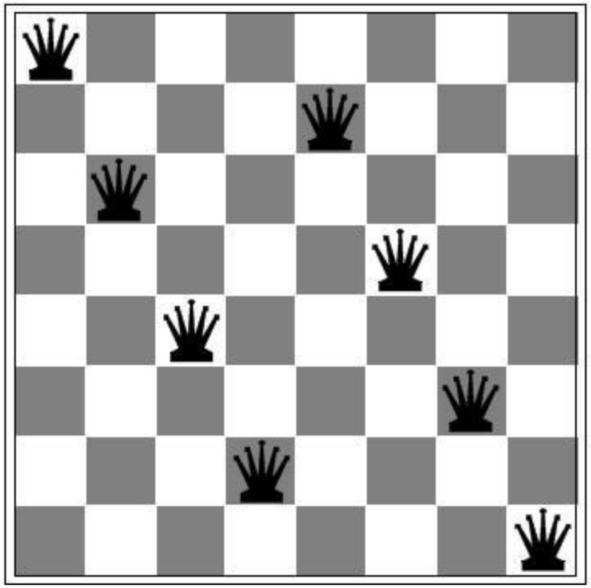
CS 561 Artificial Intelligence
Lecture

Rashmi Dutta Baruah
Dept. of Computer Science & Engineering
IIT Guwahati

Outline

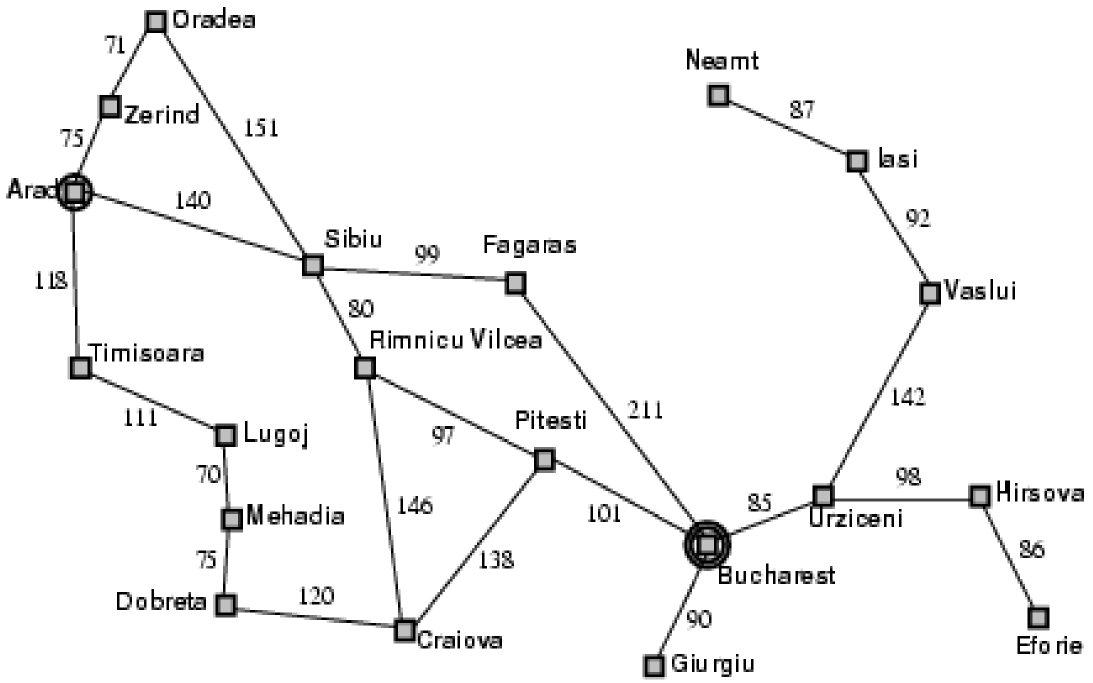
- Local Search
  - Hill climbing and variants
  - Simulated annealing
  - Local beam search and variants
  - Genetic algorithms

Introduction



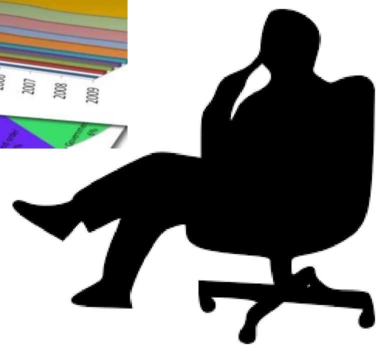
8-queens problem

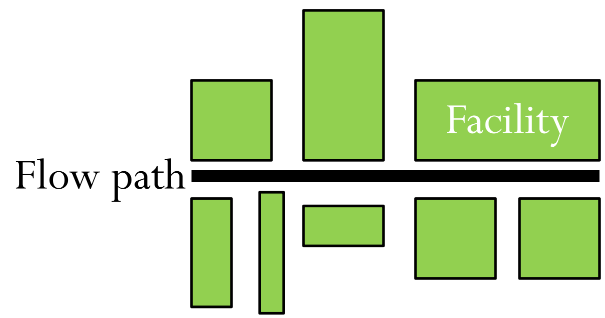
Route finding problem



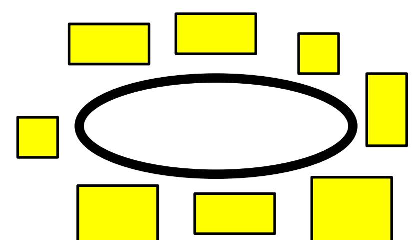
Introduction

Portfolio management





Facility layout problem



Local search algorithms

- In many optimization problems, the path to the goal is irrelevant; the goal state itself is the solution.
- Find configuration (state) satisfying constraints, e.g., n-queens
- In such cases, we can use local search algorithms keep a single "current" state, try to improve it

Hill-climbing search

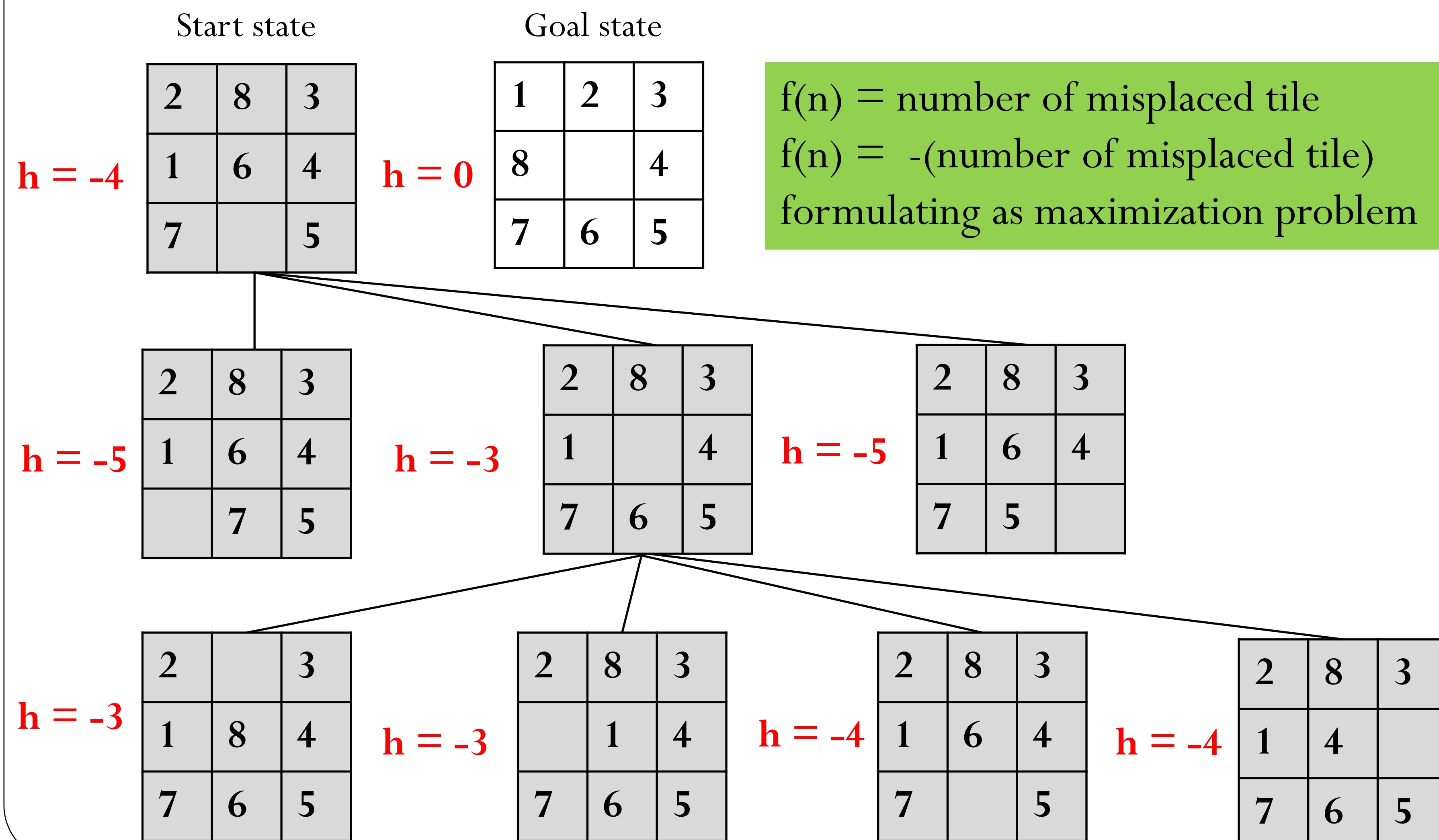
- Idea: continuously move in the direction of up-hill until "peak" is reached.
- At each step the current node is replaced by the best neighbour, neighbour with the highest VALUE (steepest ascent)
- Stop when the neighbour is not better than the current node and return the current node.

Example: hill climbing 8-puzzle problem

f(n) = number of misplaced tile

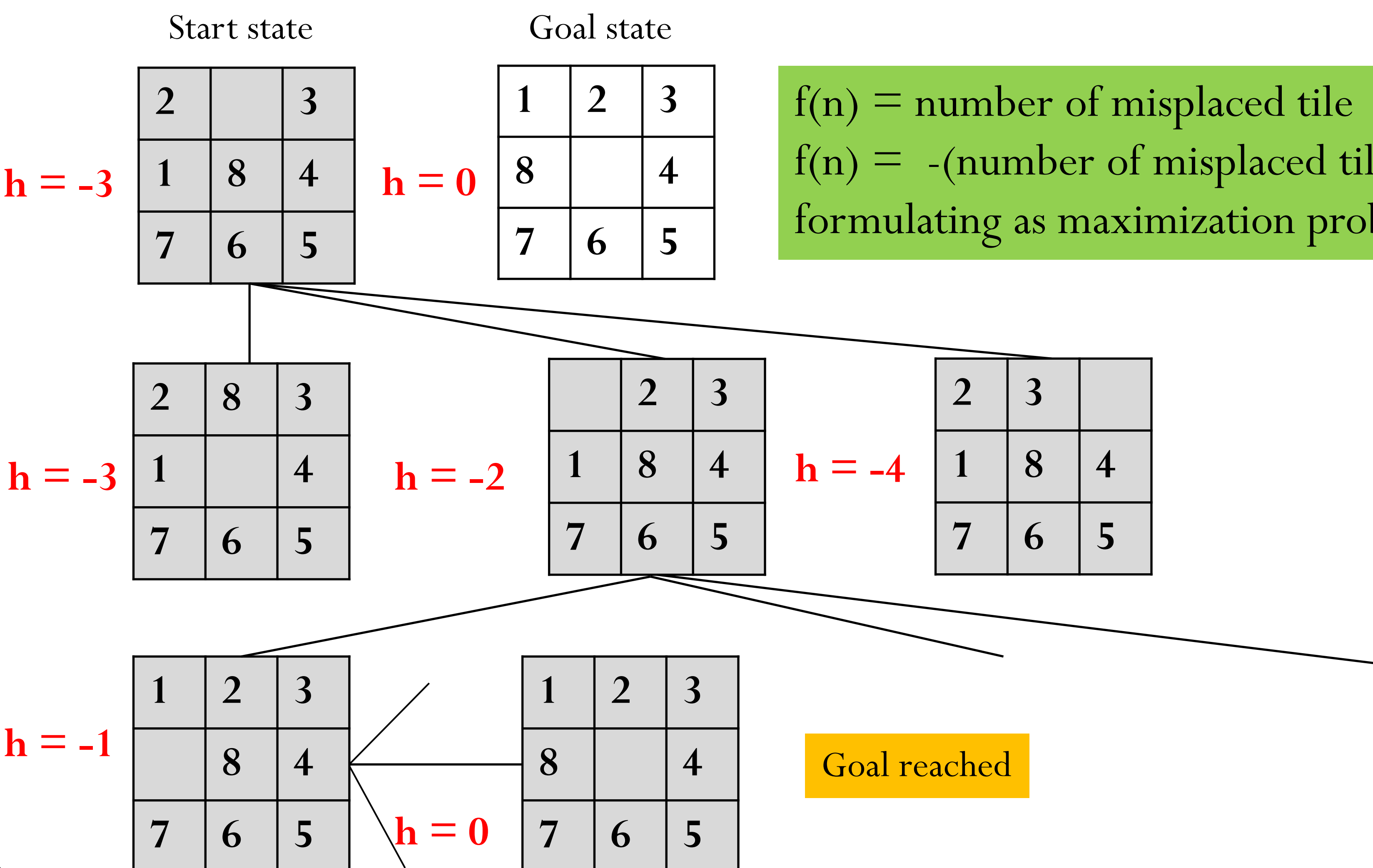
f(n) = -(number of misplaced tile)

formulating as maximization problem



Example: hill climbing 8-puzzle problem

f(n) = number of misplaced tile
f(n) = -(number of misplaced tile)
formulating as maximization problem

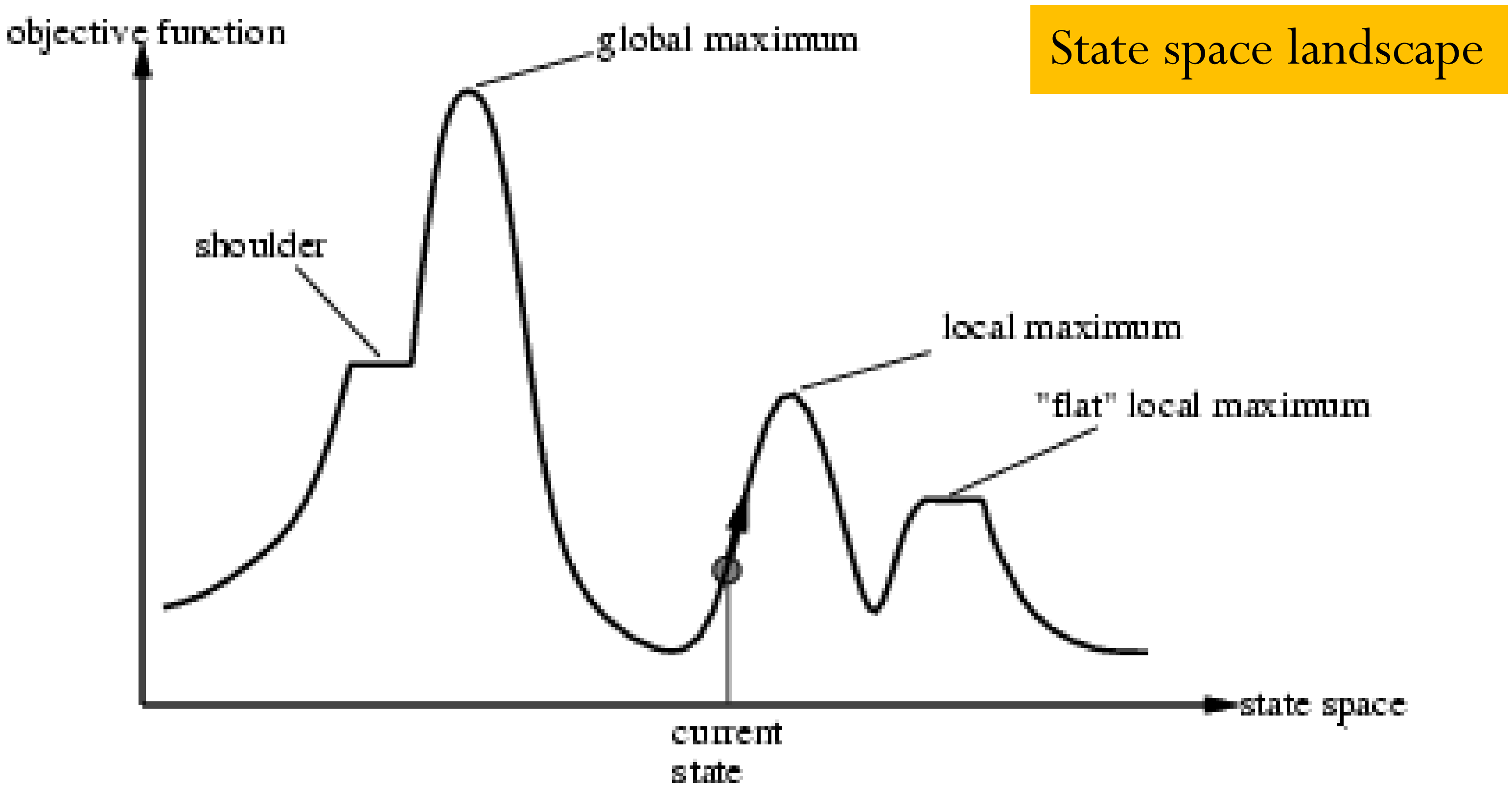


Hill-Climbing search

- Complete?
  - NO
- Optimal ?
  - NO
- Time ?
  - Linear (length of the steepest gradient ascent)
- Memory?
  - Constant (needs to keep only current node)

Hill-climbing search

- Problem: depending on initial state, can get stuck in local maxima



Hill-climbing variations

- Stochastic hill-climbing
  - Choses at random from among the up-hill moves
- Random restart hill-climbing
  - Conducts a series of hill-climbing searches from randomly generated initial states until a goal is found.

Success of hill-climbing depends on the shape of the state-space landscape

Simulated annealing search

- Idea of SA: escape local maxima by allowing some "bad" moves but gradually decrease their frequency
- It is motivated by physical annealing process where the aim is to reach to a low energy state so that a material obtains a crystalline solid structure.
- The process involves melting the substance at high temperature and then lowering the temperature slowly, spending more time near freezing point.

Physical Annealing →	**Simulate Annealing (minimisation)**
System states	Feasible solutions
Energy	Cost
Change of state	Neighbouring solutions
Temperature	Control parameter
Frozen state	Solution

Simulated annealing search

- Select a state (from candidate solutions) randomly as *current* (or start) state.
- Loop until a termination criterion is satisfied:
  - Get the temperature T from a *cooling schedule*.
  - Select a successor/neighbour of the current state randomly.
  - Compute the change in energy
    $$\Delta E = f(current) - f(neighbour)$$
  - If $\Delta E > 0$ then accept neighbour as current sate otherwise calculate the probability of accepting neighbour (bad move)
    $P = \exp\left(\frac{\Delta E}{\mathrm{T}}\right)$

Simulated annealing search

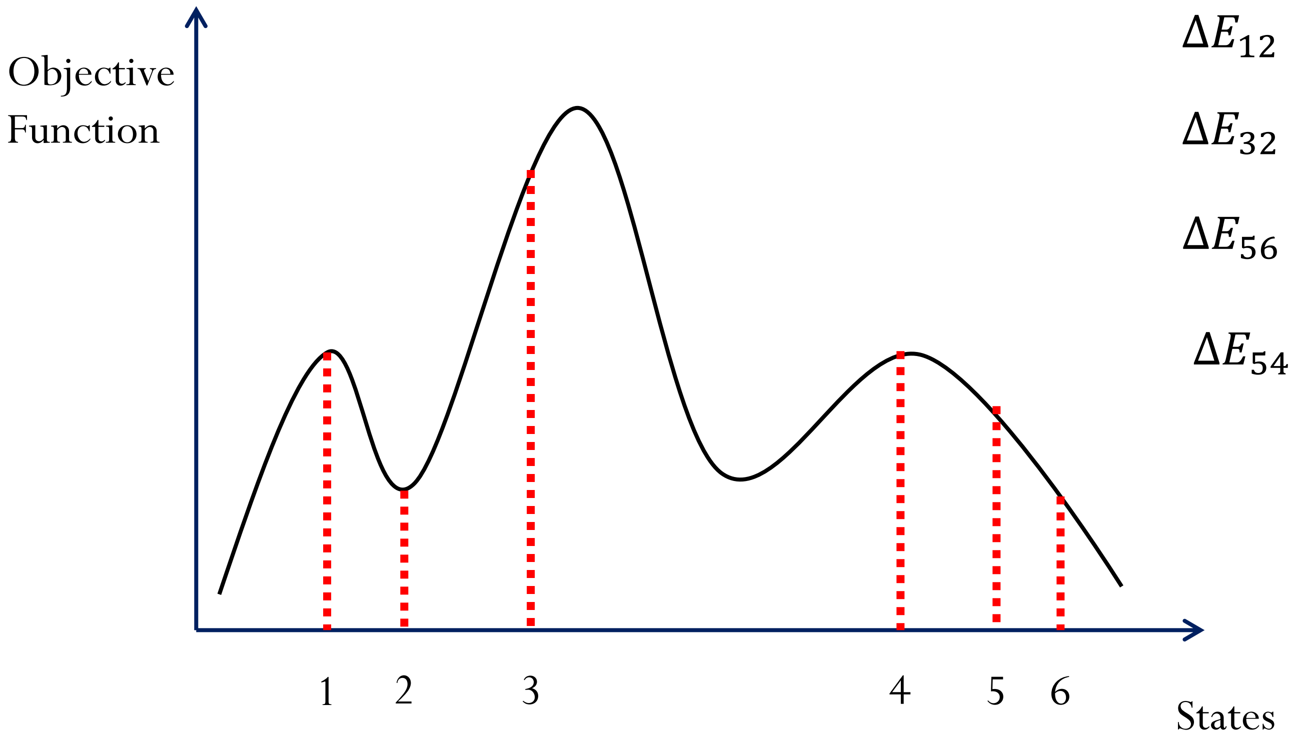
ΔE	$\exp(\Delta E\ )$	$\exp\left(\frac{\Delta E}{\mathrm{T}}\right)$, $\mathrm{T} = 100$	$\exp\left(\frac{\Delta E}{\mathrm{T}}\right)$, $\mathrm{T} = 10$
-1	0.3679	0.9900	0.9048
-2	0.1353	0.9802	0.8187
-3	0.0498	0.9704	0.7408
-4	0.0183	0.9608	0.6703
-5	0.0067	0.9512	0.6065

Observations:

- Probability of accepting a bad state is a function of both the temperature of the system and of the change in the evaluation function.
- As the temperature of the system decreases the probability of accepting a worse move is decreased.

Simulated annealing search

- What can you say about the states 1, 2, 3, 4, 5, 6 ?



Properties of simulated annealing search

- One can prove: If *T* decreases slowly enough, then simulated annealing search will find a global optimum with probability approaching 1

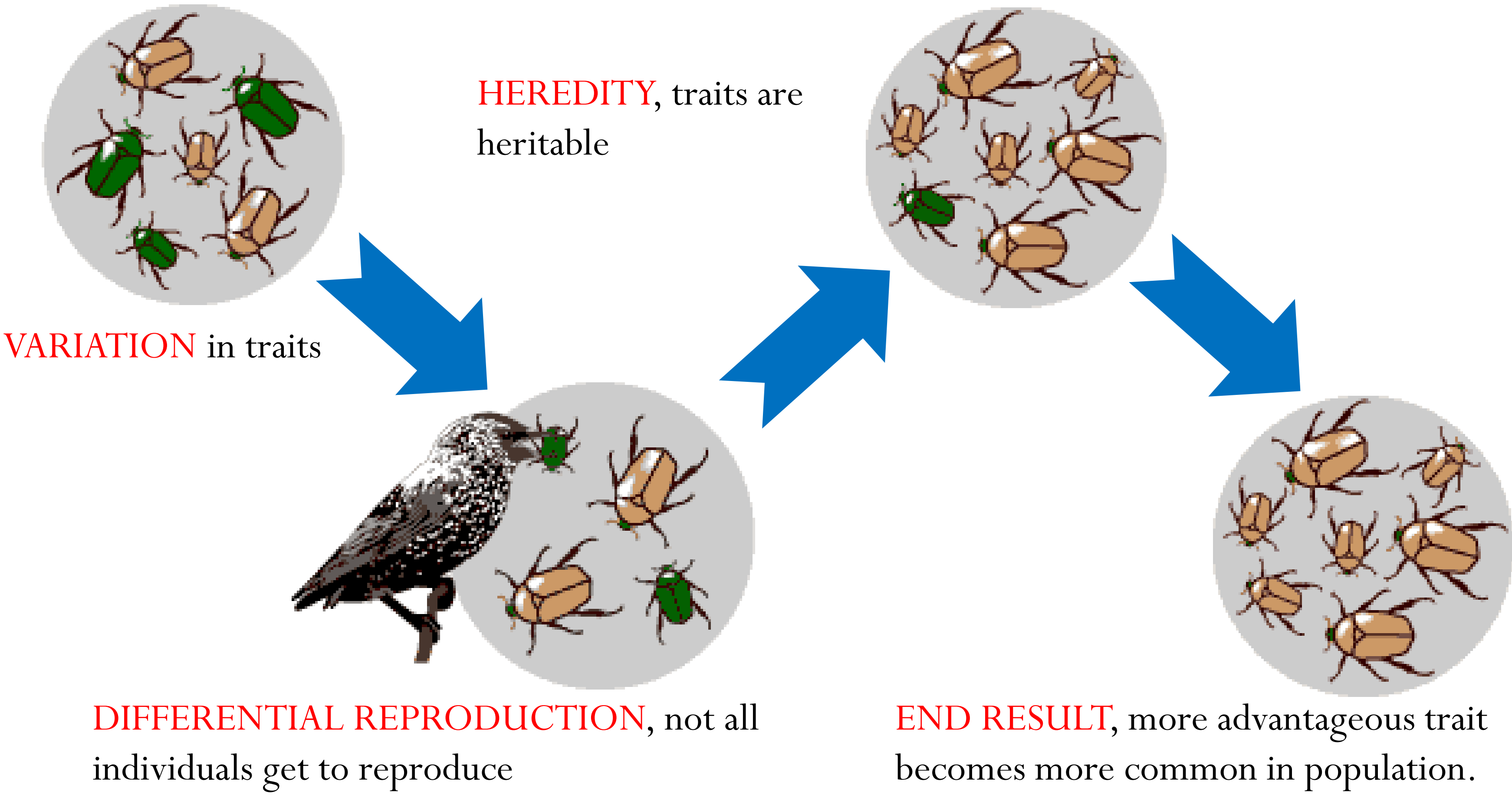
- used in airline scheduling, factory scheduling and other large-scale optimization problems.

Local beam search

- Keep track of *k* states rather than just one (k is the beam width).
- Start with *k* randomly generated states.
- At each iteration, all the successors of all *k* states are generated.
- If any one is a goal state, stop; else select the *k* best successors from the complete list and repeat.

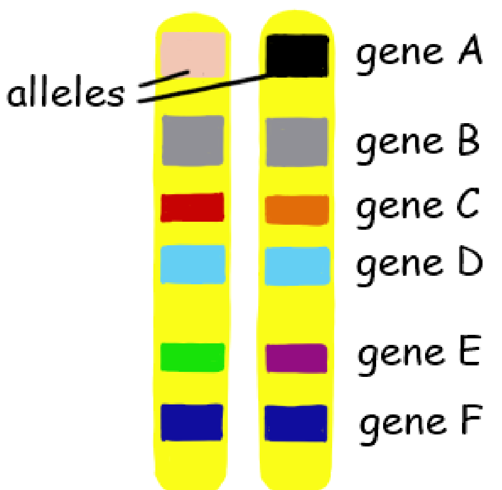
Genetic algorithms

- Genetic Algorithms (GAs) are based on the principles of natural evolution (natural selection or survival of the fittest).



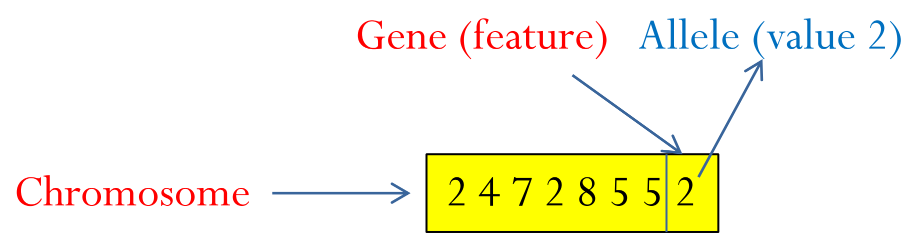
Source: University of California Museum of Paleontology's Understanding Evolution (http://evolution.berkeley.edu)

Natural Vs. GA terminologies



homologous chromosomes

* carry the same sequence of genes;
* however, not necessarily the same alleles for those genes.



Chromosomes are composed of genes, which may take on some number of values called alleles.

Natural	Genetic Algorithm
Chromosomes	Strings
Gene	Feature, Character (or variable)
Allele	Feature value

Figure Source: http://ibbiologyhelp.com/Genetics/chromosomes.html

Genetic algorithms

- How are genetic algorithms different from the search methods we have seen so far?

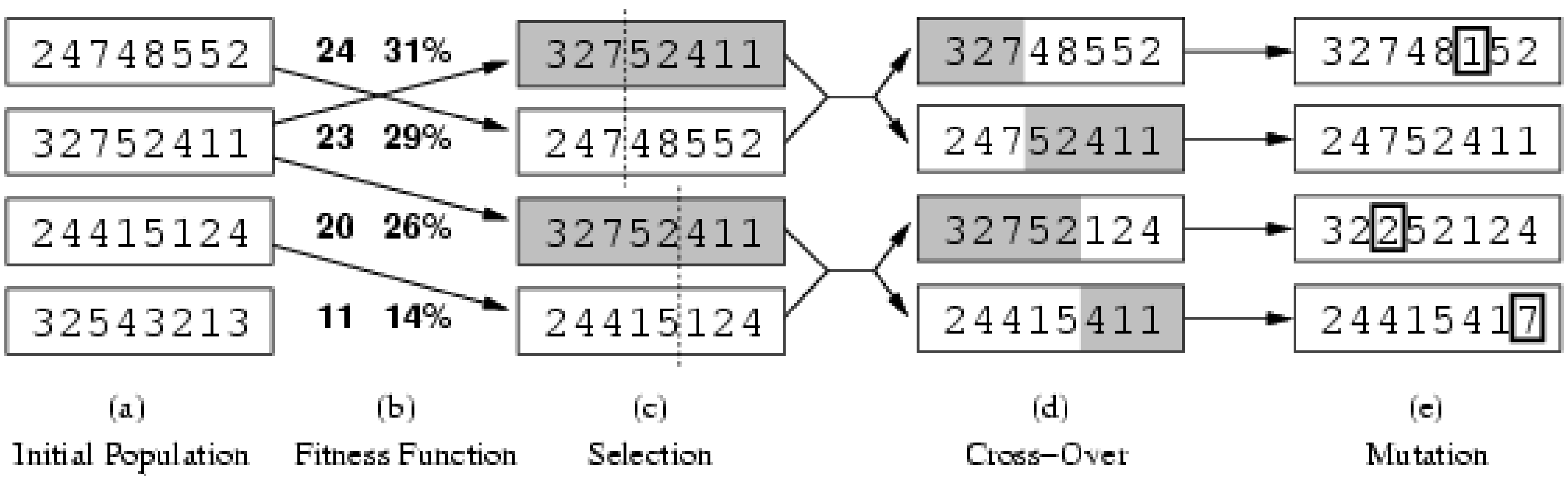
  - Sates or solutions are coded as strings
    - A state is represented as a string over a finite alphabet (often a string of 0s and 1s).
  - Search from a population of solutions
    - Start with *k* randomly generated states (population).
  - Do not require auxiliary information
    - Only requires Evaluation function (fitness function). Higher values for better states.
  - Randomized operators
    - A successor state is generated by combining two parent states. Produce the next generation of states by selection, crossover, and mutation.

GA Algorithm

- Initialise a population with k randomly generated states
- Evaluate each state (individual) in the population
- Loop until some terminating condition is satisfied
  - generate new states by mating states in the current population (using crossover and mutation)
  - delete members of the existing population to make way for the new members
  - evaluate the new members and insert them into the population
- Return the best state as the solution

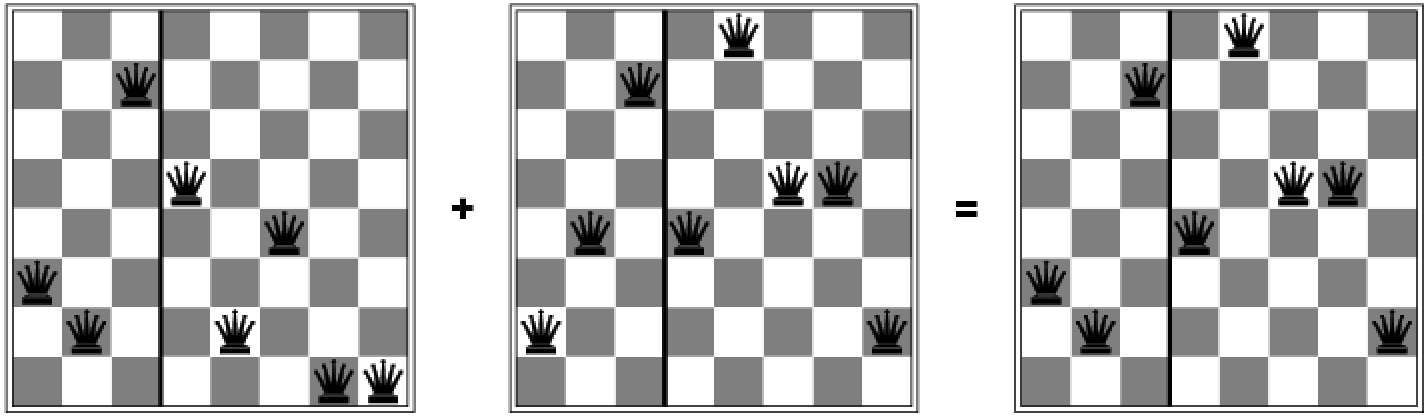
Genetic algorithms

- 8-Queens problem



- Fitness function: number of non-attacking pairs of queens (min = 0, max = $8 \times 7/2 = 28$)
- $24/(24+23+20+11) = 31\%$
- $23/(24+23+20+11) = 29\%$ etc

Genetic algorithms



What did we discuss ?

- Introduction to the problems in real-life where local search can be applied.
- Various local search algorithms and there advantages and disadvantages: Hill Climbing, Simulated annealing, Local beam search, Genetic algorithms.
- Examples of Genetic algorithms: 8-puzzle problem, function maximization.