

GenJam *Populi*: Training an IGA via Audience-Mediated Performance

John A. Biles
Rochester Institute of
Technology
(716)475-7453
jab@cs.rit.edu

William G. Eign
Xerox Corporation

(716)422-0688
weign.wbst102a@xerox.com

ABSTRACT: This demonstration will explore audience participation in the training of an interactive genetic algorithm (IGA) by concurrently allowing several attendees to serve as “human fitness functions” for GenJam, an IGA that improvises jazz solos. In addition to providing an opportunity for hands-on experience training an IGA, we will discuss the use of multiple concurrent mentors as a solution to the fitness bottleneck inherent in musical IGAs, and we will explore artistic and pragmatic aspects of audience-mediated performance.

1 IGAs and the Fitness Bottleneck

Genetic algorithms, or GAs (Goldberg, D.) apply the principles of natural genetics to artificial search problems. The paradigm can be viewed as breeding successive generations of potential or partial solutions to a problem until a solution emerges that meets some criterion. An individual member of a population is represented as a string of symbols (its genotype), which can be mapped to a potential problem solution (its phenotype). An important requirement for using a genetic algorithm is the ability to determine the *fitness* of a given individual as an indicator of how well it solves the problem at hand. If an algorithm exists that can at least determine which of two given individuals is better, the GA can be run off-line with no human intervention, and hundreds or thousands of generations of large populations are possible. In subjective domains, however, this *fitness function* often cannot be expressed algorithmically.

Interactive genetic algorithms (IGAs) use a human *mentor* to evaluate each individual and assign fitness values. In musical domains, where each individual might represent a sound sample (Takala, T.), a rhythm measure (Horowitz, D.), or a melodic fragment (Biles, J.), the human fitness function represents a limiting factor in the GA’s operation because the mentor must listen to each musical sample in real time in order to indicate its fitness. This *fitness bottleneck* severely constrained the design of GenJam, an IGA that learns to improvise jazz solos to the accompaniment of a synthesized rhythm section (Biles, J.).

GenJam’s original interface allows a single mentor to provide real-time feedback by typing ‘g’ (for good) or ‘b’ (for bad) while GenJam improvises solos. When the ‘g’ key is typed, the fitness values for the currently playing measure and phrase individuals are incremented. When the ‘b’ key is typed, they are decremented. Empirically-derived delays allow time for the mentor to perceive and react to the melodic material before giving feedback. In an attempt to address the fitness bottleneck, we have developed an enhanced mentor’s interface that allows several mentors to provide concurrent feedback to GenJam as it performs. The naive assumption is that multiple mentors parallelize the human fitness function, thereby speeding up the acquisition of fitness feedback and enhancing its reliability.

The interface uses readily available components to implement the following simple design. There are several button boxes available, one for each mentor. Each box provides a “good” and a “bad” button, each of which can be clicked at will. Holding a button down does not repeat its feedback, so mentors must explicitly click “good” or “bad” each time they wish to register a non-neutral opinion. The button boxes are connected to a micro-controller driven collector, which maps the button clicks to MIDI control change messages, which it sends over a standard MIDI cable to the Yamaha MU-80 tone generator’s MIDI IN terminal and, through the MU-80’s interface, to the host computer. The controller number used is 50, which is ignored by the MU-80. The GenJam software recognizes the control parameter messages with a *ctrlchange* event handler supported by the underlying CMU MIDI Toolkit environment (Dannenberg, R.). Even data values map to “good,” and odd values map to “bad.”

2 Audience-Mediated Performance

The development of the parallel-mentor interface to GenJam has followed a year-long series of experiments using GenJam in what we call *audience-mediated performance* situations. Three such venues have been explored, one involving a series of single-mentor training sessions and the other two using a low-technology approach to parallel mentoring. Our experiences in these venues have taught us some interesting lessons on how listeners respond to serving as GenJam's mentor.

The single-mentor training venue was an annual technology exposition conducted by the Information Technology Students' Organization at RIT. The first author maintained a booth during the one-day Expo where passersby were encouraged (or dragooned in some cases) to sit down and serve as mentor "for a tune or two." Erstwhile mentors read a simple description of the mentor's interface from a poster, and they were given a very brief verbal overview of GenJam. Over the course of the day, mentors each spent from one chorus to about a dozen solos providing feedback while GenJam played, and approximately 15 generations were bred from a totally random initial generation. After the day's training, the resulting soloist was used to perform three pieces in a recital of student works by the first author's computer music class, which was the final event in the Expo. Several of the mentors attended the recital to hear how "their" soloist sounded.

The two parallel-mentor venues were a series of lecture/demonstrations and a formal concert performance. The lecture/demonstrations were geared to technically sophisticated audiences and consisted of a half-hour talk on GenJam's design and implementation followed by several training solos. The concert venue was *A Concert of New Music*, presented by the Studio of Electronic Music, Inc. (SEMI) in Hartford, Connecticut on November 12, 1994.

For both of these venues each member of the audience was given a small cardboard sign with one red side and one green side. Audience members were instructed to hold up the green side (for "go") when they liked what they heard, hold up the red side (for "stop") when they disliked what they heard, and hide the card if they didn't feel strongly either way. The first author acted as the feedback collector and typed "g"s and "b"s at the Macintosh in response to the amount of red and/or green visible at any given time. In the SEMI concert, the audience trained a small-population version of GenJam for three solos, and the first author then traded choruses and "fours" with the resulting soloist on a fourth tune.

The major lesson learned from these experiments is best expressed in terms of user interface design (Norman, D.). Audience members form conceptual models of GenJam that are more complicated than the designer's conceptual model. Perhaps paradoxically, GenJam's system image (typing "g", "b", or nothing whenever so moved) seems too simple for most users to believe, so they construct conceptual models that reflect their expectations of hard-to-use technology. The parallel mentor interface used for this ICMC demonstration obviously automates the feedback collection task and makes the mentor's task even simpler and less public. However, it does not seem to help the problem of misleading user conceptual models. We will explore these models and other aspects of audience-mediated performance in this demonstration.

References

- (Biles, J.) John A. Biles. GenJam: A Genetic Algorithm for Generating Jazz Solos. In *Proceedings of the 1994 International Computer Music Conference*, ICMA, San Francisco, 1994.
- (Dannenberg, R.) Roger B. Dannenberg. *The CMU MIDI Toolkit, Version 3*. Carnegie Mellon University, Pittsburgh, PA, 1993.
- (Goldberg, D.) David Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, 1989.
- (Horowitz, D.) Damon Horowitz. Generating Rhythms with Genetic Algorithms. In *Proceedings of the 1994 International Computer Music Conference*, ICMA, San Francisco, 1994.
- (Norman, D.) Donald A. Norman. *The Design of Everyday Things*. Doubleday, New York, 1988.
- (Takala, T.) T. Takala, J. Hahn, L. Gritz, J. Geigel, and J. W. Lee. Using Physically-Based Models and Genetic Algorithms for Functional Composition of Sound Signals, Synchronized to Animated Motion. In *Proceedings of the 1993 International Computer Music Conference*, ICMA, San Francisco, 1993.