

Chapter 18. Meeting 18, Approaches: Genetic Algorithms

18.1. Announcements

- Next Quiz: Thursday, 15 April (inclusive)
- Sonic system draft due: 27 April
- No class Tuesday, 20 April

18.2. Genetic Algorithms

- Model states of a system (or processes) as an allele, or a fundamental unit of expression
- Two or more alleles form a chromosome; order of alleles generally is significant
- Chromosomes, representing individuals, are collected in a population
- Using a fitness function, each chromosome is given a fitness value
- Chromosomes are mated under conditions where more-fit chromosomes are more likely to mate
 - Two chromosomes can produce two offspring (replacing themselves)
 - Each new chromosome is created by either cloning parents or intermingling their alleles through one or two-point crossover
 - Each child chromosome may undergo mutation at the level of single allele changes or multiple allele changes
- The population is completely replaced through mating
- Numerous cycles of regeneration are completed
- The goal is for the population to evolve the most fit chromosome

18.3. GA History and Common Applications

- First described in depth by John Holland in 1975

Holland, J. 1975. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Michigan: The University of Michigan Press.

- Employed in tasks ranging from computational protein engineering, automatic programming, and the modeling of economic and ecological systems.

- Generally best suited for solving problems that lack rigorous definition

18.4. Encoding the Alleles and Chromosomes

- Many GA designs use binary encoding: 1s and 0s encode desired parameters
- Real-value encoding uses an alphabet of many characters or real numbers
- Many GAs use fixed length chromosomes

18.5. Mutations

- Binary GAs often perform bit-level manipulations
 - Bits can be flipped
 - Segments of bits can be deleted, repeated, or reversed
- Domain-specific GA mutations are possible

18.6. The Fitness Function and Finding Solutions

- The fitness function is the key
- The fitness function expresses the priorities of the system
- GAs can evolve toward a local fitness yet get stuck, not reaching the maximum fitness
- Some systems have employed human-mediated fitness evaluation

18.7. A GA of Pulse Triple Chromosomes

- Project conducted in 2001-2002

Ariza, C. 2002. "Prokaryotic Groove: Rhythmic Cycles as Real-Value Encoded Genetic Algorithms." In *Proceedings of the International Computer Music Conference*. San Francisco: International Computer Music Association. 561-567. Internet: <http://www.flexatone.net/docs/pgcrvega.pdf>.

- First design for sub-system models in athenaCL, exposed through ParameterObjects
- Alleles are pulse triples
- Chromosome is a sequence of alleles where order is musically performed order
- Fitness function is based on similarity to a target chromosome

- Find temporal distance of note durations, rest durations, and total duration (larger values mean greater distance)
- Find weighted duration of non-matched alleles (non-exact pulse triple matches, where count is multiplied by average allele duration)
- Find weighted duration of non-matched duration ratios (non matching pulse triple ratios, where count is multiplied by average allele duration)
- Sum of these values weighted with values found through experiment: $\text{noteDistance} * 1.50$, $\text{restDistance} * 1.50$, $\text{durDistance} * 2.33$, $\text{noMatchAlleleDistance} * 1.00$, $\text{noMatchValueDistance} * 0.66$.
- An inverse relation: the larger the value, the greater the distance from the target
- Two point crossover employed in mating
- Mutations are specific to pulse triples
 - Ratio equivalence: multiply or divide divisor or multiplier by 2 or 3
 - Divisor mutate: add or subtract 1 to divisor
 - Multiplier mutate: add or subtract 1 to multiplier
 - Flip note/rest state
 - Inversion: select to lic, reverse the segment with the retrograde of the segment
- Population is initialized through random arrangements of pulse triples found in the source
- For each generation, retain the chromosome that is the best fit (and is unique)
- After generations are complete, order best-fit chromosomes by fitness
- Example: python genetic.py

18.8. GA as ParameterObject

- The gaRhythm ParameterObject

```

:: tpv garhythm
Rhythm Generator ParameterObject
{name,documentation}
GaRhythm          gaRhythm, pulseList, crossover, mutation, elitism,
                   selectionString, populationSize
Description: Uses a genetic algorithm to create rhythmic
variants of a source rhythm. Crossover rate is a percentage,
expressed within the unit interval, of genetic crossings
that undergo crossover. Mutation rate is a percentage,
expressed within the unit interval, of genetic crossings

```

that undergo mutation. Elitism rate is a percentage, expressed within the unit interval, of the entire population that passes into the next population unchanged. All rhythms in the final population are added to a list. Pulses are chosen from this list using the selector specified by the control argument. Arguments: (1) name, (2) pulseList {a list of Pulse notations}, (3) crossover, (4) mutation, (5) elitism, (6) selectionString {"randomChoice", "randomWalk", "randomPermutate", "orderedCyclic", "orderedCyclicRetrograde", "orderedOscillate"}, (7) populationSize

18.9. Evolving African Drum Patterns with a GA

- Slow Agbekor (Chernoff 1979)

Slow Agbekor (supporting drums)

The musical score consists of six staves, each representing a different instrument in the Slow Agbekor ensemble. The time signature for all staves is 12/8. The instruments and their corresponding lyrics are:

- Bell**: etc.
- Kagan**: Mi- tso, mi- tso, mi- tso, mi- tso
- Kidi**: Kpo afe go- dzi, kpo afe go- dzi
- Kroboto**: Gbe- dzi ko ma do, gbe- dzi ko ma do
- Totogi**: Ko- ko dzi, dzi, dzi

© University of Chicago Press. All rights reserved. This content is excluded from our Creative Commons license. For more information, see <http://ocw.mit.edu/fairuse>.

- Command sequence 1: exploring two durations:
 - emo mp
 - tmo lg

- tin a 61
- *bell line, set to loop*
tie r l,[(4,4,1),(4,4,1),(4,2,1),(4,4,1),(4,4,1),(4,4,1),(4,2,1)]
- *accent the first of each articulation*
tie a bg,oc,(1,.5,.5,.5,.5,.5,.5)
- tin b 68
- *create genetic variations using a high mutation rate*
tie r gr,[(4,4,1),(4,4,1),(4,2,1),(4,4,1),(4,4,1),(4,4,1),(4,2,1)],.7,.25,0
- tie a bg,oc,(1,.5,.5,.5,.5,.5,.5)
- eln; elh
- Command sequence 2: combinations of rests and silences
 - emo mp
 - tmo lg
 - tin a 61
 - *kagan line, set to loop*
tie r l,[(4,2,0),(4,2,1),(4,2,1),(4,2,0),(4,2,1),(4,2,1),(4,2,0), (4,2,1),(4,2,1),(4,2,0),(4,2,1),(4,2,1)]
 - *accent the first of each articulation*
tie a bg,oc,(.5,1,.5, .5,.5,.5, .5,.5,.5, .5,.5,.5)
 - *turning on silence mode will use parameters even for rests*
timode s on
 - tin b 68
 - *create genetic variations using a high crossover, no mutation*
tie r gr,[(4,2,0),(4,2,1),(4,2,1),(4,2,0),(4,2,1),(4,2,1),(4,2,0),
(4,2,1),(4,2,1),(4,2,0),(4,2,1),(4,2,1)],1,0,0
 - tie a bg,oc,(.5,1,.5, .5,.5,.5, .5,.5,.5, .5,.5,.5)
 - *turning on silence mode will use parameters even for rests*

timode s on

- eln; elh
- Command sequence 3: multiple rhythmic values:

- emo mp

- tmo lg

- tin a 61

- *keroboto line, set to loop*

tie r l,[(4,3,1),(4,1,1),(4,2,1),(4,2,1),(4,1,1),(4,1,1),(4,2,1),
(4,3,1),(4,1,1),(4,2,1),(4,2,1),(4,1,1),(4,1,1),(4,2,1)]

- *accent the first of each articulation*

tie a bg,oc,(1,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5)

- tin b 68

- *create genetic variations using a high crossover and mutation rate and some elitism*

tie r gr,[(4,3,1),(4,1,1),(4,2,1),(4,2,1),(4,1,1),(4,1,1),(4,2,1),
(4,3,1),(4,1,1),(4,2,1),(4,2,1),(4,1,1),(4,1,1),(4,2,1)],.9,.25,0.1

- tie a bg,oc,(1,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5)

- eln; elh

18.10. Polyphonic African Drum Patterns with a GA

- Slow Agbekor (Chernoff 1979)

Slow Agbekor (supporting drums)

Bell etc.

Kagan

Mi- tso, mi- tso, mi- tso, mi- tso

Kidi

Kpo afe go- dzi, kpo afe go- dzi

Kroboto

Gbe- dzi ko ma do, gbe- dzi ko ma do

Totogi

Ko- ko dzi, dzi, dzi

© University of Chicago Press. All rights reserved. This content is excluded from our Creative Commons license. For more information, see <http://ocw.mit.edu/fairuse>.

- Command sequence:
 - emo mp
 - tmo lg
 - tin a 45
 - tie r gr,[(4,4,1),(4,4,1),(4,2,1),(4,4,1),(4,4,1),(4,4,1),(4,2,1)],7,15,0
 - tie a bg,oc,(1,.5,.5,.5,.5,.5,.5)
 - tin b 60
 - *create genetic variations using a high crossover, no mutation*
 - tie r gr,[(4,2,0),(4,2,1),(4,2,1),(4,2,0),(4,2,1),(4,2,1),(4,2,0),
(4,2,1),(4,2,1),(4,2,0),(4,2,1),(4,2,1)],1,0,0
 - tie a bg,oc,(.5,1,.5, .5,.5,.5, .5,.5,.5, .5,.5,.5)

- *turning on silence mode will use parameters even for rests*
- timode s on
- tin c 68
 - *create genetic variations using a high crossover and mutation rate and some elitism*
- tie r gr,[(4,3,1),(4,1,1),(4,2,1),(4,2,1),(4,1,1),(4,1,1),(4,2,1),
(4,3,1),(4,1,1),(4,2,1),(4,2,1),(4,1,1),(4,1,1),(4,2,1)],.9,.25,0.1
- tie a bg,oc,(1,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,.5)
 - eln; elh

18.11. Reading: Biles, GenJam in Perspective: A Tentative Taxonomy for GA Music and Art Systems

- Biles, J. A. 2003. "GenJam in Perspective: A Tentative Taxonomy for GA Music and Art Systems." *Leonardo* 36(1): 43-45.
- What are the alleles and chromosomes in this study?
- At what level of the chromosome do the mutations operate? What types of mutations are used
- How is fitness measured?
- How does the concept of "musically meaningful mutations" deviate from conventional GAs?
- Which does the author suggest are more solution-rich: artistic domains or non-artistic domains?

18.12. GenJam Example

- Video: Demonstration created in 2003

18.13. Reading: Magnus, Evolving electroacoustic music: the application of genetic algorithms to time-domain waveforms

- Magnus, C. 2004. "Evolving electroacoustic music: the application of genetic algorithms to time-domain waveforms." In *Proceedings of the International Computer Music Conference*. San Francisco: International Computer Music Association. 173-176.

- What are the alleles and chromosomes in this study?
- What types of mutations were explored in this study?
- Is there a distinction between genotype and phenotype?
- The author writes: “at each stage of programming, choices must be made that introduce designer bias into the system”; is this a problem?

MIT OpenCourseWare
<http://ocw.mit.edu>

21M.380 Music and Technology: Algorithmic and Generative Music
Spring 2010

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.