Derivative Free Optimization

Topic 3: Genetic Algorithms

Warren Hare

Genetic Algorithms

Survival of the fittest

- In 1864 Charles Darwin published “On the origin of species”
- In 1869, Darwin published edition 5 and include the phrase “survival of the fittest”
- The phrase is originally attributed to Herbert Spencer, an economist

GENETIC ALGORITHM philosophy
Use *survival of the fittest* to remove poor solutions and create new better solutions

Genetic Algorithms, history

- Late 1960s, Rechenberg writes "Evolutionary Strategies for Optimization" (in German) and uses then to optimize aerofoils
- 1971, Rechenberg's PhD thesis
- 1975, John Holland write "Adaptation in Natural and Artificial Systems"
- 2020, Rechenberg's thesis has 125 citations,

  Holland's book has 67,000+ citations

Genetic Algorithms, Overview

- Points in constraint set = potential members in a population
- Collection of points = population
- Function values = quality of individual (fitness)
- Algorithms Iterations = reproduction & survival

Algorithm: Genetic Algorithms (GA)

Given $f : \mathbb{R}^n \mapsto \mathbb{R}$ and an initial population $P^0 = \{x^1, x^2, \ldots, x^{\bar{p}}\}$

0. Initialize:
   - $\gamma \in (0, 1)$ — mutation probability
   - $k \leftarrow 0$ — iteration counter
1. Fitness:
   - Use $f(x)$ to assign a fitness to each individual $x \in P^k$
2. Reproduce:
   - 2a) Selection: select 2 parents from P^k and proceed to 2b)
     or select 1 survivor from P^k and proceed to 2d)
   - 2b) Crossover: use the 2 parents to create an offspring
   - 2c) Mutation: with probability γ mutate the offspring
     if the offspring is infeasible
     declare the offspring deceased and return to 2a)
     otherwise declare the offspring a survivor and proceed to 2d)
   - 2d) Update next generation: place survivor into P^{k+1}
     if $|P^{k+1}| \geq \bar{p}$ declare population complete and proceed to 3
     otherwise declare population incomplete and go to 2a)
3. Update:
   - increment $k \leftarrow k + 1$, stop or go to 1

Notes

- GA is more of a framework than an algorithm
- The great flexibility has resulted in a lot of custom-made GA

Fitness & Selection

Genetic Algorithm

To maintain the spirit of genetics,

- GA begins by assigning each point a **fitness**
- Fitness is used to select **survivors** and **parents**
- Parents create **offspring**
- Offspring are **mutated** to create more variation

Fitness

Fitness converts function value into a new number

Fitness should follow two rules

- When *minimizing*, if $f(x) < f(y)$ then x is more fit than y
  (When *maximizing*, if $f(x) > f(y)$ then x is more fit than y)
- Fitness should be strictly positive

Rank Fitness (minimization)

Definition: Given a population P^k with function values $\{f(x^i)\}$
The **rank fitness** of x^i is the number of points x^j where $f(x^i) \leq f(x^j)$

Example: Suppose $P = \{x^1, x^2, \ldots, x^5\}$ and

$$f(x^1) = -3, \quad f(x^2) = 4, \quad f(x^3) = 1, \quad f(x^4) = 1, \quad f(x^5) = 2$$

Find the rank fitness

Function Value Fitness (minimization)

Definition: Given a population P^k with function values $\{f(x^i)\}$
The **function valued fitness** is obtained by multiplying the function value by -1 and then shifting the results until the lowest value is equal to 1

$$\text{Fit}(x^i) = -f(x^i) + \bar{f} + 1 \qquad \text{where} \quad \bar{f} = \max_{i=1,2,\ldots,\bar{p}} \{f(x^i)\}$$

Example: Suppose $P = \{x^1, x^2, \ldots, x^5\}$ and

$$f(x^1) = -3, \quad f(x^2) = 4, \quad f(x^3) = 1, \quad f(x^4) = 1, \quad f(x^5) = 2$$

Find the function value fitness

Selection

Given population P^k

In selection we create two subsets

1. Survivors $S^k \subseteq P^k$
2. Parents $T^k \subseteq \{P^k, P^k\}$

To maintain the spirit of genetics, typically some randomness is used

Elitism Selection

Definition: Given a population P^k with fitness values $\{f^i = \text{Fit}(x^i)\}$ **Elitism selection** of size s selects the s individuals with the highest fitness level with ties broken by random selection with uniform probability

---

Example: Suppose $P = \{x^1, x^2, \ldots, x^5\}$ and

$$\text{Fit}(x^1) = 3, \quad \text{Fit}(x^2) = 2, \quad \text{Fit}(x^3) = 1, \quad \text{Fit}(x^4) = 5, \quad \text{Fit}(x^5) = 3$$

Describe the result if elitism selection is used to select 2 individuals

Roulette Wheel Selection

Definition: Given a population P^k with fitness values $\{f^i = \text{Fit}(x^i)\}$ **Roulette wheel selection** randomly selects individuals with probability

$$\text{prob}\{x^i\} = f^i / \sum_{j=1}^{\bar{p}} f^j$$

Example: Suppose $P = \{x^1, x^2, \ldots, x^5\}$ and

$$\text{Fit}(x^1) = 3, \quad \text{Fit}(x^2) = 2, \quad \text{Fit}(x^3) = 1, \quad \text{Fit}(x^4) = 5, \quad \text{Fit}(x^5) = 3$$

Describe the result if roulette wheel selection is used to select 1 individual

Tournament Selection

Definition: Given a population P^k with fitness values $\{f^i = \text{Fit}(x^i)\}$
Given a **tournament size** $1 \leq N \leq \bar{p}$
Tournament selection first randomly chooses N *distinct* individuals with equal probability, then selects from this subset using either elitism or roulette selection

---

Example: Suppose $P = \{x^1, x^2, \ldots, x^5\}$ and

$$\text{Fit}(x^1) = 3, \quad \text{Fit}(x^2) = 2, \quad \text{Fit}(x^3) = 1, \quad \text{Fit}(x^4) = 5, \quad \text{Fit}(x^5) = 3$$

Describe the result if tournament selection is used to select 2 individuals when

1. the tournament size is 2 and elitism is used
2. the tournament size is 2 and roulette wheel is used
3. the tournament size is 3 and roulette wheel is used

Tournament Selection, size 2

Tournaments of size 2 for $P = \{x^1, x^2, \ldots, x^5\}$

Tournament	Probability
$\{x^1, x^2\}$	$1/10$
$\{x^1, x^3\}$	$1/10$
$\{x^1, x^4\}$	$1/10$
$\{x^1, x^5\}$	$1/10$
$\{x^2, x^3\}$	$1/10$
$\{x^2, x^4\}$	$1/10$
$\{x^2, x^5\}$	$1/10$
$\{x^3, x^4\}$	$1/10$
$\{x^3, x^5\}$	$1/10$
$\{x^4, x^5\}$	$1/10$

Tournament Selection, Elitism size 2

Elitism winners when
$\mathrm{Fit}(x^1) = 3, \quad \mathrm{Fit}(x^2) = 2, \quad \mathrm{Fit}(x^3) = 1, \quad \mathrm{Fit}(x^4) = 5, \quad \mathrm{Fit}(x^5) = 3$

Tournament	Probability	Winner
$\{x^1, x^2\}$	$1/10$	x^1
$\{x^1, x^3\}$	$1/10$	x^1
$\{x^1, x^4\}$	$1/10$	x^4
$\{x^1, x^5\}$	$1/10$	x^1 and x^5 tie
$\{x^2, x^3\}$	$1/10$	x^2
$\{x^2, x^4\}$	$1/10$	x^4
$\{x^2, x^5\}$	$1/10$	x^5
$\{x^3, x^4\}$	$1/10$	x^4
$\{x^3, x^5\}$	$1/10$	x^5
$\{x^4, x^5\}$	$1/10$	x^4

Tournament Selection

Tournaments of size 3 for $P = \{x^1, x^2, \ldots, x^5\}$

Tournament	Probability
$\{x^1, x^2, x^3\}$	$1/10$
$\{x^1, x^2, x^4\}$	$1/10$
$\{x^1, x^2, x^5\}$	$1/10$
$\{x^1, x^3, x^4\}$	$1/10$
$\{x^1, x^3, x^5\}$	$1/10$
$\{x^1, x^4, x^5\}$	$1/10$
$\{x^2, x^3, x^4\}$	$1/10$
$\{x^2, x^3, x^5\}$	$1/10$
$\{x^2, x^4, x^5\}$	$1/10$
$\{x^3, x^4, x^5\}$	$1/10$

Encoding

What is encoding

To maintain the spirit of genetics, most GA start by encoding the points as *chromosomes*

$$x \in \mathbb{R}^n \quad \mapsto \quad c \in \{0,1\}^d$$

This is called **encoding**

Example:

Suppose $\Omega = [0,1]$
Encode $x \in \Omega$ by

$$\begin{array}{ccc} x < \frac{1}{2} & \mapsto & c = 0 \\ x \geq \frac{1}{2} & \mapsto & c = 1 \end{array}$$

What is encoding

To maintain the spirit of genetics, most GA start by encoding the points as *chromosomes*

$$x \in \mathbb{R}^n \quad \mapsto \quad c \in \{0,1\}^d$$

This is called **encoding**

Example:

Suppose $\Omega = [0,1]$

Encode $x \in \Omega$ by

$$\begin{array}{rcl} x < \frac{1}{2} & \mapsto & c = 0 \\ x \geq \frac{1}{2} & \mapsto & c = 1 \end{array}$$

Alternately

$$\begin{array}{rcl} 0 \leq x < \frac{1}{4} & \mapsto & c = [0,0] \\ \frac{1}{4} \leq x < \frac{1}{2} & \mapsto & c = [0,1] \\ \frac{1}{2} \leq x < \frac{3}{4} & \mapsto & c = [1,0] \\ \frac{3}{4} \leq x \leq 1 & \mapsto & c = [1,1] \end{array}$$

Natural encodings

Some problems have a *natural structure* that makes encoding easy

Example:

$$\min \left\{ \textit{expected overtime} : \begin{array}{l} \textit{all shifts filled} \\ \textit{staff are `happy'} \end{array} \right\}$$

uses the variables

$$x_{i,j} \in \{0, 1\}$$

with $x_{i,j} = 1$ if staff member i works shift j

(See MATH 441 for more info on scheduling problems)

Natural encoding for bounded variables

Definition: If $x \in \mathcal{Z}^n$ with $\ell \leq x \leq u$, then the **natural encoding for bounded integer variables** replaces each x_i with the binary string $c^i \in \{0,1\}^{d_i}$ where d_i is the number of bits required
The final chromosome is defined by $c = [c^1, c^2, \ldots, c^n]^\top$

Example: Let $S = \{x \in \mathbb{R}^2 : 0 \leq x_1 \leq 8, -2 \leq x_2 \leq 1, x_i \in \mathcal{Z}\}$
Develop the natural encoding for S

Discretization encoding

Definition: Suppose $x \in \mathbb{R}^1$ with $\ell \leq x \leq u$
Select d and set $\Delta = (u - \ell)/(2^d)$
Separate $[l, u]$ into 2^d equal sized subintervals

$$[\ell, \ell + \Delta) \;\; [\ell + \Delta, \ell + 2\Delta) \;\; \ldots \;\; [\ell + (2^d - 1)\Delta, \ell + (2^d)\Delta]$$

The **discretization encoding** is created by enumerating the interval list and encoding the integer enumeration
If $x \in \mathbb{R}^n$, the encode each $x_i \mapsto c^i$, then set $c = [c^1, c^2, ..c^n]^\top$

Example: Let $S = \{x \in \mathbb{R}^2 : 0 \leq x_1 \leq 8, -2 \leq x_2 \leq 1\}$
Develop a discretization encoding using $2^2 = 4$ equally spaced subintervals in each coordinate

Notes

- Using 2^d subintervals in a discretization encoding avoids waste
- Another option is to not encode (leave $x \in \mathbb{R}^n$)

Reproduction

Reproduction

Survivors of iteration k are placed into P^{k+1}

Parents are combined to create *offspring*

$$\text{parent1} \oplus \text{parent2} \to \text{child}$$

To maintain the spirit of genetics...

Crossover

Definition: Given parents $p = [p_1, p_2, \ldots, p_d]^\top$ and $q = [q_1, q_2, \ldots, q_d]^\top$ **Single-point crossover** selects a random number $X \in \{1, 2, \ldots, d-1\}$, and creates the child

$$[p_1, p_2, \ldots, , p_X, q_{X+1}, q_{X+2}, \ldots, q_d]^\top$$

Example: Let $p = [5, 2, -3, 5, 9, 1]^\top$ and $q = [3, 0, 0, 5, 2, 1]^\top$
Describe the result of a single-point crossover

Crossover

Definition: Given parents $p = [p_1, p_2, \ldots, p_d]^\top$ and $q = [q_1, q_2, \ldots, q_d]^\top$ **2-point crossover**, selects 2 random numbers $X_1 \in \{1, 2, \ldots, d-2\}$ and $X_2 \in \{X_1+1, X_1+2, \ldots, d-1\}$ and creates the child

$$[p_1, p_2, \ldots, , p_{X_1}, q_{X_1+1}, q_{X_1+2}, \ldots, q_{X_2}, p_{X_2+1}, p_{X_2+2}, \ldots, p_d]^\top$$

Example: Let $p = [5, 2, -3, 5, 9, 1]^\top$ and $q = [3, 0, 0, 5, 2, 1]^\top$
Describe the result of a 2-point crossover

Crossover

Definition: Given parents $p = [p_1, p_2, \ldots, p_d]^\top$ and $q = [q_1, q_2, \ldots, q_d]^\top$ ***n*-point crossover**, selects n random numbers ...

Probabilistic gene selection

Definition: Given parents $p = [p_1, p_2, \ldots, p_d]^\top$ and $q = [q_1, q_2, \ldots, q_d]^\top$
Select a probability parameter $\theta \in (0, 1)$
Probabilistic gene selection produces child c such that

$$\mathrm{prob}(c_i = p_i) = \theta \quad \text{and} \quad \mathrm{prob}(c_i = q_i) = (1 - \theta)$$

Example: Let $p = [5, 2, -3, 5, 9, 1]^\top$ and $q = [3, 0, 0, 5, 2, 1]^\top$
Describe the result of a probabilistic gene selection for $\theta = 0.25$

Weighted Average

Definition: Given parents $p \in \mathbb{R}^d$ and $q \in \mathbb{R}^d$
Select a weight parameter $\theta \in (0,1)$
Weighted average produces the child

$$c = \theta p + (1-\theta)q$$

Example: Let $p = [5, 2, -3, 5, 9, 1]^\top$ and $q = [3, 0, 0, 5, 2, 1]^\top$
Suppose $\mathrm{Fit}(p) = 6$ and $\mathrm{Fit}(q) = 4$
Describe the result of a weighted average where $\theta = \frac{\mathrm{Fit}(p)}{\mathrm{Fit}(p)+\mathrm{Fit}(q)}$

Mutation

Mutation

Offspring in natural are never exact copies of their parents

To maintain the spirit of genetics, *mutation* is applied

Bit Inversion

Definition: Given offspring $c = [c_1, c_2, \dots, c_d]^\top$ where $c_i \in \{0, 1\}$
Fix a *mutation probability* $\delta \in (0, 1)$
The **bit inversion** mutation creates *mutated offspring* m such that

$$\mathrm{prob}(m_i = c_i) = (1 - \delta) \quad \text{and} \quad \mathrm{prob}(m_i \neq c_i) = \delta.$$

Example: Suppose $c = [1, 0, 0]^\top$ and let $\delta = 1/3$
Describe the mutated offspring of a bit inversion mutation

Coordinate perturbation

Given offspring $c = [c_1, c_2, \ldots, c_d]^\top$ where $c_i \in \mathbb{R}$
Fix a *mutation probability* $\delta \in (0, 1)$
Fix a *mutation probability distribution* function F
Coordinate perturbation creates *mutated offspring* m such that

$$\mathrm{prob}(m_i = c_i) = (1 - \delta) \quad \text{and} \quad \mathrm{prob}(m_i \neq c_i) = \delta$$

and

$$\text{if } m_i \neq c_i, \text{then } m_i = c_i + \xi \text{ where } \xi \text{ follows the distribution } F$$

Example: Suppose $c = [0.5, 0, 0.3]^\top$ and let $\delta = 1/3$
Let F be uniform in $[-0.1, 0.1]$
Describe the mutated offspring of a coordinate perturbation

Convergence

Stochastic

GA are stochastic

- i.e., the algorithm contains random subroutines, so running it twice on the same problem can get different results

So, we aim for *probabilistic convergence*

- i.e., we try to prove $\mathrm{prob}(\lim_{k\to\infty} f^k_{\texttt{best}} = f^*) = 1$

Required assumptions

Definition: (Monotonicity)
The **monotonicity** assumption for GA is that the best function value for population $k+1$ is at least as good as the best function value for population k

$$\min\{f(x^i) \ : \ x^i \in P^{k+1}\} \ \leq \ \min\{f(x^i) \ : \ x^i \in P^k\}$$

Definition: (Positive probabilities - binary encoded)
Suppose points are binary encoded
The **positive probabilities** assumption states that the set of encoded points is finite and there exists $\varepsilon > 0$ such that

$$\mathrm{prob}(c \in P^k) \geq \varepsilon \quad \text{for any encoded point } c \text{ and iteration } k$$

* Positive probabilities can be rephrased for unencoded or other encodings

Monotonicity

Theorem: Suppose a GA is used such that

- each generation has at least 1 survivor picked by elitism

Then the *monotonicity assumption holds*

Positive probabilities

Theorem: Suppose a GA with binary encoding is used such that

- at least one offspring is generated per iteration
- bit inversion mutation is used

Then the *positive probabilities assumption holds*

* Similar results hold for unencoded or other encodings

Convergence

Theorem: Suppose a GA is used such that

- the points are binary encoded
- the monotonicity and positive probability assumptions hold

Let

$$f^* = \min_x\{f(x) \;:\; x \in \Omega\}, \qquad X^* = \operatorname{argmin}_x\{f(x) \;:\; x \in \Omega\},$$
$$f^k_{\texttt{best}} = \min\{f(x^i) \;:\; x^i \in P^k\} \text{ and } \quad x^k_{\texttt{best}} \in \operatorname{argmin}\{f(x^i) \;:\; x^i \in P^k\}.$$

Then

$$\mathrm{prob}\left(\lim_{k\to\infty} f^k_{\texttt{best}} = f^*\right) = 1 \quad \text{and} \quad \lim_{k\to\infty} \mathrm{prob}(P^k \cap X^* \neq \emptyset) = 1.$$

* Similar results hold for unencoded or other encodings

Final Remarks

Genetic Algorithms

Pros:

- Very popular
- Very flexible
- Theoretical proof that GA probably converges

Cons:

- Requires infinite time
- No way to know when you can stop
  - Popular stopping methods are fixed budget and improvement stagnation

Conclusion

- Not really a DFO method
- Can be useful for BBO

Assignment 3

Assignment 3

MATH 462

- Textbook # 4.1, 4.4, 4.5, 4.8, 4.9

COSC 419K

- Textbook # 4.2, 4.3, 4.4, 4.5, 4.8

MATH 562

- All MATH 462 and COSC 419K questions
- Textbook # 4.6b, 4.11