Evolution strategies

Chapter 4

ES quick overview

- Developed: Germany in the 1970's
- Early names: I. Rechenberg, H.-P. Schwefel
- Typically applied to:
  - numerical optimisation
- Attributed features:
  - fast
  - good optimizer for real-valued optimisation
  - relatively much theory
- Special:
  - **self-adaptation** of (mutation) strategy parameters

ES technical summary tableau

Representation	Real-valued vectors
Recombination	Discrete or intermediary (arithmetic)
Mutation	Gaussian perturbation
Parent selection	Uniform random
Survivor selection	(μ,λ) or $(\mu+\lambda)$
Specialty (Feature)	Self-adaptation of mutation step sizes σ

Introductory example

- Task: minimise $f : R^n \rightarrow R$
- Algorithm: "two-membered ES" , i.e. (1+1)-ES
  - Vectors from R^n directly as chromosomes
  - Population size: 1
  - Only mutation creating one child
  - Greedy selection

Introductory example: pseudocde

Begin
  Set $t = 0$
  Create an initial point $x^t = \langle x_1^t,\ldots,x_n^t \rangle \in R^n$
  REPEAT UNTIL (*TERMIN.COND* satisfied) DO
    Draw z_i from a normal distri. for all i = 1,…,n
    $y_i^t = x_i^t + z_i$
    IF $f(x^t) < f(y^t)$ THEN $x^{t+1} = x^t$
      ELSE
        $x^{t+1} = y^t$
    FI
    Set $t = t+1$
  OD
End

➔ minimization

Illustration of Gaussian distribution

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

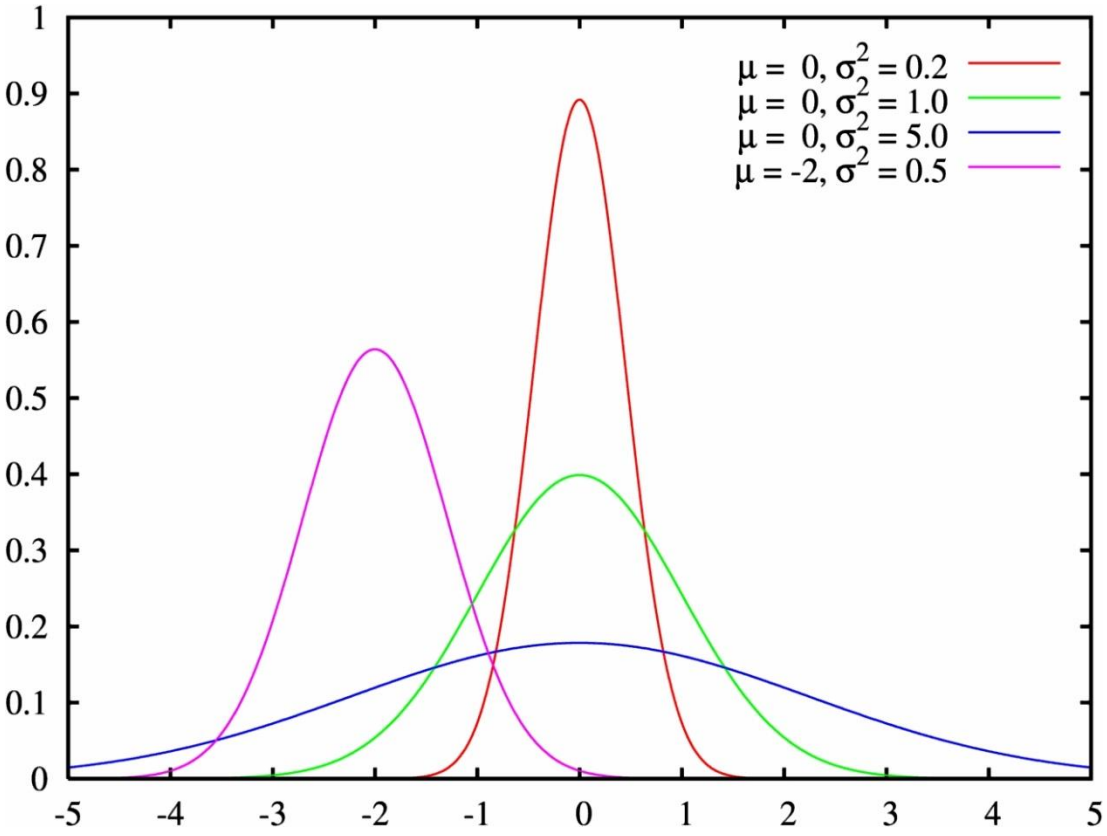
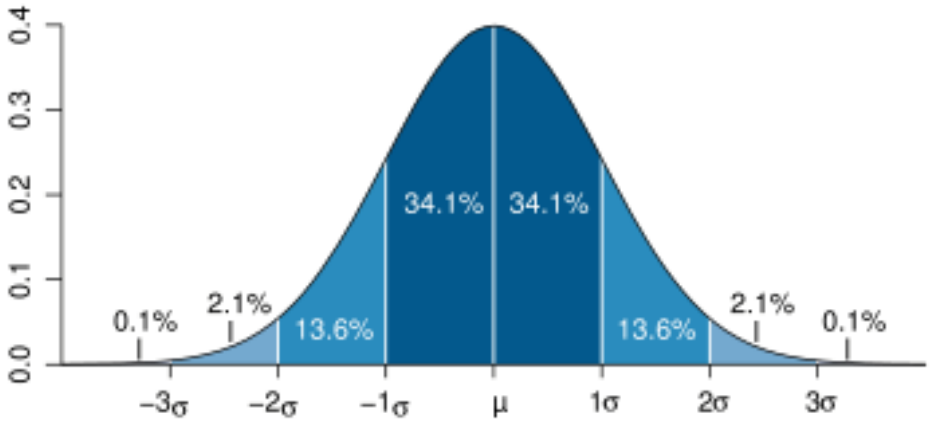


Illustration of normal distribution with zero mean and standard deviation



Dark blue is less than one standard deviation from the mean. For the normal distribution, this accounts for about 68% of the set (dark blue) while two standard deviations from the mean (medium and dark blue) account for about 95% and three standard deviations (light, medium, and dark blue) account for about 99.7%.

$$N(0,\tau) = \tau N(0,1)$$

N(0,1)	N(0,0.1)	N(0,0.01)	N(0,0.001)
randn(20,1)	0.1 * randn(20,1)	0.01 * randn(20,1)	0.001 * randn(20,1)
-0.6730	0.0808	-0.0021	0.0002
-0.1493	0.0041	-0.0020	0.0007
-2.4490	-0.0756	0.0031	-0.0006
0.4733	-0.0089	-0.0057	-0.0010
0.1169	-0.2009	-0.0098	-0.0002
-0.5911	0.1084	-0.0045	-0.0011
-0.6547	-0.0981	0.0108	-0.0001
-1.0807	-0.0688	0.0237	0.0003
-0.0477	0.1339	0.0023	0.0014
0.3793	-0.0909	-0.0027	0.0002
-0.3304	-0.0413	0.0070	-0.0005
-0.4999	-0.0506	-0.0049	0.0016
-0.0360	0.1620	0.0186	0.0008
-0.1748	0.0081	0.0111	0.0002
-0.9573	-0.1081	-0.0123	0.0007
1.2925	-0.1125	-0.0067	-0.0005
0.4409	0.1736	0.0134	0.0009
1.2809	0.1937	0.0039	0.0003
-0.4977	0.1635	0.0039	0.0006
-1.1187	-0.1256	-0.0171	-0.0010

Introductory example: mutation mechanism

- z values drawn from normal distribution $N(\mu,\sigma)$
  - mean μ is set to 0
  - Standard deviation (variation) σ is called **mutation step size**
- σ is varied across generations by the "1/5 success rule" of Rechenberg
- This rule resets σ after every *k* iterations by
  - $\sigma = \sigma / c$, if $p_s > 1/5$ (wider search step, exploration)
  - $\sigma = \sigma \cdot c$, if $p_s < 1/5$ (search around the current solution, exploitation)
  - $\sigma = \sigma$, if $p_s = 1/5$
- where p_s is the % of successful mutations over a number of trials, $0.8 \leq c \leq 1$

- Step sizes change based on the feedback from the search process
- Schwefel (1981) suggested a factor of c=0.817
- If the ratio is greater than 1/5, the step size should be increased to make a wider search of the space
- If the ratio is less than 1/5, the step size should be decreased to concentrate the search around the current solution.

➔ 1/5 success rule applied to (1+1)-ES

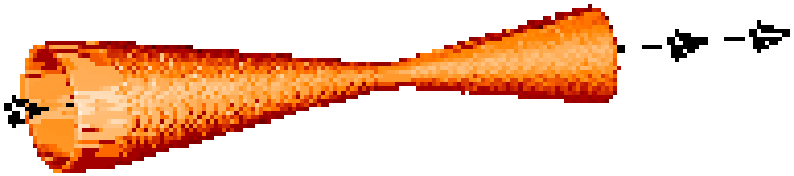
➔ ES uses self-adaption nowadays

Essential Characteristics of Evolution Strategies:

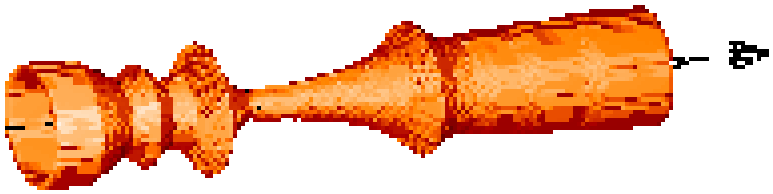
1. Typically used for continuous parameter optimization
2. Strong emphasis on mutation for creating offspring
3. Mutation is implemented by adding some random noise drawn from Gaussian distribution
4. Mutation parameters are changed during a run of the algorithm

Another historical example: the jet nozzle experiment

Task: to optimize the shape of a jet nozzle
Approach: random mutations to shape + selection

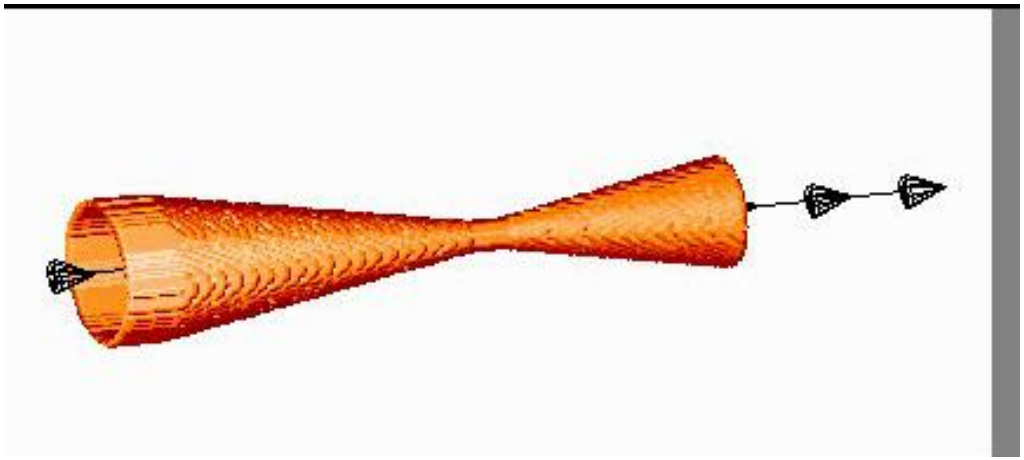


Initial shape



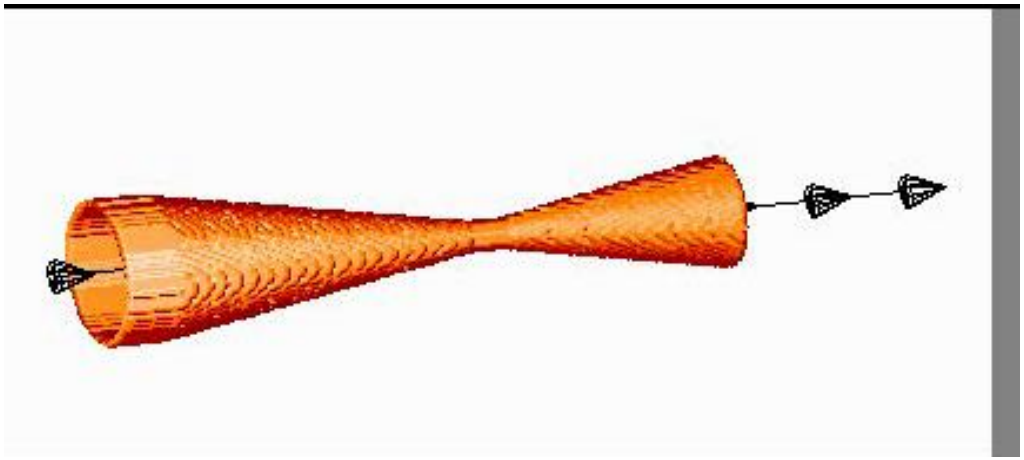
Final shape

Another historical example: the jet nozzle experiment cont'd



Jet nozzle: the movie

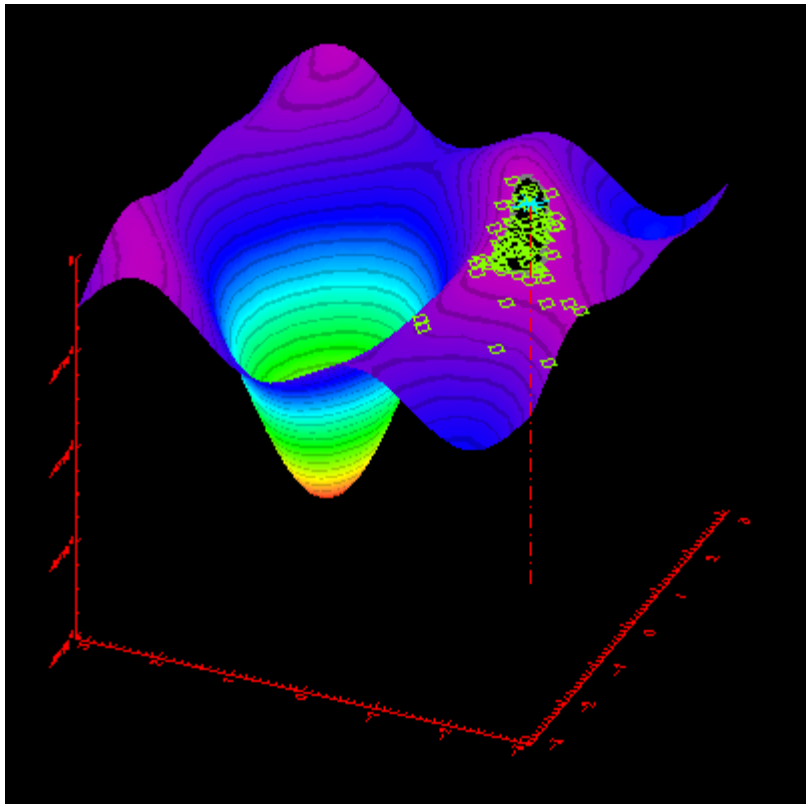
The famous jet nozzle experiment (movie)



Moving Optimum

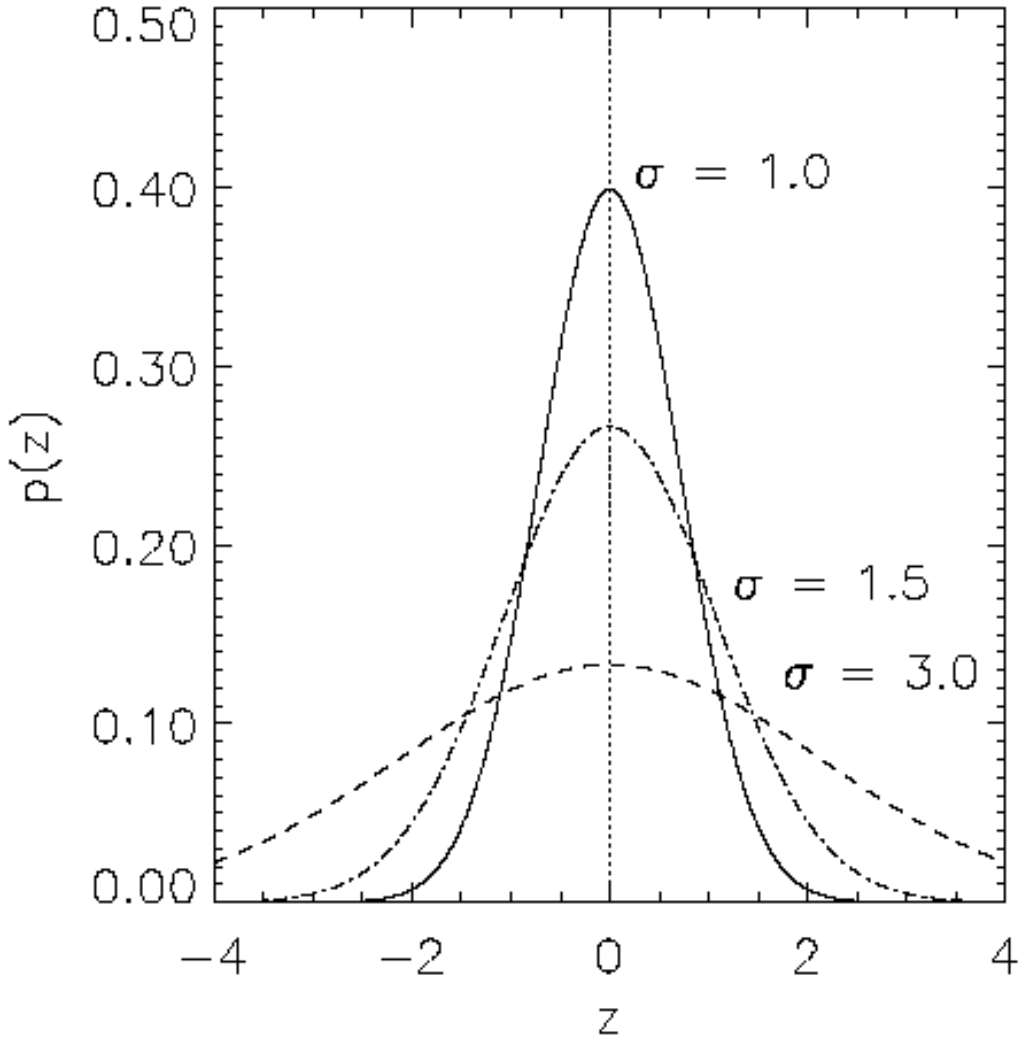
• The animation below depicts a (15, 100)-ES following the moving optimum (indicated by a light blue star) of a (modified) 30-dimensional Fletcher-Powell function.

• Due to memory limits and in order not to overcrowd the pictures with green diamonds each of the 70 frames only displays the best five individuals of 10 consecutive generations.



Genetic operators: mutations (2)

The one-dimensional case



Representation

- Chromosomes consist of three parts:
  - Object (floating-point) variables: $x_1,\ldots,x_n$
  - Strategy parameters:
    - Mutation step sizes: $\sigma_1,\ldots,\sigma_n$
    - Rotation angles: $\alpha_1,\ldots,\alpha_k$
- Not every component is always present
  - At least $\langle x_1,\ldots,x_n, \sigma \rangle$
- Full size: $\langle x_1,\ldots,x_n, \sigma_1,\ldots,\sigma_n, \alpha_1,\ldots,\alpha_k \rangle$
- where $k = n \cdot (n-1)/2$

Mutation

- Main mechanism: changing value by adding random noise drawn from normal distribution $N(\mu,\sigma)$

$$N(\mu,\sigma), \text{where } \mu: \text{mean}, \ \sigma: \text{standard deviation}$$

- $x'_i = x_i + N(0,\sigma)$
  - $N(0,\sigma)$ is a random number drawn from a Gaussian distribution with zero mean and standard deviation σ
  - Small mutations are more likely than large ones
- Key idea:
  - σ is part of the chromosome $\langle x_1,\ldots,x_n, \sigma \rangle$
  - σ is also mutated into σ' (see later how)
- Thus: mutation step size σ is co-evolving with the solution (object) x ➔ self adaptation

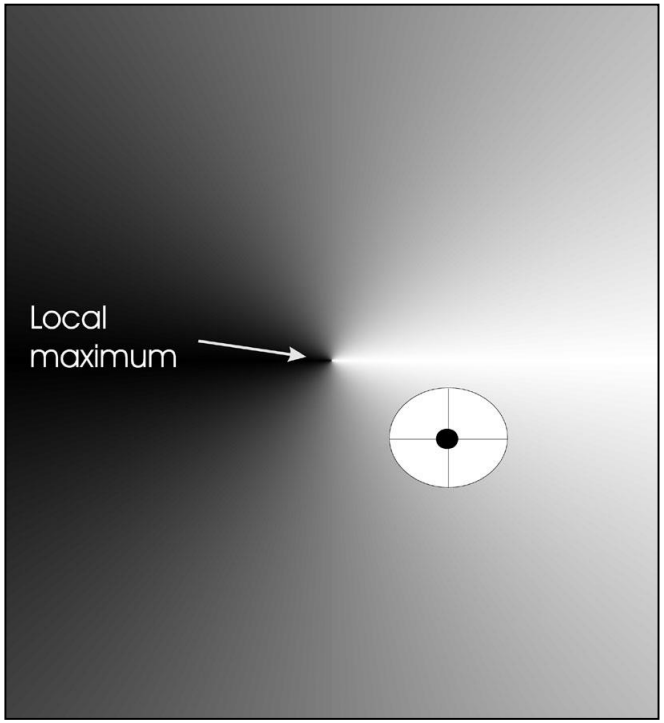
Mutate σ first

- Net mutation effect:
  - $\langle x_1,\ldots,x_n, \sigma \rangle \rightarrow \langle x_1',\ldots,x_n', \sigma' \rangle$
- Mutation size is NOT set by the user, but coevolving with the solutions
- Order is important:
  - first $\sigma \rightarrow \sigma'$ (see later how)
  - then $x \rightarrow x' = x + N(0,\sigma')$
- Rationale: new $\langle x', \sigma' \rangle$ is evaluated twice
  - Primary: x' is good if $f(x')$ is good
  - Secondary: σ' is good if the x' it created is good
- Reversing mutation order this would not work

Mutation case 1: Uncorrelated mutation with one σ

- The same distribution is used to mutate each x_i
- Chromosomes: $\langle x_1,\ldots,x_n, \sigma \rangle$
  - $\sigma' = \sigma \cdot \exp(\tau \cdot N(0,1))$ or $(\sigma' = \sigma \cdot \exp(N(0, \tau)))$
  - $x'_i = x_i + \sigma' \cdot N_i(0,1)$ or $(x'_i = x_i + N_i(0, \sigma'))$
- Typically the "learning rate" $\tau \propto 1/ n^{½}$, an external parameter set by the user
- And we have a boundary rule $\sigma' < \varepsilon_0 \Rightarrow \sigma' = \varepsilon_0$
  - σ' should be larger than 0
  - Very small σ' have a negligible effect, thus, unwanted

Mutants with equal likelihood



$n=2,\ n_\sigma=1,\ n_\alpha=0$

Chromosomes:
$\langle x_1, x_2, \sigma \rangle$

- The same step size for all dimensions
- Black dot: individual
- Points of offspring ➔ circle
- Mutation size is the same in each direction
- Circle: mutants having the same chance to be created
- Probability of moving along x and y axis is the same

Mutation case 2: Uncorrelated mutation with *n* σ's

- Use *n* step sizes to treat dimensions differently
- Chromosomes: $\langle x_1,\ldots,x_n,\ \sigma_1,\ldots,\sigma_n \rangle$
  - $\sigma'_i = \sigma_i \cdot \exp(\tau' \cdot N(0,1) + \tau \cdot N_i(0,1))$
  - $x'_i = x_i + \sigma'_i \cdot N_i(0,1)$

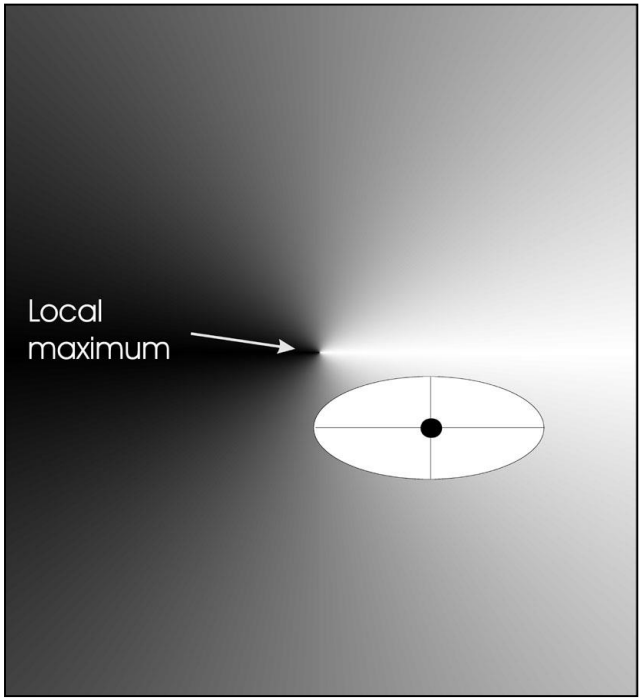
$$\sigma_i' = \sigma_i \cdot \exp(\tau \cdot N_i(0,1))$$
$$x'_i = x_i + \sigma_i' \cdot N_i(0,1)$$

←

$$\sigma' = \sigma \cdot \exp(\tau \cdot N(0,1))$$
$$x'_i = x_i + \sigma' \cdot N_i(0,1)$$

- Two learning rate parameters:
  - τ' overall learning rate (全體), individual level
  - τ coordinate-wise learning rate (個別維度)
- $\tau' \propto 1/(2n)^{1/2}$ and $\tau \propto 1/(2n^{1/2})^{1/2}$
- And $\sigma_i' < \varepsilon_0 \Rightarrow \sigma_i' = \varepsilon_0$

Mutants with different likelihood



$n=2,\ n_\sigma=2,\ n_\alpha=0$

Chromosomes:
$\langle\ x_1, x_2, \sigma_1, \sigma_2\ \rangle$

- Different step size for each dimension (axis)
- Black dot: individual
- Points of offspring ➔ Ellipse
- Ellipse: mutants having the same chance to be created
- Probability of moving along *x* is larger than that moving along *y* axis

Mutation case 3: Correlated mutations

- Allow ellipses to have any orientation (方向) by rotating them with a rotation
- Chromosomes: $\langle x_1,\ldots,x_n, \sigma_1,\ldots, \sigma_n , \alpha_1,\ldots, \alpha_k \rangle$ where $k = n \cdot (n-1)/2$
- The mutation mechanism is then:

$$\begin{aligned}
\sigma_i^{(t+1)} &= \sigma_i^{(t)} \exp\left(\tau' N(0,1) + \tau N_i(0,1)\right), \\
\alpha_j^{(t+1)} &= \alpha_j^{(t)} + \beta_\alpha N_j(0,1), \\
\mathbf{x}^{(t+1)} &= \mathbf{x}^{(t)} + \mathbf{N}\left(\mathbf{0}, C(\boldsymbol{\sigma}^{(t+1)}, \boldsymbol{\alpha}^{(t+1)})\right),
\end{aligned}$$

$\mathbf{N}\left(\mathbf{0}, C(\boldsymbol{\sigma}^{(t+1)}, \boldsymbol{\alpha}^{(t+1)})\right)$ is a realization of a normally distributed correlated mutation vector with zero mean vector and a covariance matrix C

Mutation case 3: Correlated mutations

$\boldsymbol{C}$ is the covariance matrix after mutation of the α values, defined as

$$c_{ii} = \sigma_i^2$$
$$c_{ij} = \tfrac{1}{2} (\sigma_i^2 - \sigma_j^2) \bullet \tan(2 \alpha_{ij})$$

Example: Chromosome $\langle x_1 , x_2 , x_3, x_4, \sigma_1,\ldots, \sigma_n, \alpha_1,\ldots, \alpha_k \rangle$

where $k = n \cdot (n-1)/2$

$$<x_1, x_2, x_3, x_4, \sigma_1,\sigma_2,\sigma_3,\sigma_4,\alpha_{12},\alpha_{13},\alpha_{14},\alpha_{23},\alpha_{24},\alpha_{34}>$$

Correlated mutations cont'd

$$C=\begin{bmatrix} \sigma_1^2 & c_{12} & c_{13} & c_{14} \\ c_{21} & \sigma_2^2 & c_{23} & c_{24} \\ c_{31} & c_{32} & \sigma_3^2 & c_{34} \\ c_{41} & c_{42} & c_{43} & \sigma_4^2 \end{bmatrix}$$

$$c_{12} = \frac{1}{2}(\sigma_1^2 - \sigma_2^2) \cdot \tan(2\alpha_{12})$$

$$c_{13} = \frac{1}{2}(\sigma_1^2 - \sigma_3^2) \cdot \tan(2\alpha_{13})$$

$$\vdots$$

$$c_{21} = \frac{1}{2}(\sigma_2^2 - \sigma_1^2) \cdot \tan(2\alpha_{12})$$

$$\vdots$$

The mutation mechanism is then:

$\sigma'_i = \sigma_i \cdot \exp(\tau' \cdot N(0,1) + \tau \cdot N_i(0,1))$

$\alpha'_j = \alpha_j + \beta \cdot N(0,1)$

$\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{N(0,C)}$

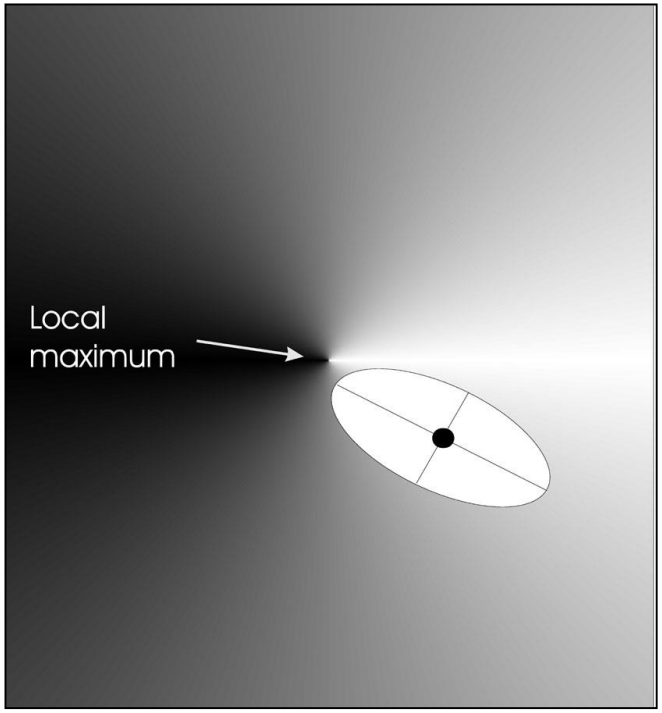
- $\boldsymbol{x}$ stands for the vector $\langle x_1,\ldots,x_n \rangle$

$\tau' \propto 1/(2\ n)^{½}$ and $\tau \propto 1/(2\ n^{½})^{½}$ and $\beta \approx 5^{\circ} = 0.0873$

$\sigma_i' < \varepsilon_0 \Rightarrow \sigma_i' = \varepsilon_0$ and

$|\ \alpha'_j\ | > \pi \Rightarrow \alpha'_j = \alpha'_j - 2\ \pi\ \mathrm{sign}(\alpha'_j)$, since the rotation angles lie in the range $[-\pi, +\pi]$

Rotated Ellipse



$n=2,\ n_\sigma=2,\ n_\alpha=1$

Chromosomes:

$\langle\ x_1,\ x_2,\ \sigma_1,\sigma_2,\ \alpha\ \rangle$

- Black dot: individual
- Points of offspring ➔ rotated Ellipse
- Probability of moving in the direction of the steepest ascent is now larger than that for other directions.

Recombination

- ES involves two parents to create one child
- Perform λ times to generate λ offspring
- Local recombination:
  - Averaging parental values (intermediary)
  - one of the parental alleles is randomly chosen with equal chance (discrete)
- Global recombination (multi-parent recombination)
  - A set of ρ randomly chosen parents (eg. 2 parents x_i and y_i) are drawn from the whole population for each position $i \in \{1,2,3,\ldots,n\}$
  - Using the selected parents to make a child, i.e. more than two individuals contributing to the offspring
    - Averaging parental values
    - one of the parental alleles is randomly chosen with equal chance

Recombination

Two fixed parents x_i and y_i	Two parents randomly selected for each position i	ρ randomly chosen parents for each i (multi-parent)
Local intermediary (whole arithmetic) $z_i = (x_i + y_i)/2$	Global intermediary	Global intermediary $y = \frac{1}{\rho}\sum_{i=1}^{\rho} x^{(i)}$
Local discrete z_i is x_i or y_i chosen randomly	Global discrete	Global discrete

Global **discrete** recombination for $\rho=4$

Parent 1:	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	$x_4^{(1)}$	$x_5^{(1)}$	$x_6^{(1)}$
Parent 2:	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	$x_4^{(2)}$	$x_5^{(2)}$	$x_6^{(2)}$
Parent 3:	$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	$x_4^{(3)}$	$x_5^{(3)}$	$x_6^{(3)}$
Parent 4:	$x_1^{(4)}$	$x_2^{(4)}$	$x_3^{(4)}$	$x_4^{(4)}$	$x_5^{(4)}$	$x_6^{(4)}$
Recombinant:	$x_1^{(2)}$	$x_2^{(3)}$	$x_3^{(4)}$	$x_4^{(2)}$	$x_5^{(4)}$	$x_6^{(3)}$

- ES typically uses global recombination
- Different recombination is used for the object variable part, i.e. object variables and strategy parameters
- (Global) Discrete recombination is recommended for object variable part
- (Global) intermediary recombination is recommended for strategy parameter part (to assure a more cautious adaption of strategic parameters)

Parent selection

- Parents are selected by uniform random distribution whenever an operator needs one/some
- Thus: ES parent selection is unbiased - every individual has the same probability to be selected (cf: parent selection is based on fitness for GA)
- In ES, “parent” means a population member (in GA’s: a population member selected to undergo variation)

Survivor selection

- Applied after creating λ children from the μ parents by mutation and recombination
- Deterministically chops off the "bad stuff" by selecting best μ of them
- Basis of selection is either:
  - The set of children only: (μ,λ)-selection
    - Select μ from λ
  - The set of parents and children: $(\mu+\lambda)$-selection
    - Select μ from $(\mu+\lambda)$
- Selection schemes are strictly based on rank rather than absolute fitness value

Survivor selection cont'd

- $(\mu+\lambda)$-selection is an elitist strategy
- (μ,λ)-selection can "forget", i.e. discarding all parents
- Often (μ,λ)-selection is preferred over $(\mu+\lambda)$
  - Better in leaving local optima, in case of multi-modal topologies
  - Better in following moving optima, where the fitness function is not fixed, where the $(\mu+\lambda)$ selection might preserve outdated solutions
  - Using the + strategy, bad σ values can survive in $\langle x,\sigma \rangle$ longer, resulting in bad offspring

- Selective pressure in ES is very high
- $\lambda \approx 7 \bullet \mu$ is a common setting
- Typically, μ=15 and λ =100

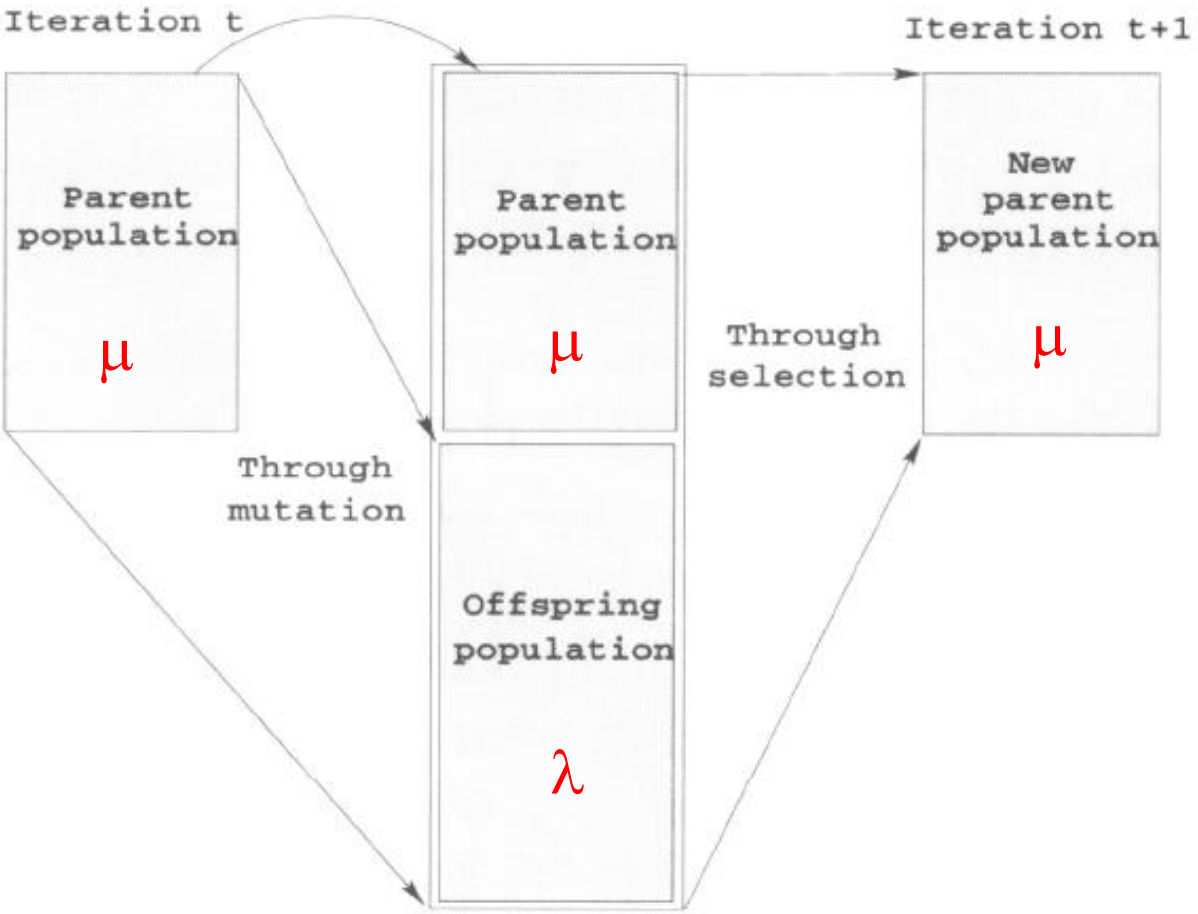


Figure 74 The procedure for a $(\mu + \lambda)$-ES.

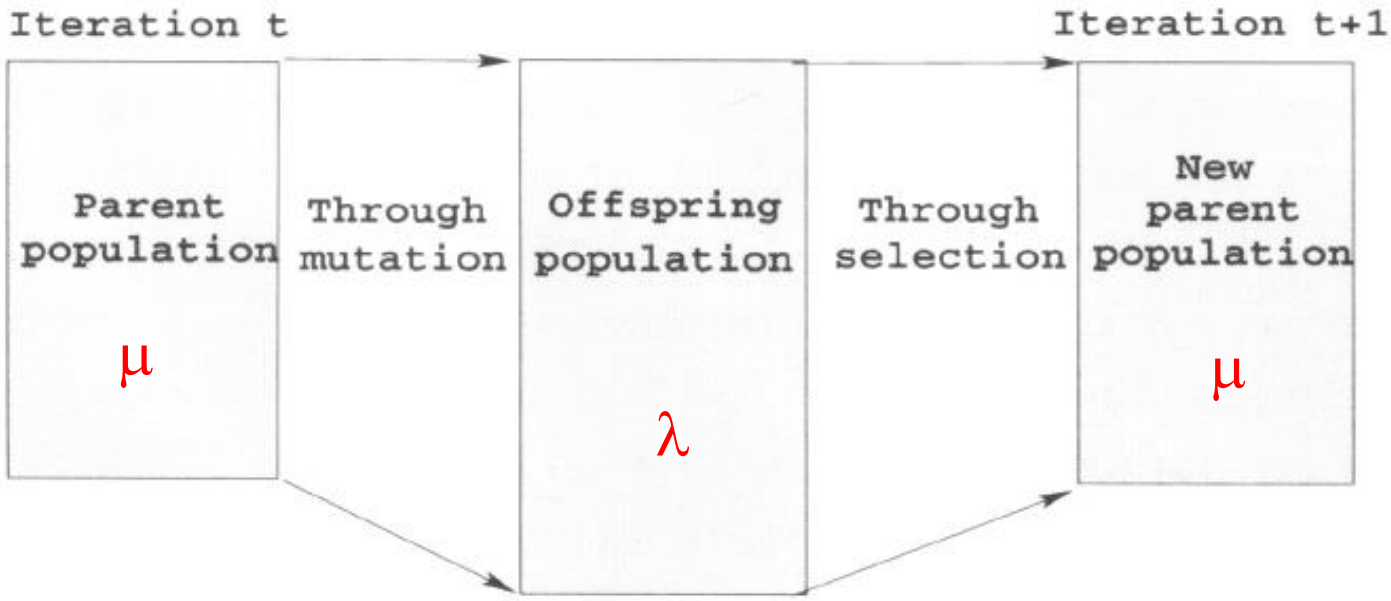


Figure 75 The procedure for a (μ, λ)-ES.

Self-adaptation illustrated

- ES with self-adaptation outperforms the same ES without self-adaptation
- Supported by theoretical and experimental results, where theoretical optimal step sizes can be calculated for a objective function f: $R^n \rightarrow R$
- Given a dynamically changing fitness landscape (optimum location shifted every 200 generations)

Self-adaptive ES is able to

- follow the optimum and
- adjust the mutation step size after every shift !

- For a successful run, the step size σ must decrease over time

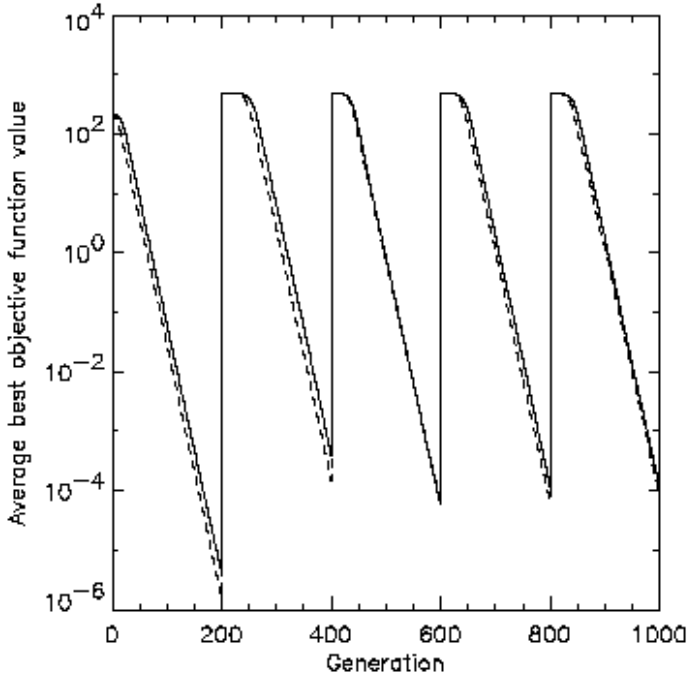
  - In the beginning of a search process, a larger σ is preferred to allow a explorative search to locate promising regions

  - In the later stage, a smaller σ is preferred to fine-tune the individuals

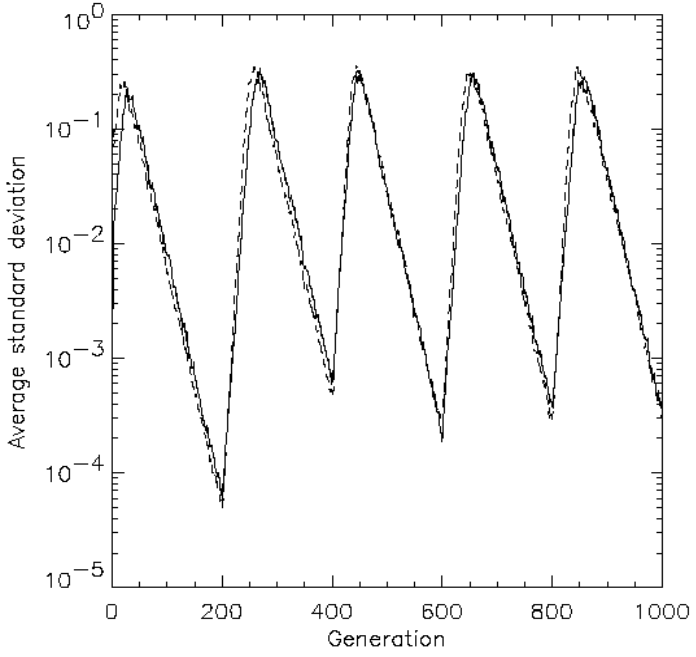
Self-adaptation illustrated cont'd

- Changing fitness landscapes:
- optimum location shifted every 200 generations

Objective value



Step size



$n=30$

$n_\sigma=1$

$n_\alpha=0$

Changes in the fitness (objective) values (left) and the mutation step sizes (right)

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i^2$$

- Unimodal with only one minimum point x^*=(0,0,0,…..,0)
- Global minimum point is shifted every 200 generations
- μ=8 and λ =50

Recommendations for self-adaptation

- Accumulated acknowledge over the last decade to identify necessary conditions for self-adaptation:
  - $\mu > 1$ to carry different strategies
  - $\lambda > \mu$ to generate offspring surplus
  - Not "too" strong selection, e.g., $\lambda \approx 7 \cdot \mu$ (e.g. <15,100>)
  - (μ,λ)-selection to get rid of mis-adapted σ's
  - Recombination also on strategy parameters (especially by intermediary recombination)

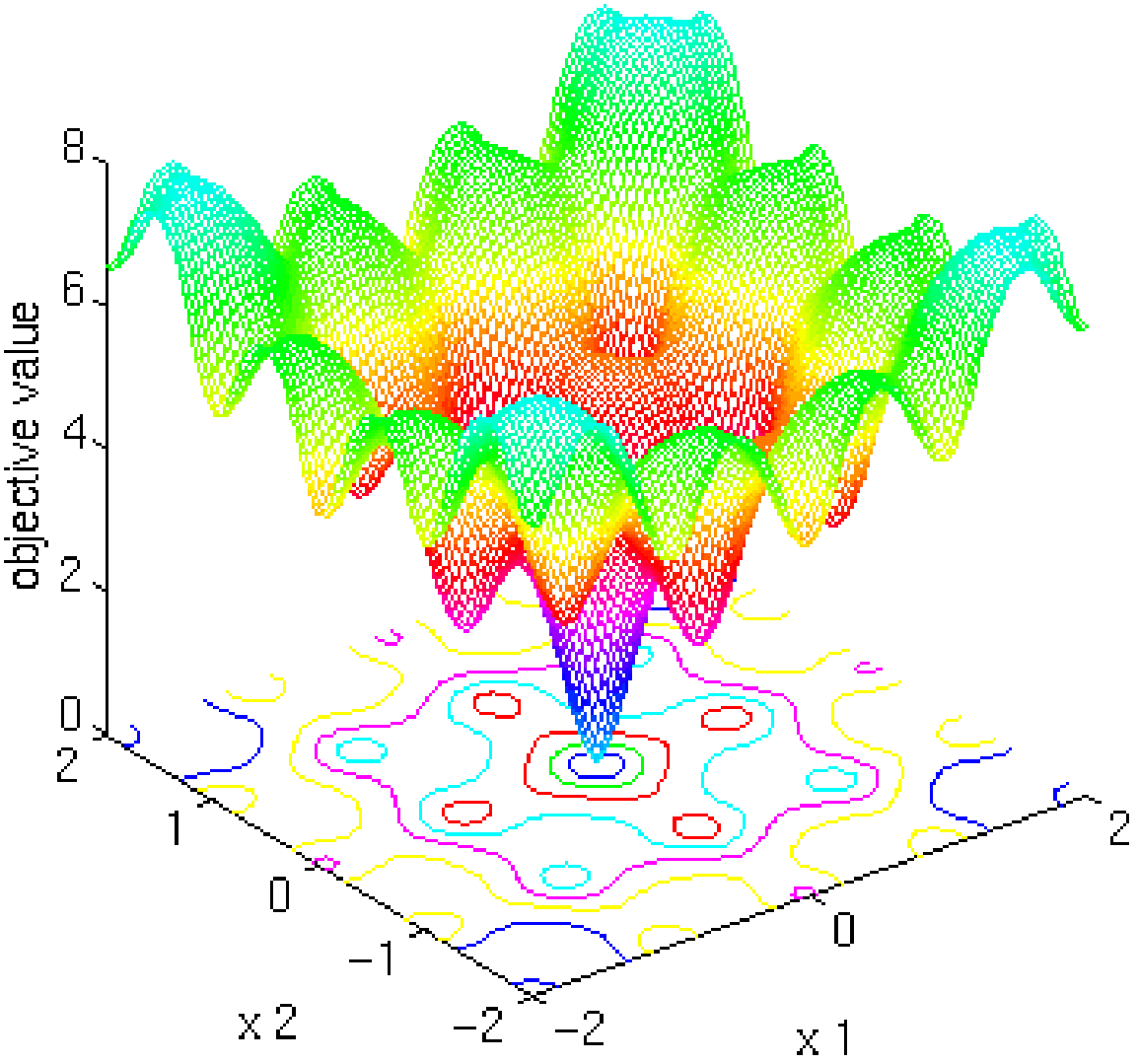
Example application: the Ackley function (Bäck et al '93)

A widely used multimodal test function

$$f(\vec{x}) = -c_1 \cdot \exp\left(-c_2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\cdot\sum_{i=1}^{n}\cos(c_3\cdot x_i)\right) + c_1 + e$$

$$c_1 = 20; c_2 = 0.2; c_3 = 2\pi; n = 30; -30 \leq x_i \leq 30.$$

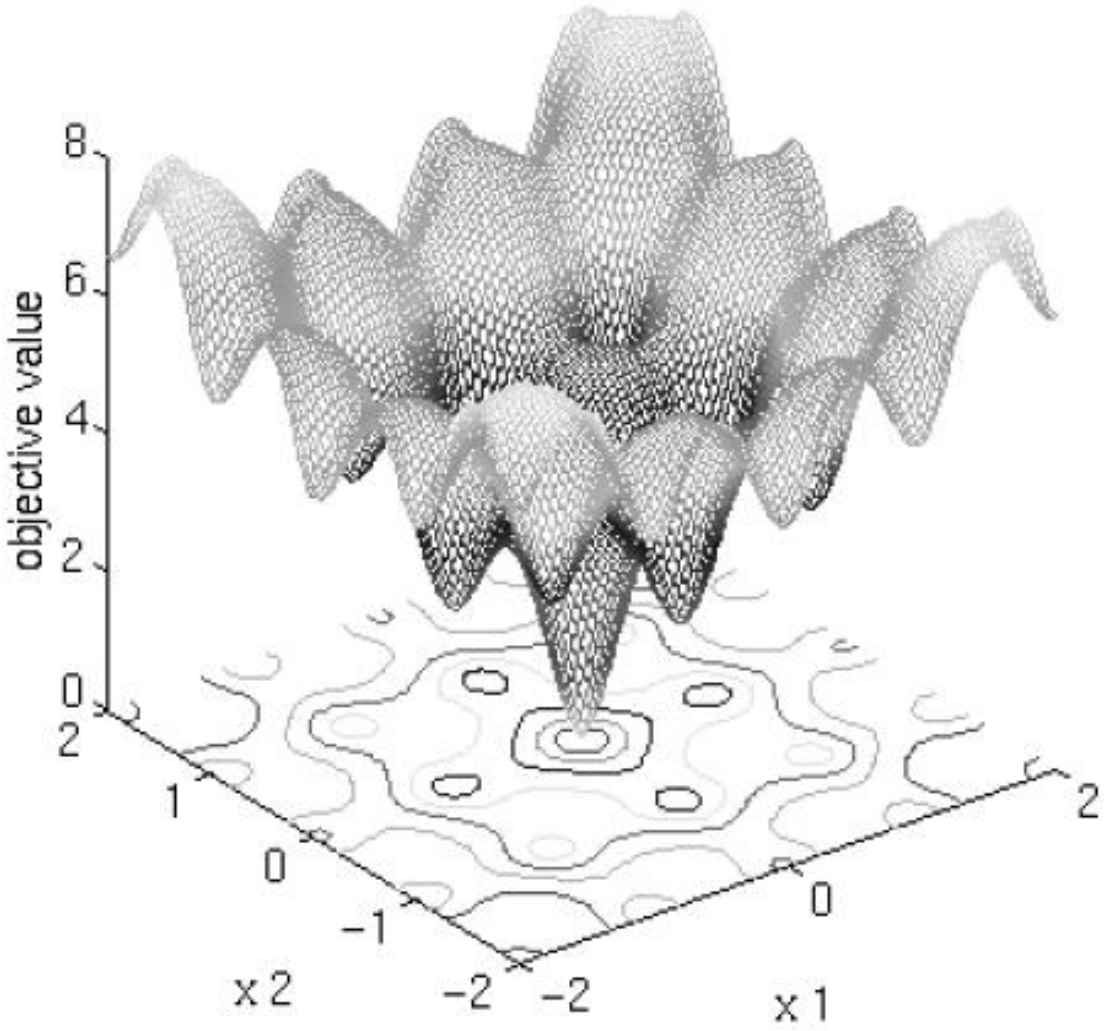
The optimum solution is the vector v = (0,...,0) with F(v) = 0.



- The Ackley function (here used with *n* =30):

$$f(x) = -20 \cdot \exp\left(-0.2\sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$$

- Evolution strategy:
  - Representation:
    - $-30 < x_i < 30$ (coincidence of 30's!)
  - (30,200) selection
  - Discrete recombination for object variable, and global intermediate recombination for strategy parameters
  - Termination : after 200000 fitness evaluations (FES)
  - Results: average best solution is $7.48 \cdot 10^{-8}$ (very good)


objective value
8
6
4
2
0
2
1
0
−1
−2
x 2
−2
0
2
x 1

Prof. Dr. rer. nat. habil. Hans-Georg Beyer

http://www2.staff.fh-vorarlberg.ac.at/~hgb/downloads.html

simple self-adaptive (mu/mu_I, lambda)-sigma-SA-ES

simple Covariance Matrix Adaptation ES (CMA-ES)

Comparison of ES and GA

ES: developed for numerical optimization, later for discrete optimization (integer, permutation, etc)

GA: general purpose search technique, including real parameter optimization

➔ Unfair to compare time and precision performance using a numerical function, which is the focus (strength) of ES.

Differences between ES and GA

1. Representation of individuals:
   - floating point vectors or binary (SGA) vectors by GA
   - Co-evolution of strategy parameters by ES
2. Selection process:

   ES:

   - μ parents ➔ λ offspring ➔ intermediate population of $(\mu+\lambda)$ or λ ➔ reduced to μ individuals by removing the least fit individuals from $(\mu+\lambda)$ or λ
   - Deterministic: best μ out of $(\mu+\lambda)$ or λ
   - Static, extinctive, no repetition!

GA:

- pop_size individuals are selected from the population
- strong individuals have good chance to be selected several times, i.e. repetition
- Even the weakest individual has a chance of being selected
- random!
- Dynamic, preservative, with repetition!

3. Order of selection and recombination

ES: recombination & mutation operators ➔ intermediate population ($\mu+\lambda$) or λ ➔ selection (μ)

GA: selection ➔ intermediate population ➔ recombination & mutation operators

4. Reproduction parameters:

GA: pm, pc remain constant

ES: σ and α change all the time

Really that different?

1. Complement each other over the past years

   • GA borrows adaptation from ES, eg. Non-uniform mutation

   • ES borrows crossover (recombination) from GA

   ➔ Even score!

2. Operators introduced simultaneously into GA and ES

$$\mathbf{y}_1 = a\mathbf{x}_1 + (1-a)\cdot\mathbf{x}_2$$
$$\mathbf{y}_2 = a\mathbf{x}_2 + (1-a)\cdot\mathbf{x}_1$$

   • GA: guaranteed average crossover or arithmetical crossover

   • ES: an intermediate crossover