Genetic Algorithms and their use in Astronomy

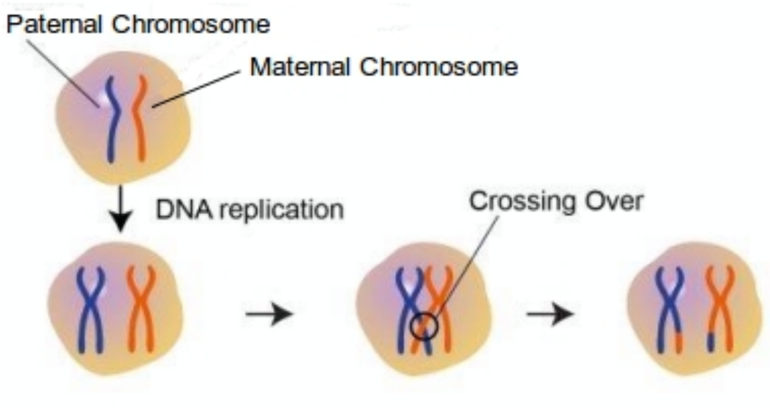
Kyle Roell

What is a Genetic Algorithm (GA)?

- An algorithmic technique used to help solve optimization problems
- Uses principles from genetics and evolutionary biology
  - Natural Selection
  - Inheritance
  - Mutations
- First majorly developed and publicized in the 1970s

Background Genetic Terms

- Genotype – the genetic makeup of an organism
- Phenotype – observable, physical characteristics (based on genotype and environment)
- Chromosome – a “package” of DNA (23 pairs in humans)
- Inheritance – the passing of DNA from parents to offspring
- Recombination / Crossover – exchange of genetic material across chromosomes or from different regions of the same chromosome
- Mutation – permanent alteration in the genetic sequence

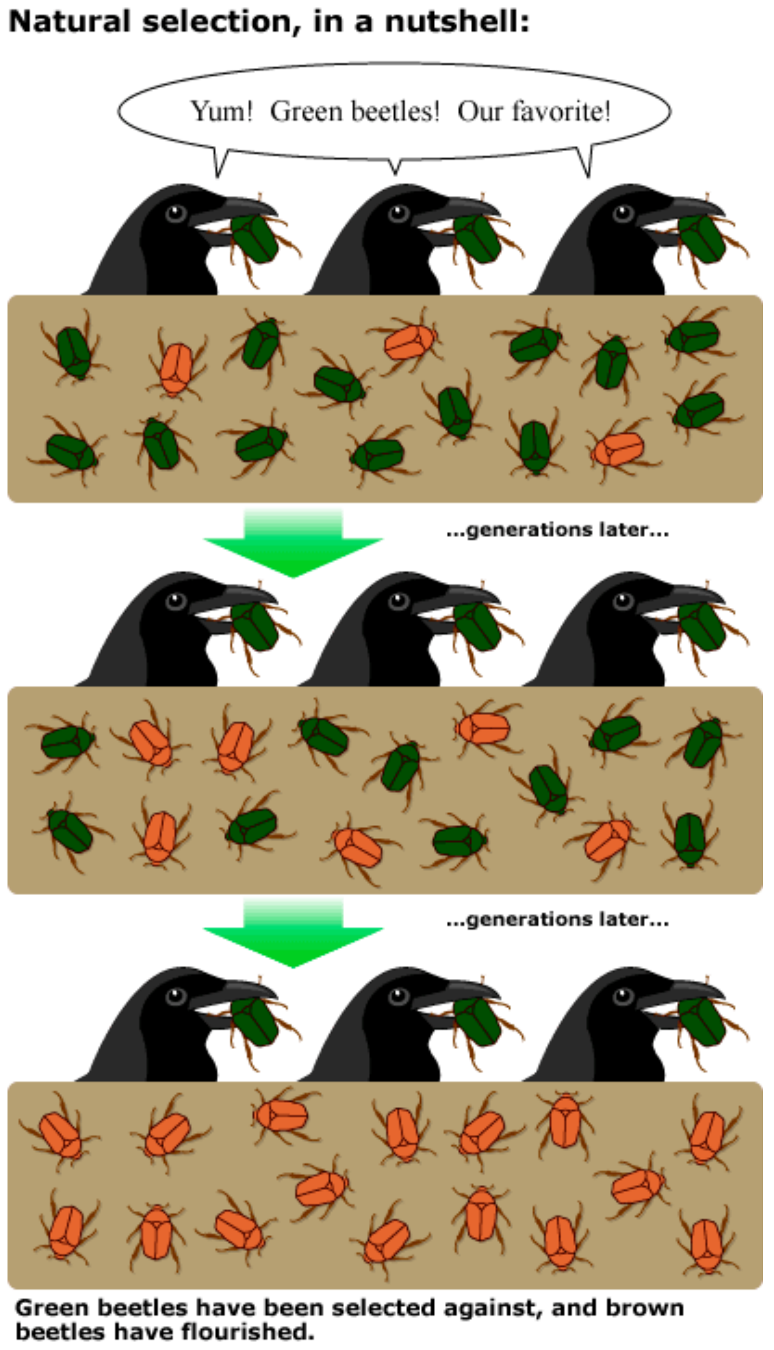


Recombination between 2 homologous chromosomes

http://biology.about.com/od/genetics/ss/genetic-recombination.htm

Idea of Natural Selection

- Variation in a population
  - green vs orange beetles
- Some selection process
  - birds eating beetles
- Heredity (genetic basis for trait)
  - color is passed on to future generations



http://evolution.berkeley.edu/evolibrary/article/bergstrom_02

Back to Computation…

- We want to find the solution to some (optimization) problem!

1. Start with some population of candidate solutions
2. The “fitness” of each solution is determined by some function
3. Select candidate solutions with probability determined by fitness
4. Encode selected candidates and use “recombination” and “mutation” operations on them
5. Continue from step 2 until termination condition attained

Example of GA

- To better illustrate the steps of a GA, let's look at an example
- Say, we want to maximize the function:
  - $f(x) = 120x - x^2$ over the domain $0 \leq x \leq 127$
- We can easily find the solution to this
  - Set derivative equal to 0
  - Obtain $x = 60$
- But it's more fun and easier to let a GA solve this (assuming you already have one ready to go!)

Starting Population and Encoding

- Starting population is often generated randomly throughout the entire search space of solutions (phenotypes)
- Population of solutions must be encoded (genotypes or chromosomes)
  - Ex: binary encoding (0’s and 1’s)
  - Stored as an array of bits
- *Our example, numbers encoded as binary*
  - *1 = 0000001, 127 = 1111111*

Fitness Function

- The core part of the algorithm
  - Determines how “good” a solution is
  - Selects which solutions “reproduce” to keep algorithm going
- Designed by the programmer
- If designed poorly can lead to inappropriate or non-optimal solutions
- Often is the additive inverse of the cost function to be optimized

- *Our example, we want to maximize* $f(x) = 120x - x^2$

Selection of Solutions

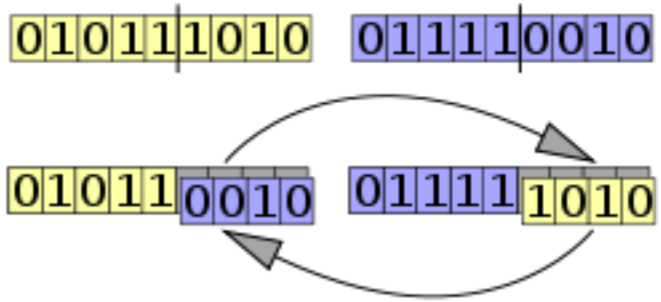
- Solutions need to be selected (with replacement) for future generations
- Various methods of selection
  - Roulette – apply probabilities of being selected directly based on fitness
  - Rank – rank solutions by fitness and then select
    - Good for population with very close fitness values
  - K-way Tournament – randomly select K individuals and compare fitness directly against each other to select most fit parent, repeat for every parent
  - Random – simply randomly select solutions from population
    - No selection pressure towards fitness so generally not useful
  - Elitism – keep one or more best solutions unchanged in future generation

- *We'll use tournament style in our example since we can have negative fitness*

Applying Genetic Operators

- The two genetic operators are generally crossover and mutation
- Each have a certain probability of occurrence associated with them
  - A parameter of the algorithm
- Idea of crossover is to (hopefully) combine best parts of each solution
- Mutation randomly changes part of a solution
  - Get out of local extremes
  - Too much mutation can lead to simply a random search (not what we want)

- *We'll use a crossover rate of .7 and mutation rate of .001*



http://creationwiki.org/Genetic_algorithm

Termination Conditions

- Termination hopefully occurs when an optimal solution is found
- General conditions for stopping
    - Length of time running / number of iterations
    - No improvement for specified number of iterations
    - Achievement of some predetermined fitness value

- *Our termination condition will be finding known solution or some number of runs without improvement*

Pros and Cons GAs

- Can be faster and potentially more efficient than other methods
- Provides numerous "good" solutions
- Parallel computing possible

- Won't guarantee the optimal solution
- Can be computationally taxing if incorrect parameters or fitness function are chosen
- May not converge if incorrectly implemented

Questions to Consider

- What type of encoding to use?
- How to decide on the proper fitness function?
- Which genetic operators should be used?
- When should the algorithm terminate?

- Answers depend on the individual problem to be solved

Toy Example in R!

Examples in Astronomy

- Wahde and Donner used a GA for determining the orbital parameters of interacting galaxies and applied their method to both artificial and real data
- Bogdanos and Nesseris used GAs to analyze Type Ia SNe data and to extract model-independent constraints on the evolution of the dark energy equation of state
- Baier et al. were able to combine radiative transfer codes with a GA to produce an automated procedure for fitting the dust spectra of AGB stars

Rajpaul et al. Genetic algorithms in astronomy and astrophysics. Proceedings of SAIP2011. pp. 519-524.

GA for Light Curve Optimization

- Metcalfe developed a genetic algorithm to optimize parameters of a model based on the Wilson-Devinney (W-D) code
  - W-D code is a procedure for calculating light curves
- Problems with parameter space
  - Where to start?
  - Subjectivity from using past personal experience
  - Large search space hard to efficiently search
- GA to help solve some of these common problems

Observations and Data

- Metcalfe wanted to calculate light curves for observations of the eclipsing binary star BH Cassiopeiae
- Light curves from previous data were also used to constrain the fit
  - V-band from 432 data points
  - B-band from 1107 data points
  - U-band from 1041 data points

GA Overview

- Define search space for (5) parameters of model
- Randomly generate 1000 trial parameter sets
- Calculate light curves based on observational data for each of these trial parameter sets
- Compute variance between these observed data light curves and the three generated from previous data
- Average these variances to get the fitness of a given parameter set

GA Overview

- Pass along one copy of the most fit parameter set to the next generation
- Additionally, randomly select two sets to potentially crossover and mutate
  - Crossover rate set to .65 and mutation rate to .003
- Repeat this until the fractional difference between the average fitness and maximum fitness in a generation is less than 1%
- Final parameter estimates are then taken as averages of all the estimates for the final, converged generation

Results

- Metcalfe used the standard differential corrections procedure supplied in the W-D code to validate the GA results
- The results from the GA fit are in “excellent agreement” with the results produced by the W-D code calculated

$$q^{(1)} = 0.474 \pm 0.002\ , \quad \Omega^{(1)} = 2.798 \pm 0.015\ ,$$
$$i^{(1)} = 69^{\circ}\!.52 \pm 1^{\circ}\!.42\ ,$$
$$[T_1/T_2]^{(1)} = 0.953 \pm 0.005\ , \quad T_1^{(1)} = 4788 \pm 106\ \mathrm{K}\ .$$

Results from GA

$$q = 0.474 \pm 0.002\ , \quad \Omega = 2.801 \pm 0.003\ ,$$
$$i = 70^{\circ}\!.1 \pm 0^{\circ}\!.2\ ,$$
$$T_1 = 4790 \pm 10\ \mathrm{K}\ , \quad T_2 = 4980 \pm 10\ \mathrm{K}\ .$$

Results after applying W-D corrections procedure

Bibliography

- http://creationwiki.org/Genetic_algorithm
- http://www.obitko.com/tutorials/genetic-algorithms
- https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_quick_guide.htm
- https://en.wikipedia.org/wiki/Genetic_algorithm
- Mitchell, M. An Introduction to Genetic Algorithms. 1999.
- Rajpaul et al. Genetic algorithms in astronomy and astrophysics. Proceedings of SAIP2011. pp. 519-524.
- Metcalfe T. Genetic-Algorithm-based Light-Curve Optimization Applied to Observations of the W Ursae Majoris Star BH Cassiopeiae. The American Astronomical Society. 1999.
- http://biology.about.com/od/genetics/ss/genetic-recombination.htm

Questions?