Nature-inspired Algorithms

Lecture 2

Algorithm Engineering Group
Hasso Plattner Institute, University of Potsdam

15 May 2017

Let's look at different approaches

Assumptions

1. Solutions encoded as length-n bitstrings (elements of $\{0,1\}^n$),
2. want to maximize some $f\colon \{0,1\}^n \to \mathbb{R}$.

Random Search

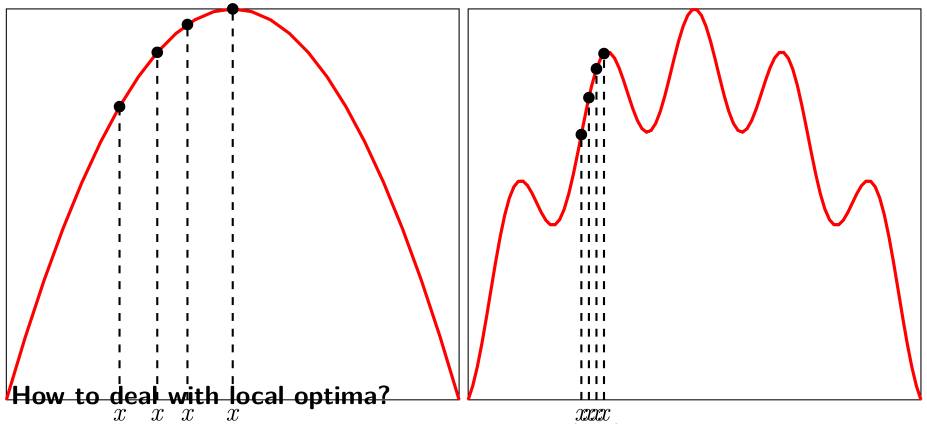
Choose x uniformly at random from $\{0,1\}^n$;
while *stopping criterion not met* **do**
  Choose y uniformly at random from $\{0,1\}^n$;
  if $f(y) \geq f(x)$ **then** $x \leftarrow y$;
end

Heuristics

Random(ized) Local Search (RLS)

Choose x uniformly at random from $\{0,1\}^n$;
while *stopping criterion not met* **do**
  $y \leftarrow x$;
  Choose i uniformly at random from $\{1,\ldots,n\}$;
  $y_i \leftarrow (1-y_i)$;
  if $f(y) \geq f(x)$ **then** $x \leftarrow y$;
end

Local Optima



How to deal with local optima?

- Restart the process when it becomes trapped (ILS)
- Accept disimproving moves (MA, SA)
- Take larger steps (EA, GA)

Simple Randomized Search Heuristics

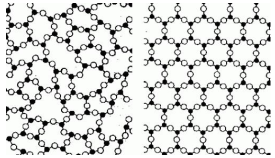
Metropolis Algorithm

Choose x uniformly at random from $\{0,1\}^n$;
while *stopping criterion not met* **do**
  $y \leftarrow x$;
  Choose i uniformly at random from $\{1,\ldots,n\}$;
  $y_i \leftarrow (1-y_i)$;
  if $f(y) \geq f(x)$ **then** $x \leftarrow y$;
  else $x \leftarrow y$ with probability $e^{(f(y)-f(x))/T}$;
end

- Method developed for generating sample states of a thermodynamic system (1953)
- T is **fixed** over the iterations

More inspiration from nature: biology to metallurgy

- want: a flawless crystal structure (lowest energy state)
- atoms move randomly, settle into low energy states as system cools
- sometimes, atoms can't move to their optimal positions (other atoms are in the way)
  - reheat the material slightly
  - provides just enough energy to "bump" atoms into place



Simulated annealing

- molecular configurations → solutions
- energy → objective function
- warm system → many random jumps
- carefully cooled system → lowest energy state (best objective)

Simple Randomized Search Heuristics

Simulated Annealing

Choose x uniformly at random from $\{0,1\}^n$;
while *stopping criterion not met* **do**
  $y \leftarrow x, t \leftarrow 0$;
  Choose i uniformly at random from $\{1,\ldots,n\}$;
  $y_i \leftarrow (1-y_i)$;
  if $f(y) \geq f(x)$ **then** $x \leftarrow y$;
  else $x \leftarrow y$ with probability $e^{(f(y)-f(x))/T_t}$;
  $t \leftarrow t+1$;
end

- Heating and controlled cooling of a material to increase crystal size and reduce their defects.
- High temperature ⇒ many random state changes
- Low temperature ⇒ system prefers "low energy" states (high fitness)
- Idea is to carefully settle the system down over time to its lowest energy state (highest fitness) by **cooling**
- T_t is **dependent on** t, typically decreasing.

How simulated annealing proceeds ...

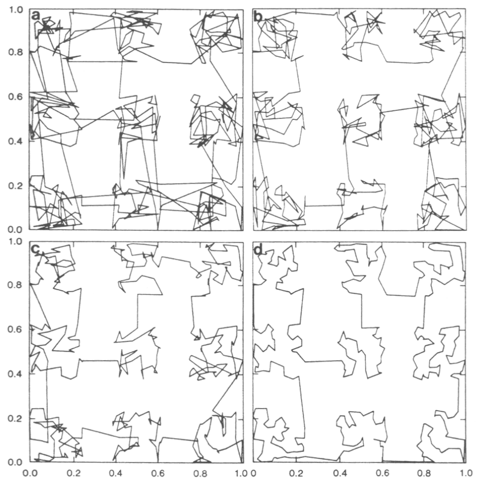


Fig. 9. Results at four temperatures for a clustered 400-city traveling salesman problem. The points are uniformly distributed in nine regions. (a) $T = 1.2$, $\alpha = 2.0567$; (b) $T = 0.8$, $\alpha = 1.515$; (c) $T = 0.4$, $\alpha = 1.055$; (d) $T = 0.0$, $\alpha = 0.7839$.

Theory gives a boost

Simulated annealing

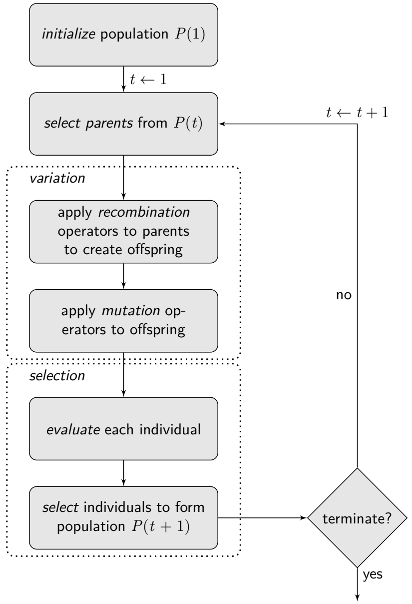
1987 - Laarhoven and Aarts proved under relatively simple conditions,

$$\lim_{t \to \inf} \Pr(\text{solution is found}) = 1$$

by time t!

Implication: general purpose methods can be created that are guaranteed to converge.

Evolutionary Algorithms



Evolutionary Algorithms

- Allow larger jumps
- Long (destructive) jumps should be rare

(1+1) EA

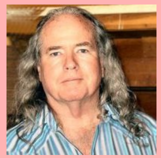
Choose x uniformly at random from $\{0,1\}^n$;
while *stopping criterion not met* **do**
  $y \leftarrow x$;
  foreach $i \in \{1,\ldots,n\}$ **do**
    With probability $1/n$, $y_i \leftarrow (1-y_i)$;
  end
  if $f(y) \geq f(x)$ **then** $x \leftarrow y$;
end

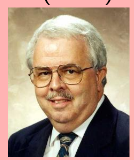
From physics/evolution to social behavior

Marco Dorigo (1992)

Ant Colony Optimization: a set of agents called *ants* that heuristically construct solutions

James Kennedy and **Russell Eberhart** (1995)





Particle Swarm Optimization: inspired by social behavior of bird flocking and fish schooling

From physics/evolution to social behavior

"Swarm intelligence" gets attention in popular press

SECTIONS HOME SEARCH The New York Times

SCIENCE

From Ants to People, an Instinct to Swarm

By CARL ZIMMER NOV. 13, 2007

A school of bigeye trevally in Malaysia. Peter Scoones/Getty Images

If you have ever observed ants marching in and out of a nest, you might have been reminded of a highway buzzing with traffic. To Iain D. Couzin, such a comparison is a cruel insult — to the ants.

The Economist World politics Business & finance Economics Science & technology

Artificial intelligence

Riders on a swarm

Mimicking the behaviour of ants, bees and birds started as a poor man's version of artificial intelligence. It may, though, be the key to the real thing

Aug 12th 2010 | ROME | From the print edition Timekeeper Like 830 Tweet

Russell Cobb

ONE of the bugaboos that authors of science fiction sometimes use to scare their human readers is the idea that ants may develop intelligence and take over the Earth. The purposeful collective activity of ants and other social insects does, indeed, look intelligent on the surface. An illusion, presumably. But it might be a good enough illusion for computer

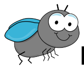
A cautionary tale

Inspired by the success of SA/EAs/ACO/PSO, some researchers begin looking elsewhere for inspiration

 Bees optimization (Nakarani and Tovey, 2004)

 Artificial bee colony optimization (Karaboga, 2005)

 Honey bee mating optimization (Haddad et al. 2006)

 Flies (Abadin et al. 2010)

 Fruit flies (Pan, 2011)

A cautionary tale

 Termites (Hedayatzadeh et al. 2010)

 Fireflies (Lukasik and Zak, 2009)

 Glow worms (Krishnanand and Ghose, 2005)

 intelligent water droplets (Shah-Hosseini, 2009)

 cuckoo search (Yang and Deb 2009)

 harmony search (Geem et al.)

A cautionary tale

Harmony search: "a metaheuristic framework based on the principle of jazz musicians playing together"

Components

- harmony → solution
- note, pitch → decision variable
- sounds better → better objective function value
- harmony memory → population

Procedure

- Generate a set of random initial solutions
- Find better solutions by combining existing ones and changing variables

Weyland (2010)

harmony search is a special case of the $(\mu{+}1)$ ES (Rechenberg, 1973).

A cautionary tale

Why did this happen?

- poor research practices (up the wall games)
- initial success of earlier algorithms created a kind of “market bubble”
- moral: algorithms research should be conducted carefully and scientifically

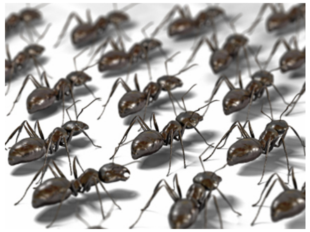
Further reading: Kenneth Sörensen, *Metaheuristics—the metaphor exposed*, International Transactions in Operational Research (2013)

Swarms

- As in genetic algorithms, we want to take *inspiration* from nature;
- In this case from *swarms*;
- We are interested in how *groups of automomous agents* find *good/optimal* solutions;
- Main approaches:

PSO

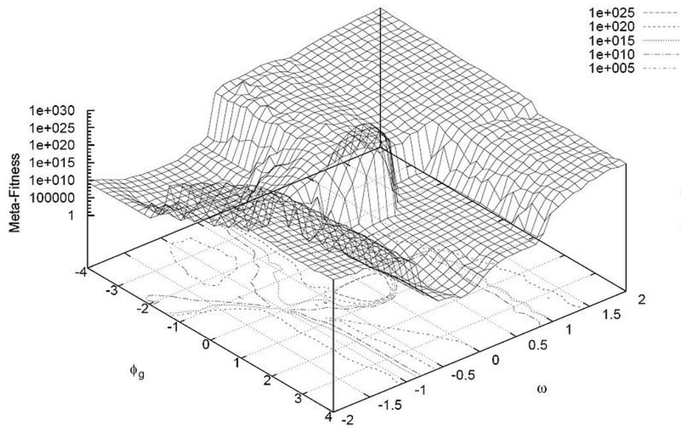
ACO



PSO

- Many *autonomous agents*;
- Usually *continuous* search space;
- *Each* agent searches for a good solution;
- Direct communication with (some) other agents.

PSO

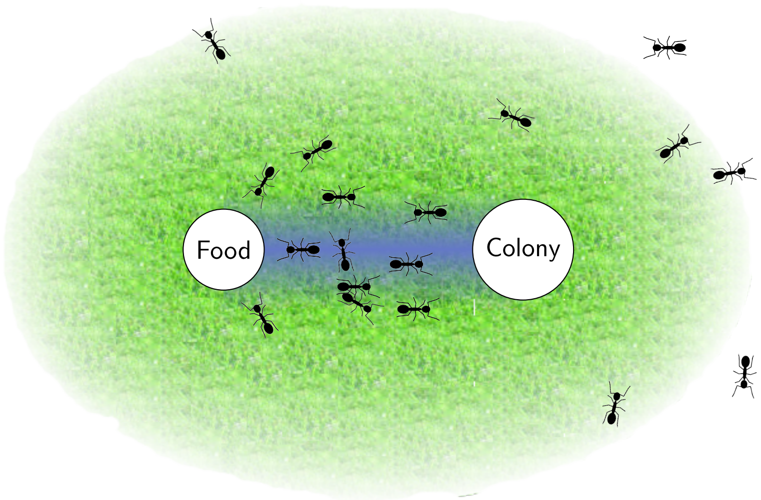


PSO

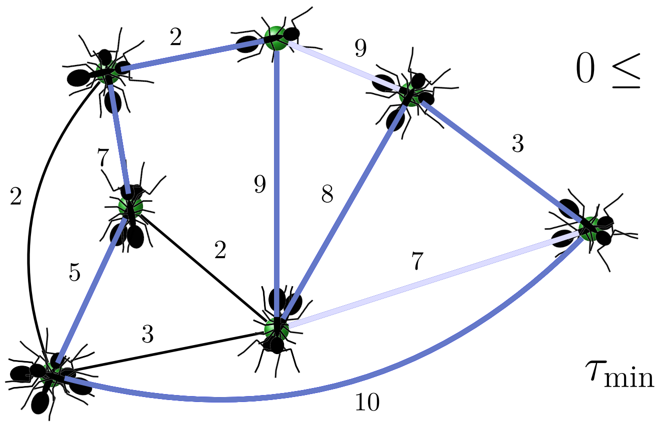
ACO

- Usually *discrete* search space;
- Main example: *Path finding* problems;
- *Indirect* communication with other agents. . .

Pheromones



Classic ACO

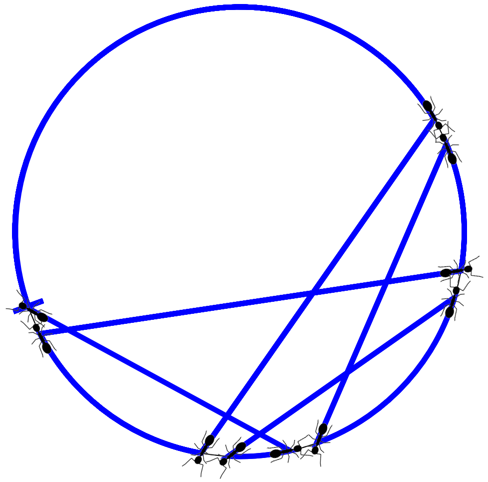


$$0 \le \rho \le 1$$

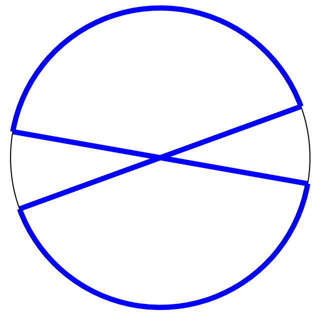
$$\tau_{\text{min}}, \tau_{\text{max}}$$

$$\frac{1}{n^2}, 1 - \frac{1}{n}$$

Exchanges of Edges



Alternative



$E =$ edges of degree < 2 that would not create a premature cycle;

$R =$ total sum of pheromones of edges from E.

Choose edge e with probability of its pheromone divided by R.

Pseudo-Boolean Functions

We can use ants for optimization of *pseudo-Boolean* functions, e.g., OneMax:

$$\mathrm{OneMax}(x) = \sum_{i=1}^{n} x_i.$$