13: Parameter Control

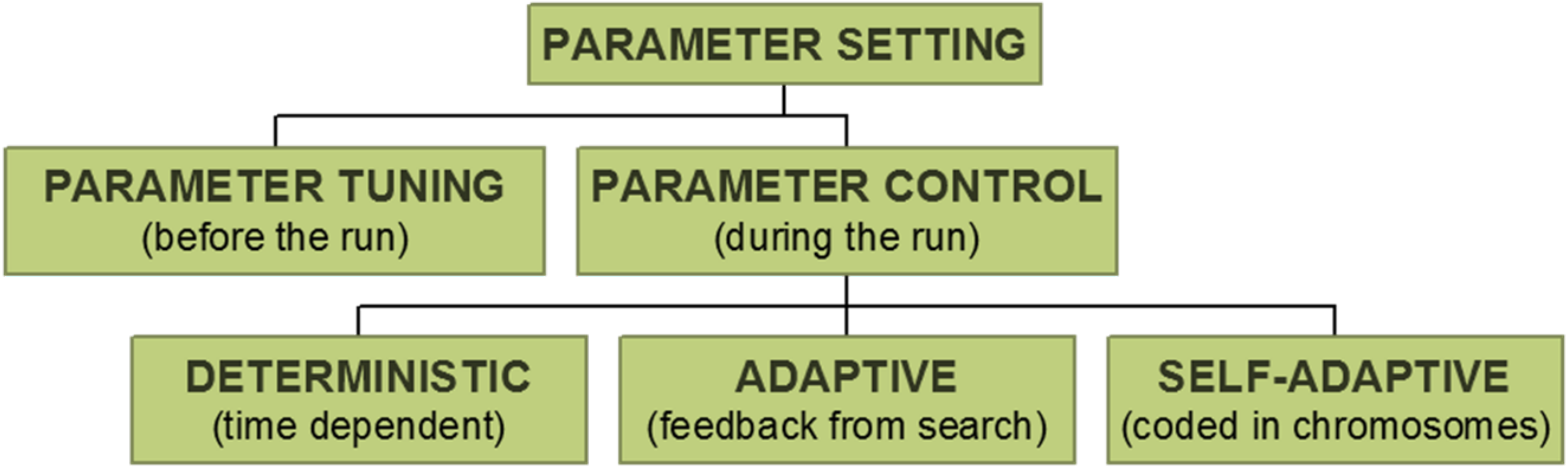
- Motivation
- Parameter control
- Example on varying mutation step size
- Example on varying penalty in fitness evaluation
- Example on varying population size
- Textbook Chapter 8

Prof. Ting Hu, School of Computing, Queen’s University

Motivation

- An EA has many strategy parameters, e.g., mutation operation and mutation rate
- Good parameter values facilitate good performance
- EA parameters are often constant during a run
- An EA is a dynamic, adaptive process
- Optimal parameter values may vary during a run

Taxonomy


PARAMETER SETTING
PARAMETER TUNING
(before the run)
PARAMETER CONTROL
(during the run)
DETERMINISTIC
(time dependent)
ADAPTIVE
(feedback from search)
SELF-ADAPTIVE
(coded in chromosomes)

Parameter tuning

- Test and compare different values before the "real" run
- Challenges
    - user mistakes in settings can be sources of errors or suboptimal performance
    - costs time
    - parameters interact: exhaustive search is not feasible; can't test one at a time
    - requirements may change during a run

Parameter control

- Set values on-line, during a run
- Challenges
  - hard to optimize parameters with a time-varying schedule
  - hard to optimize user-defined feedback mechanism
  - hard to select for parameter optimization

Example - Varying mutation step size

- Task to solve
  - min $f(x_1, x_2, \ldots, x_n)$
- Algorithm
  - EA with real-valued representation $(x_1, x_2, \ldots, x_n)$
  - Arithmetic averaging crossover
  - Gaussian mutation, $x_i' = x_i + N(0,\sigma)$, SD σ is called mutation step size

Example - Varying mutation step size option 1

- Replace the constant σ by a function $\sigma(t)$
  - $\sigma(t) = 1 - 0.9 \times \dfrac{t}{T}$
  - t is the current generation number and T is the generation limit
- Features
  - changes in σ are independent from the search progress
  - strong user control of σ
  - σ is fully predictable
  - a given σ acts on all individuals of the population

Example - Varying mutation step size option 2

- Replace the constant σ by a function $\sigma(t)$ updated after every k generations using the $\frac{1}{5}$ success rule
  - $$\sigma(t) = \begin{cases} \frac{\sigma(t-k)}{c} & \text{if } p_s > 0.2 \\ \sigma(t-k) \times c & \text{if } p_s < 0.2 \\ \sigma(t-k) & \text{otherwise} \end{cases}$$
  - $0.817 \leq c \leq 1$

- Features
  - changes in σ are based on feedback from the search progress
  - some user control of σ
  - σ is not predictable
  - a given σ acts on all individuals of the population

Example - Varying mutation step size option 3

- Assign a σ to each individual and incorporate it into the chromosome
  - $(x_1, x_2, \ldots, x_n, \sigma)$
  - $\sigma' = \sigma \times e^{N(0,\tau)}$
  - $x_i' = x_i + N_i(0,\sigma')$

- Features
  - changes in σ are results of natural selection
  - (almost) no user control of σ
  - σ is not predictable
  - a given σ acts on one individual

Example - Varying mutation step size option 4

- Assign a σ to each variable in each individual and incorporate them into the chromosome
  - $(x_1, x_2, \ldots, x_n, \sigma_1, \sigma_2, \ldots, \sigma_n)$
  - $\sigma_i' = \sigma_i \times e^{N_i(0,\tau)}$
  - $x_i' = x_i + N_i(0,\sigma')$
- Features
  - changes in σ are results of natural selection
  - (almost) no user control of σ
  - σ is not predictable
  - a given σ acts on one individual

Example - Varying the penalty in fitness evaluation

Constraints

- $g_i(x) \le 0$ for $i = 1,\ldots,q$ inequality constraints
- $h_i(x) = 0$ for $i = q+1,\ldots,m$ equality constraints

are handled by penalties:

$$eval(x) = f(x) + W \times penalty(x)$$

where

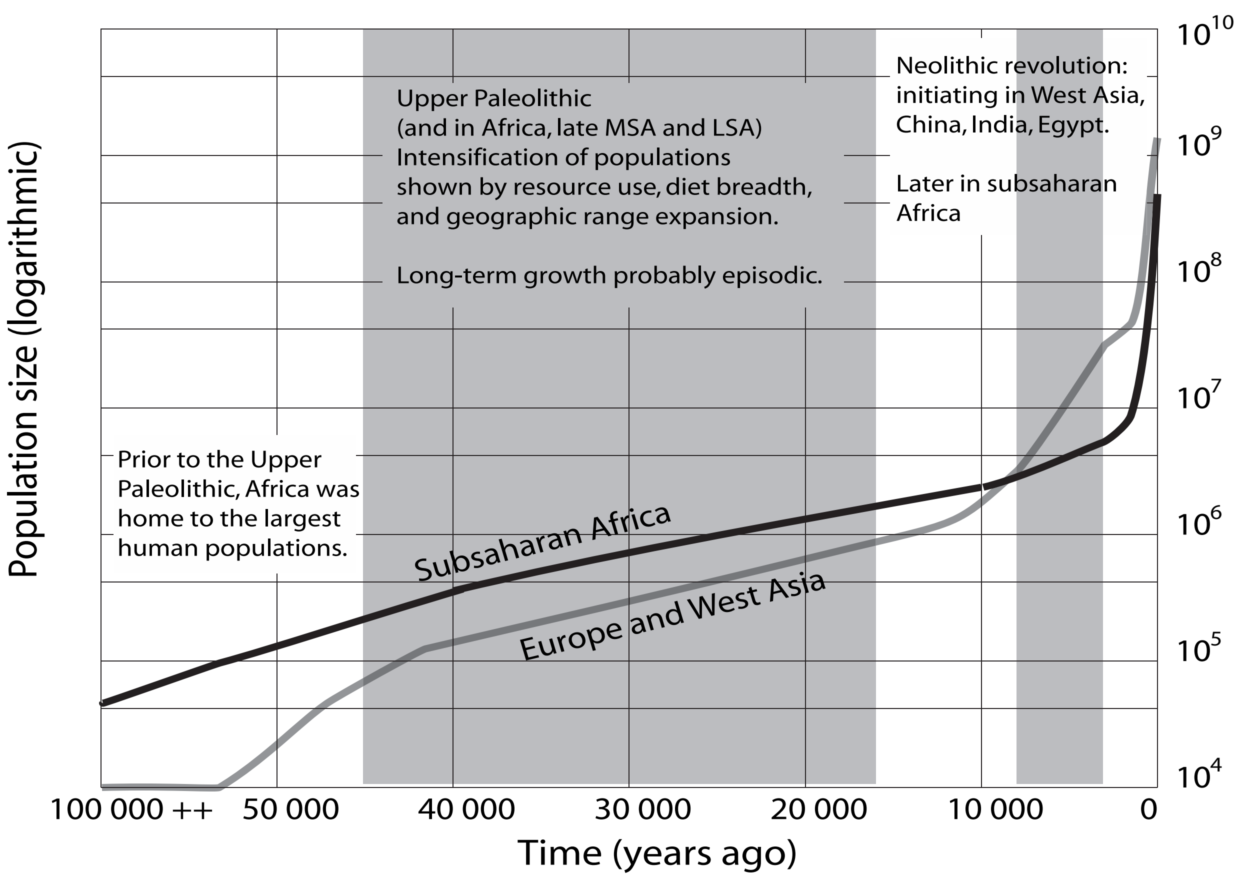
$$penalty(x) = \sum_{j=1}^{m} \begin{cases} 1 & \text{for violated constraint} \\ 0 & \text{for satisfied constraint} \end{cases}$$

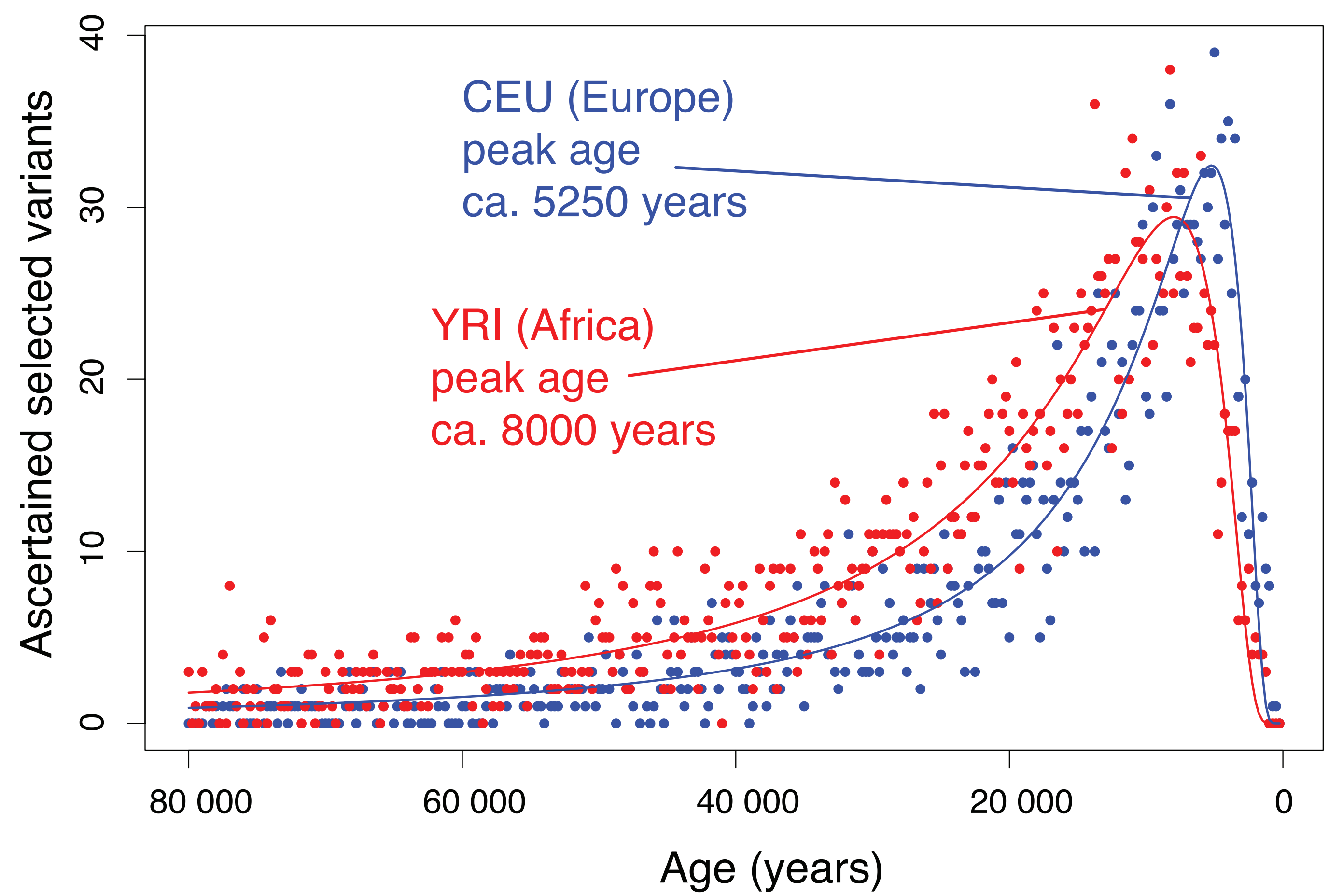
Example - Varying population size

- Population provides diversity for the search
- Determine population size based on problem difficulty
- Adjust population size during a run
- Feedback
  - Individual's persistence in the population
  - fitness progression
  - rate of evolution

Population size and evolution rate

- Does increase population size accelerate or slow down evolution?
- Is it a monotonic relationship?
- Examples:
  - Island species vs. related mainland species
  - recent rapid human evolution

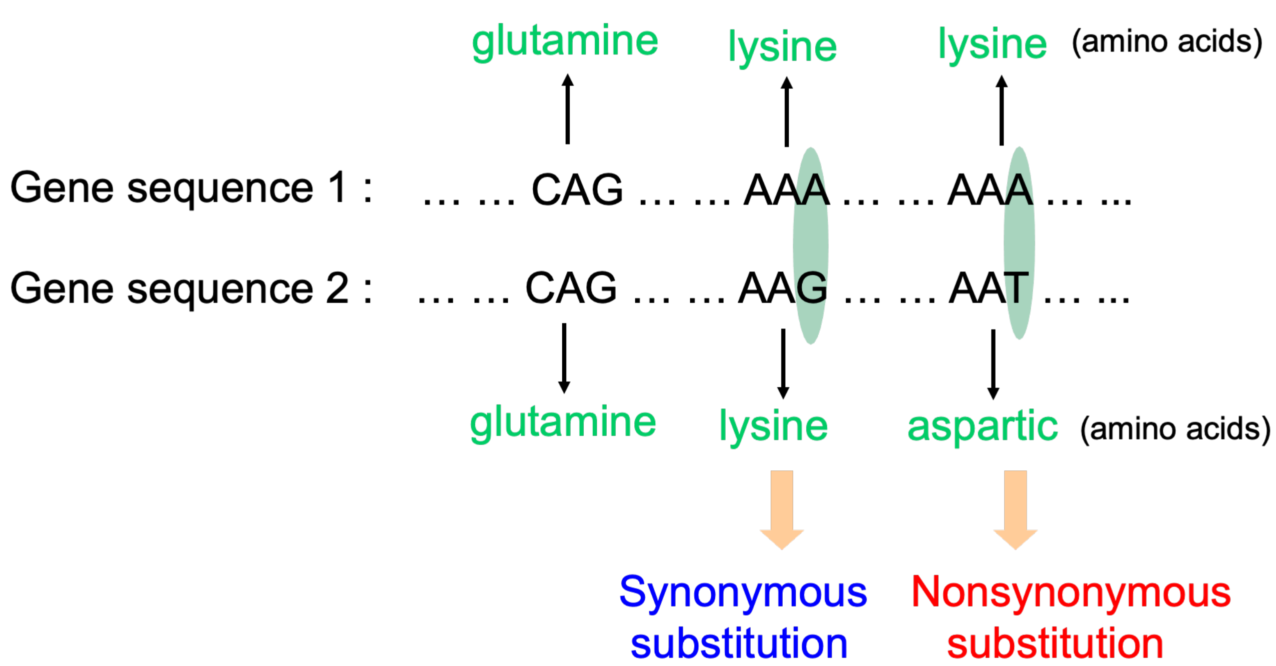




Hawks et al.: Recent acceleration of human adaptive evolution, 2007

Measure rate of evolution

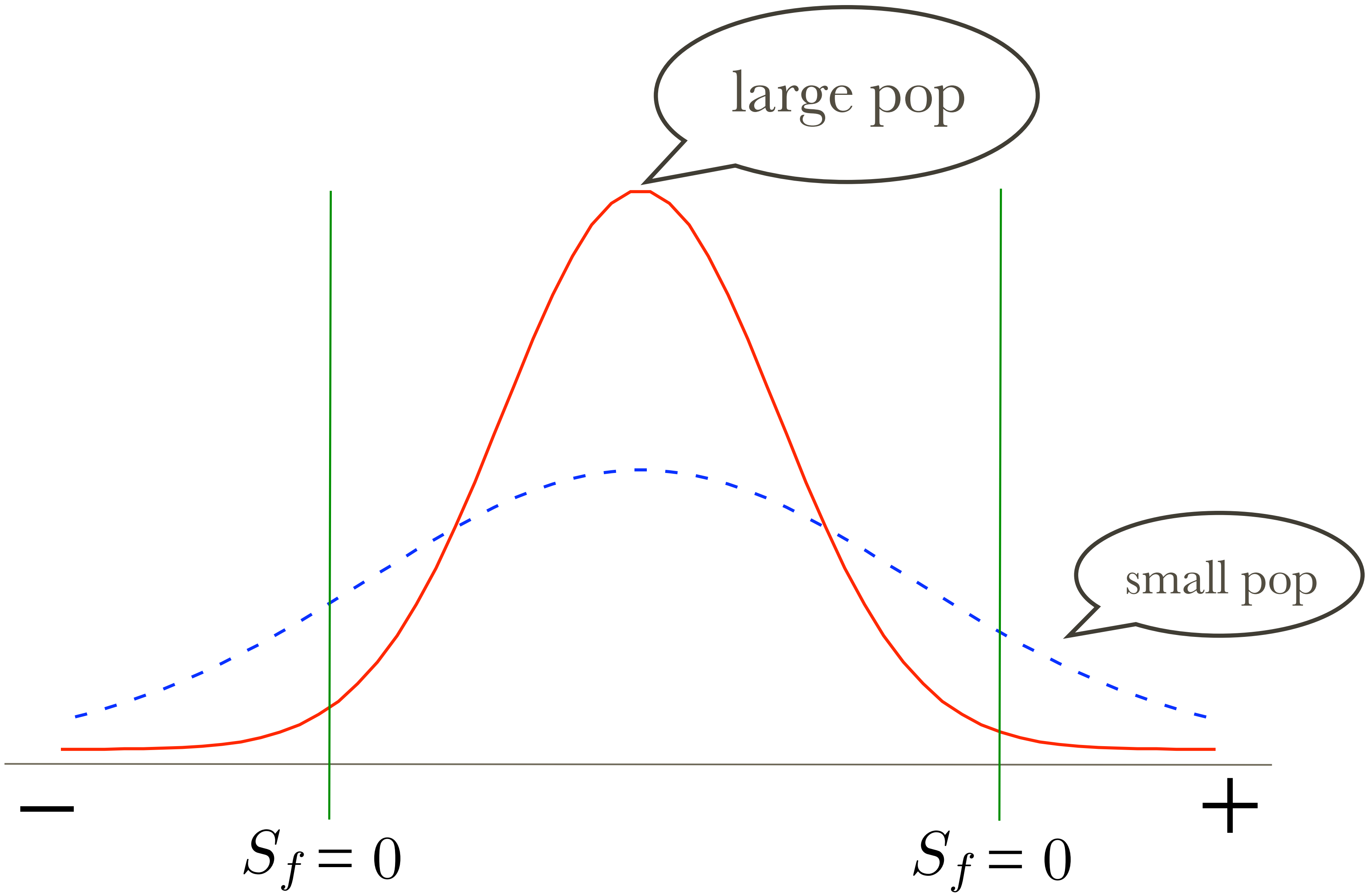
- Non-synonymous to synonymous substitution rate k_a/k_s
- k_a : rate of accepting non-synonymous genetic changes
  - observable adaptive evolution
- k_s : rate of accepting synonymous genetic changes
  - background "clock ticks"



Nearly Neutral Theory of molecular evolution

- Proposed by Tomoko Ohta in 1973
- Nearly neutral mutations are slightly deleterious and slightly advantageous mutations
- Substantial number of nearly neutral mutations exist in molecular evolution
- Accepting nearly neutral mutations provides variation potential
- Population size ca influence the rate of molecular evolution

Positive vs. negative selection



positive selection negative selection

Population size and rate of evolution

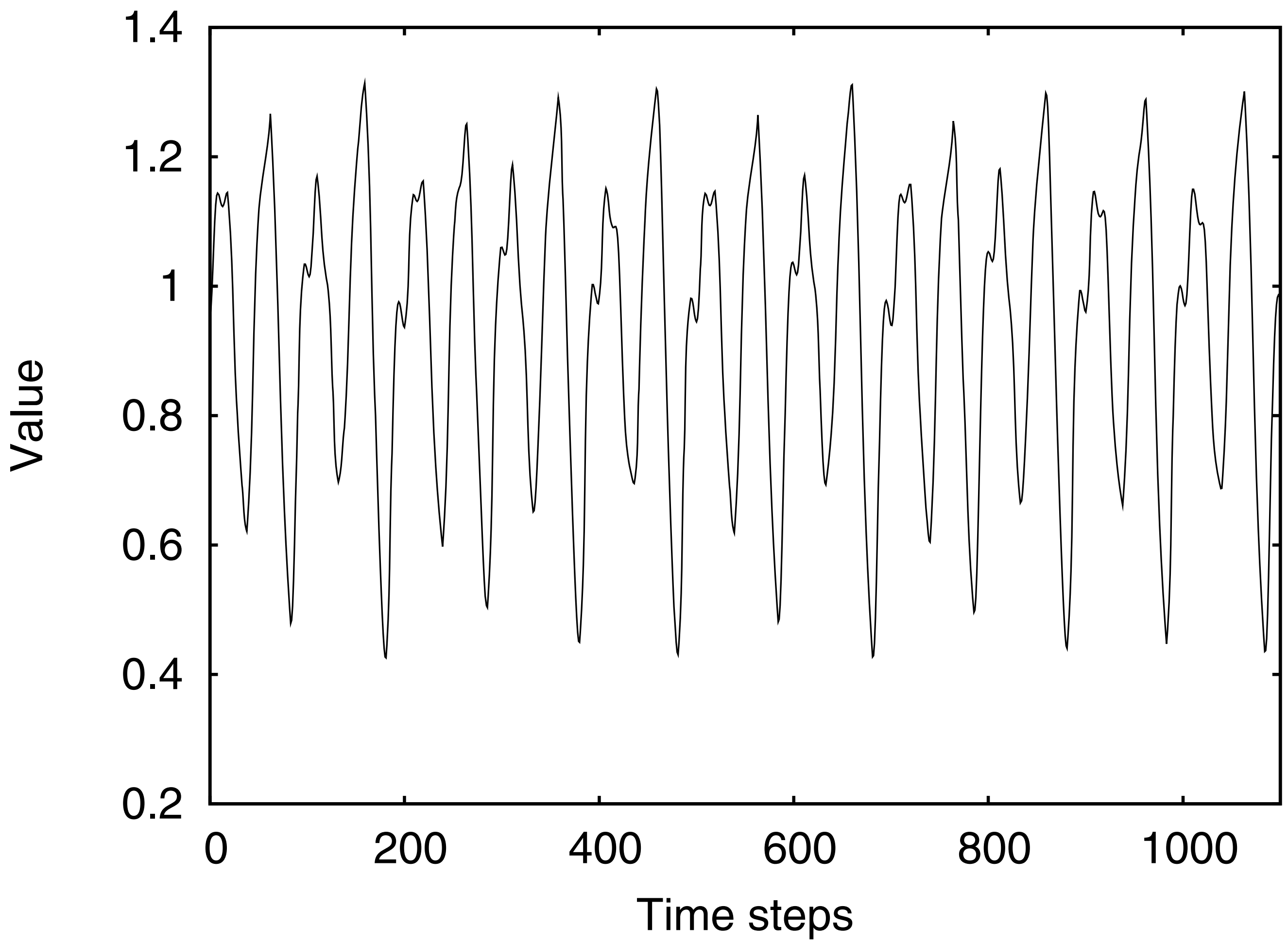
- Under positive selection
  - increase population size can accelerate the rate of accepting adaptive mutations
- Under negative selection
  - decrease population size can allow accepting mutations (mostly deleterious)
  - adjust population size to compensate selection pressure
  - stabilize the rate of accepting adaptive mutations

Adjust population size during a run

- At generation t
  - if $k_a/k_s(t) > 1$, increase population size
  - if $k_a/k_s(t) = 1$, keep the same size
  - if $k_a/k_s(t) < 1$, increase size if $k_a/k_s(t)$ increases; decrease size if $k_a/k_s(t)$ decreases

Experiment

- Problem: Mackey-Glass time series prediction
- Algorithm: Tree-based genetic programming

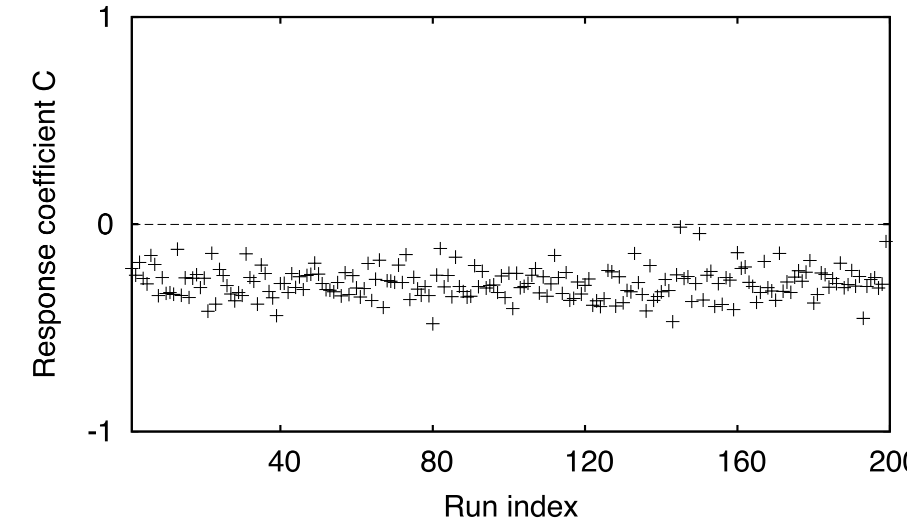


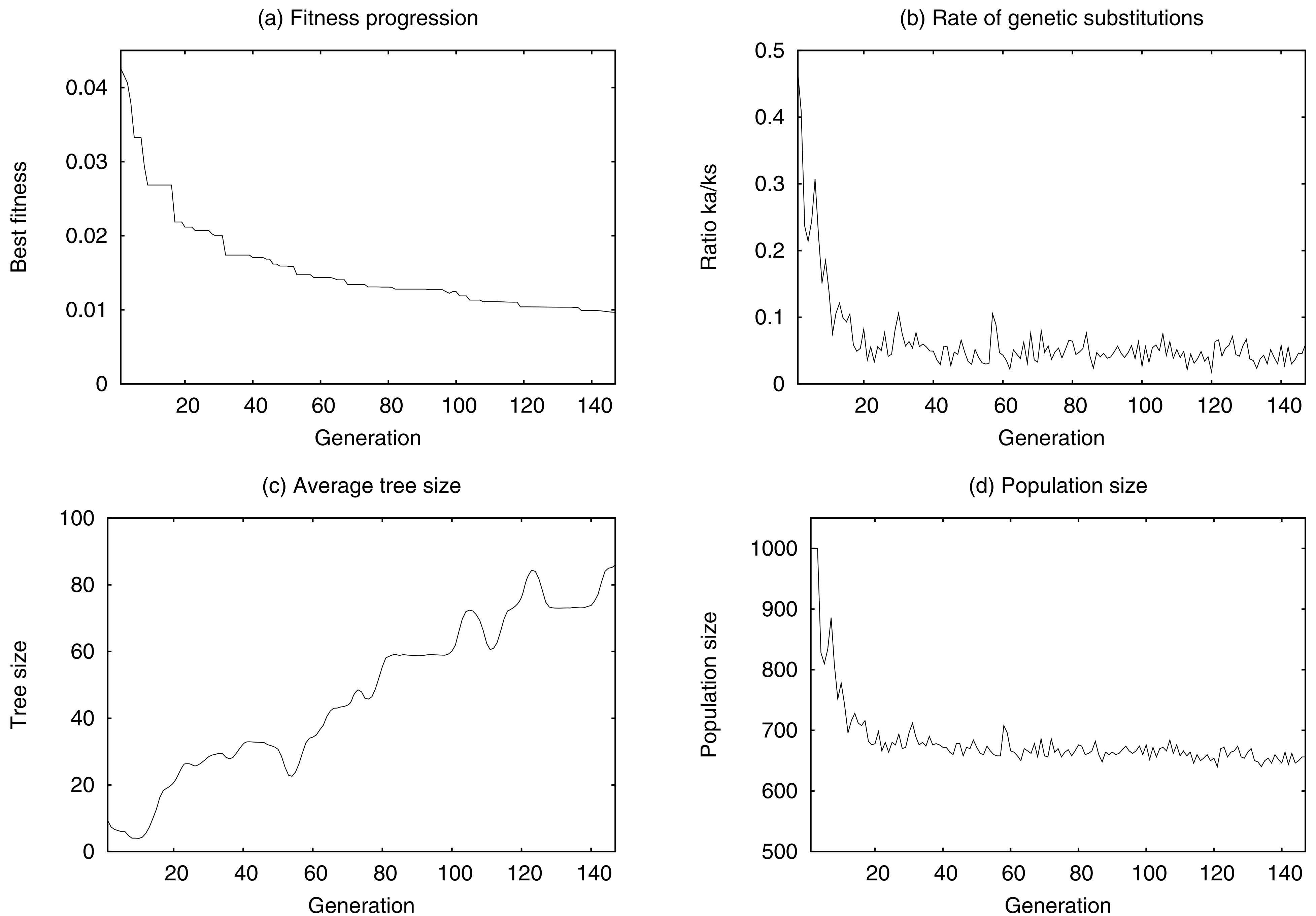
$$x_{t+1} = x_t - b \times x_t + \frac{a \times x_{t-\tau}}{1 + (x_{t-\tau})^{10}},$$

$$x_0 = 1,$$

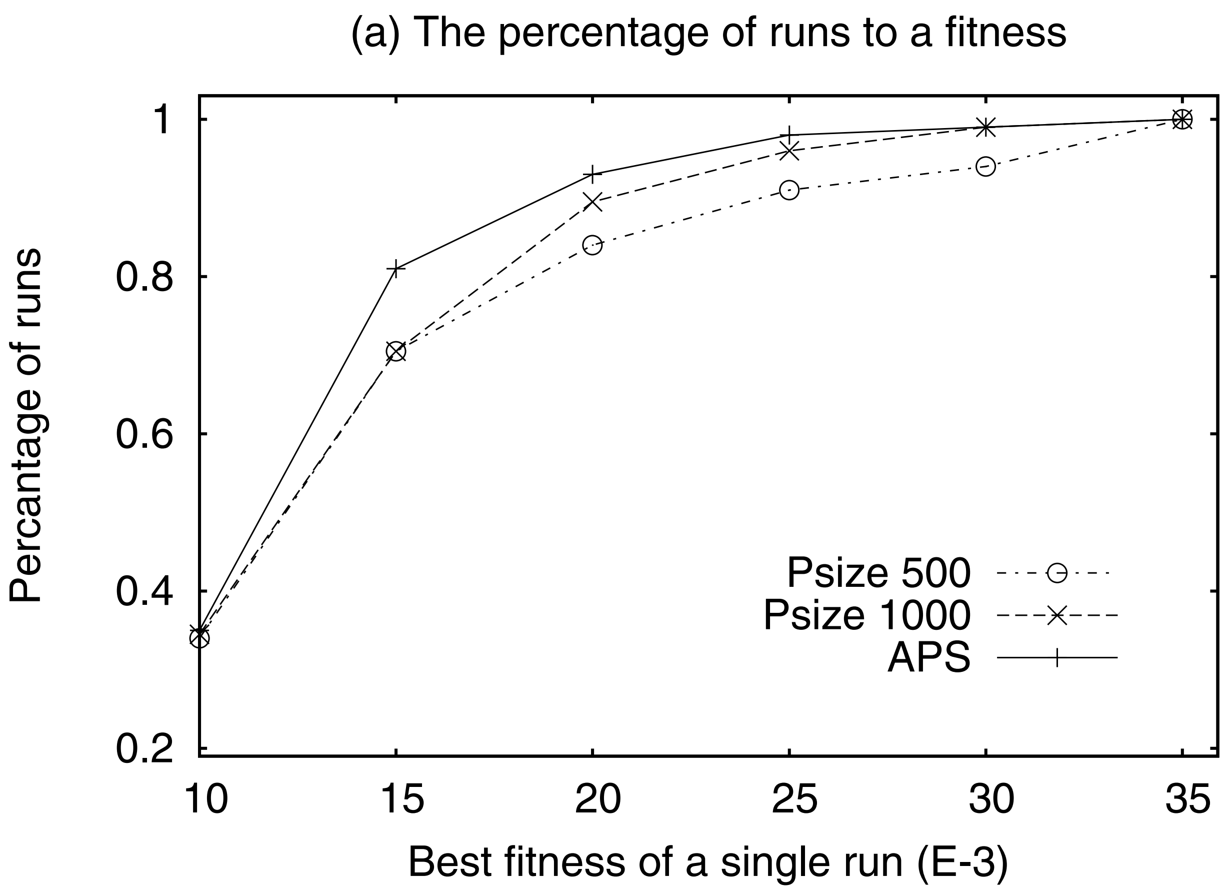
$$a = 0.2, b = 0.1, \tau = 17.$$

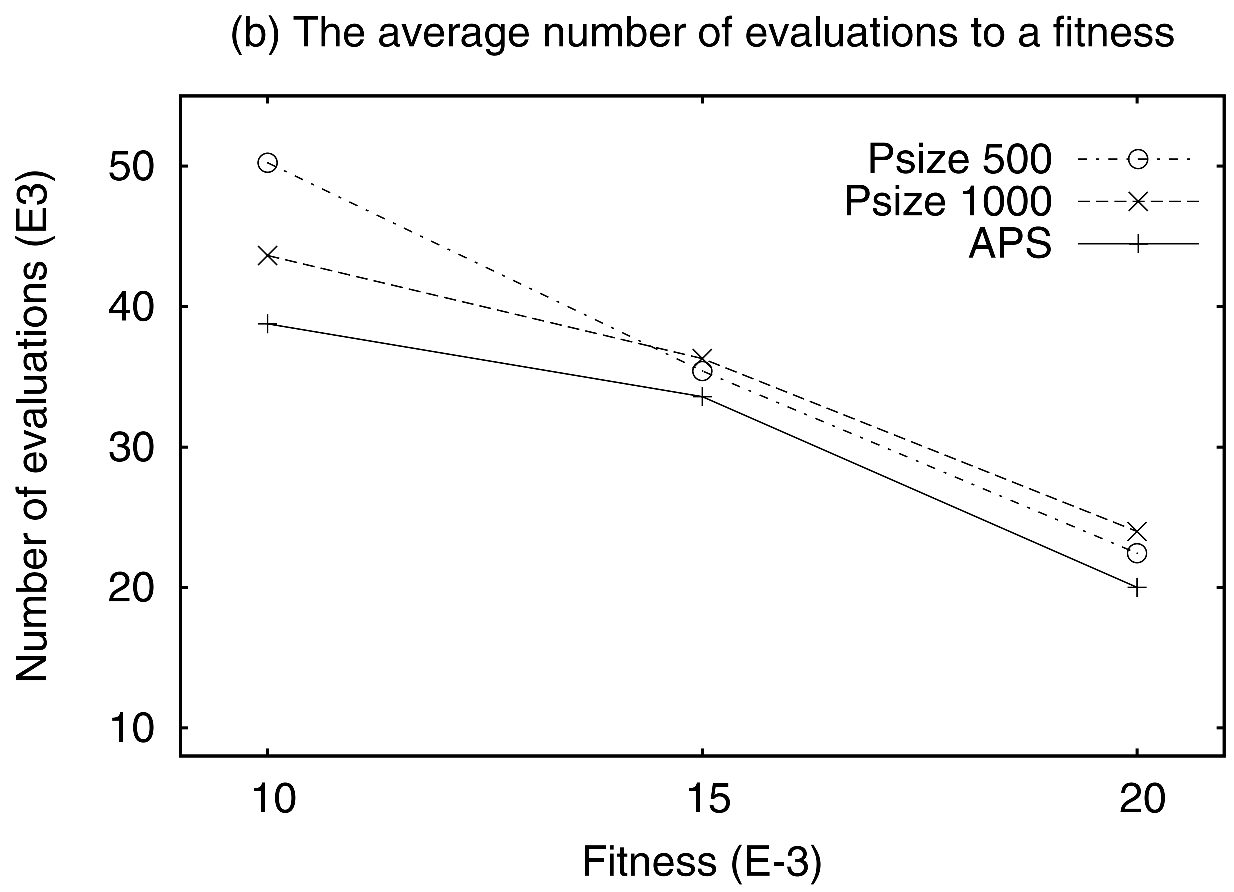
An example run





Performance comparison





Parameter control techniques

- Deterministic - some rule modifies strategy parameters without feedback from the search
- Adaptive - feedback rule based on some measure monitoring search progress
- Self-adaptive - parameter values evolve along with solutions; encoded onto chromosomes

Parameter control categorization

- Various forms of parameter control can be distinguished by
  - primary features
    - *what* component of the EA is changed
    - *how* the change is made (deterministic, adaptive, self-adaptive)
  - secondary features
    - evidence upon which the change is carried out (absolute, relative)
    - level/scope of change (one parameter, one individual, entire population)

Karafotias et al.: Trends and challenges in evolutionary algorithms parameter control, 2015