The Decentralized Pianola

Evolving Mechanical Music Instruments Using A Genetic Algorithm

Assaf K. Talmudi
Institute of Sonology
Royal Conservatory
The Hague, The Netherlands
aktalmudi@yahoo.com

ABSTRACT

This paper presents computer experiments concerning the decentralized pianola, a hypothetical mechanical music instrument, whose large-scale musical behavior is the result of local physical interactions between simple elements. Traditional mechanical music instruments like the pianola and the music box rely for their operation on the separation between a sequential memory unit and an execution unit. In a decentralized mechanical instrument, musical memory is an emergent global property of the system, undistinguishable from the execution process. Such a machine is *both* a score *and* an instrument. The paper starts by discussing the difference between sequential memory systems and systems exhibiting emergent decentralized musical behavior. Next, the use of particle system simulation for exploring virtual decentralized instruments is demonstrated, and the architecture for a simple decentralized instrument is outlined. The paper continues by describing the use of a genetic algorithm for evolving decentralized instruments that reproduce a given musical behavior.

Keywords

Emergence, Decentralized Processes, Genetic Algorithms.

1. SCORE/INSTRUMENT DUALISM

With its paper rolls acting as a soul, and its pneumatic mechanism acting as a body, the pianola is a perfect mechanical representation of instrument/score dualism - a fundamental concept in the history of Western music reproduction. According to this concept the reproduction of music depends upon two distinct units: a sequential memory unit – a score, and an execution unit – an instrument. These two units are connected by a one-way translation mechanism. The instrument itself is considered as capable of executing only isolated local musical events. The temporal arrangement of these events is created in advance, appropriately encoded and stored as a score to enable performance at a later time. During music reproduction, the score is decoded sequentially and translated into actions of the instrument.

The score/instrument dualism has proved to be an immensely powerful concept and has made possible the great developments and achievements of western music over the last five centuries. Distribution of musical knowledge, the study of music and the possibility of coordinating large groups of musicians are all to some extent dependent on this dualistic and hierarchical paradigm of music reproduction.

But this dualism has its limits, both as an approach towards generating music and as an analytical tool. These limits have been exposed by developments in music during the last century. Many forms of twentieth century music have been consciously or unconsciously challenging the validity of the dualistic hierarchical model of music reproduction, the obvious examples being improvised music and electronic music. The score/instrument dualism has also proved to offer little possibility for interaction with a piece of music. I would like to add another challenge, this time attacking the score-instrument dualism at its mechanical core.

2. SEQUENTIAL MEMORY vs. EMERGENT BEHAVIOR

What will a mechanical music instrument that does not obey the score/instrument dualism look like? As a matter of fact, many such instruments already exist. Consider wind-chimes hanging from a window: when a light breeze blows through the chimes they start colliding with each other and a melody is produced. Where is this melody stored? Explicitly it is not stored anywhere. This melody is an emergent phenomenon arising as a result of the interactions between chimes, wind, gravity and all the other elements of the system. There is one problem with this example of a decentralized music instrument: depending on the point of view, it is either infinite or else it contains stochastic elements (it really depends on the way wind is considered). However, by means of computer simulations, it is possible to investigate the properties of much simpler systems that exhibit a behavior similar to wind chimes but do not contain any stochastic element. Using computer simulations of a system that obeys simple Newtonian physics, I will sketch a basic architecture for such an instrument.

3. CONSTRUCTING A BASIC DECENTRALIZED INSTRUMENT

Three basic building blocks will be used in constructing all the following instruments:

1. A bell, simplified and modeled as a particle with a fixed mass. In the model, a bell rings when its acceleration reaches a peak and that peak is above a preset threshold.
2. A spring with a specific spring coefficient K.
3. A fixed point

These building blocks are combined in a two dimensional space which is assumed to have small friction and no gravity. Bells may be connected to other bells or to a fixed point with springs. An external impulse can be exerted to one bell to set the system in motion. The behavior of the system is then simulated by the numerical solution of the governing set of ordinary differential equations [1] [2]. During the simulation, peaks in the acceleration of the different bells are found and a bell sound with an appropriate pitch is synthesized when a bell in the model rings. The simulation runs in real time, offering audible and visual representation. Thus a music machine constructed out of bells and springs can be designed, simulated and its audible result can be examined.

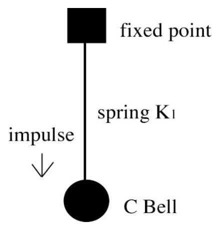

Figure 1: the simplest mechanical music instrument

The first instrument is the simplest possible (figure 1). It is composed of a single bell, which is tuned to the note C. The bell is connected by a spring with spring coefficient K1 to a fixed point. At time t, an external impulse force is exerted to the bell. Imagine listening to such a device. What melody will it play? Since the bell is going up and down and the acceleration is oscillating periodically**,** it will simply repeat the note of C at a constant tempo, gradually getting softer, loosing energy as a result of friction until the notes die out (figure 2).

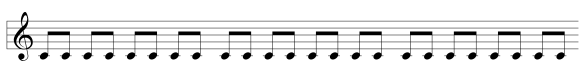

Figure 2: the melodic impulse response of the system shown in figure 1.

Throughout the rest of the paper I will refer to the perceived musical behavior of such a system, as the melodic impulse response of the system. This term describes the melody that the instrument plays when the C bell is hit with an impulse. A melodic impulse response will be given in traditional music notation. Note that the melodic impulse response of an instrument is related to the physical impulse response, but the two are not identical.

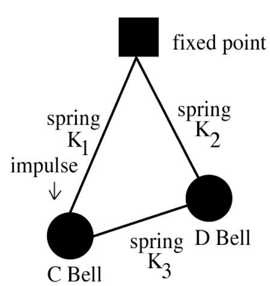

Figure 3: a slightly more complex instrument

The next step is to build a slightly more complicated music machine by adding another bell, tuned to D, and connecting it both to the first bell and to the fixed mass (figure 3). Hitting the C bell with an impulse will produce the following response:

Figure 4: the melodic impulse response of the instrument shown in figure 3

The exact periods of the notes will depend on the spring coefficients K1, K2, K3 and on the friction.

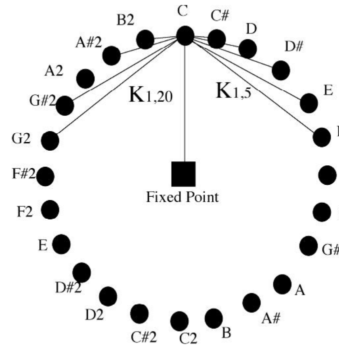

Figure 5: a two octave music instrument, only 11 springs out of 264 are shown

Now consider two chromatic octaves, or 24 different bells (figure 5). Each bell is connected to 10 neighboring bells, and all are connected to the same fixed point (only the first 11 connections are shown in the diagram). In the next simulation the spring coefficients are randomly distributed between 0.1 and 5.0, all masses are equal to 1.0, and friction is set to 0.9999. Hitting the C bell with an impulse and running the simulation for 1000 steps yields the following melodic impulse response:

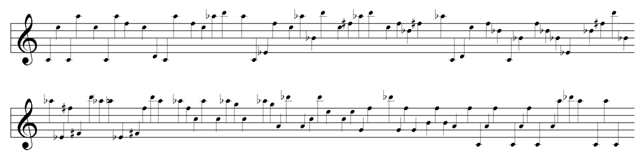

Figure 6: the melodic impulse response of the instrument from figure 5

Listening to the result of such a model makes it clear that it has a very rich musical behavior, which on the one hand is not periodic but on the other hand does not sound like a random sequence. Two interesting facts arise while considering the richness of the musical behavior generated. The first is that the musical behavior depends only on local mechanical interactions between the parts, and is not directed by high-level rules. Although it clearly contains certain melodic and rhythmic motives (rhythm is not notated above to avoid a much more complex figure), these are generated bottom-up, without any global coordinating mechanism. The second interesting fact is that the interaction is deterministic. No stochastic processes are needed for generating such a rich behavior. Given the same configuration of spring coefficients, the sequence can be precisely reproduced. In the case of the pianola, the melody was stored on a roll of punched paper, and instructions were translated into musical phenomenon. In the case of the melody above, it is quite hard to say where it is stored, although from the fact that it can be accurately reproduced one has to agree that it is stored somewhere, somehow. In the system above, the memory unit can not be distinguished from the execution mechanism. The instrument and the score are in fact just two different ways of looking at the same object. Changing the instrument, for example by changing some of the spring coefficients, will change its "score" - its large-scale musical behavior.

I now have the basic design for a two-octave decentralized mechanical music instrument, capable of some global behavior. The next step is to ask whether it is possible to tune such an instrument, for example by finding specific configuration of spring coefficients, so it will exhibit a *given* global musical behavior. Is it possible to design an instrument that will play in C major? Or one that, when hit, will play the Star Spangled Banner? One possible global characteristic of a music fragment is the note distribution in the fragment. For example, the note distribution in the melody of The Star Spangled Banner is given in figure 8(a).

The next section will describe the use of a genetic algorithm to design an instrument that will exhibit the same distribution of notes as that of *The Star Spangled Banner*.

4. EVOLVING INSTUMENTS WITH A GA

A genetic algorithm (see [3][4] for the basic technique and [5][6] for music-specific applications) was used to evolve mechanical music instruments to exhibit specific note probability distribution in their melodic impulse response. The target note distribution was chosen to be the note distribution of *The Star Spangled Banner*. During a GA run, a population of instruments is searching the space of possible configurations of spring coefficients. Each candidate instrument in the population is constructed according to its genome. This instrument is then played, and an assessment of its statistical similarity to the target melody is made. Candidates that exhibit higher similarity to the target melody have higher chances to survive and reproduce

4.1 Population

A population is defined as a group of candidate instruments. All the instruments in the population are of the form depicted in figure 4, each constructed out of 24 bells of different tones, or two octaves. Each bell is connected to 10 neighboring bells and to the fixed point. In total there are 264 springs in an instrument. Different instruments have different configurations of spring coefficients. The description of an instrument, its genome, is simply a list of 264 spring constants, represented as a list of 16bit floating point numbers. This list describes the coupling between the bells in the machine. That is not to say that all the springs are actually contributing to the dynamics and to the melodic impulse response. During the evolution of the population, the value of some spring coefficients can become as small as being practically zero, and have no effect on the behavior of the instruments.

4.2 Fitness Evaluation

For each candidate instrument, the melodic impulse response of an instrument is obtained*:* given a list of 264 spring coefficients describing an instrument, a model of the instrument is constructed. An impulse force is then exerted to the C bell and the simulation is calculated for 1000 time steps, with friction set to 0.999, and the sequence of notes played by the instrument is collected. From the melodic impulse response of the candidate, a note probability distribution P(inst) is calculated. The fitness function of a candidate instrument is defined to be the mean square difference between P(inst) and the note probability distribution of the target melody P(target).A low fitness function means a good correspondence between the distribution of notes in the melodic impulse response of the instrument and that of the target melody. For example, in *The Star Spangled Banner* there is no occurrence of the note Eb. An instrument whose impulse response contains Eb will have a higher fitness function, and so will be *less* likely to reproduce and survive than an instrument with the same impulse response only without the Eb..

4.3 Selection and Reproduction

The genetic algorithm uses an elite selection mechanism, in which for each generation, only a fraction of the population, containing the instruments with the highest fitness, is chosen for survival and reproduction. This fraction was set to be 20 percent in the experiment. From this elite group, random pairs of candidates are combined to produce new candidates. Each point in the new candidate's genome is randomly chosen from either one parent or the other. The resulting new candidates are then mutated. Since the genome is comprised of floating point numbers, a distinction is introduced between the chance of a point in the genome to be mutated and the range of possible mutation. The mutation process randomly selects a fraction of the points in the genome, the mutation percentage, each chosen point is then multiplied by a random value between 0.0 and the mutation factor.

4.4 Results

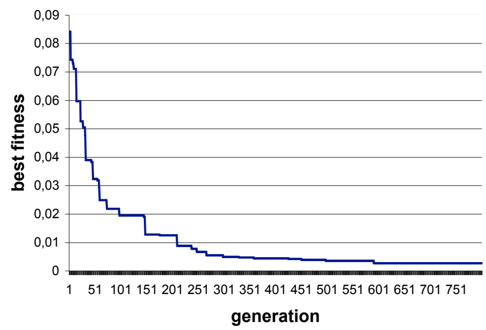

Figure 7: best fitness per generation for a GA run of 800 generations

The genetic algorithm experiment was ran for 800 generations, with a random initial population of 1000 candidate instruments. The elite fraction was set to be 0.2, mutation chance was 0.1, and mutation factor was 0.9. Several runs of the experiment exhibited similar results. As expected, during the first generations, no member of the population exhibited much statistic similarity with the target melody (if it did it would have been by pure chance.)

During the first generations, instruments with only the slightest resemblance to the target melody are taking over the population. At different points in the run, evolutionary leaps appear. Both mutation and sexual reproduction appear to be contributing to these leaps, with sexual reproduction appearing to be responsible for the bigger leaps. After 800 generations**,** an instrument with high statistical similarity to the target melody is found. Some runs have evolved instruments exhibiting note distributions that were almost identical to the target distribution, achieving fitness values as low as 0.002, with less than one percent of the notes outside the desired distribution. Listening to the result of such an instrument makes it clear that it bears statistical resemblance to the Star Spangled Banner.

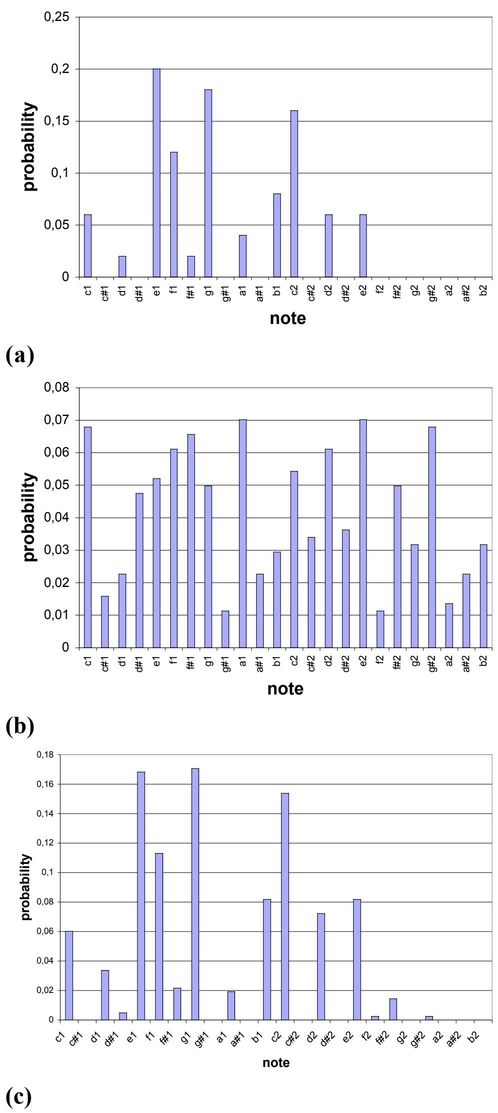

Figure 8: (a) note distribution of the first nine bars of *The Star Spangled Banner* by J. Stafford Smith. (b) note distribution of the best instrument after 800 generations, with fitness 0.002 (c) note distribution of the target melody, compared with figure b the increase in similarity is visible

5. CONCLUSIONS

In this paper I have shown that using computer simulations and genetic algorithm strategies it is possible to design a simple mechanic instrument which reproduces a certain global musical behavior without the need of a central coordination unit, or any kind of sequential memory. The experiment described evolved instruments that reproduce a given note distribution. It is likely that other high level musical features can be implemented in an instrument using the same approach.
In order to explain the potential relevance of these results, it is convenient to go back to the pianola once more. Consider a melody reproduced by a pianola and the same melody reproduced by a decentralized mechanic instrument made of springs and bells. What is the difference in the *way* in which the two instruments accomplish the same task? In the case of the pianola, the melody is fully encoded in a temporal manner in the perforations of the paper roll which functioning as a central sequential memory unit. The reproduction process does not depend on the melody, indeed it is *indifferent* to the melody. The decentralized instrument does not have such a memory unit. Musical reproduction is achieved because the dynamics of the instrument are to a certain extent analogous to the dynamics of the original melody. The instrument captures the dynamics of the melody. When wind chimes are hit by wind, a melody emerges. When people sing together, a melody emerges. Intuitively and traditionally, we think of the two processes underlying the two melodies as being very different. We explain the melody of the wind chimes in terms of the dynamics of the physical system involved. When we think of a human song we tend to think about it in terms of higher-level cognitive and cultural concepts such as musical context, musical syntax and so on. The idea of a decentralized music instrument, whose melody *is* just an emergent property of its simple dynamics does not need such concepts. Even at the current heuristic stage, the simulation of a decentralized music instrument suggests that the conventional instrument-score dualism is not inevitable: indeed in a decentralized instrument the distinction between the two fundamental elements of a musical performance becomes blurred. In doing so, it supports a reductionist approach to music which might shed further light on the information processes underlying music creation and music appreciation.

6. ACKNOWLEDGEMENTS

The research discussed in the paper was carried at the Santa Fe Institute during the Complex Systems Summer School, June-July 2003. I thank Tom Carter and Neta Cohen who provided advice and guidance in designing the experiments, and Paul Berg and Eytan Kazav for many fruitful discussions.

7. REFERENCES

[1] Castagne, N. amd Cadoz. C Phyiscal Modeling Synthesis: Balance between Realism AND Computing Speed,. In Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, Italy, 2000 http://profs.sci.univr.it/~dafx/Final-Papers/pdf/Castagne_DAFx2000.pdf

[2] Muth D., E. Burton, Sodaconductor. In Proceedings of the New Interfaces for Musical Expression (NIME'03), Montreal, Canada: McGill University, 2003.

[3] Koza, J. R. Genetic Programming: *On the Programming of Computers by Means of Natural Selection.* Cambridge, MA: MIT Press, 1992

[4] Mitchell M., J. P. Crutchfeild, and R. Das, Evolving cellular automata with genetic algorithms: a review of recent work. In *Proceedings of the First International Conference on Evolutionary Computation and Its Applications (EvCA'96).* Moscow, Russia: Russian Academy of Sciences, 1996

[5] Biles, J. 1994. GenJam: A genetic algorithm for generating jazz solos. In Proceedings of the 1994 International Computer Music Conference (ICMC'94), Aarhus, Denmark, International Computer Music Association, 1994.

[6] Miranda, E., The Music of Emergent Behaviour, What Can Evolutionary Computation Bring to the Musician, www.csl.sony.fr/downloads/papers/2000/miranda-gecco2000.pdf, 2000.