Introduction to Evolutionary Computation

Dr. David Fernández Barrero
Dra. Mª Dolores Rodríguez Moreno

Universidad de Alcalá
ISG

Objectives

Specific Objectives

- Introduce artificial evolution
- Justify the utility of artificial evolution from an engineering perspective
- Overview the components of an Evolutionary Algorithm

Source

- Stuart Russell & Peter Norvig (2009). Artificial Intelligence: A Modern Approach. (3rd Edition). Ed. Pearsons.
- Eiben, A.E. and Smith, J. E. (2003). Introduction to evolutionary computing. Springer

Outline

- **Introduction**
- Historical review
- Biological Background
- Evolutionary Algorithm
- Case studies
- Conclusions

Introduction (I)

- Basic Evolutionary Computation (EC) approach
  - Evolutionary strategies (Rechenberg, 1973)
  - Evolutionary programming (Fogel et al., 1966)
  - Genetic algorithms (Holland, 1975)
  - Genetic programming (Koza, 1992, 1994)
- EC techniques abstract evolutionary principles into algorithms that may be used to search for optimal solutions to a problem
- Evolutionary search is generally better than random search and is not susceptible to the hill-climbing behaviors of gradient-based search

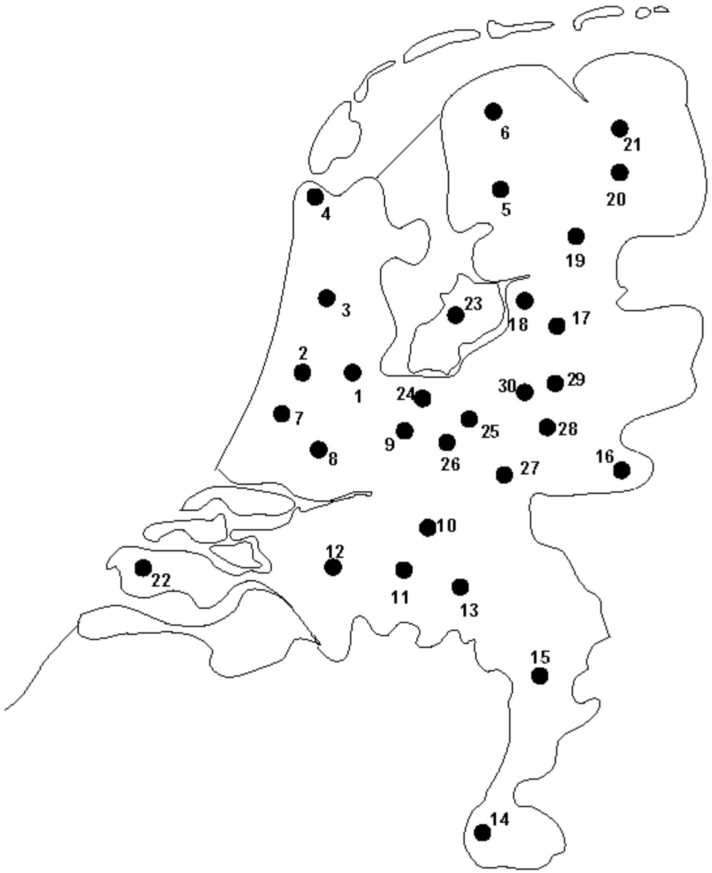
Introduction (II)

- Examples of applications
  - Optimization: the model and the O is known, we need to find the I
    - I.e.: The TSP (shortest tour around cities). The desired output property is optimality, that is, minimal length, and we are looking for inputs realising this
  - Modelling: I and O are known, we need to know the model of the system.
    - The stock exchange: Dow Jones index is the O, economic index as I. If we can find a correct model for the known data (the past) we can have a prediction tool for the value of the Dow Jones for new data
  - Simulation: the model and the I are known.
    - An electronic circuit for signal filtering (this is cheaper than building it)

Introduction (III)

- Problem:
  - Given n cities
  - Find a complete tour with minimal length
- Encoding:
  - Label the cities 1, 2, … , *n*
  - One complete tour is one permutation (e.g. for n =4 [1,2,3,4], [3,4,2,1] are OK)
- Search space is BIG:

  for 30 cities there are $30! \approx 10^{32}$ possible tours



Outline

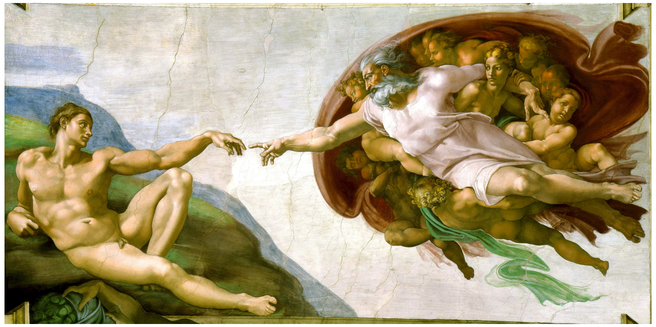
- Introduction
- **Historical review**
- Biological Background
- Evolutionary Algorithm
- Case studies
- Conclusions

Historical review (I)

- Anaximander of Miletus (610 – 546 BC)
  - First animals come from water
  - Man come from fishes
- Plato (428/427 – 348/347 BC)
  - Demiurgo created the cosmos
  - Theory of Ideas
- Aristotle (384 – 322 BC)
  - Spontaneous generation
  - Strong influence in Europe

Historical review (II)

- Creationism: God created all the species
  - Literal interpretaton of the Genesis
  - Species are hierarchical
  - Man has a superior position
- Main school in Europe for centuries



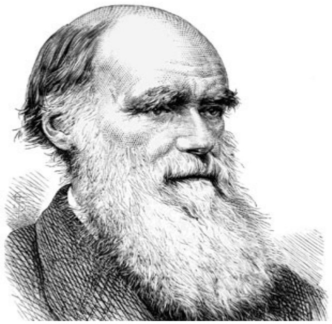
Historical review (III)

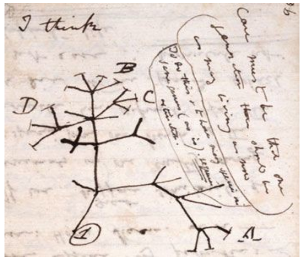
- Georges Louis Leclerc (1707 - 1788)
  - Especulated that species change
  - Noticied the similarities between men and apes
  - Could not provide a theory
- Jean-Baptiste Lamarck (1744 - 1829)
  - First to propose a theory of evolution
  - Transmutation of Species
  - Use strengthens/weakens organs
  - Heritability of acquired characteristics

Historical review (IV)

- Charles Darwin (1809-1882)
  - Published in "On the Origin of the Species" in 1859

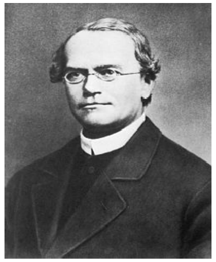
    ("On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life")
- Introduced natural selection … and applies it to human being
  - Natural selection = Variability + selection
- He did not explain the source of variation

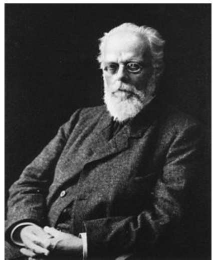


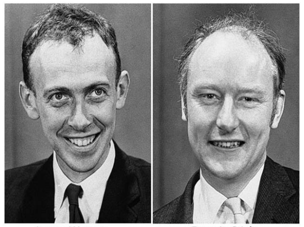


Historical review (V)

- Gregor Mendel (1822 - 1884)
  - Mendelian inheritance
  - Recessive and dominant traits
- August Weismann (1834 - 1914)
  - Germ plasm theory
  - Germ and somatic cells
  - End of Lamarkism
- J. Watson (1928) and F. Crick (1916 - 2004)
  - Discovery of DNA
  - Central Dogma of molecular biology (cells system interactions)







James Watson Francis Crick

Historical review (VI)

- Neo-Darwinism: Darwin + Mendel + Weismann
  - … also called Theory of Evolution
  - Variability + selection = evolution
- There is variation among individuals
  - Sexual reproduction, mutation and gene flow
- There is a selection of those individuals
  - Natural selection
  - Artificial selection
  - Sexual selection
  - Genetic drift (deriva genética)
- The fittest is the one that survives (not the strongest!)

Outline

- Introduction
- Historical review
- **Biological Background**
- Evolutionary Algorithm
- Case studies
- Conclusions

Biological Background (I)

- In nature, the evolutionary process occurs when the following four conditions are satisfied:
  1. An entity has the ability to reproduce itself
  2. There is a population of such self-reproducing entities
  3. There is some variety among the self-reproducing entities
  4. Some difference in ability to survive in the environment is associated with the variety
- Entities that are better able to perform tasks in their environment (i.e. fitter individuals) survive and reproduce at a higher rate (*Darwin postulation*)

Biological Background (II)

- All living organisms consist of cells
  - In each cell there is the same set of chromosomes
  - A chromosome consists of genes (blocks of DNA)
- Each gene encodes a particular protein (or trait) i.e. color of eyes
  - Possible settings for a trait (e.g. blue, brown) → alleles
  - Each gene has its own position in the chromosome→ locus
- Complete set of genetic material (all chromosomes) → **genome**
- Particular set of genes in the genome → **genotype**
- The genotype is with later development after birth base for the organism's **phenotype**

Biological Background (III)

- During reproduction, occurs recombination or crossover
- Genes from parents form in some way the whole new chromosome
- The new created offspring can then be mutated: the elements of DNA are a bit changed. This changes are mainly caused by errors in copying genes from parents
- The **fitness** of an organism is measured by success of the organism in its life

Biological Background (IV)

- Given a population ...
  1. There are differences among individuals
  2. Fittest individuals more likely to reproduce
  3. Go to 1
- We are interested in applying this to Engineering
- Biological evolution is, in essence, an optimization algorithm
  - ... it optimizes the survival probability
  - Optimizing is *search*ing the maximum

Outline

- Introduction
- Historical review
- Biological Background
- Evolutionary Algorithm
  - **Basics**
  - Exploration and exploitation
  - Components
- Case studies
- Conclusions

Basics (I)

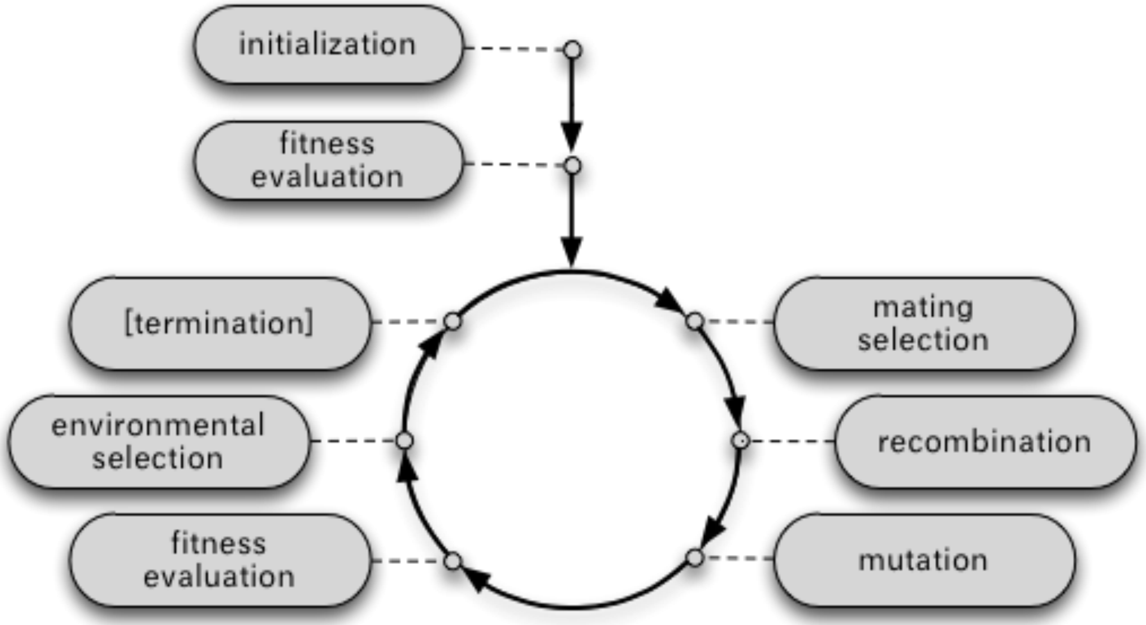
- A successor state is generated by combining two parent states
- Start with *k* randomly generated states (population)
- Evaluation function (fitness function) → Higher values for better states
- Produce the next generation of states by selection, crossover, and mutation

Basics (II)

- Large number of EAs
  - There is no “canonical” algorithm
  - They all imitate biological evolution
  - They use a population, where each individual represents a (potential) solution
  - Multiple representations
- Population is modified
  - Mutation (1 individual)
  - Crossover (>1 individuals)
  - Multiple genetic operators
- Selection imitates natural selection, based on a fitness function
- Iterative process

Basics (III)

Possible basic algorithm



Basics (IV)

- Initialization is usually random
  - Random population
  - Domain-dependent heuristics may be used
  - Known solutions might be injected into the initial population
- Termination criteria
  - Get a desired fitness
  - Maximum number of iterations (or generations)
  - Loss of genetic diversity
  - Lack of fitness improvement

Outline

- Introduction
- Historical review
- Biological Background
- Evolutionary Algorithm
  - Basics
  - **Exploration and exploitation**
  - Components
- Case studies
- Conclusions

Exploration and exploitation

- Balance between exploitation and exploration
  - These are opposite objectives
  - Need of trade-off
- Exploration: Search of new regions
  - Explore the search space
  - Performed, mostly, by mutation
- Exploitation: Search of local (or global) maximum
  - Exploit the acquired knowledge
  - Performed, mostly, by crossover

Outline

- Introduction
- Historical review
- Biological Background
- Evolutionary Algorithm
  - Basics
  - Exploration and exploitation
  - **Components**
- Case studies
- Conclusions

Components

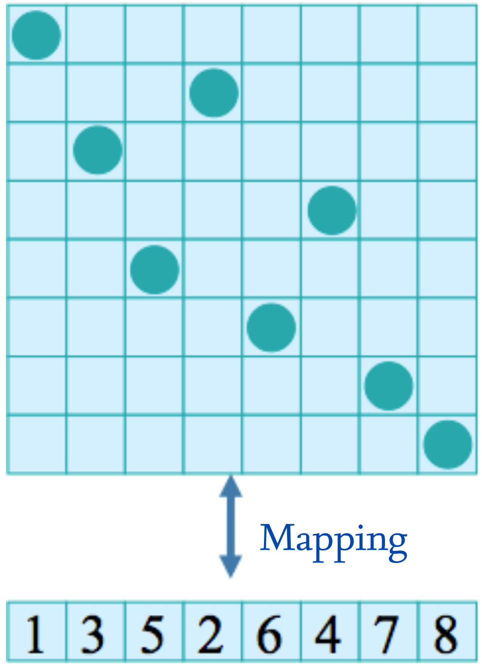
- Representation
- Evaluation
- Selection
- Genetic operators

Representation

- What type?
  - Strings: Genetic Algorithms (GA)
  - Real vectors: Evolution Strategies (ES)
  - State machine: Evolutive Programming (EP)
  - Trees: Genetic Programming (GP)
- How to select parents for crossover?
  - The better parents with better fitness (in hope that the better parents will produce better offspring)
  - Making new population only by new offspring can cause lost of the best chromosome from the last population → **elitism** is often used
  - At least one best solution is copied without changes to a new population, so the best solution found can survive to end

Representation: Example (I)

- Example: 8 queens with a GA
  - Phenotype: Board position
  - Genotype: Integer vector



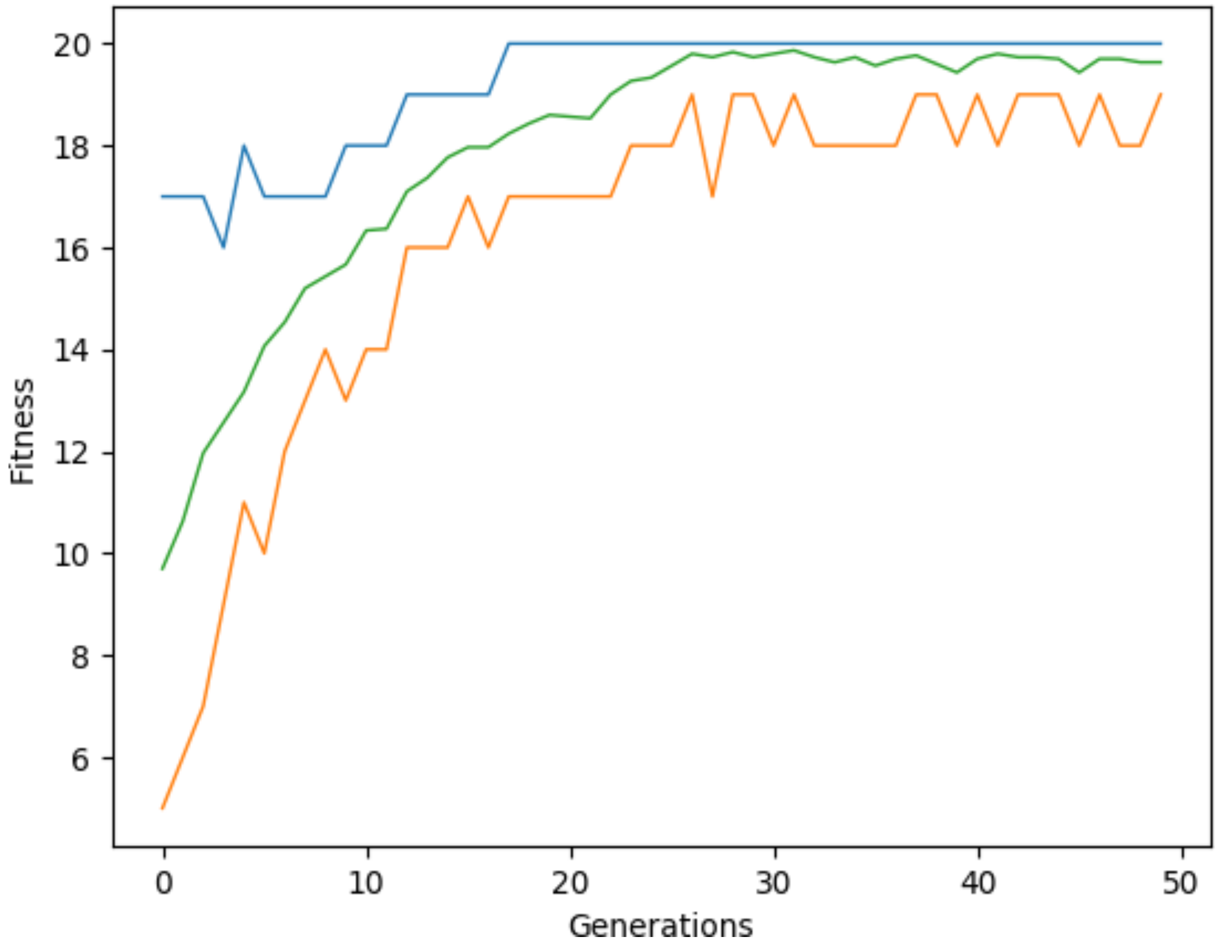
Evaluation

- Individuals quality is assessed by a fitness function
  - Individual = Potential solution
- The fitness assigns a numerical value to a phenotype
  - Caution, phenotype, not genotype
  - Multi-objective algorithms use several fitnesses
- Evaluation uses to be a bottleneck
  - Many times it involves simulating a system
  - Minimize number of evaluations

Evaluation: Example

- Example: 8 Queens
  - The fitness may be the number of threaded pieces
  - Objective: Minimize fitness (minimization problem)
- Example: Regression
  - The fitness may be the quadratic average error
  - Objective: Minimize fitness (minimization problem)

Fitness evolution



(Car design)

Selection (I)

- Selection operator "selects" individuals for reproduction
  - Imitates natural selection
  - Higher reproduction probability for high fitness individuals
  - Randomness helps avoiding local minima
- Selection is done in phenotypic space!
- Selection does not take into account how representation is
- Introduces selective pressure

Selection (II)

- High selective pressure reduces genetic diversity
  - Faster evolution, higher probability of local maxima
  - Eliminates low fitness individuals
  - Potentially valuable genetic material can be lost
  - Selection operators: Tournment size n, roulette-wheel, rank-based, …

Tournment size *n*

1. Take randomly *n* individuals
2. Compute their fitness
3. Select the highest fitness

Variable selective pressure depending on *n*

Selection (III)

- Replacement strategy: select which individual replace
- Two basic strategies
    - Generational algorithms: replace all the offspring
        - Iterations are named generations
        - Time is usually measured in generations
    - Steady-stade: replace part of the offscript
        - Criteria: Age, fitness, selection, etc
        - Lower memory consumption
- Hybrid strategy: Elitism
    - Replace the population, except the n fittest individuals
    - n fittest individuals guaranteed to survive

Genetic operators

- Genetic operators build new individuals
- Two basic operators:
  - Mutation
  - Crossover
- Open discussion (=research) about their role
  - Mutation enhances exploration
  - Crossover enhances exploitation
- Both are used

Genetic operators: Mutation

- It takes a genotype and returns another one
  - It has a stochastic behaviour
  - Used to maintain genetic diversity
- Guarantees search space connectivity
- It plays a disruptive role
- Moves population to new regions

Genetic operators: Crossover

- Fuse information from the parents (sexual reproduction)
  - Randomness has a place
- Offspring uses to be worse than its parents
  - With luck, good components of the parents are joined ...
  - ... and this is something that happens
- Crossover has a constructive role
  - Join pre-existent components
  - Does not generate new genetic material
  - Encourages exploitation

Examples

- (Genetic Algorithm Walkers)
- (Smart rockets)
- (Learn to walk)
- (Flexible Muscle-Based Locomotion for Bipedal Creatures)
- (MarI/O - Machine Learning for Video Games)
- (A genetic algorithm learns how to fight!)
- (Evolved Electrophysiological Soft Robots)

Outline

- Introduction
- Historical review
- Biological Background
- Evolutionary Algorithm
- **Case studies**
- Conclusions

Case Study I: Restaurant

- **Hamburger Restaurant:** a strategy for running a restaurant will consist of making three binary decisions that produces the highest profit
  - Price for hamburger
    - 1 = $ 0.50 price
    - 0 = $10.00 price
  - Drink
    - 1 = Coca Cola
    - 0 = Wine
  - Ambiance
    - 1 = Fast snappy service
    - 0 = Leisurely service with tuxedoed waiter

Search Space

- Alphabet size K=2, Length L=3
- Size of search space: $K^L=2^L=2^3=8$

1	000
2	001
3	010
4	011
5	100
6	101
7	110
8	111

Initial Population

Restaurant number	Price	Drink	Speed	Representation
1	High	Cola	Fast	011
2	High	Wine	Fast	001
3	Low	Cola	Leisurely	110
4	High	Cola	Leisurely	010

Generations

Generation 0		**Generation 1**
Individuals	**Fitness**	**Offspring**
011	\$3	
001	\$1	
110	\$6	
010	\$2	

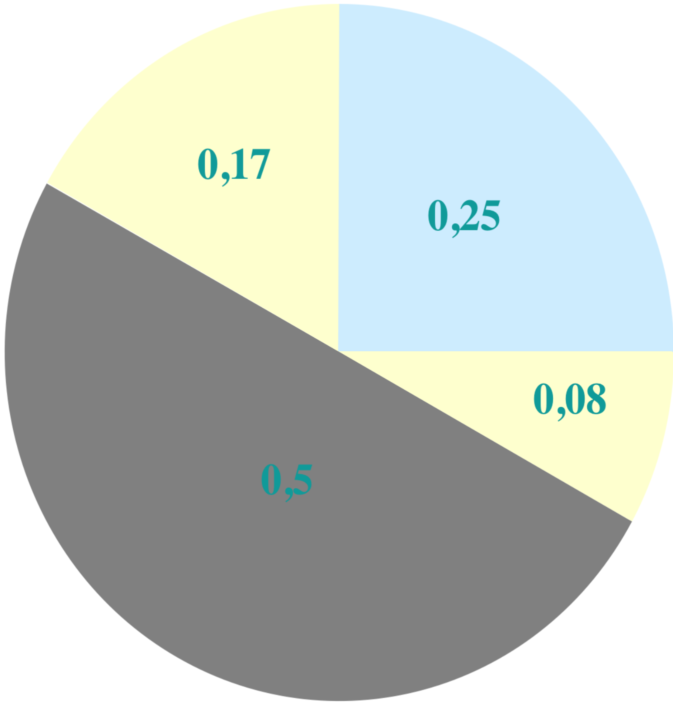
Generation 0

	Generation 0		
1	011	3	
2	001	1	
3	110	6	
4	010	2	
Total			
Worst			
Average			
Best			

Probabilistic selection based on fitness

- Better individuals are preferred
- Best is not always picked
- Worst is not necessarily excluded
- Nothing is guaranteed

Probabilistic selection based on fitness



Fitness-proportionate reproduction

	Generation 0			**Mating pool**	
1	011	3	.25	011	3
2	001	1	.08	110	6
3	110	6	.50	110	6
4	010	2	.17	010	2
Total		12			17
Worst		1			2
Average		3.00			4.5
Best		6			6

Mutation operation

- Parent chosen probabilistically based on fitness

Parent
010

- Mutation point chosen at random

Parent
--0

- One offspring

Offspring
011

After mutation

	Generation 0			Mating pool		Generation 1		
1	011	3	.25	011	3			
2	001	1	.08	110	6			
3	110	6	.50	110	6			
4	010	2	.17	010	2	---	011	3
Total		12			17			
Worst		1			2			
Average		3.00			4.5			
Best		6			6			

Crossover …

- 2 parents chosen probabilistically based on fitness

Parent 1	Parent 2
011	110

Crossover

- Interstitial point picked at random

Fragment 1	Fragment 2
01-	11-

- 2 remainders

Remainder 1	Remainder 2
- - 1	- - 0

- 2 offspring produced by crossover

Offspring 1	Offspring 2
111	010

After crossover

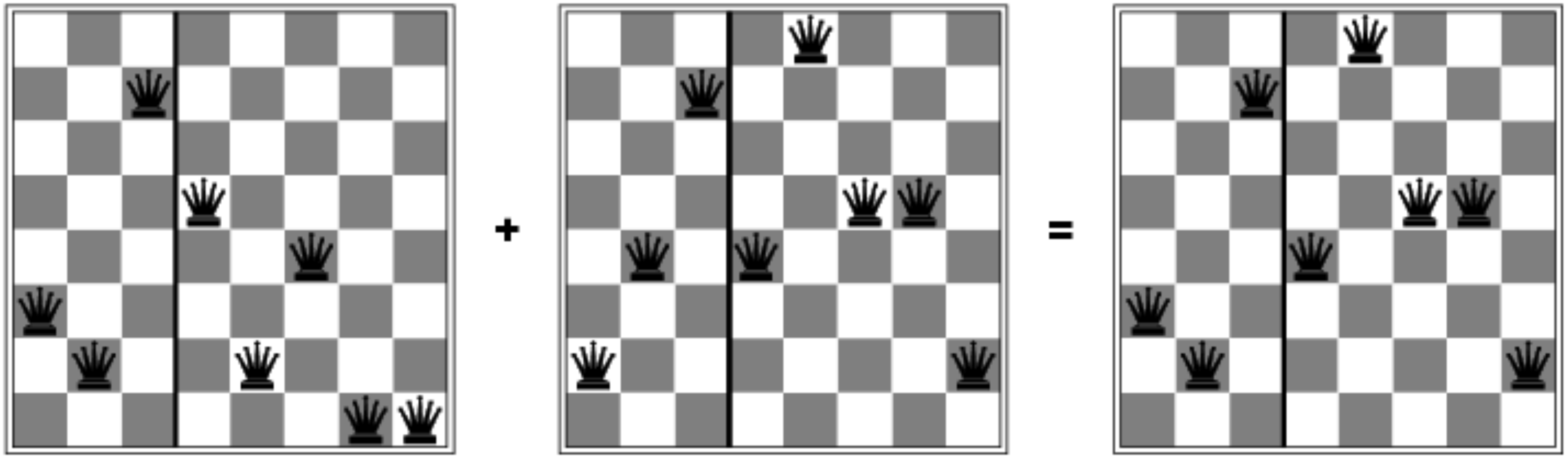
	Generation 0			Mating pool		Generation 1		
1	011	3	.25	011	3	2	111	7
2	001	1	.08	110	6	2	010	2
3	110	6	.50	110	6	---	110	6
4	010	2	.17	010	2	---	011	3
Total		12			17			18
Worst		1			2			2
Average		3.00			4.5			4.5
Best		6			6			7

Genome of the global optimum

McDONALD's

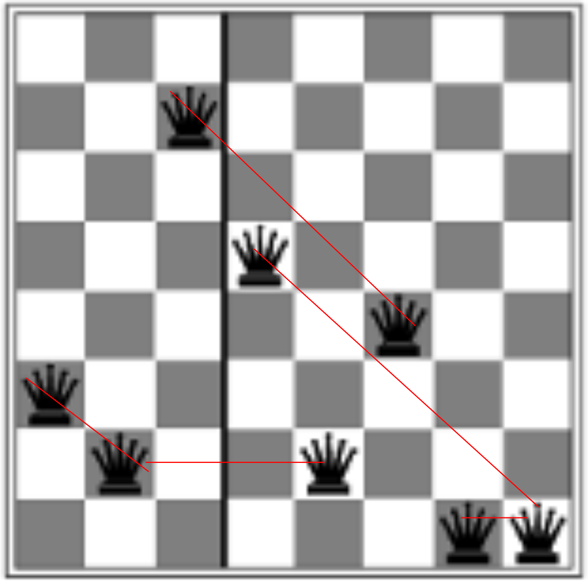
1	**1**	**1**

Case Study II: 8-queens



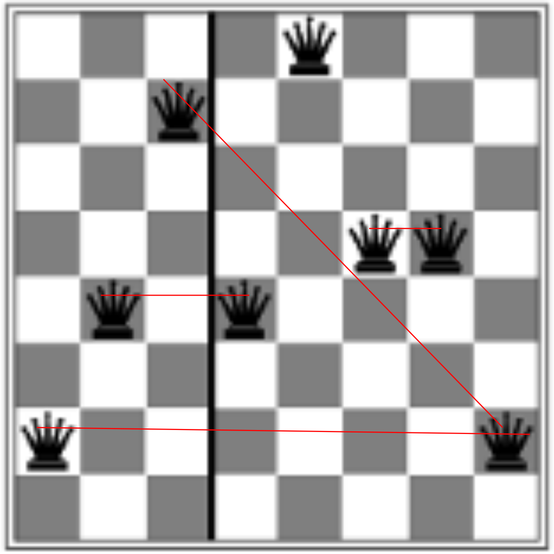
- The initial population is represented by 8-digit strings representing 8-queen states
- Fitness function: number of non-attacking pairs of queens (min = 0, max = 8 x 7/2 = 28)

Case Study II: 8-queens



- Fitness function: number of non-attacking pairs of queens (min = 0, max = 8 x 7/2 = 28)
- 327 52411 (5 attacking queens) = 28 – 5 = 23

Case Study II: 8-queens



- Fitness function: number of non-attacking pairs of queens (min = 0, max = 8 x 7/2 = 28)
- 247 48552 (4 attacking queens) = 28 – 4 = 24

Case Study II: 8-queens

- Fitness function
  - 24/(24+23+20+11) = 31%
  - 23/(24+23+20+11) = 29%
  - 20/(24+23+20+11) = 26%
  - 11/(24+23+20+11) = 14%

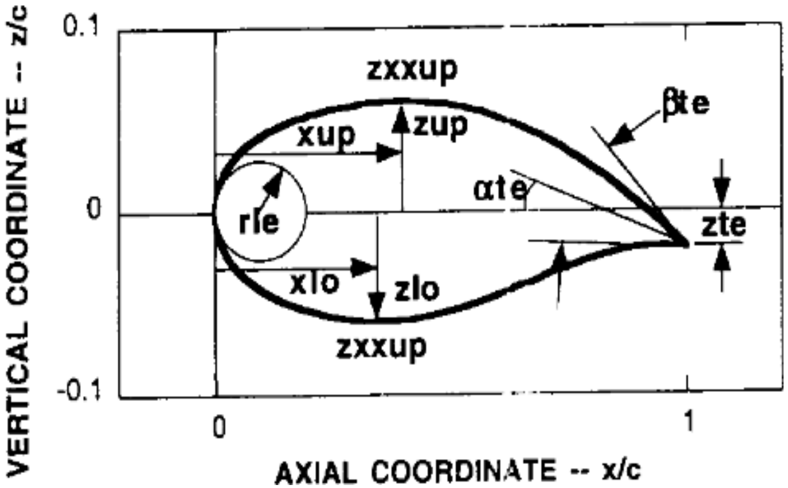
Case study III: 9th Global Trajectory Optimization Competition

- GTOC: Global Trajectory Optimization Competition
  - Proposed by ESA Advanced Concepts Team
  - Difficult trajectory optimization problems
- GTOC 9: The Kesser Run
  - 123 orbiting debris
  - Remove debris
  - Design multiple missions
- (Video) (Solution)

Case study IV: Transonic wing shape optimization

Problem: Design a wing shape for transonic flight

- Maximize lift



Holst T.L., Pulliam T.H. (2003) *Transonic Wing Shape Optimization Using a Genetic Algorithm.* In: IUTAM Symposium Transsonicum IV. Fluid Mechanics and its Applications, vol 73. Springer.

Case study V: Mars orbital insertion

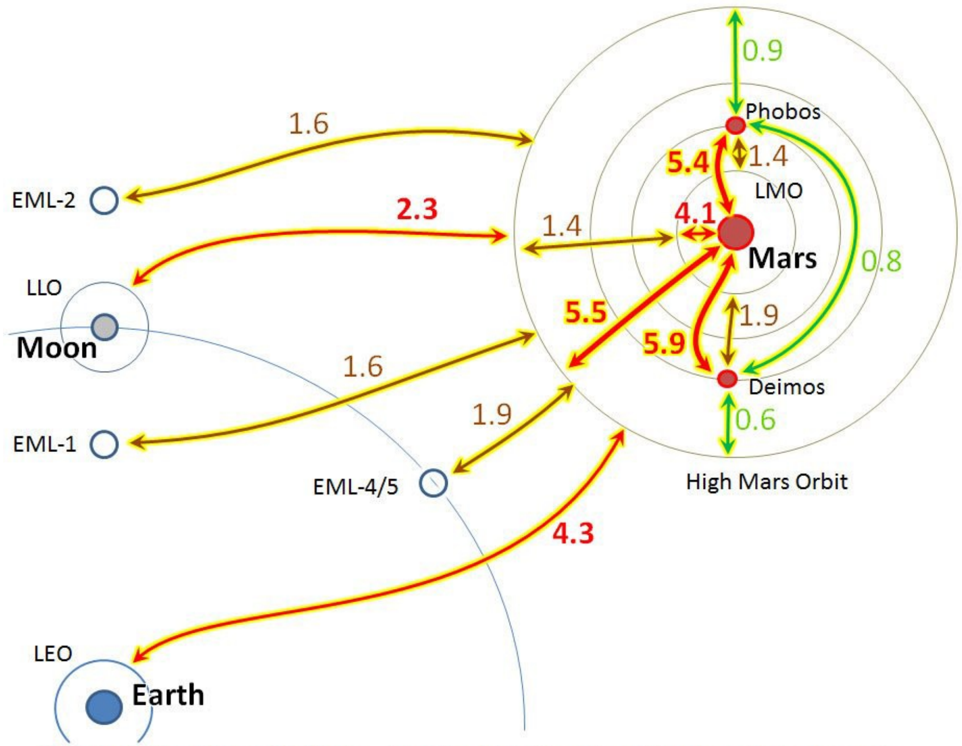


Chart by Richard Penn CC-BY, data from David Hollister hopsblog-hop.blogspot.co.uk

Outline

- Introduction
- Historical review
- Biological Background
- Evolutionary Algorithm
- Genetic Algorithms
- Case studies
- **Conclusions**

Conclusions

- EC is a search heuristic that mimics the process of natural evolution
- It uses techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover
- The heuristic is used to generate useful solutions to optimization and search problems