.* New ed. 1989, Oxford; New York: Oxford Univ. Press.
19. Miranda, E.R., *Readings in music and artificial intelligence*. Contemporary music studies, v. 20. 2000, Amsterdam: Harwood Academic.
20. Miranda, E.R., *Composing music with computers*. Music technology series. 2001, Oxford ; Boston: Focal Press.
21. Moroni, J. et al, *Vox Populi: Evolutionary Computation for Music Evolution.*
22. Rowe, R., *Interactive music systems : machine listening and composing.* 1993, Cambridge, Mass.: MIT Press.
23. Lotman, I.F.U.M., *Universe of the mind : a semiotic theory of culture.* 1990, Bloomington: Indiana University Press.
24. Degazio, B., Evolution of Musical Organisms, Leonardo Music Journal, MIT Press, 1997
25. Hendriks-Jansen, H., *Catching ourselves in the act : situated activity, interactive emergence, evolution, and human thought*. Complex adaptive systems. 1996, Cambridge, Mass.: MIT Press. xii, 367.
26. Todd, P.; Werner, G.M., Frankensteinian methods for evolutionary music composition in Griffith, N. and P.M. Todd, Musical networks : parallel distributed perception and performance. 1999, Cambridge, Mass.: MIT Press. xv, 385.