Algorithms in Nature

Nature inspired algorithms

http://www.cs.cmu.edu/~02317/

Ziv Bar-Joseph
zivbj@cs.cmu.edu
GHC 8006

Matt Ruffalo
mruffalo@andrew.cmu.edu
GHC 7715

Topics

- Introduction (1 Week)
- Classic algorithms (4 weeks)
- Bi-directional studies (4 weeks)
- Student presentations (4 weeks)
- Poster session (1 week)

Class overview

- 2 problem sets
- Project (and poster)
- Class presentation of a paper (only for those registered to the masters / grad version)
- Class attendance and participation

Class grades

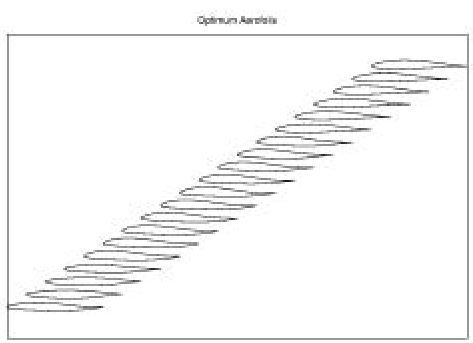
- Project (40%)
- Problem sets (20%)
- Class participation (10%)
- Class presentation (30%)
- (for those not presenting, % will be adjusted according to the weighting above)

Overview

- Why learn from nature?
- Nature inspired / learned algorithms
  - Differential Evolution algorithm
  - Other optimization
  - -Bi-directional studies
- Applications

Learning from nature

- Nature evolved efficient methods to address information processing problems
- Processes imitating such natural processes are often denoted as ‘nature inspired’
- Engineering example: Aircraft wing design



(Another) engineering example: Bullet train

Train's nose is designed after the beak of a kingfisher, which dives smoothly into water. **(Source: Popular Mechanics)**

Optimization

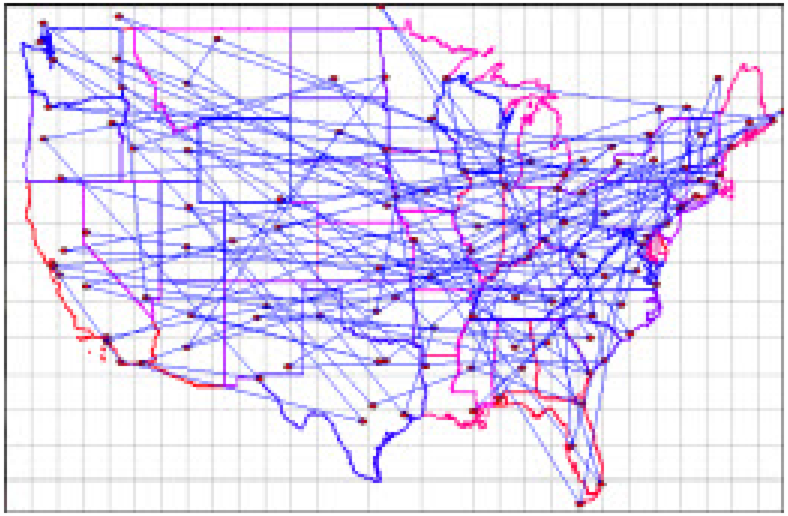
– An act, process, or methodology of making something as fully perfect, functional, or **effective as possible.** (webster dictionary)

- **Birds:** Minimize drag.
- Consider an optimization problem of the form:

$$\begin{aligned} &Min \quad \{f(x)\} \\ &s.t. \quad x \in S \subset R^n \end{aligned}$$

Optimization problem: Example

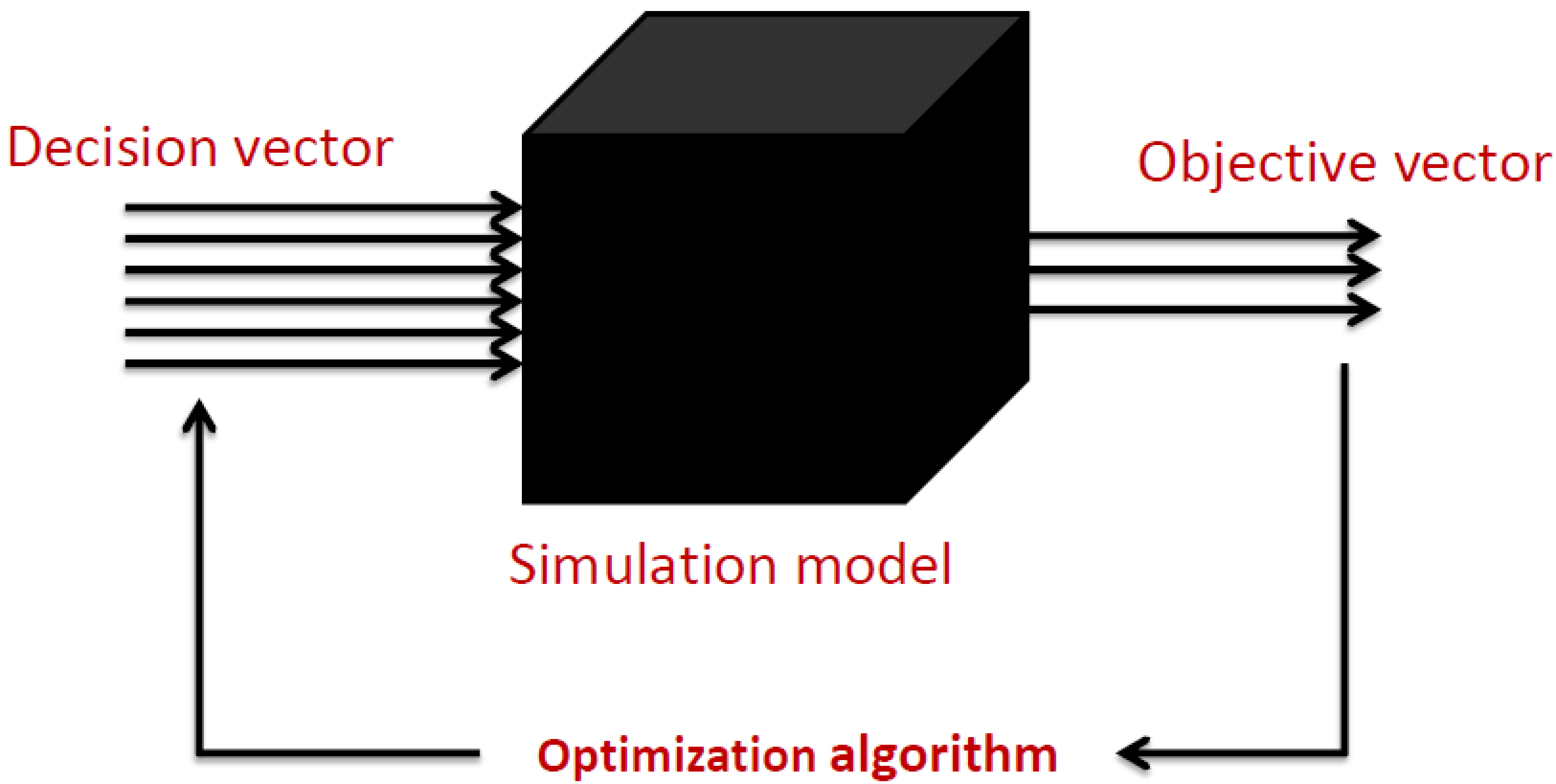
Fastest / cheapest way of visiting all 50 state capitals



Characteristics of common optimization problems

- Objective and constraint functions can be non-differentiable.
- Constraints nonlinear.
- Discrete/Discontinuous search space.
- Mixed variables (Integer, Real, Boolean etc.)
- Large number of constraints and variables.
- Objective functions can be multimodal with more than one optima
- Computationally expensive to compute in closed form

Iteratively solving optimization problems

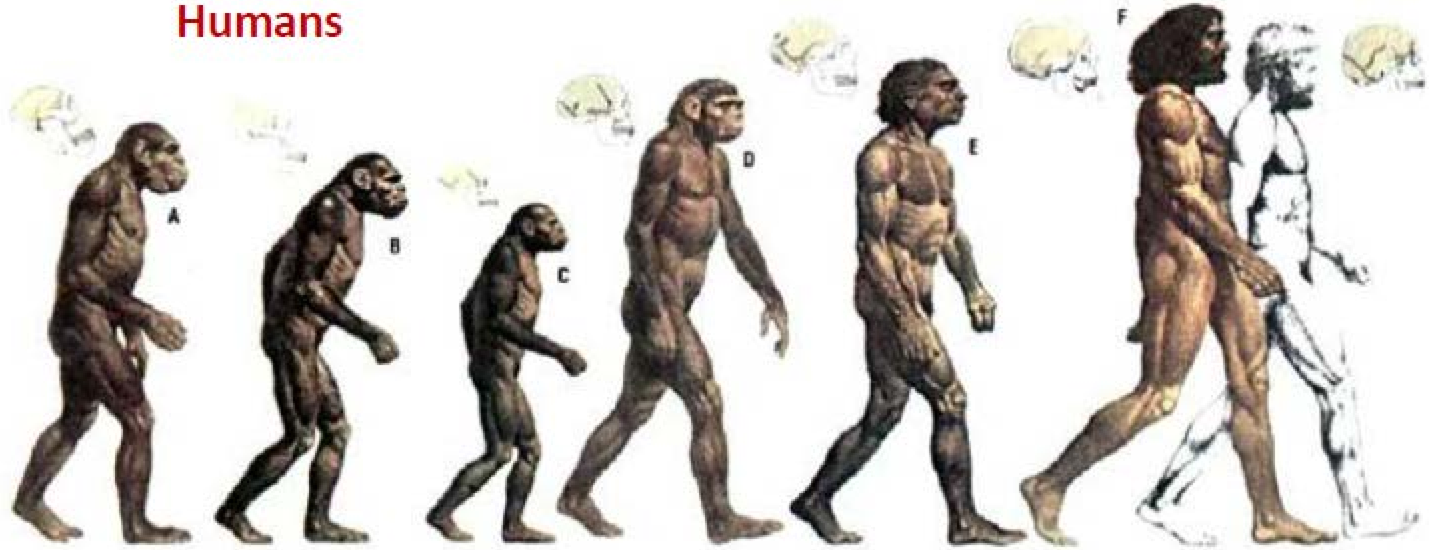


Solving optimization problems

- Different methods for different types of problems.
- Often get stuck in local optima (lack global perspective).
- Some (for example regression based on gradient descent) need knowledge of first/second order derivatives of objective functions and constraints.

Evolution

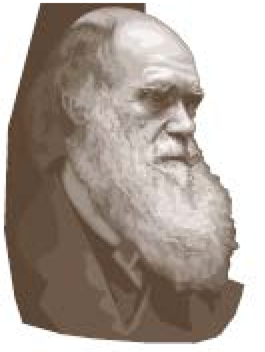
Humans



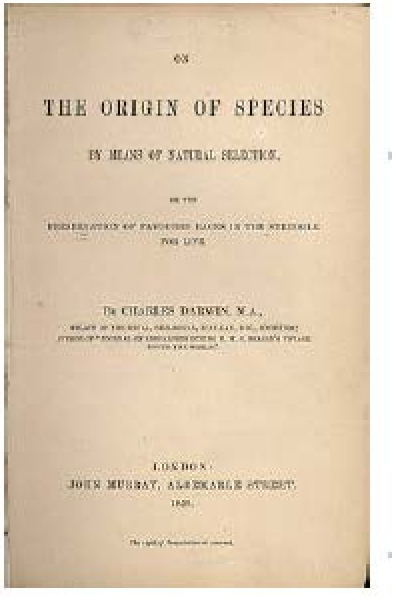
Macintosh

Evolutionary algorithms

- Offsprings created by reproduction, mutation, etc.
- Natural selection - A guided search procedure
- Individuals suited to the environment survive, reproduce and pass their genetic traits to offspring
- Populations adapt to their environment. Variations accumulate over time to generate new species



Charles Darwin



THE ORIGIN OF SPECIES

BY MEANS OF NATURAL SELECTION,

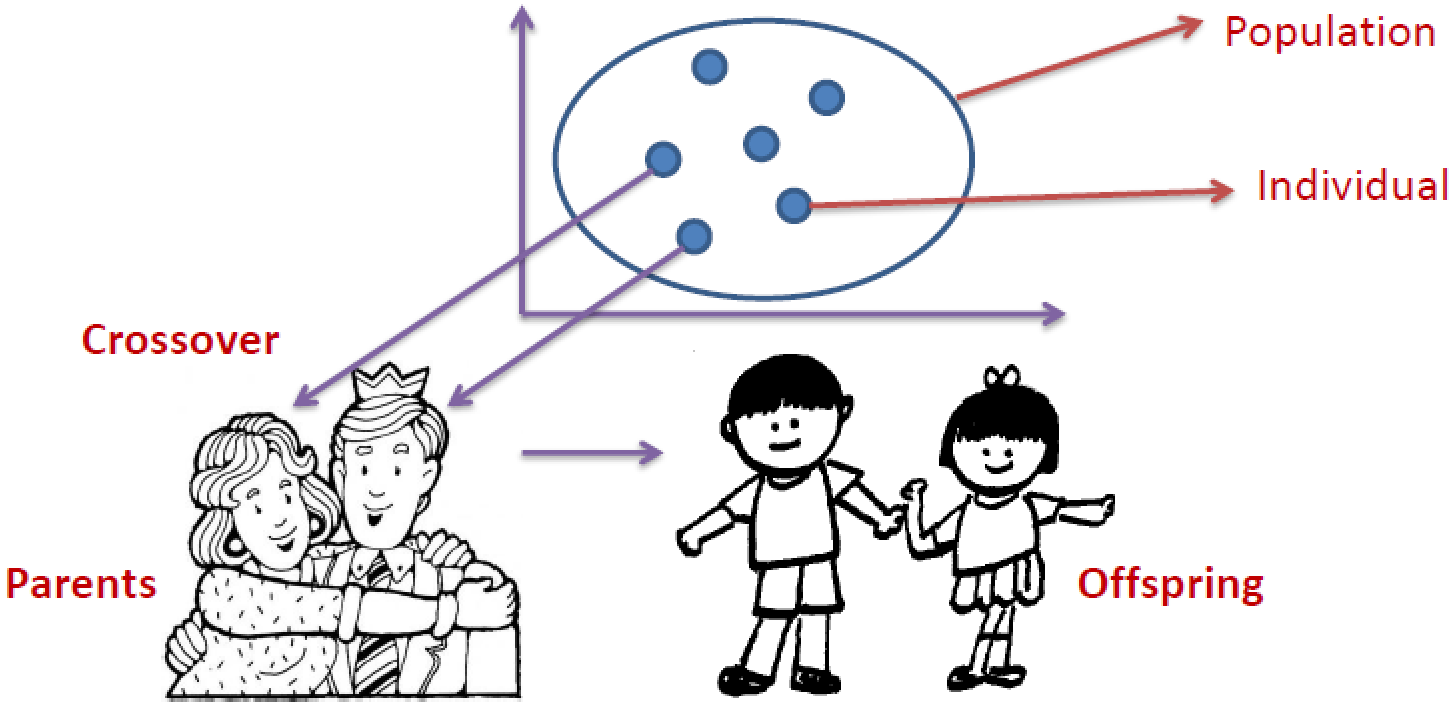
LONDON:

JOHN MURRAY, ALBEMARLE STREET.

Evolutionary algortithms

Terminology

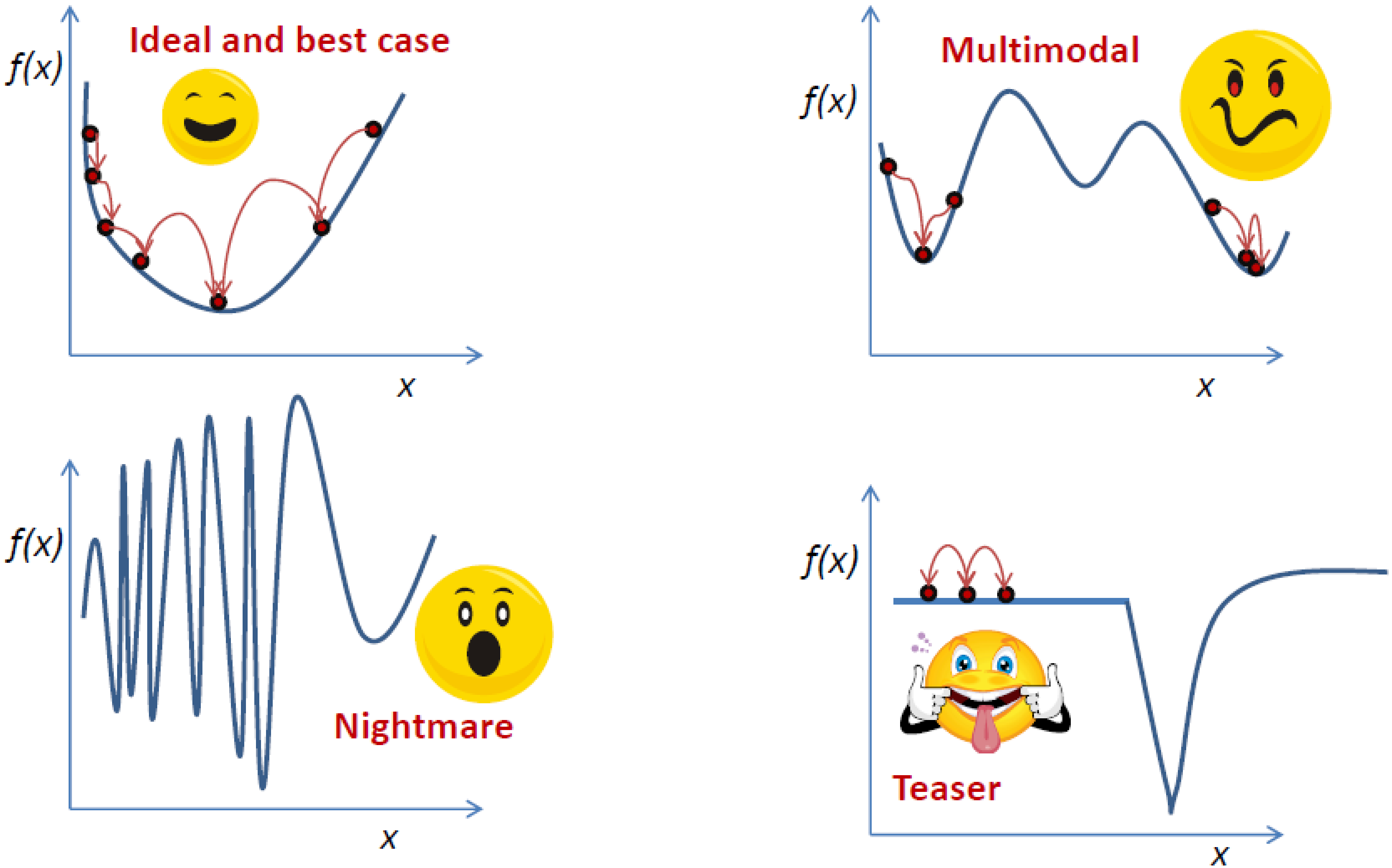
1. Individual - carries the genetic information (chromosome). It is characterized by its state in the search space and its fitness (objective function value).
2. Population - pool of individuals which allows the application of genetic operators.
3. Fitness function - The term “fitness function” is often used as a synonym for objective function.
4. Generation - (natural) time unit of the EA, an iteration step of an evolutionary algorithm.


Population
Individual
Crossover
Parents
Offspring

Overall idea

- Selection - Roulette wheel, Tournement, steady state, etc.
- Motivation is to preserve the best (make multiple copies) and eliminate the worst
- Crossover – simulated binary crossover, Linear crossover, blend crossover, etc.
- Create new solutions by considering more than one individual
  - Global search for new and hopefully better solutions
- Mutation – Polynomial mutation, random mutation, etc.
- Keep diversity in the population
  - 010110 →010100 (bit wise mutation)

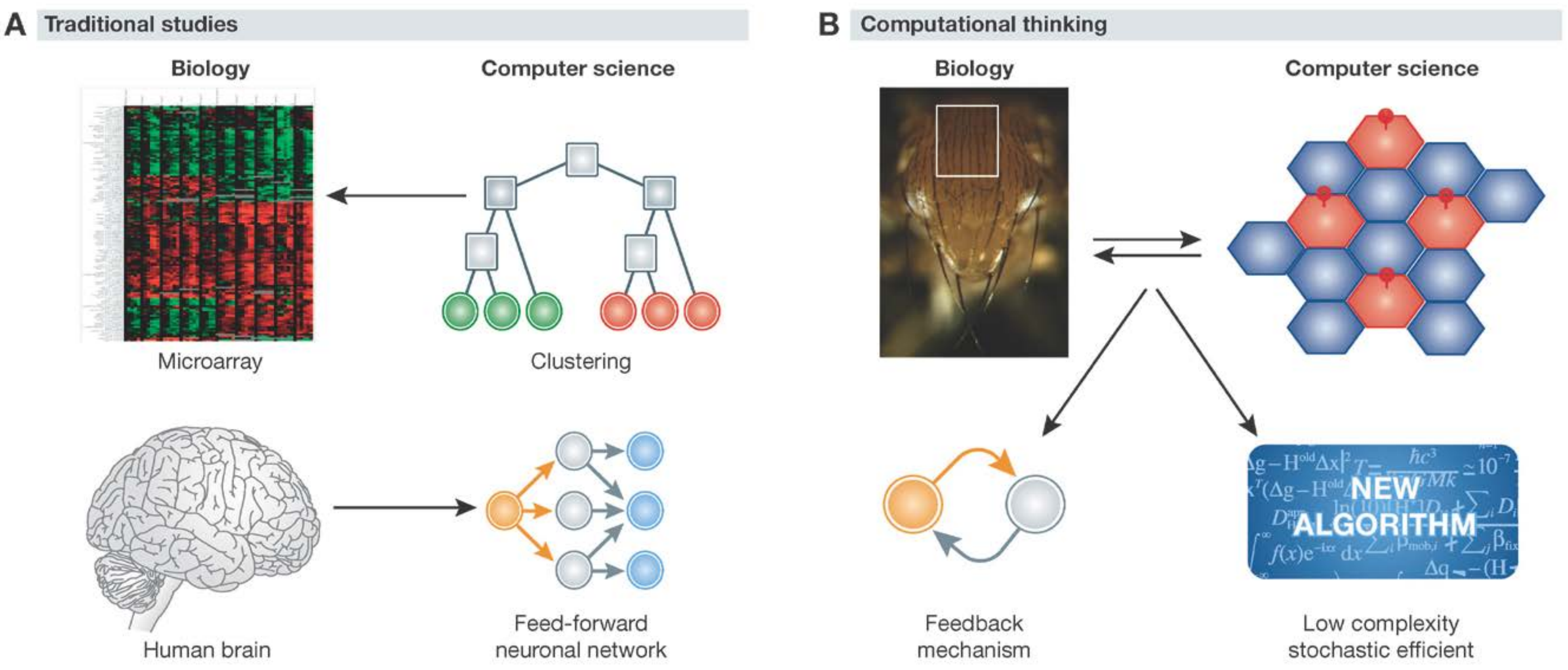
Evolutionary vs. gradient descent based methods



Limitations

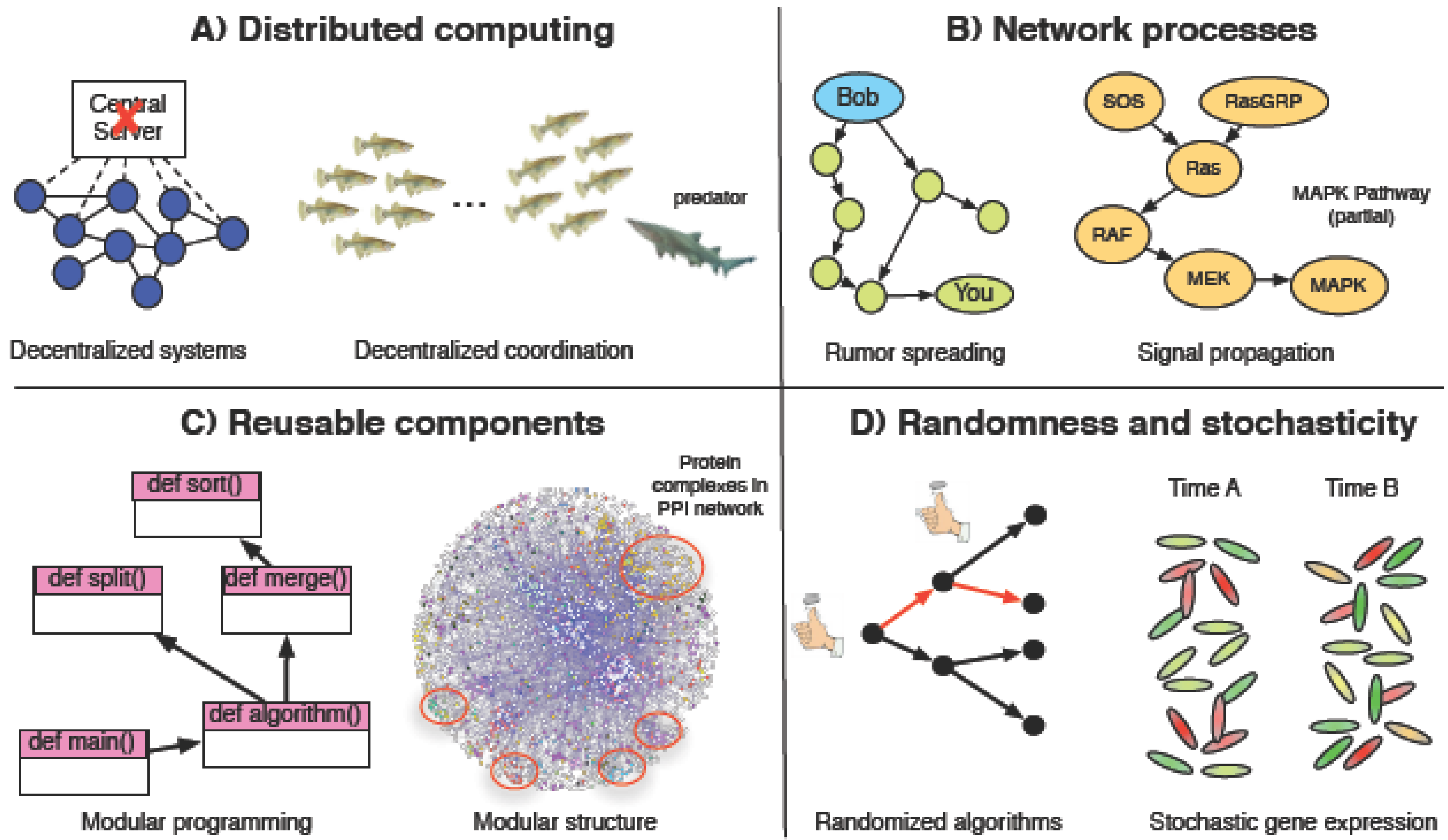
- No guarantee of finding an optimal solution in finite time
- Relatively little in terms of convergence guarantees
- Could ne computationally expensive

Bi-directional studies



Navlakha and Bar-Joseph *Nature MSB* 2011

Algorithms in nature: Shared principles between CS and Biology

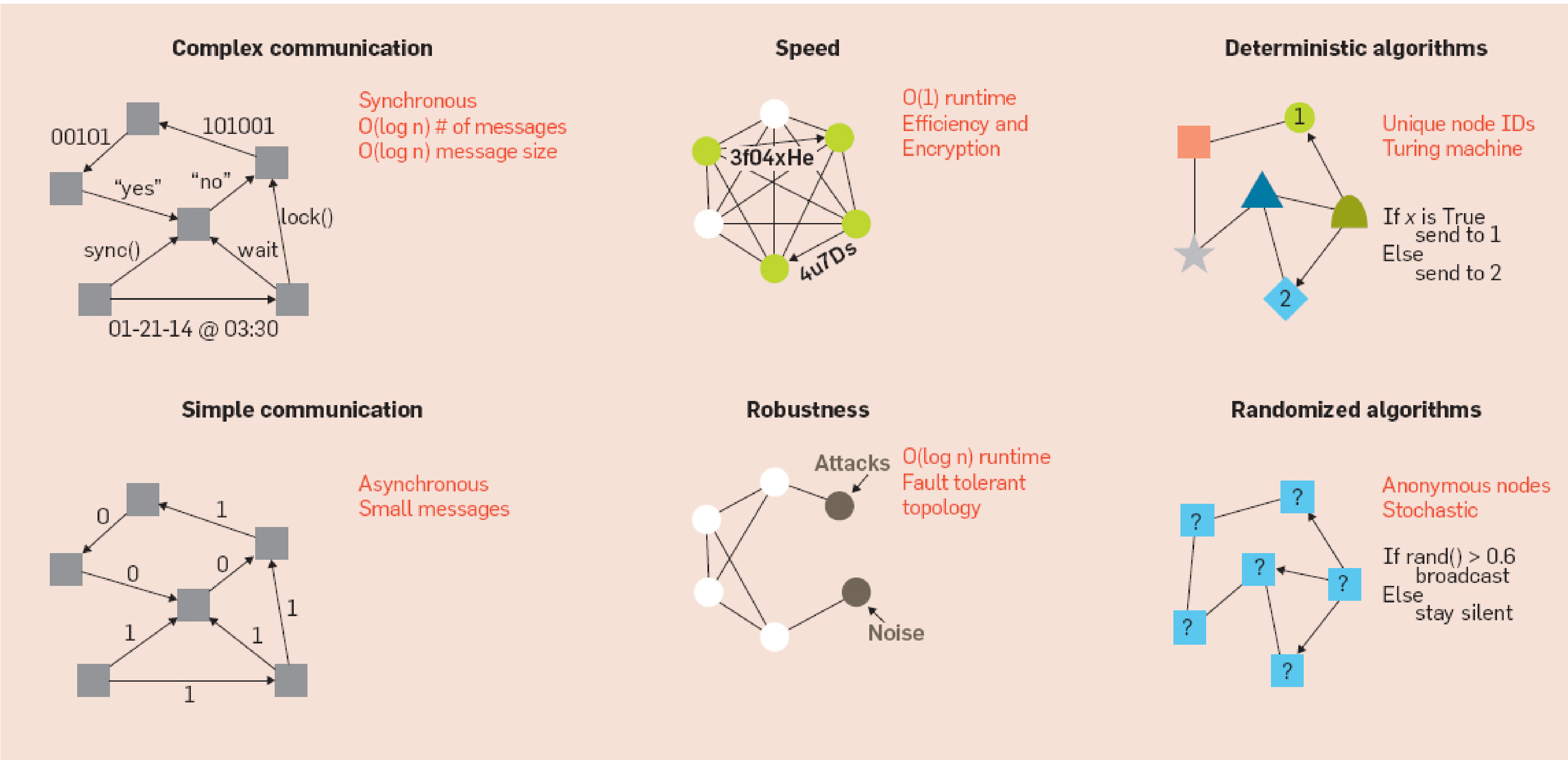


Movie

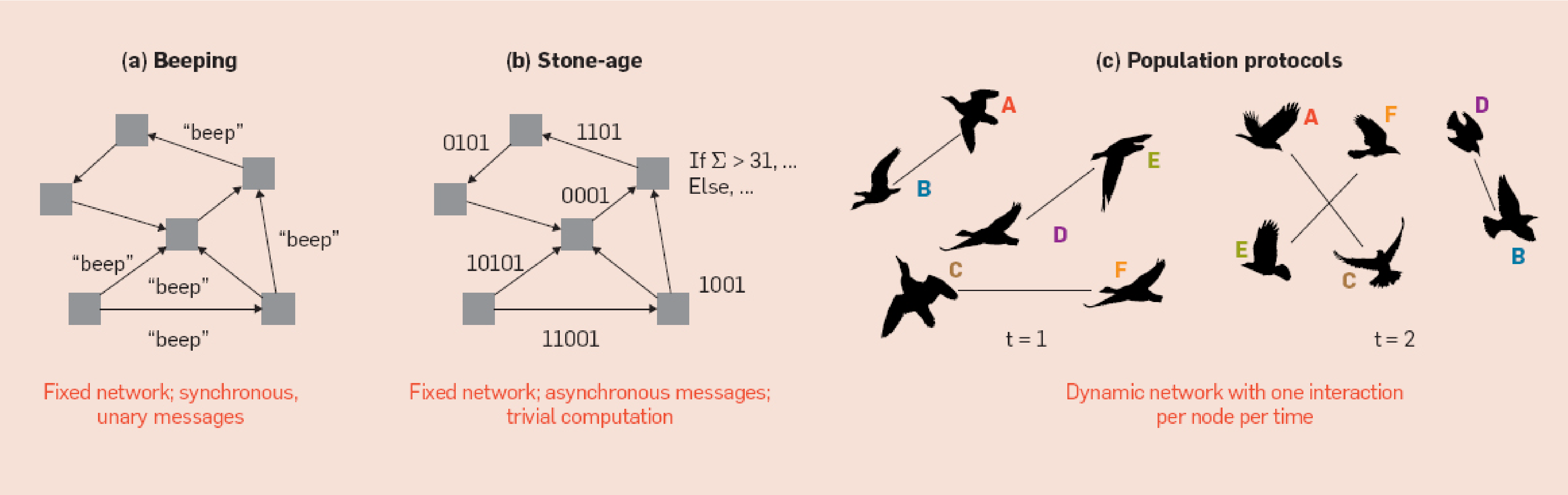
http://cacm.acm.org/magazines/2015/1/181614-distributed-information-processing-in-biological-and-computational-systems/fulltext

But there are also differences …

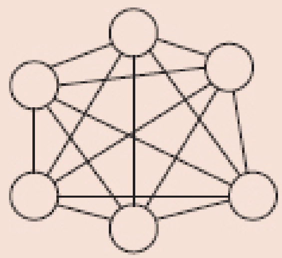
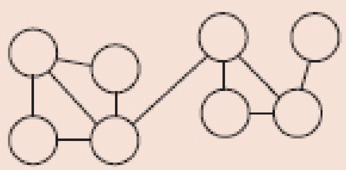
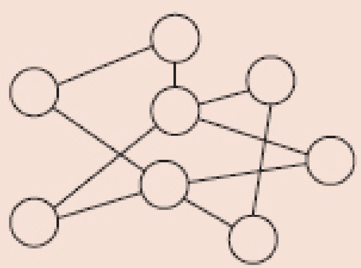
Tradeoffs between key design issues



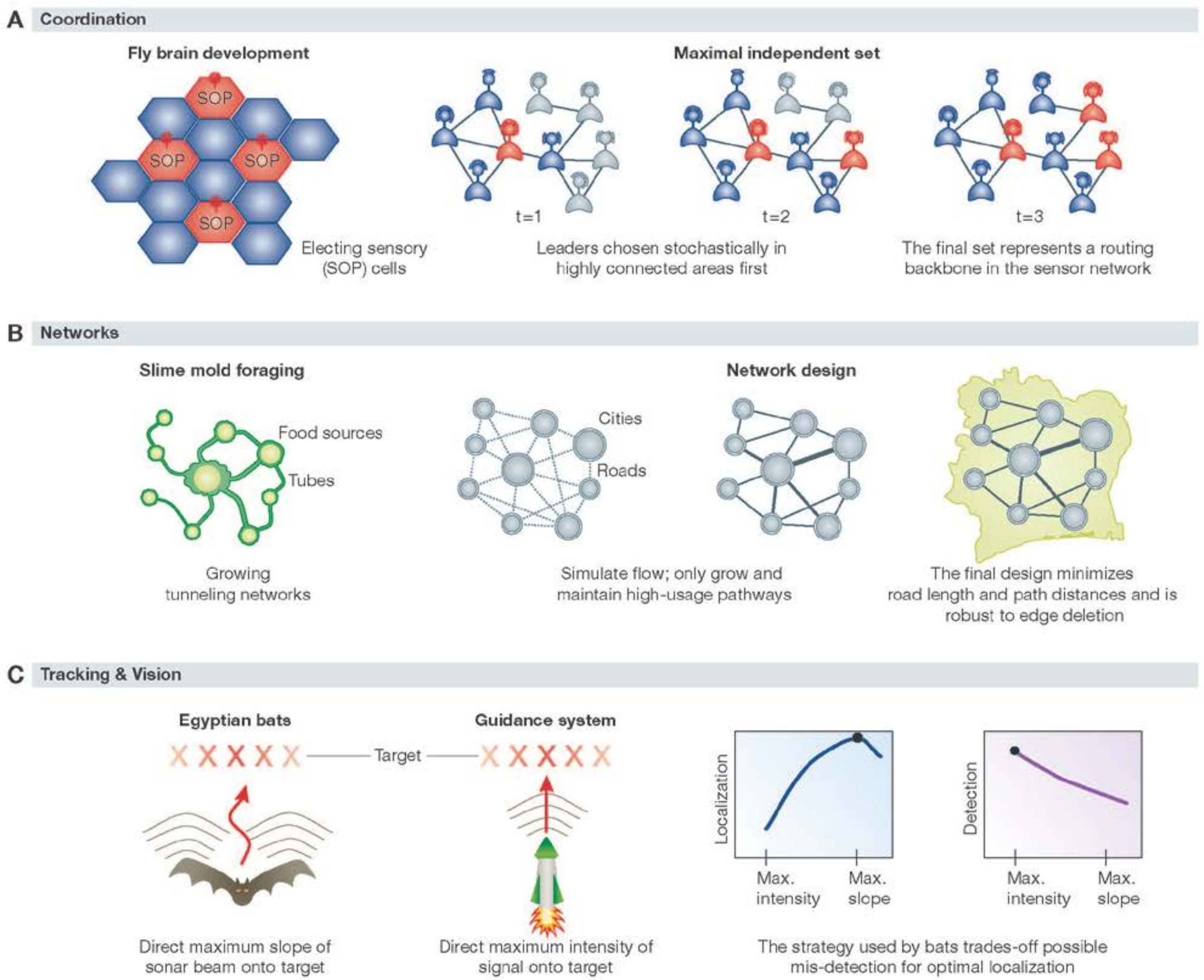
Communication models for biological processes



Network topologies

		Speed	Robust (single-point attack)	Robust (cascading failure)
Densely connected (cliques)		✓	✓	✗
Weakly-connected modules		✓	✗ (bottlenecks)	✓
Sparsely connected		✗	✓	✓

Examples of bi-directional studies



Details of models used

Biological System	Computational Problem	Communication	Topology	Stochastic?	Alg.?	Refs.
Slime mold	Network routing	Stone-Age	Incomplete	Yes	Yes	35,48,52 48
Fly brain	Max. independent set	Beeping	Sparse	Yes	Yes	1,2
Harvester ants	TCP congestion control	Population	Random	No	Yes	46
Ants, swarms	Distributed search	Population	Random	Yes	Yes	23,50
Plants	Consensus	Stone-age	Incomplete	No	Yes	21
Fish schools	Consensus	Population	Random	Yes	No	29
Cell cycle switch	Approximate majority	Population	Random	Yes	Yes	13
Spiking neurons	Probabilistic inference	Stone-Age	Incomplete	No	Yes	11,45
Dendritic branching	Distributed MSTs	Stone-Age	Incomplete	No	Yes	18
Gannet colonies	Space partitioning	Population	Random	No	Yes	54
Protein interactions	Network design	Population	Random	Yes	Yes	43