

ALHE

Jarosław Arabas

Metaheurystyki w R^n

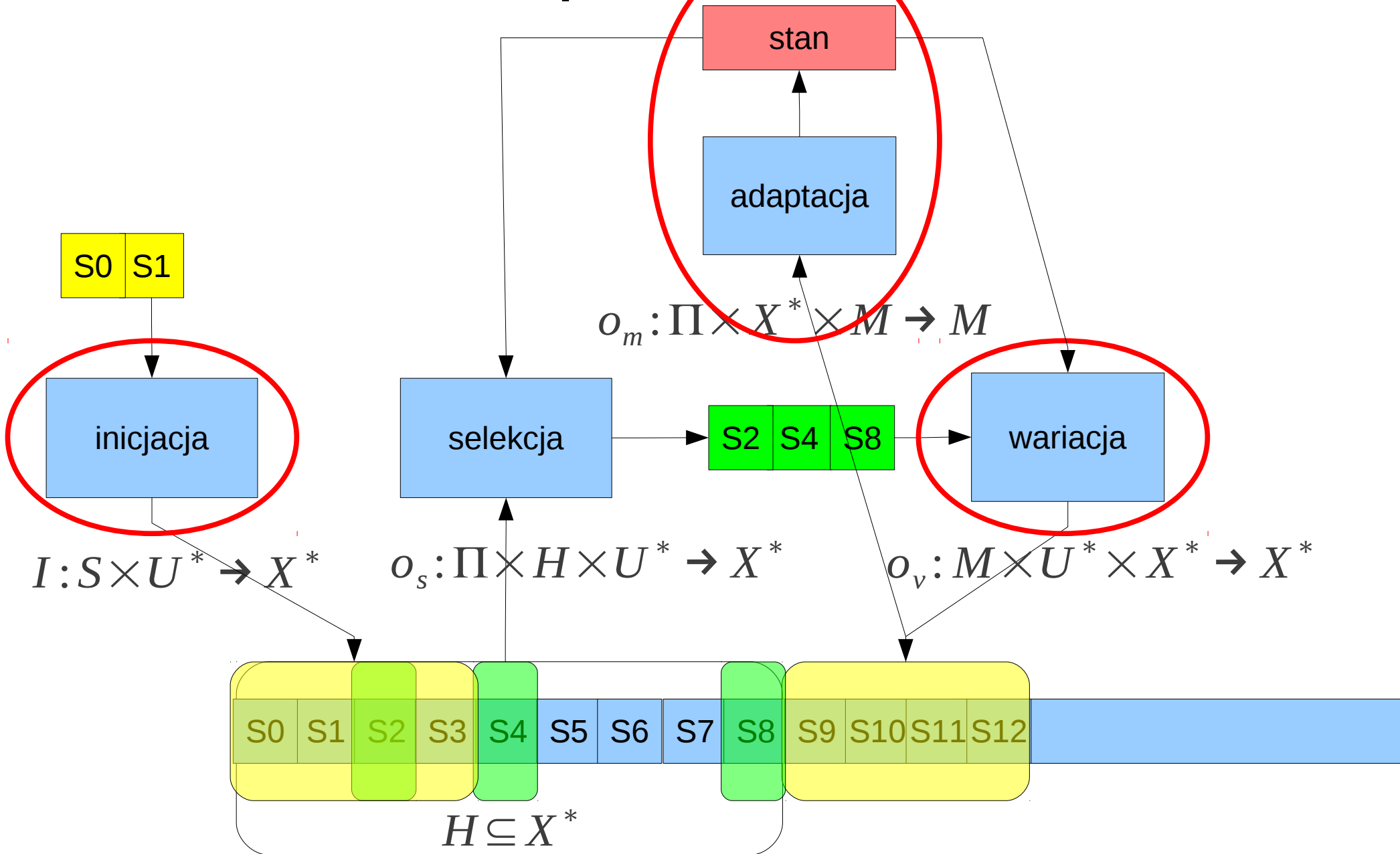
Ewolucja różnicowa

EDA

CMAES

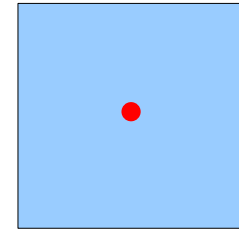
Rój cząstek

Metoda przeszukiwania

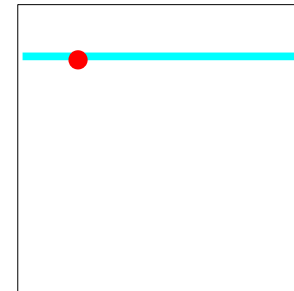


Typowe sposoby generacji sąsiadów

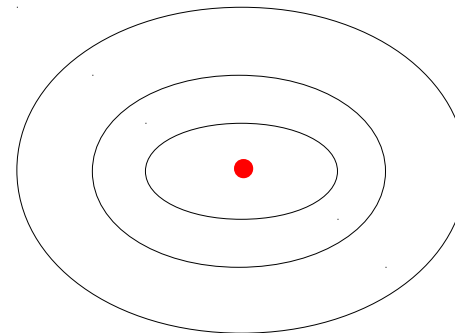
- Rozkład jednostajny w kostce



- Rozkład jednostajny w podprzestrzeni

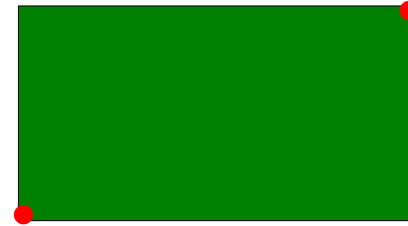


- Rozkład normalny

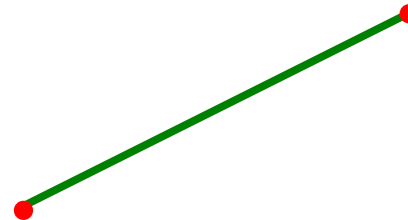


Typowe sposoby generacji punktów “pomiędzy”

- Rozkład jednostajny w kostce



- Rozkład jednostajny na odcinku łączącym



- Rozkład jednostajny na zbiorze narożników kostki

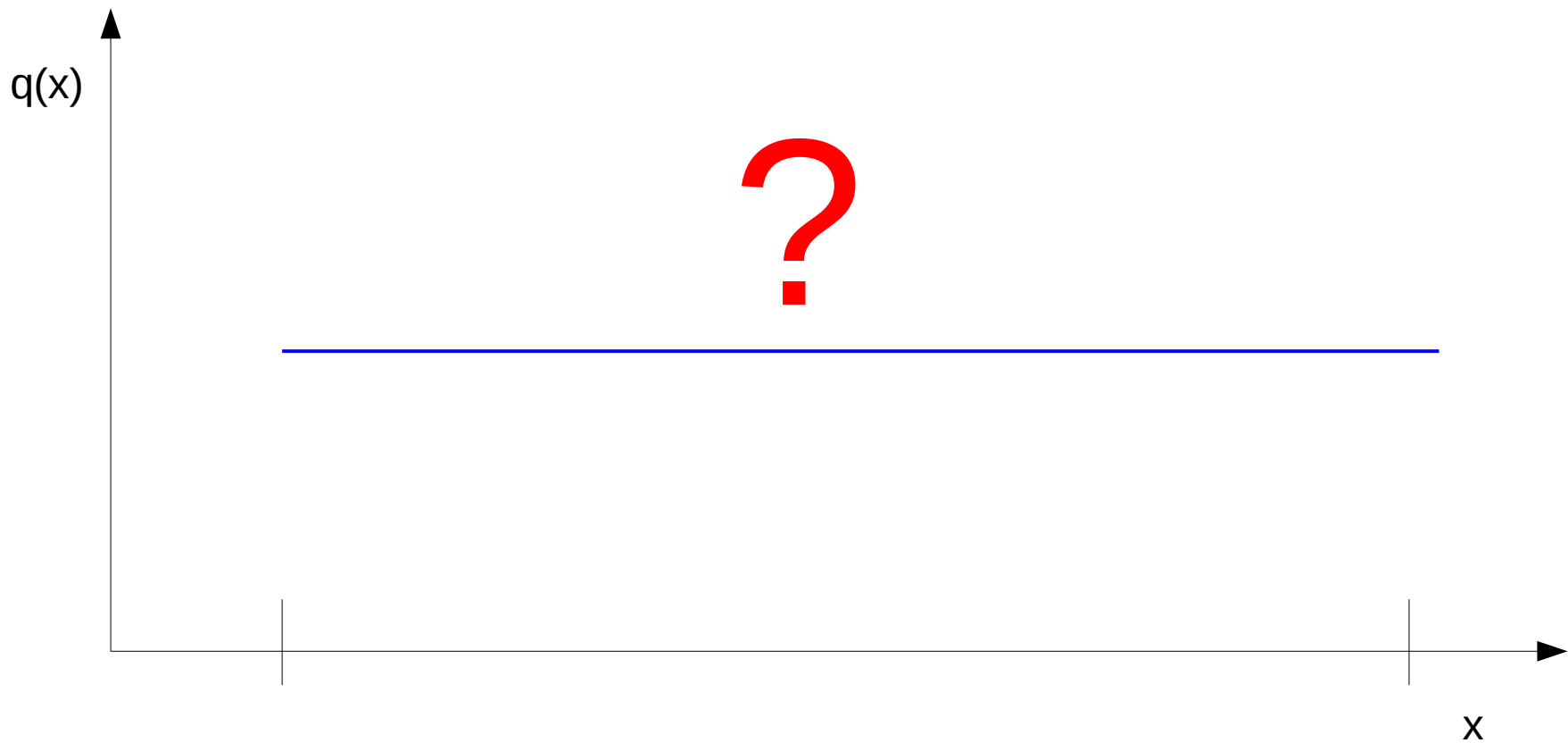


Rozkład próbkowania

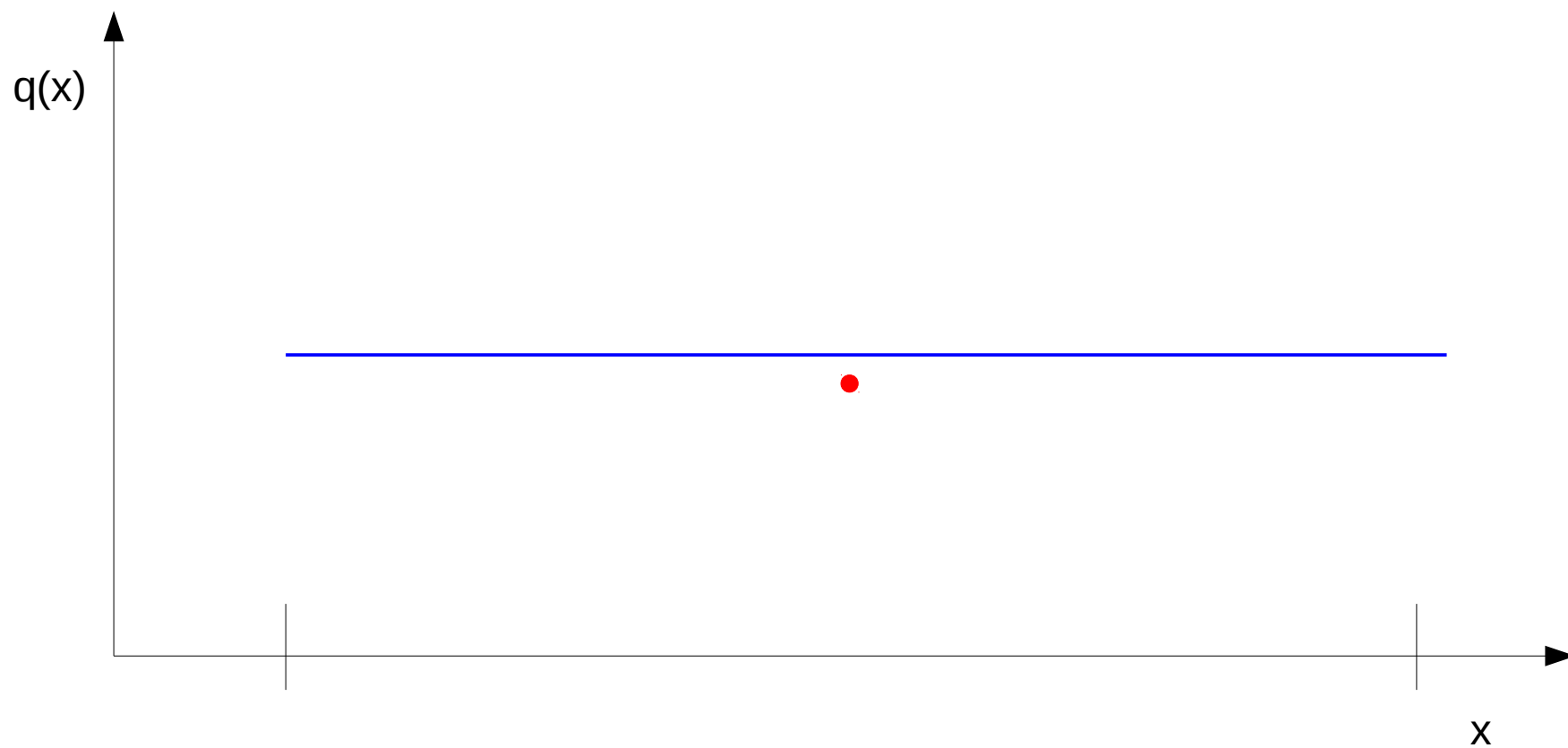
- Sposób tworzenia nowych punktów można opisać za pomocą zmiennej losowej, której rozkład da się analizować
- Jest to rozkład próbkowania
- Rozkład próbkowania może być założony z góry lub może się adaptować
- Przykłady?

Optymalizacja jako proces adaptacji

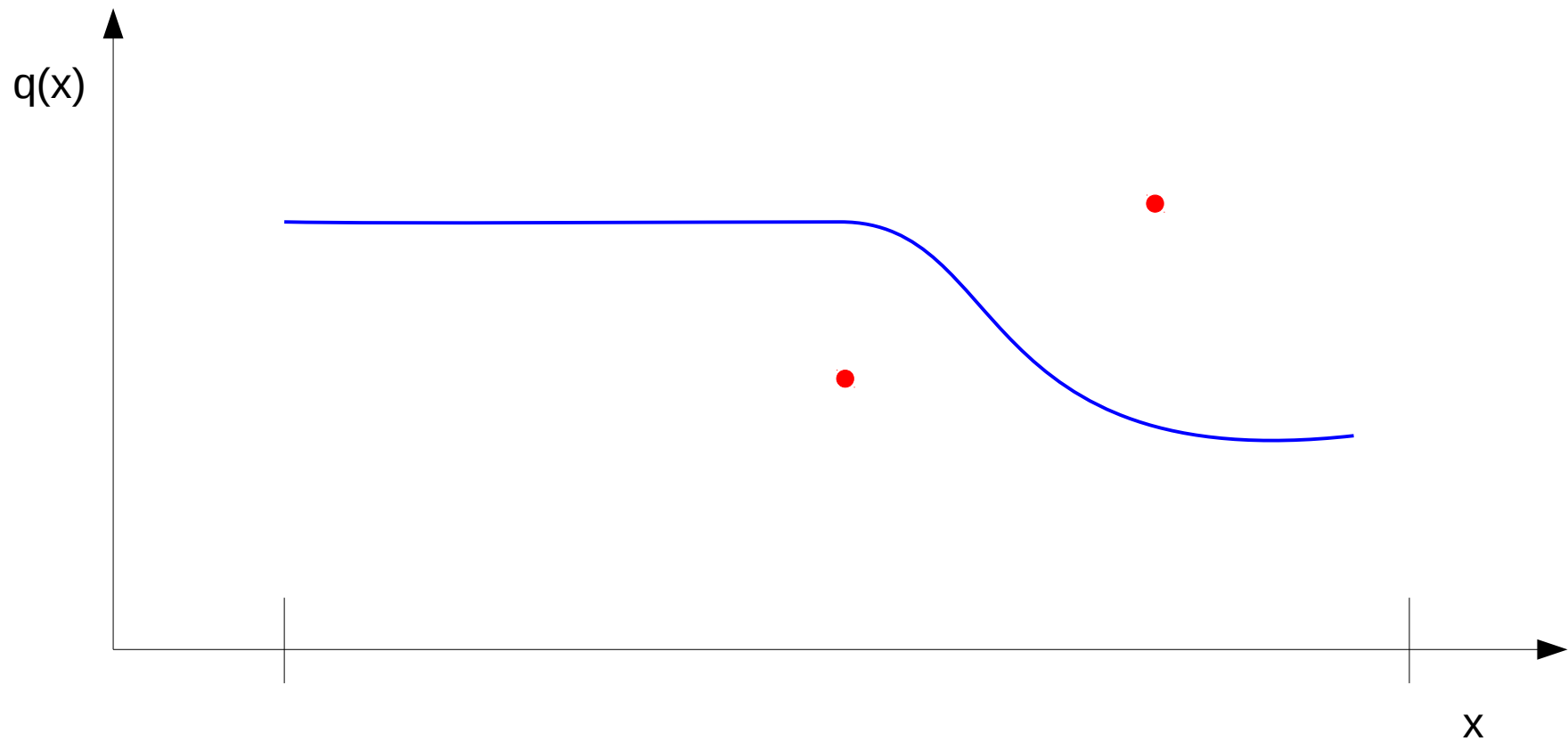
- Przekonanie o położeniu optimum



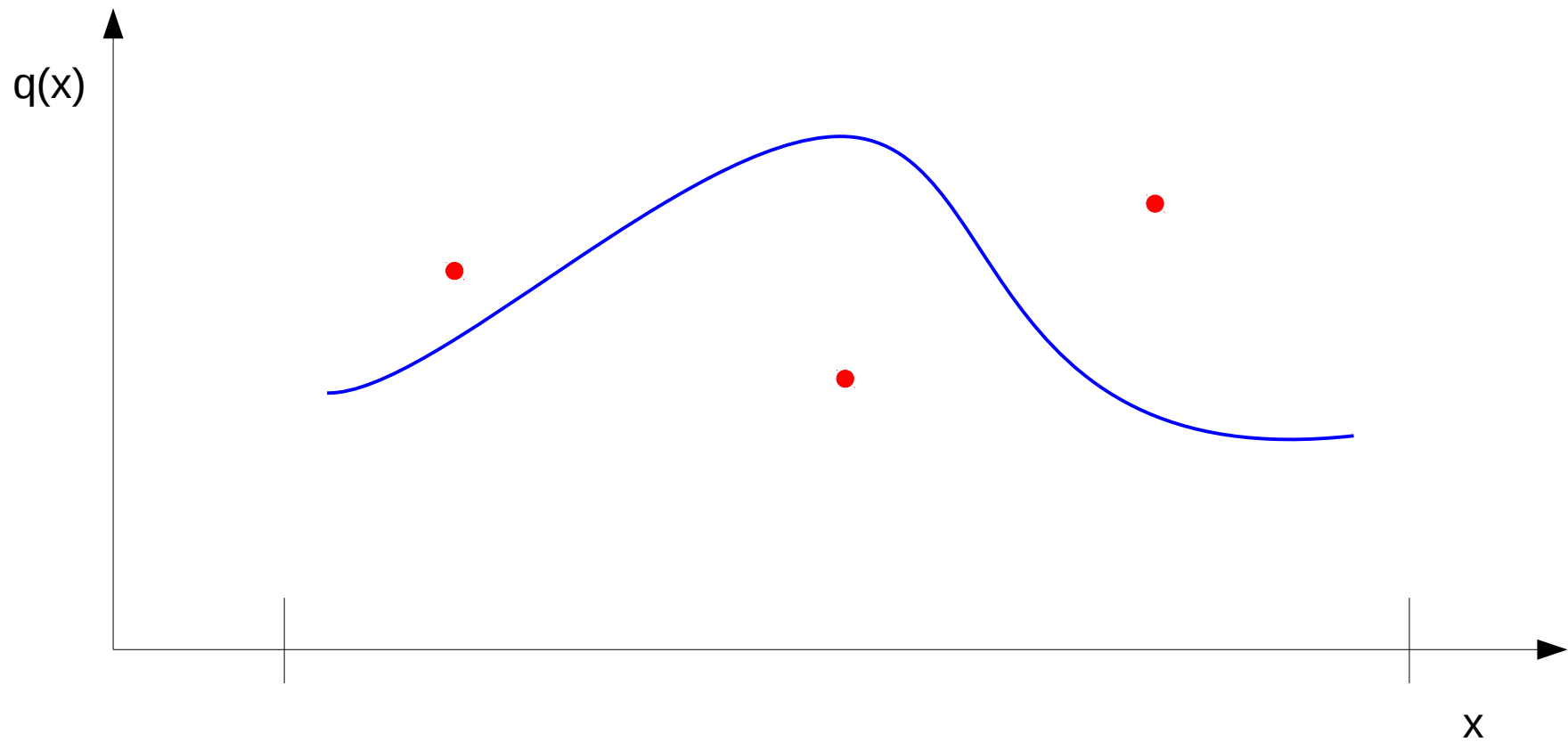
Optymalizacja jako proces adaptacji



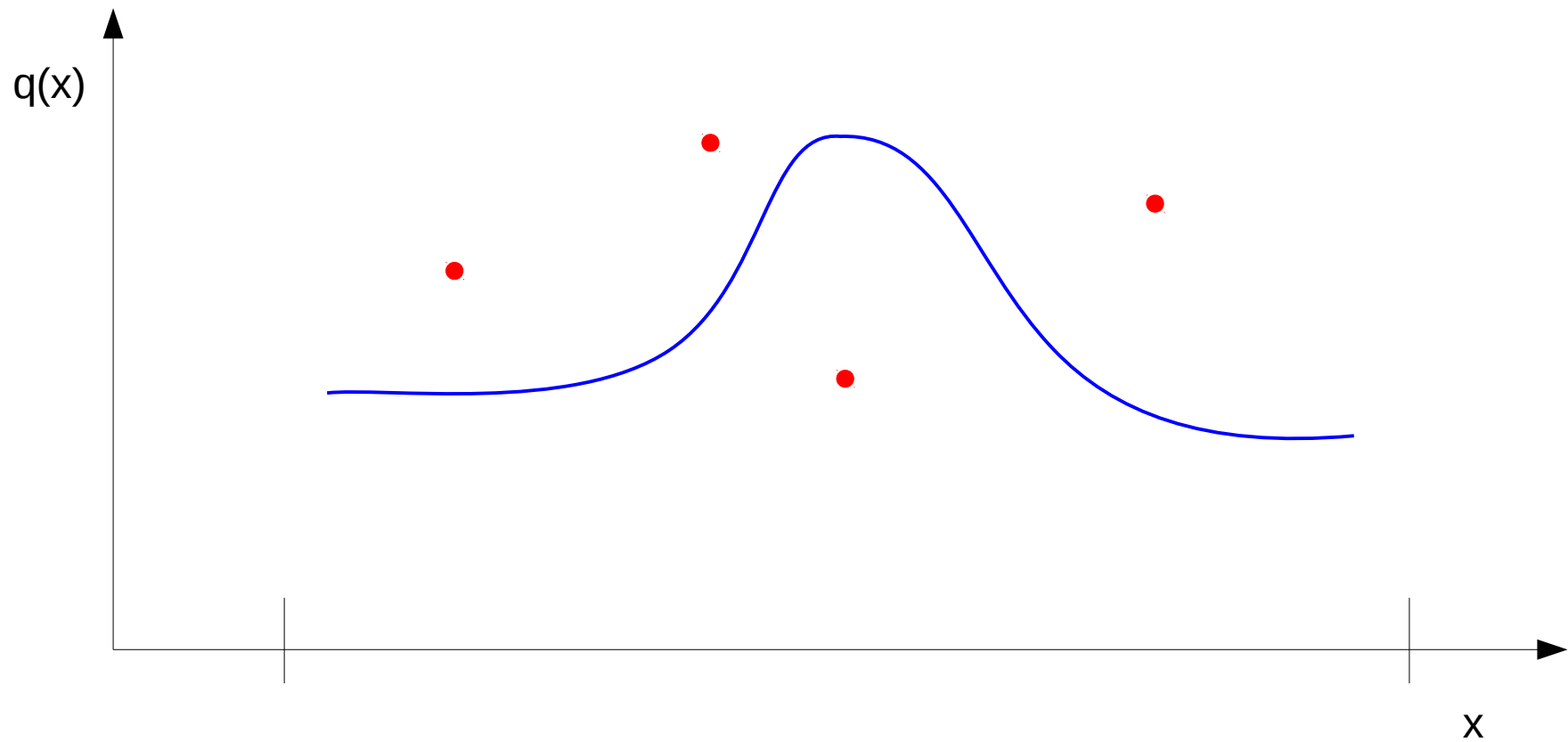
Optymalizacja jako proces adaptacji



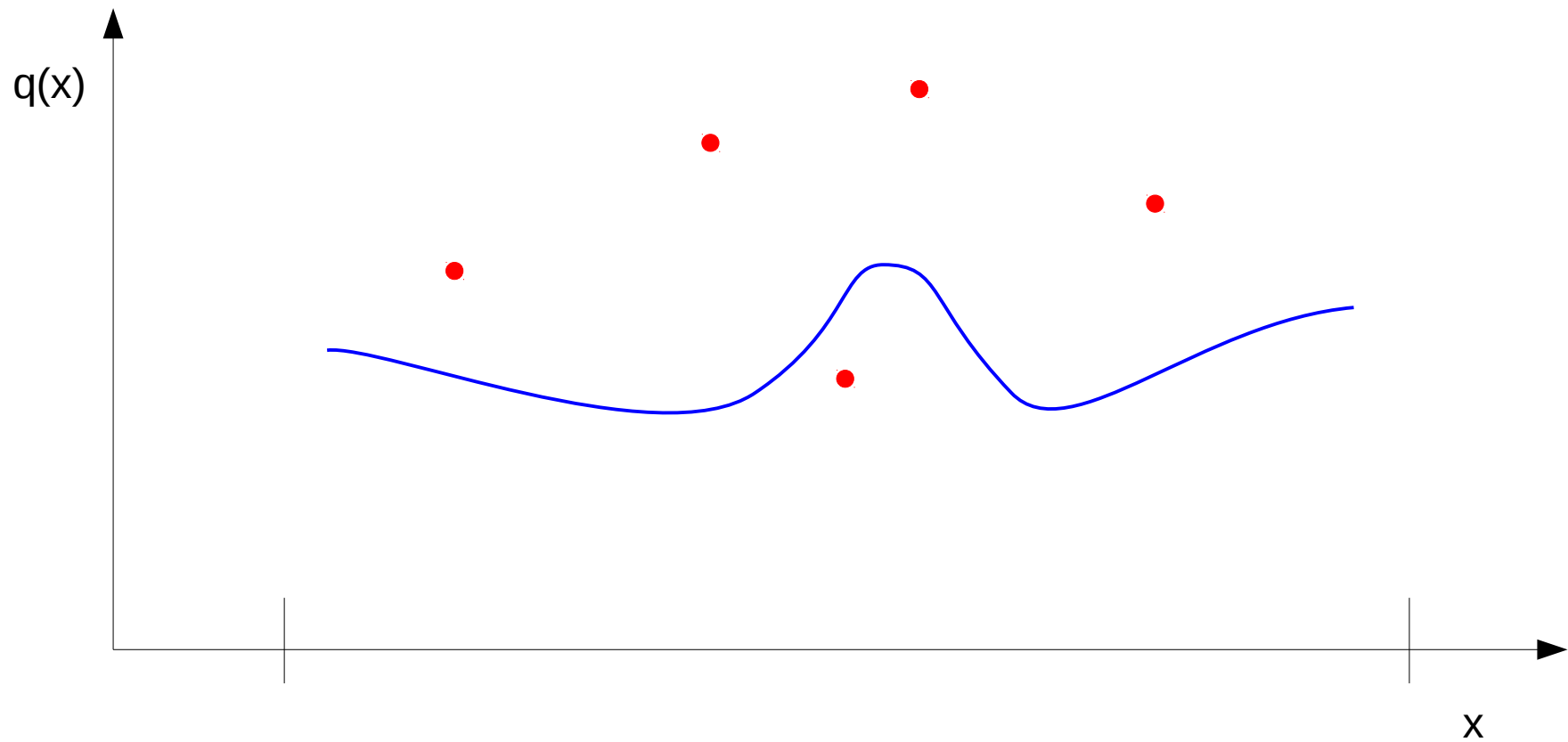
Optymalizacja jako proces adaptacji



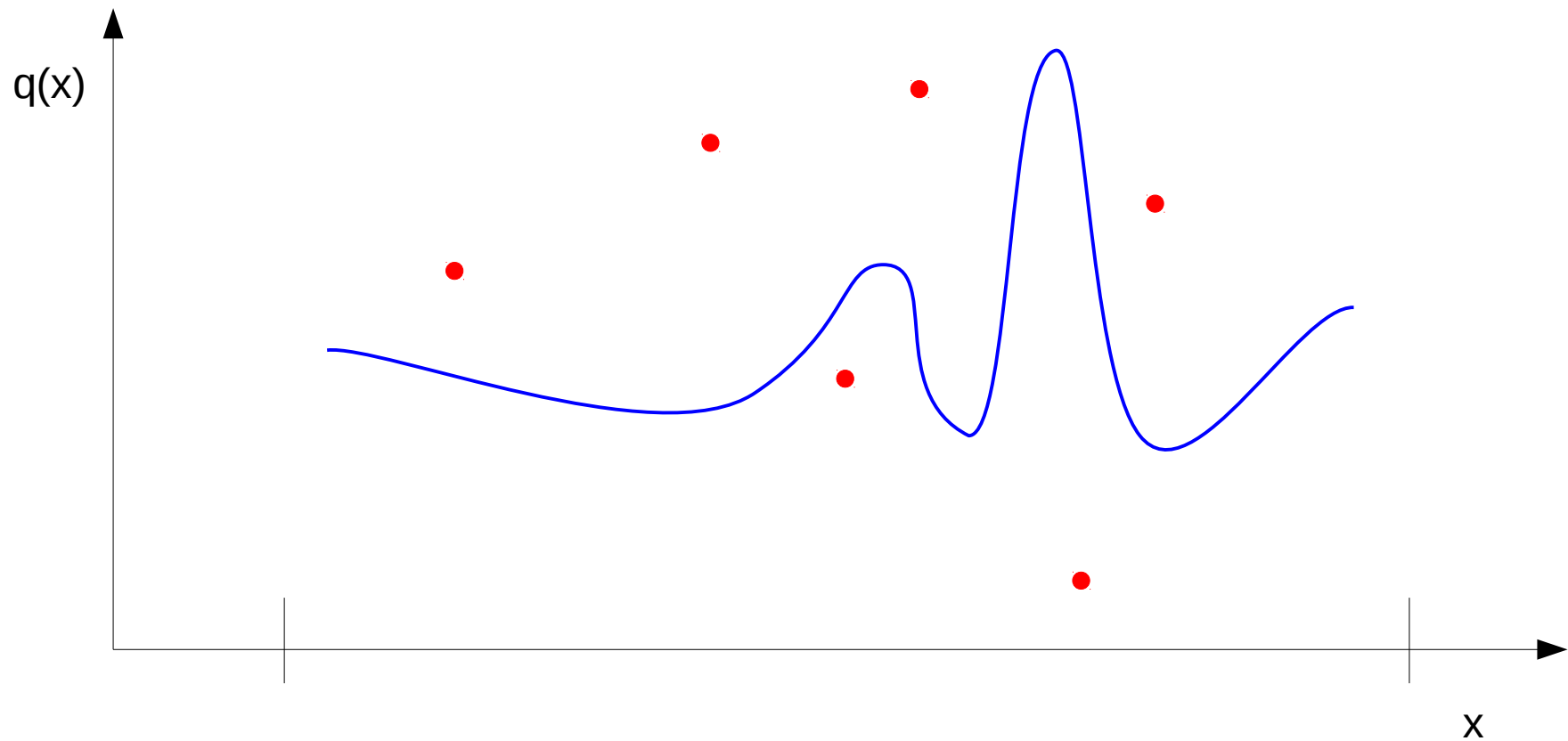
Optymalizacja jako proces adaptacji



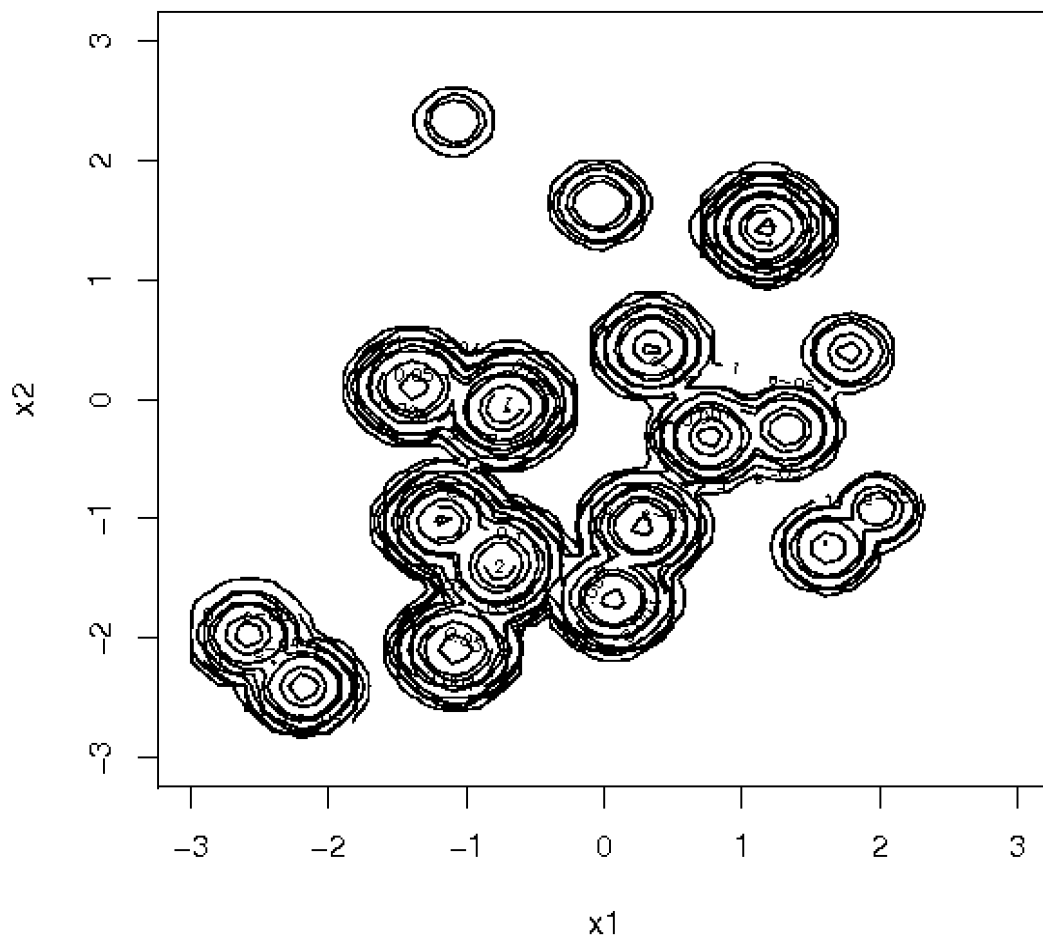
Optymalizacja jako proces adaptacji



Optymalizacja jako proces adaptacji

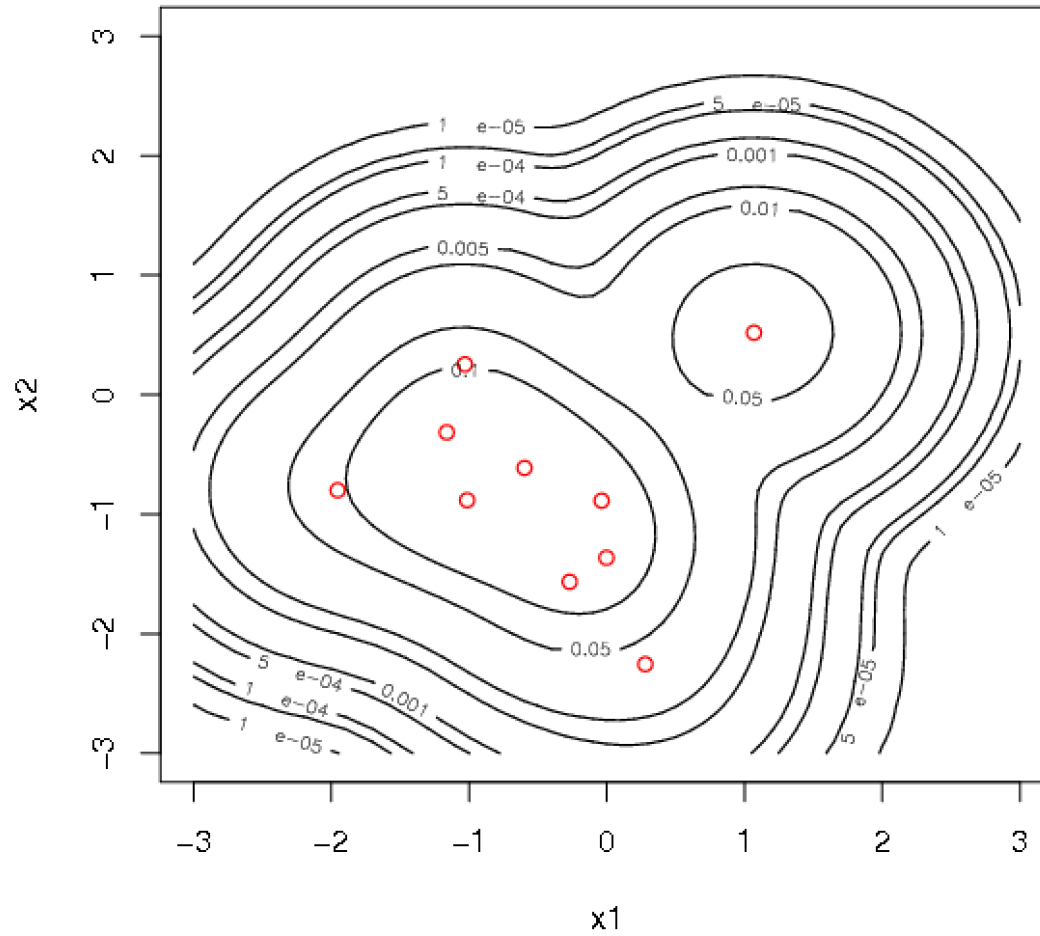


- Obserwujemy wartości funkcji celu w wielu punktach
- Okolice każdego punktu są tym chętniej próbkowane, im jest on lepszy

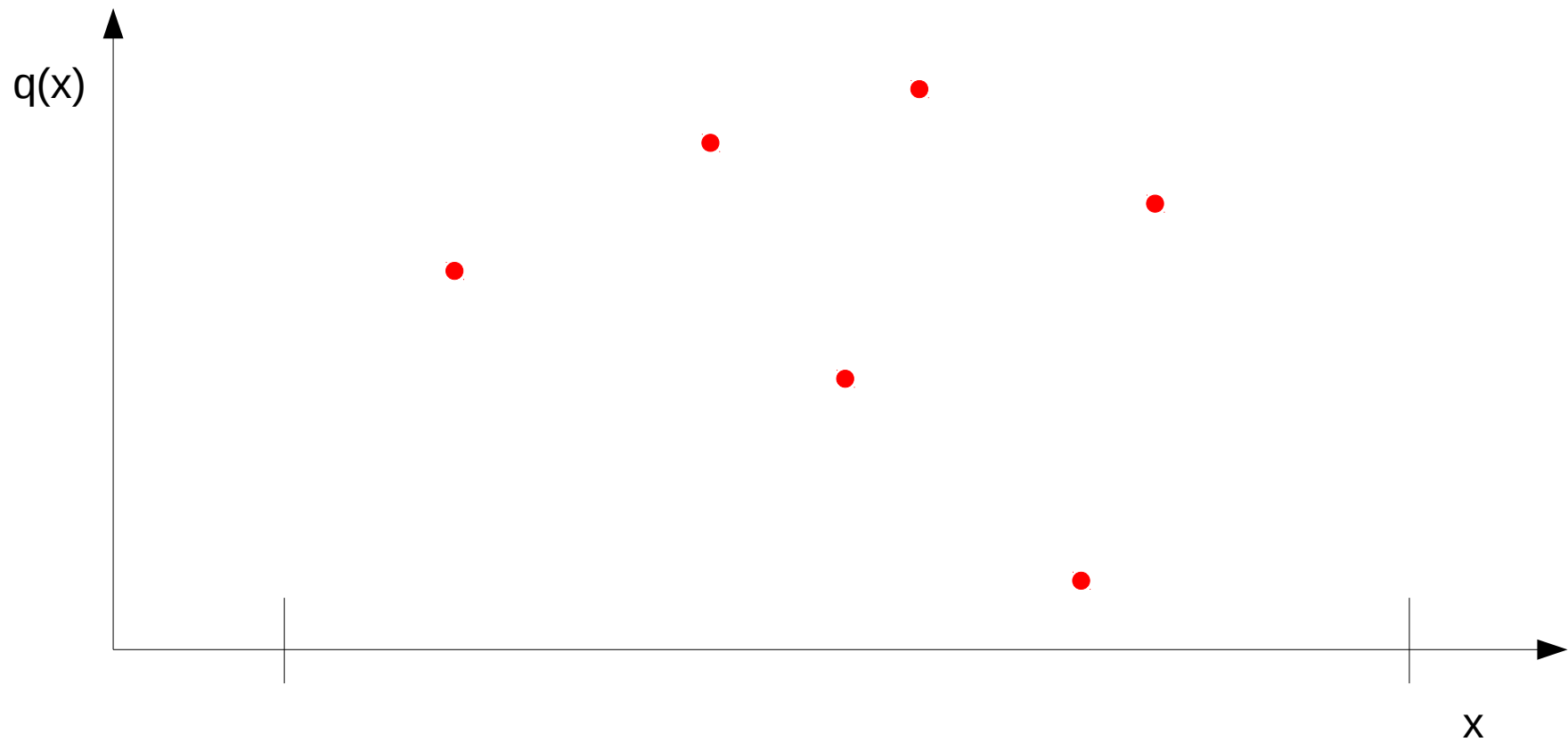


- Obserwujemy wartości funkcji celu w wielu punktach
- Okolice każdego punktu są tym chętniej próbkowane, im jest on lepszy

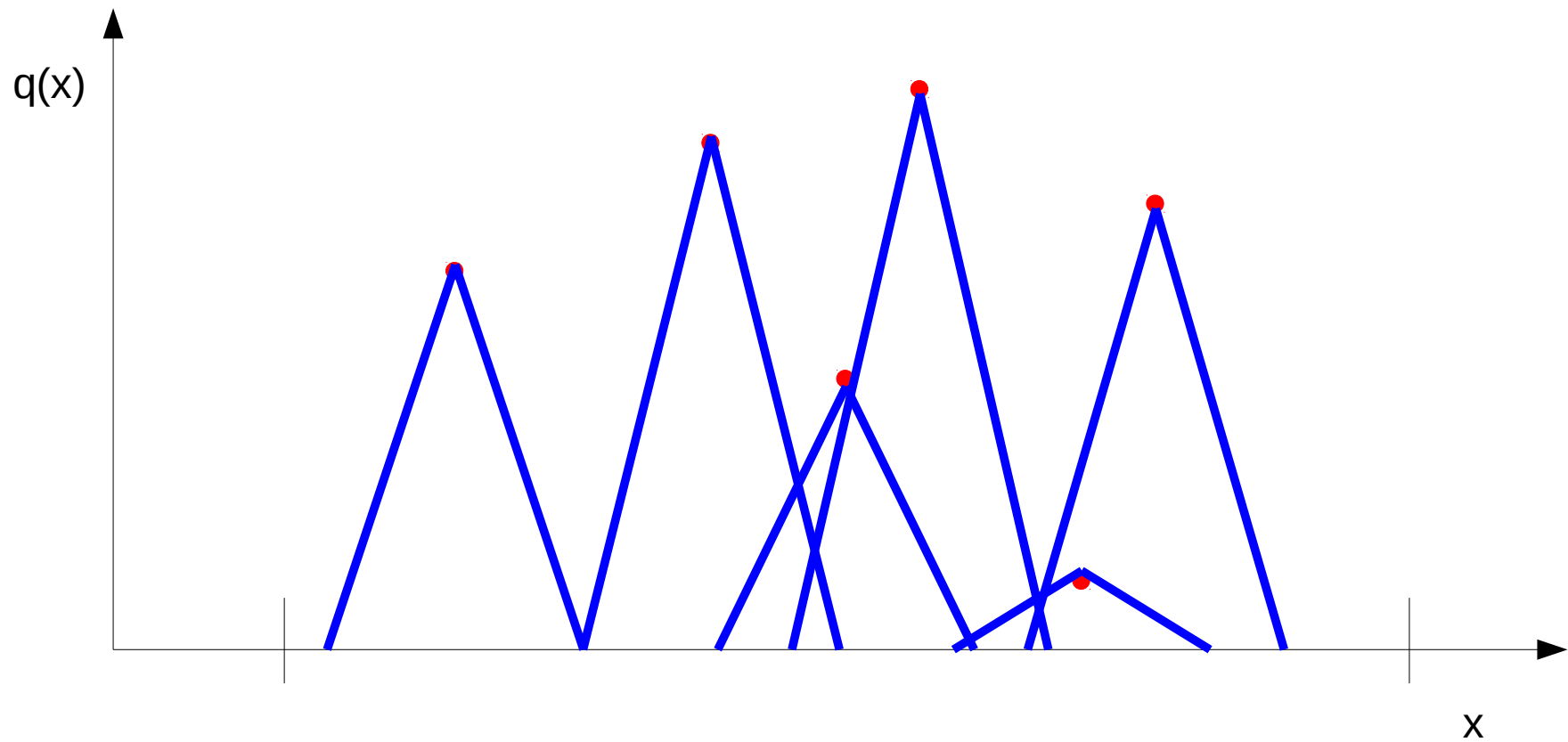
Po pewnym czasie



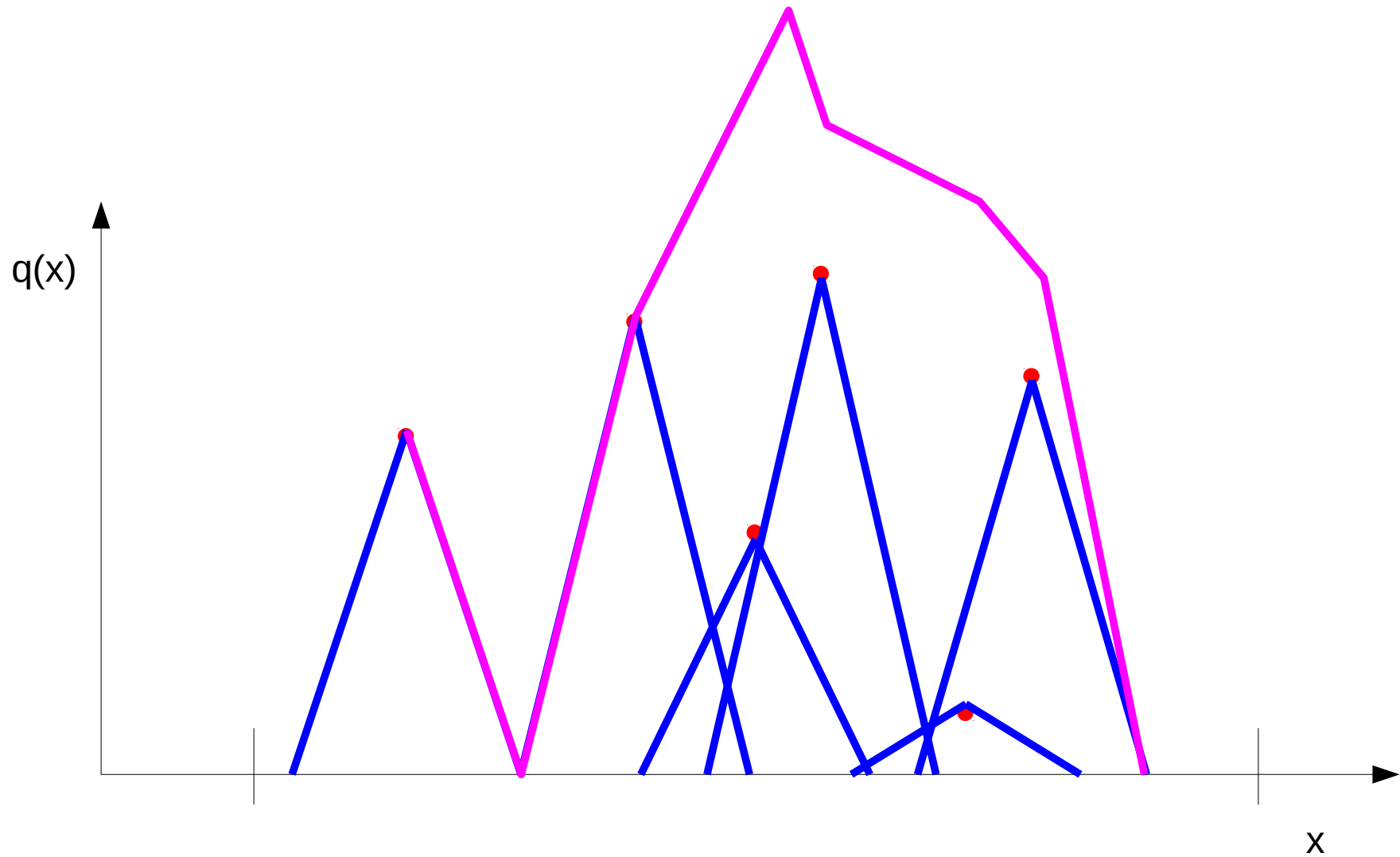
Mieszanka rozkładów p-stwa



Mieszanka rozkładów p-stwa



Mieszanka rozkładów p-stwa



Ewolucja różnicowa

algorytm differential evolution

inicjuj $P^0 \leftarrow \{P_1^0, P_2^0 \dots P_\mu^0\}$

$H \leftarrow P^0$

$t \leftarrow 0$

while ! stop

for ($i \in 1:\mu$)

$P_j^t \leftarrow \text{select}(P^t)$

$P_k^t, P_l^t \leftarrow \text{sample}(P^t)$

$M_i^t \leftarrow P_j^t + F(P_k^t - P_l^t)$

$O_i^t \leftarrow \text{crossover}(P_i^t, M_i^t)$

$H \leftarrow H \cup \{O_i^t\}$

$P_i^{t+1} \leftarrow \text{tournament}(P_i^t, O_i^t)$

$t \leftarrow t + 1$

sample jest procesem wyboru pary punktów z jednakowym p-stwem

crossover jest operacją krzyżowania wymieniającego

Typy ewolucji różnicowej - klasyka

- Typ selekcji
 - wybór losowego (rand)
 - wybór najlepszego w populacji (best)
- Typ krzyżowania
 - dwumianowe (bin)
 - wykładnicze (exp)
- Liczba par różnicowanych punktów – 1 albo 2
- Konwencja oznaczeń: DE/rand/1/bin

Typy krzyżowania

procedure binomial crossover

arguments : x , y

for ($i \in 1:n$)

if $a < c_r$

$z_i \leftarrow y_i$

else

$z_i \leftarrow x_i$

return z

procedure exponential crossover

arguments : x , y

$i \leftarrow 1$

while ($i \leq n$)

if $a < c_r$

$z_i \leftarrow y_i$

else break

while ($i \leq n$)

$z_i \leftarrow x_i$

return z

a jest zmienną losową o rozkładzie jednostajnym w $(0,1)$

c_r jest parametrem

Krzyżowanie wykładnicze a jednopunktowe

0.0759	0.062	-1.893	0.053
--------	-------	--------	-------

Rodzic 1

1	1	1	0
---	---	---	---

wagi

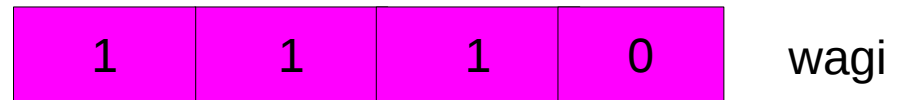
0.328	0.631	-0.299	0.194
-------	-------	--------	-------

Rodzic 2

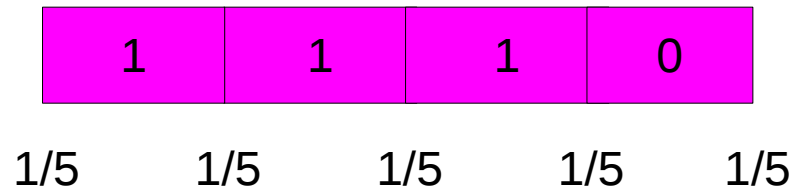
0.328	0.631	-0.299	0.053
-------	-------	--------	-------

Potomek

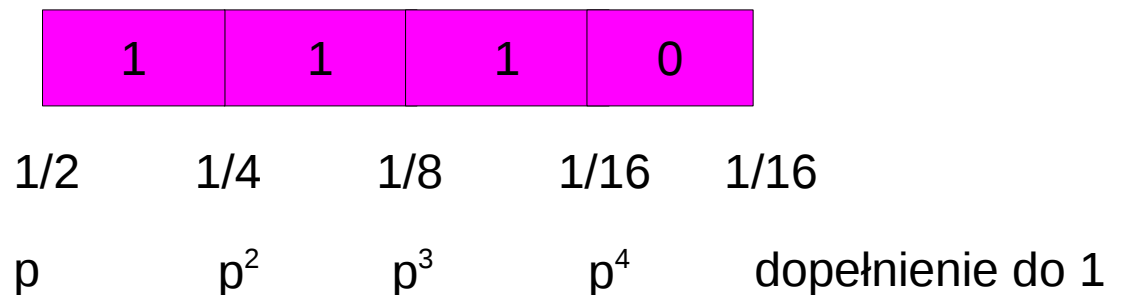
Krzyżowanie wykładnicze a jednopunktowe



W krzyżowaniu **jednopunktowym** rozkład prawdopodobieństwa pojawienia się przejścia między jedyką a zerem jest rozkładem **jednostajnym**



W krzyżowaniu wykładniczym rozkład ten jest rozkładem (prawie) wykładniczym



Krzyżowanie równomierne a dwumianowe

0.0759	0.062	-1.893	0.053
--------	-------	--------	-------

Rodzic 1

0	1	1	0
---	---	---	---

wagi

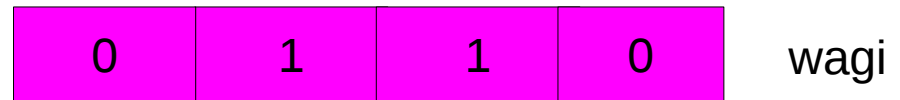
0.328	0.631	-0.299	0.194
-------	-------	--------	-------

Rodzic 2

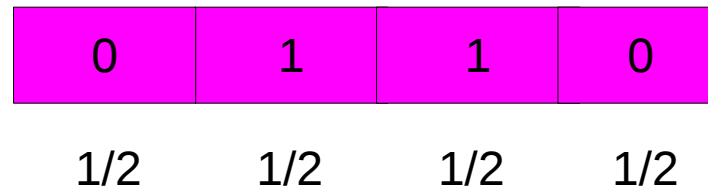
0.0759	0.631	-0.299	0.053
--------	-------	--------	-------

Potomek

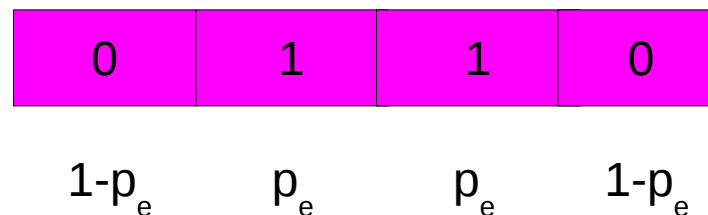
Krzyżowanie równomierne a dwumianowe



W krzyżowaniu **równomiernym** prawdopodobieństwo pojawienia się jedynek i zera na każdej pozycji jest równe $1/2$

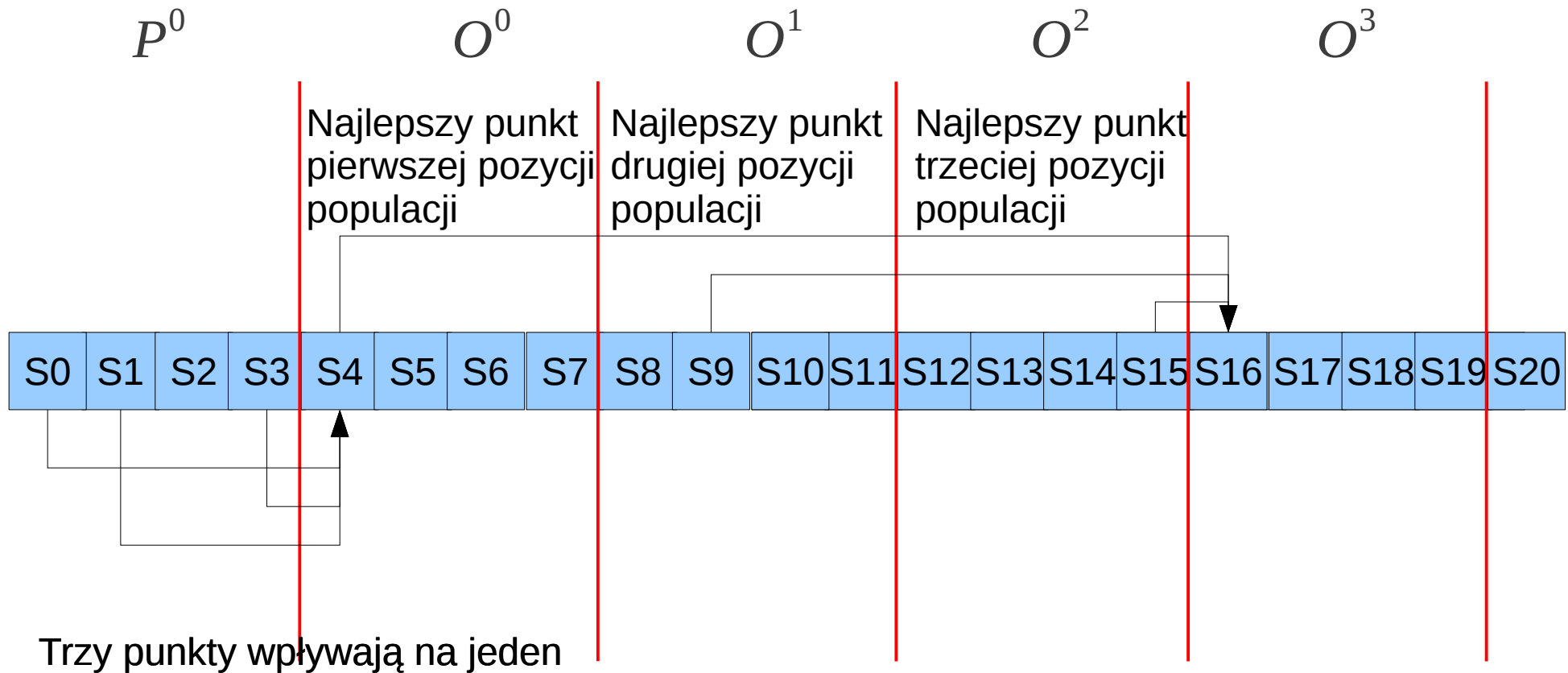


W krzyżowaniu **dwumianowym** te p-stwa są różne

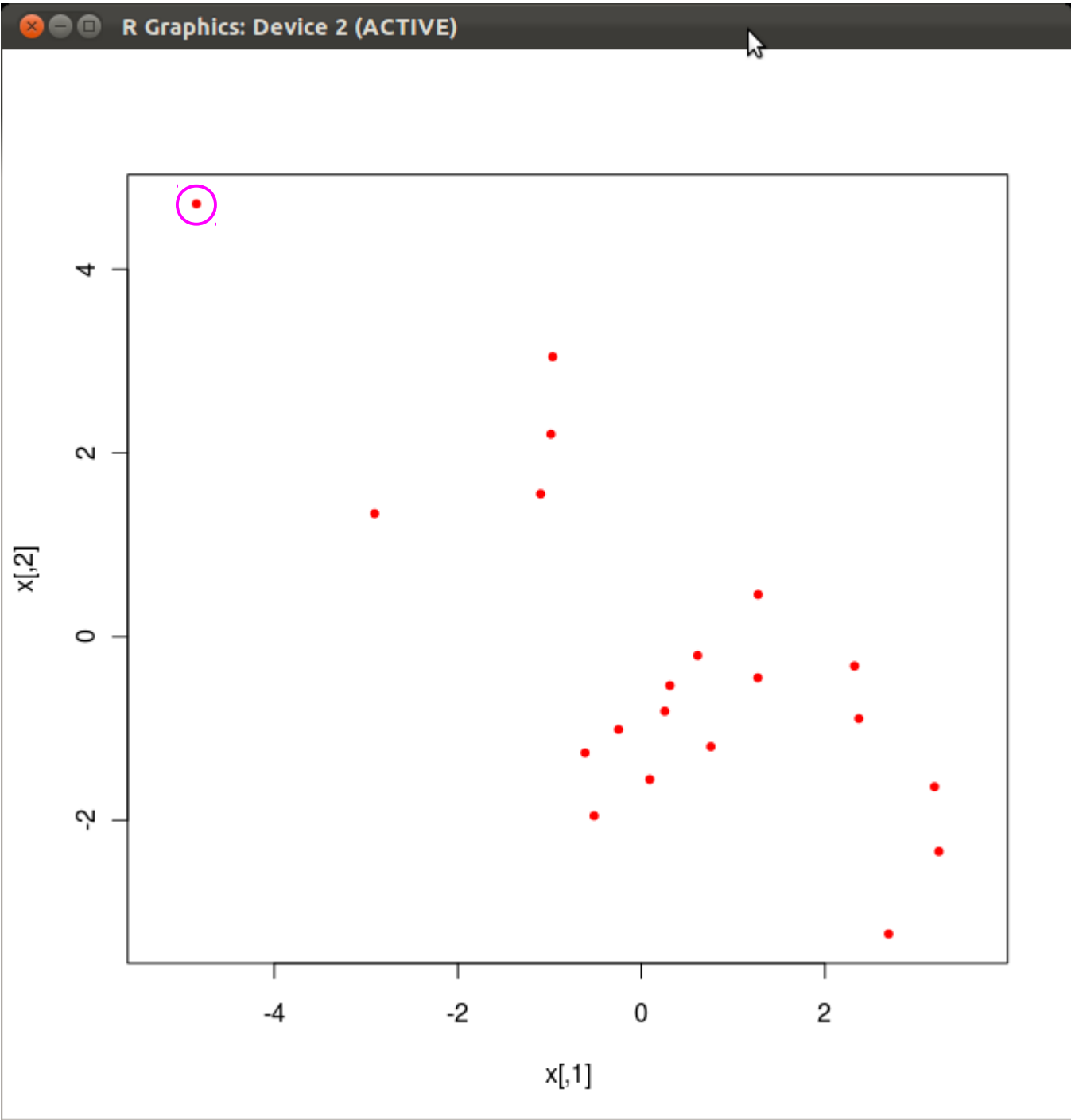


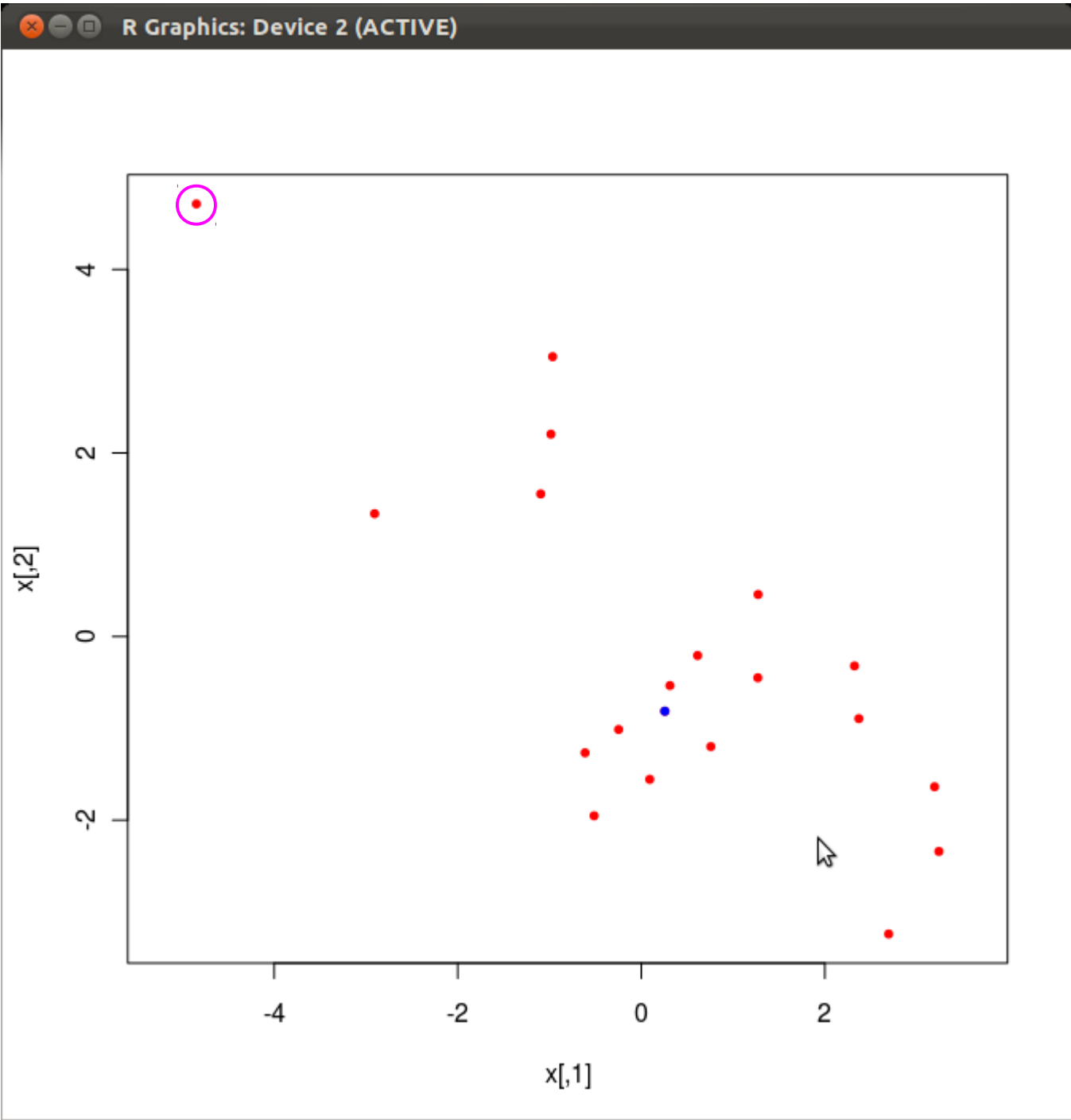
W obu przypadkach, rozkład p-stwa pojawienia się k jedynek i n-k zer jest rozkładem Bernoulliego (wg angielskiej nomenklatury dwumianowym)

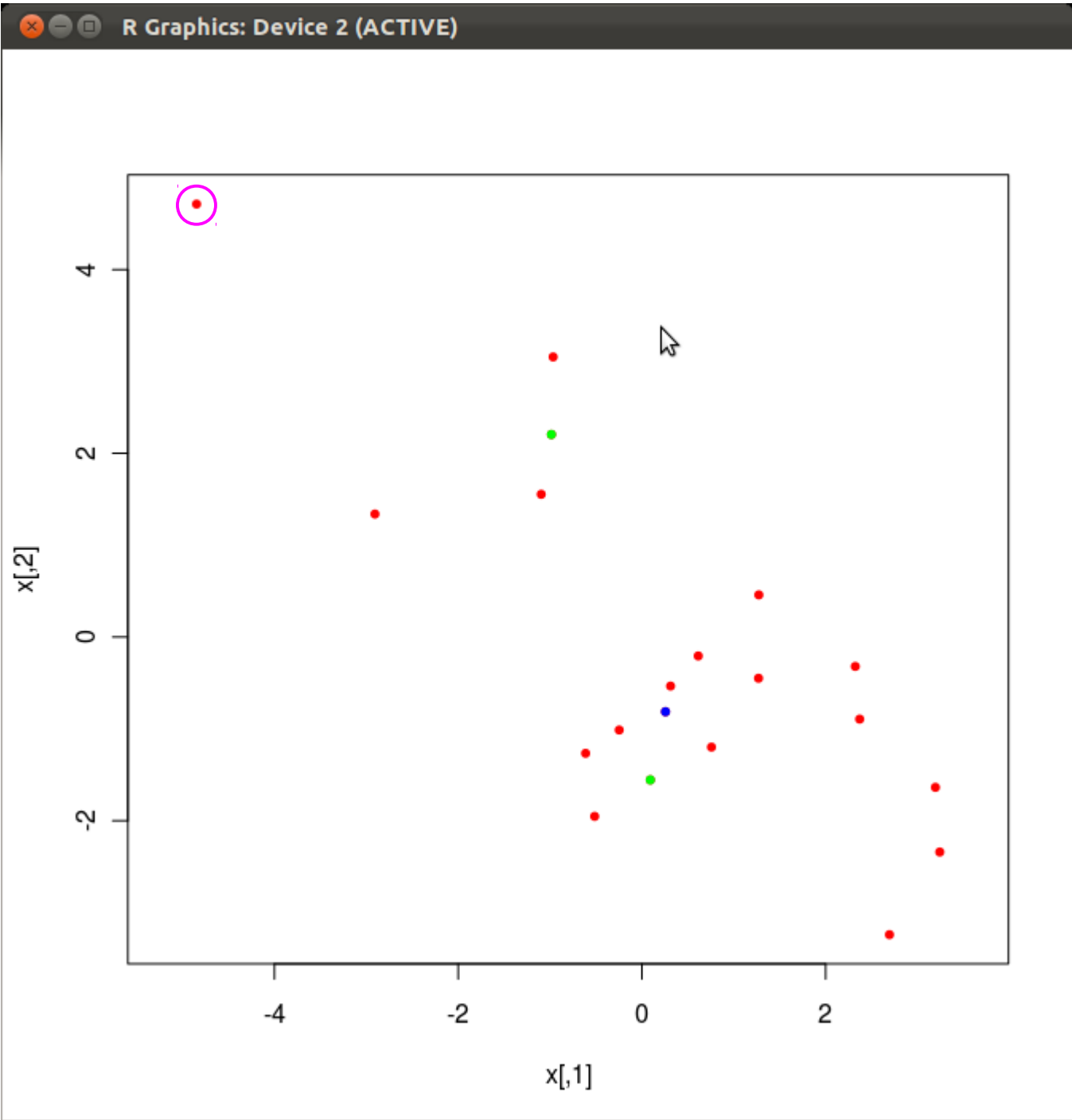
DE/rand/1

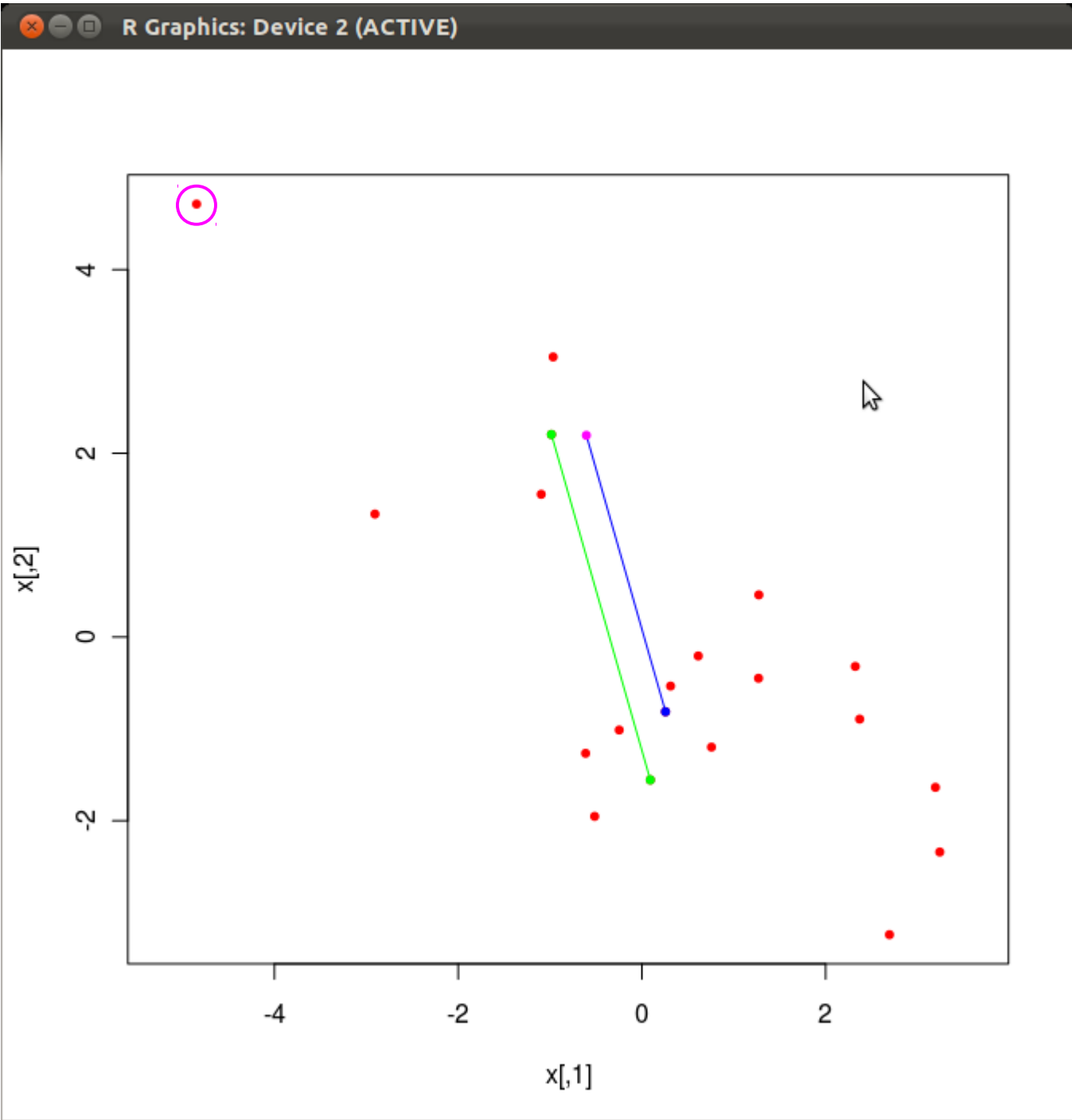


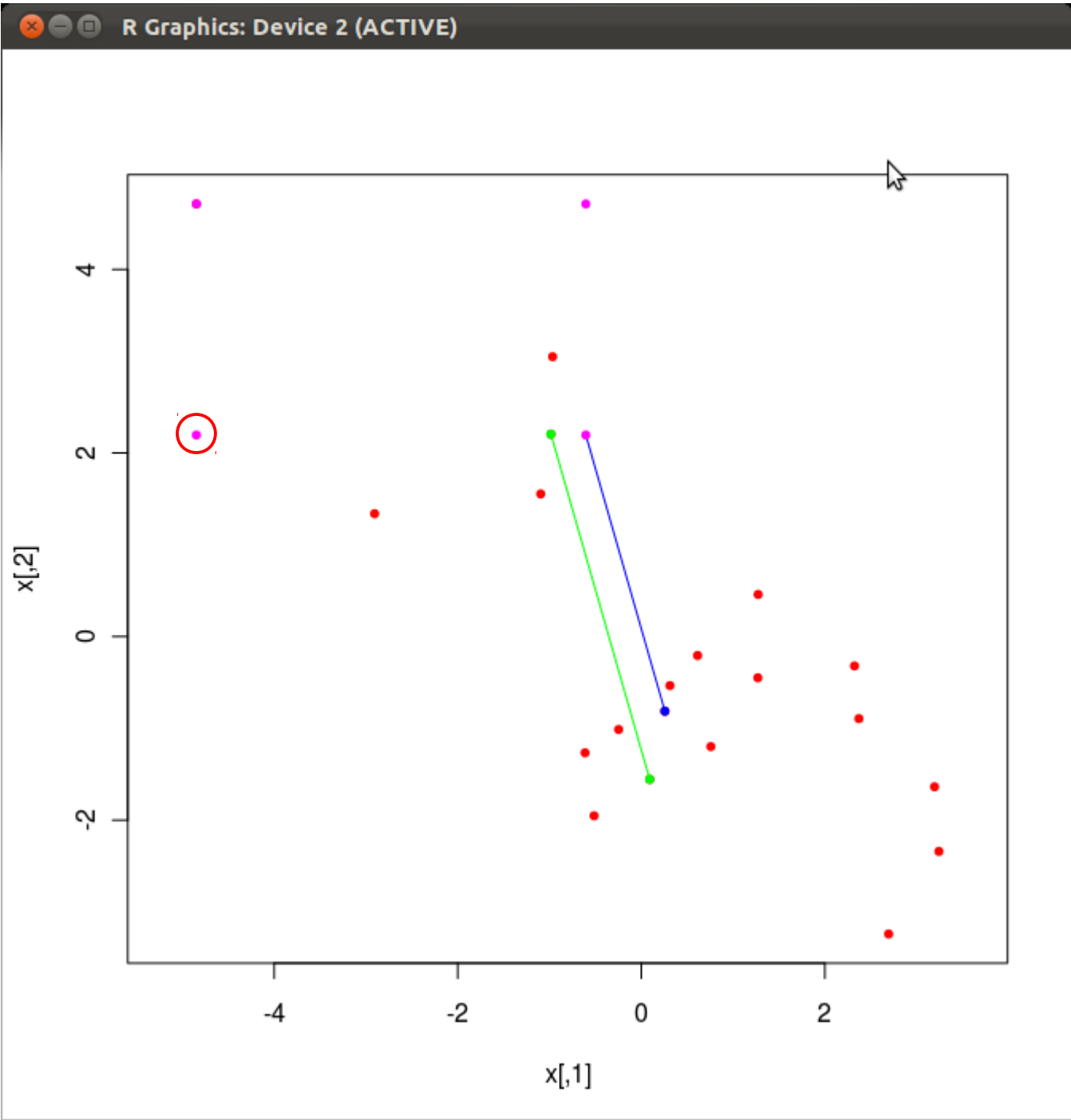
Strzałki między punktami S_x oraz S_y oznaczają, że punkt S_y jest lokalną modyfikacją punktu S_x

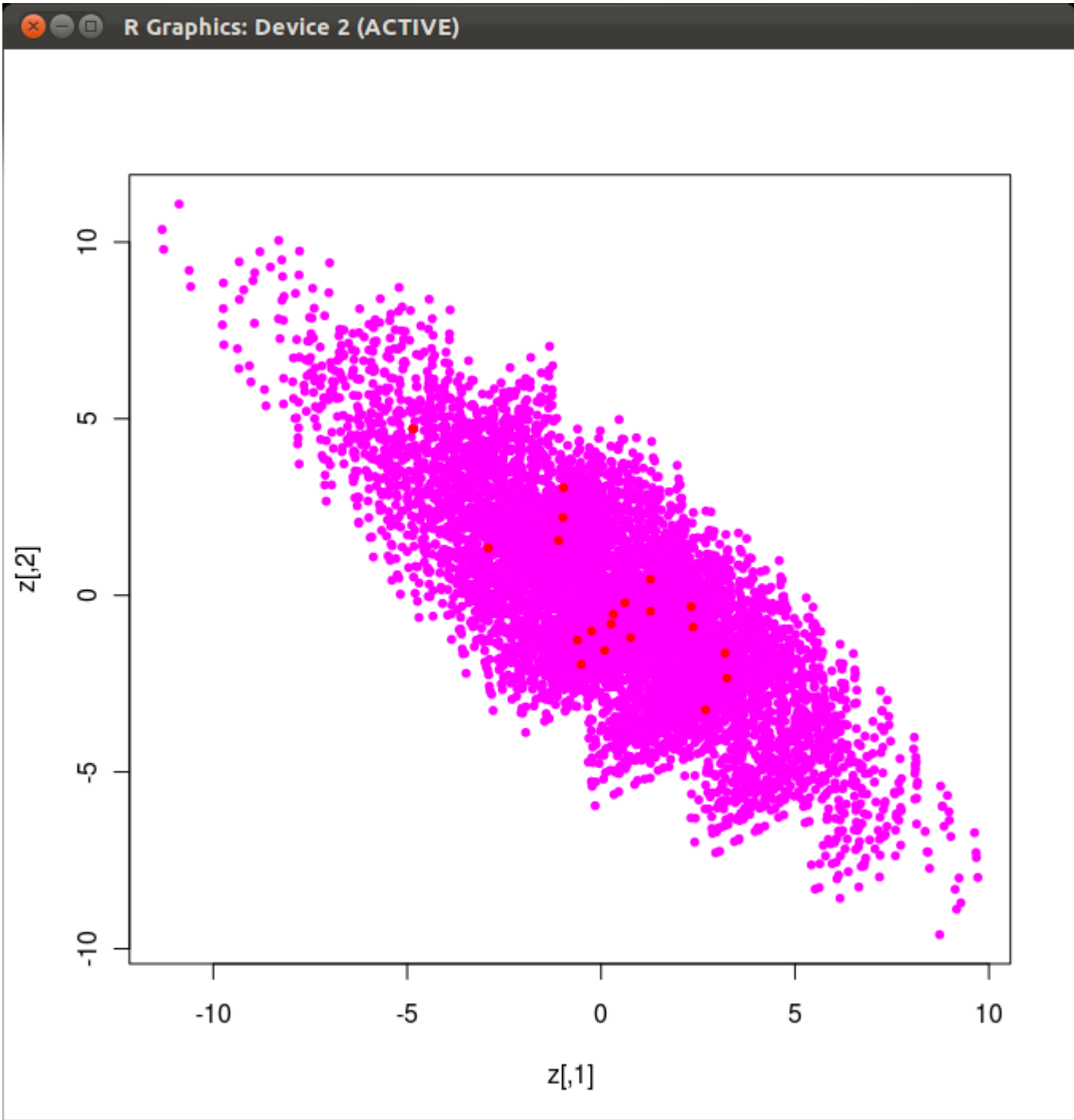


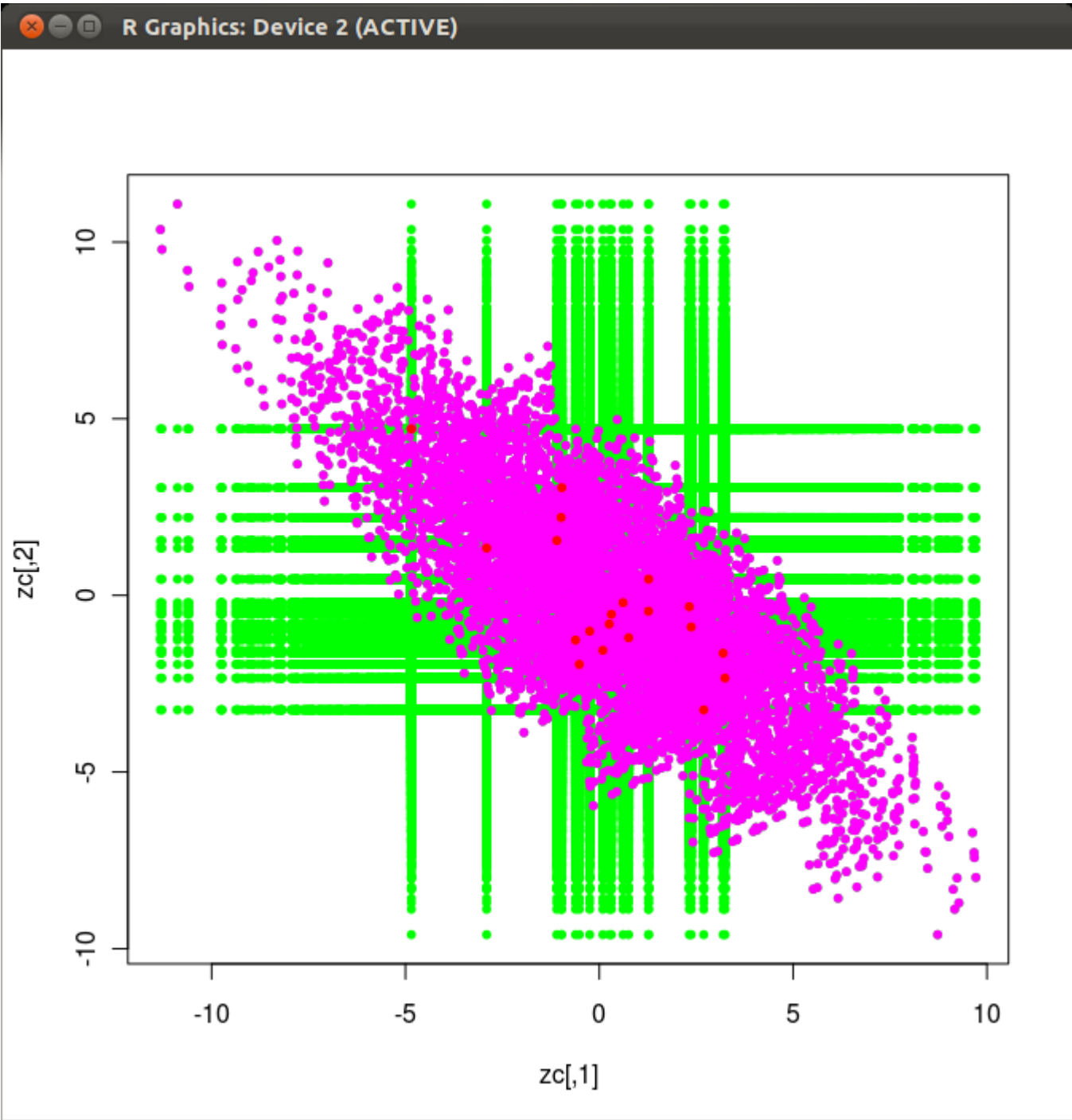












Ewolucja różnicowa

- Inne metody selekcji

- current-to-best $KP_i^t + (1 - K)P_{best}^t$

- current-to-rand $KP_i^t + (1 - K)P_j^t$

- rand-to-best $KP_{best}^t + (1 - K)P_j^t$

- Krzyżowanie uśredniające $z = KP_i^t + (1 - K)v$

- DE/either-or

$$z = \begin{cases} P_i^t + F(P_j^t - P_k^t) & \text{z } p\text{-stwem } p_F \\ KP_i^t + (1 - K)(P_j^t + P_k^t) & \text{z } p\text{-stwem } 1 - p_F \end{cases}$$

Ewolucja różnicowa

- Poinformowanie T
- Determinizm N
- Typ stanu pamięciowy
- Wielkość modelu licznosc populacji
- Lokalność generacji T
- Miękkosc selekcji N
- Okno historii nieskończone
- Zupelnosc N

Algorytm ewolucyjny wypukła funkcja celu

- Model populacji nieskończonej
- Dystrybuanta empiryczna punktów populacji (skokowa) → dystrybuanta rozkładu próbkowania (ciągła)

DE/rand/1

wypukła funkcja celu

- Wariancja punktów po selekcji v_P
- Wariancja punktów po mutacji

$$v_O = v_P + F^2(v_P + v_P) = v_P(1 + 2F^2)$$

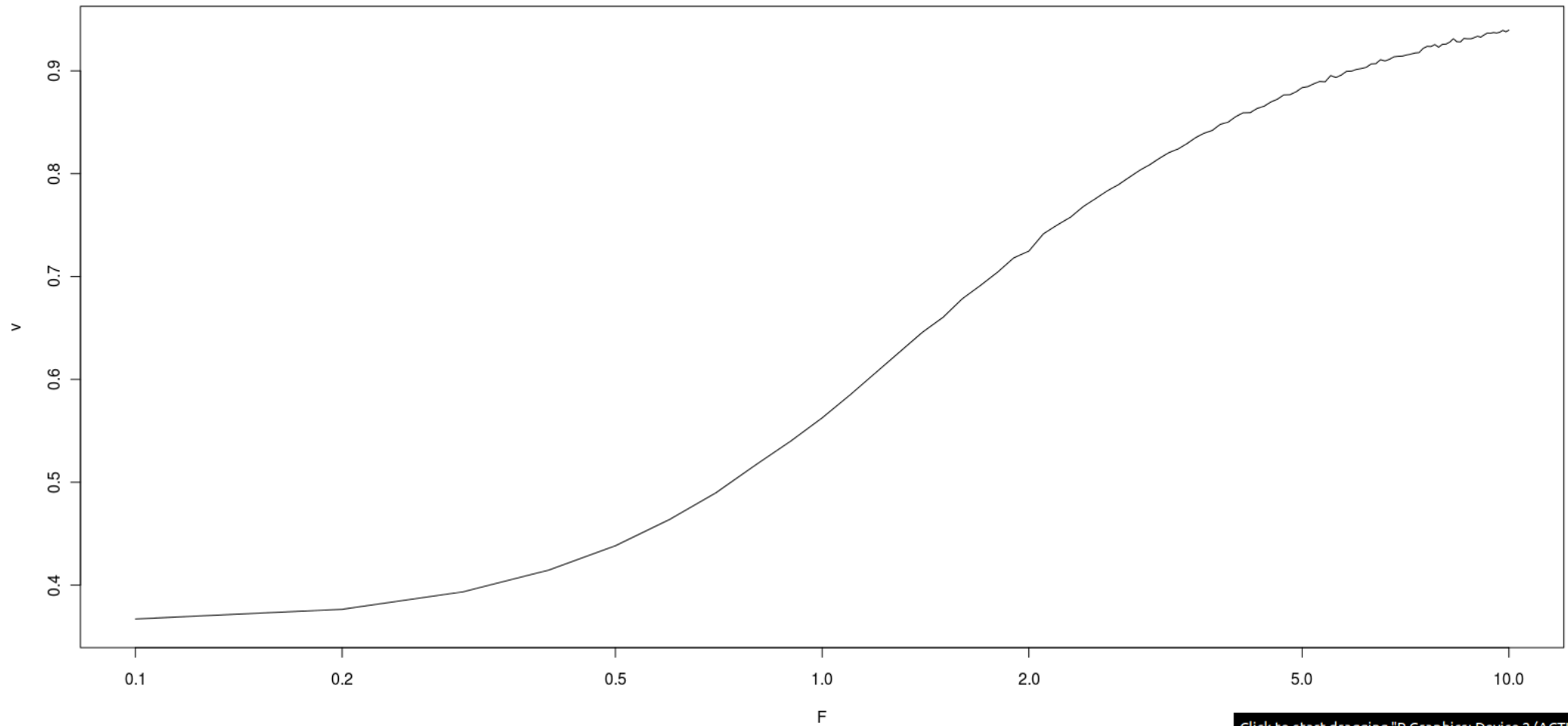
- Krzyżowanie zmienia wariancję (wzór dla bin)

$$v_C = (1 + 2F^2(1 - c_r))v_P$$

-

DE/rand/1

wypukła funkcja celu -
wariancja po sukcesji



DE/rand/1

wypukła funkcja celu

- Wariancja punktów po sukcesji

$$v_P(t+1) = k(F)v_P(t) \quad 0 < k < 1$$

- Równowagowa wariancja populacji:

$$v_P(\infty) = 0$$

- A dla alg. ewolucyjnego
(np. selekcja turniejowa, $s=2$, $pc=0$)

$$v_P(\infty) = \frac{\pi}{2} v_m$$

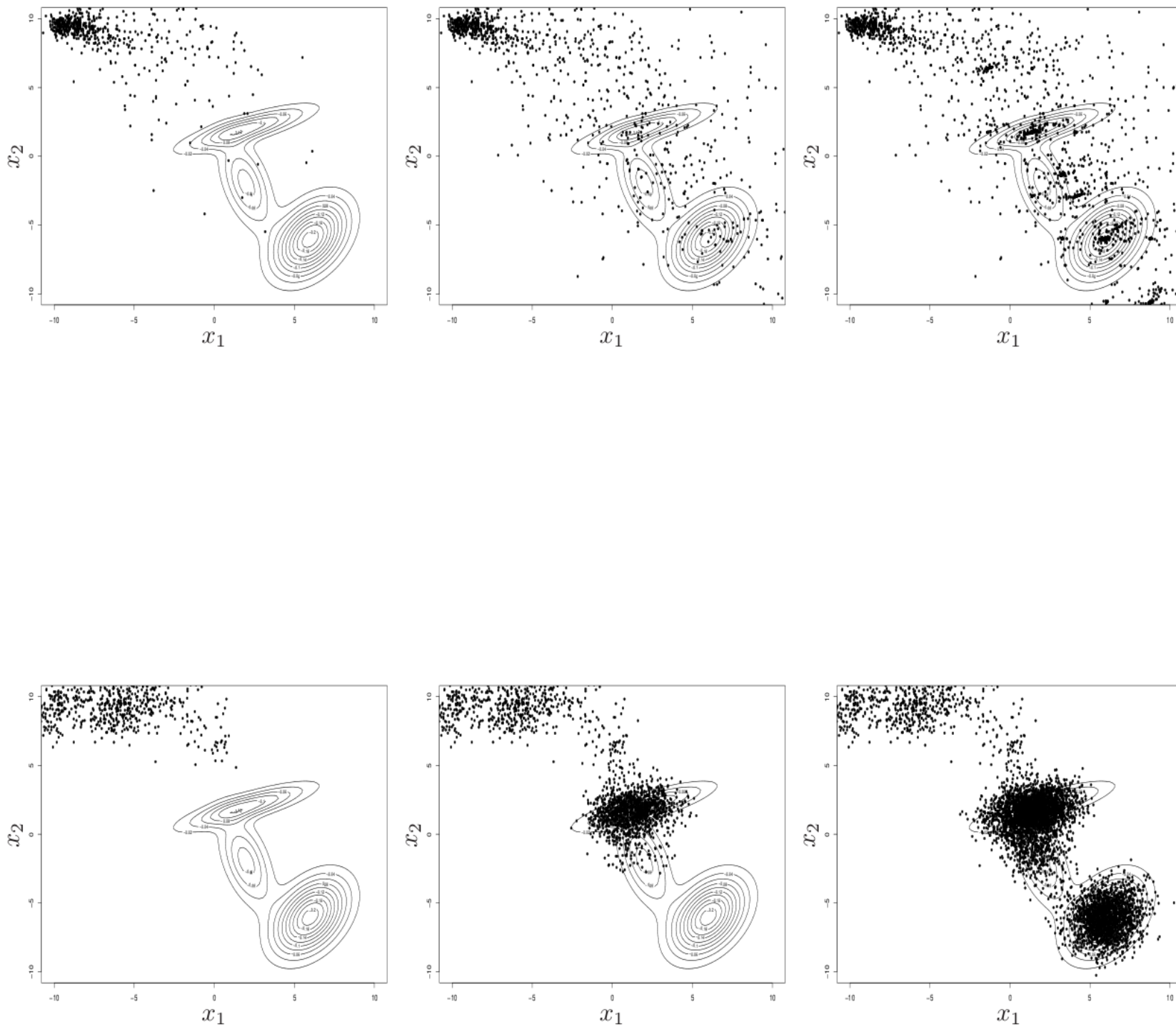
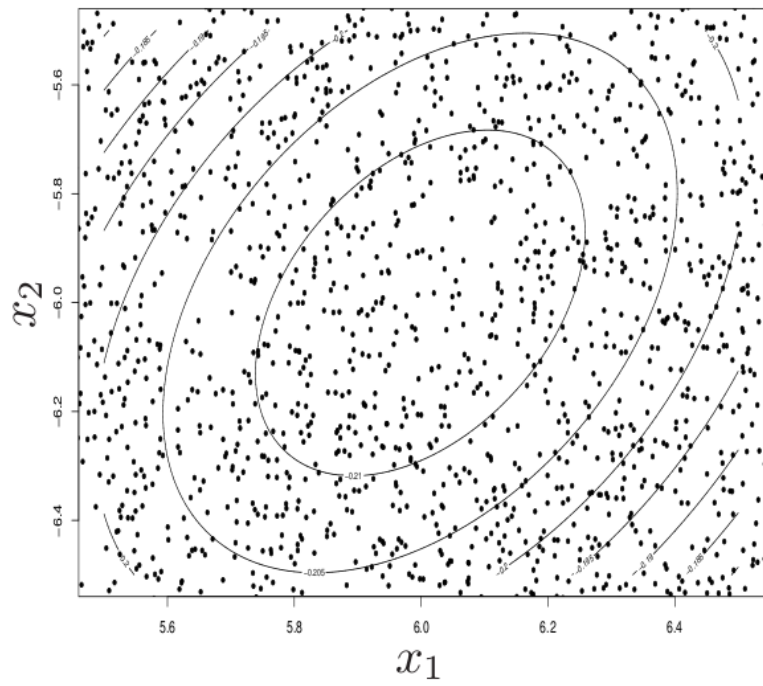
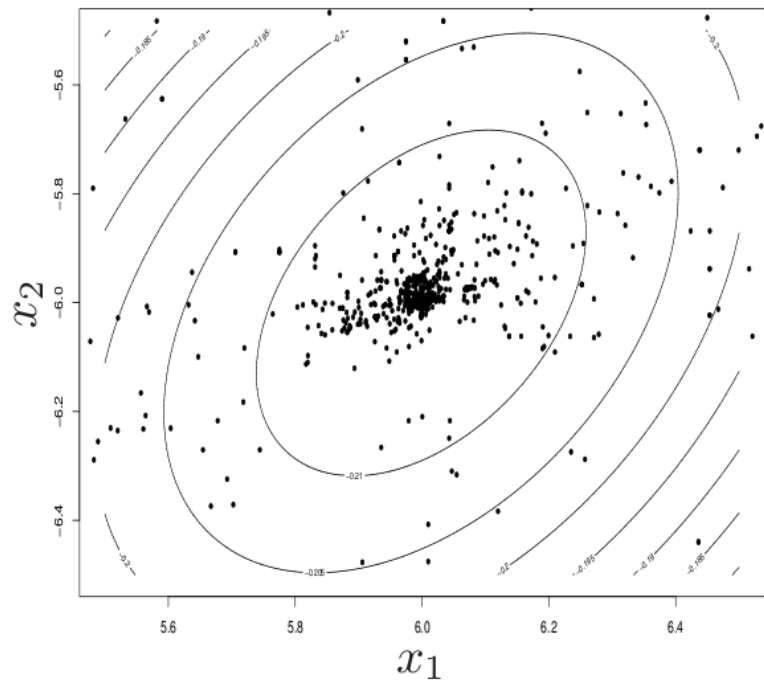


Fig. 6 Set of chromosomes generated by the GMEA algorithm with $\mu = 16$ points in population and mutation standard deviation equal to 0.6 : consecutive



Algorytm ewolucyjny

- Algorytm ewolucyjny jest techniką **adaptacji rozkładu populacji**
- Celem jest maksymalizacja wartości oczekiwanej jakości generowanych punktów
- Środek populacji – najlepszy estymator ekstremum lokalnego dla funkcji symetrycznej
- *Survival of the fittest vs. survival of the flattest*

Metoda EDA

Estimation of Distribution Algorithm

algorithm EDA

initialize($\mathbf{m}^0, \mathbf{C}^0$)

$H \leftarrow \emptyset$

$t \leftarrow 0$

while ! stop

$P^t \leftarrow \text{sample } N(\mathbf{m}^t, \mathbf{C}^t)$

$H \leftarrow H \cup P^t$

$O^t \leftarrow \text{select}(P^t)$

$(\mathbf{m}^{t+1}, \mathbf{C}^{t+1}) \leftarrow \text{update}(O^t, \mathbf{m}^t, \mathbf{C}^t)$

$t \leftarrow t + 1$

Wariant z rozkładem normalnym

Metoda EDA

Estimation of Distribution Algorithm

- UMDA (Univariate Marginal Distribution)
- Wartość oczekiwana i wariancja estymowana z próby jako

$$m(t+1)_j \leftarrow \sum_{i=1}^{\mu} w(i) P_{ij}^t$$

$$C(t+1)_{jj} \leftarrow \sum_{i=1}^{\mu} w(i) (P_{ij}^t - m(t+1)_j)^2$$

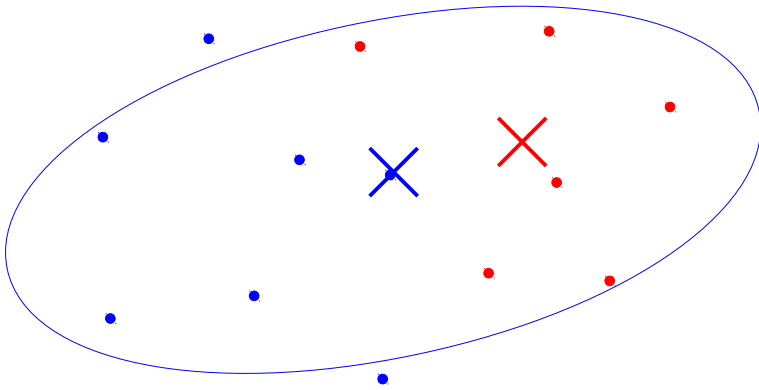
$$C(t+1)_{ij} = 0 \quad i \neq j$$

$$w(i) = \frac{q(P_i^t)}{\sum q(P_i^t)}$$

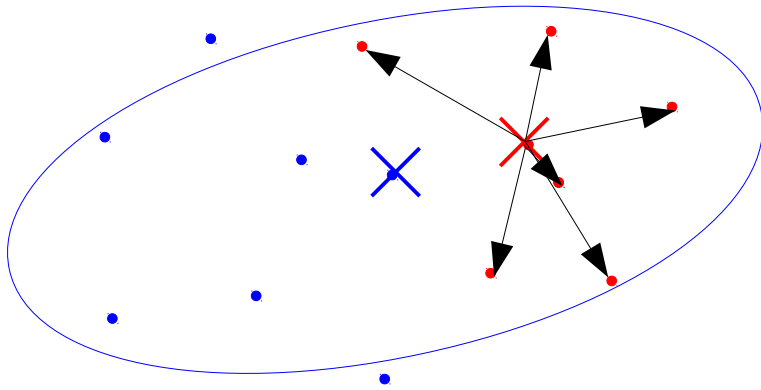
Metoda EDA

- Poinformowanie T
- Determinizm N
- Typ stanu zagregowany
- Wielkość modelu 1
- Lokalność generacji N
- Miętkość selekcji raczej N (selekcja?)
- Pamiętliwość tak, horyzont zależny od metody update
- Zupełność asymptotyczna

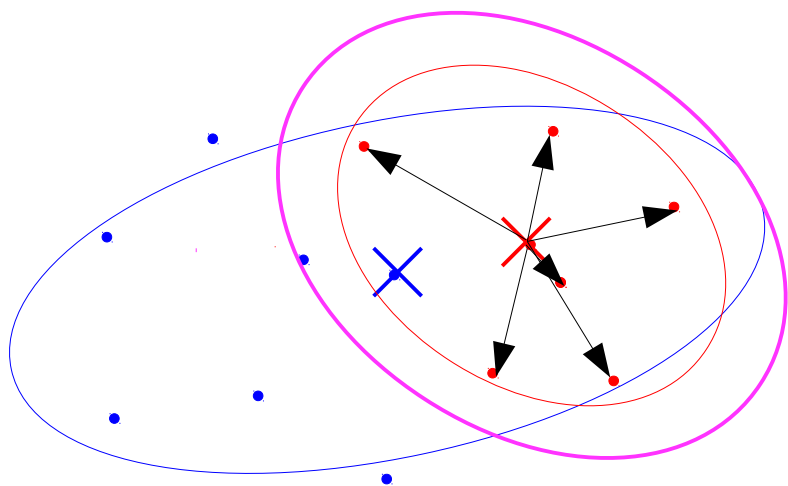
EDA



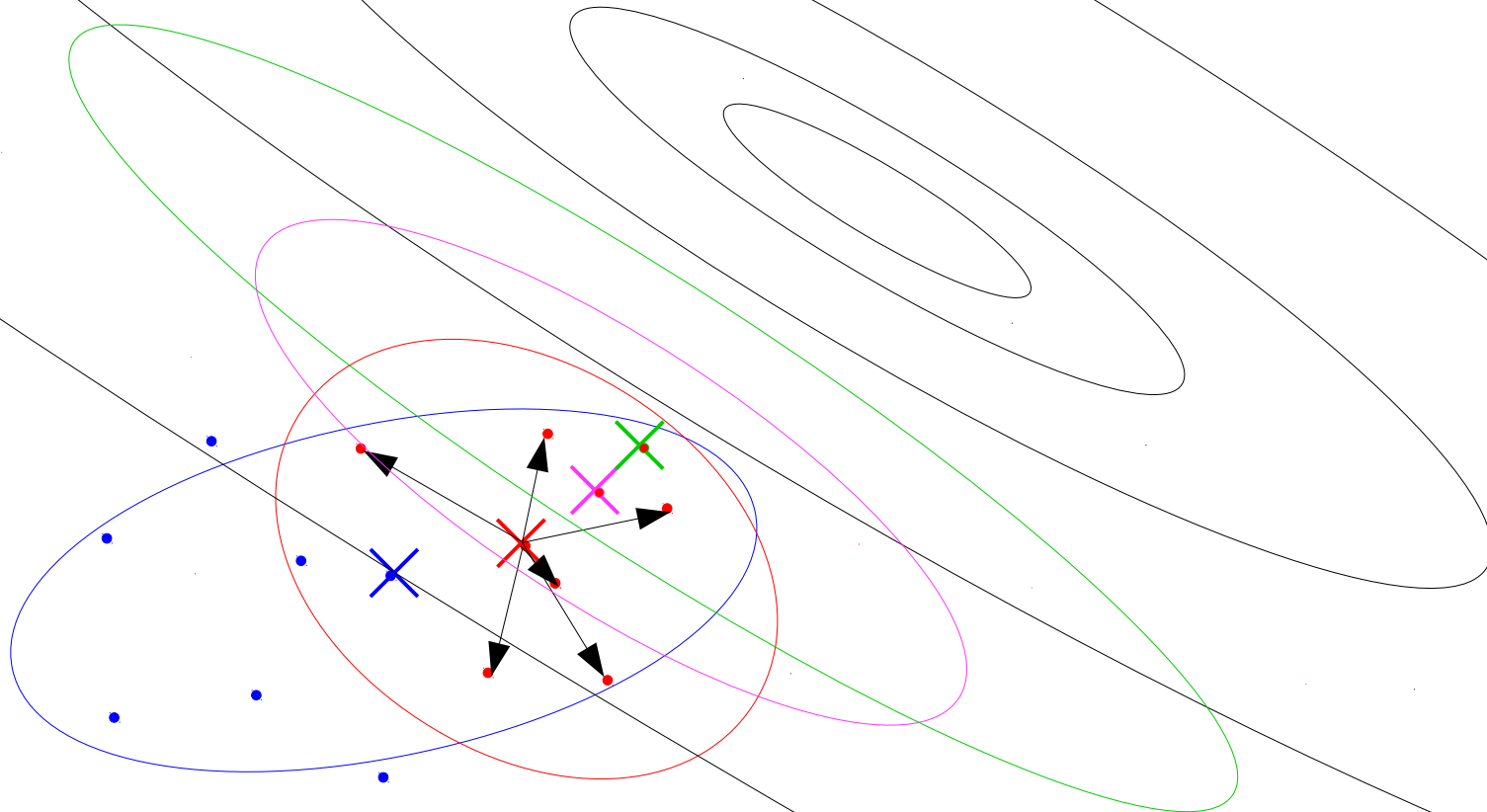
EDA



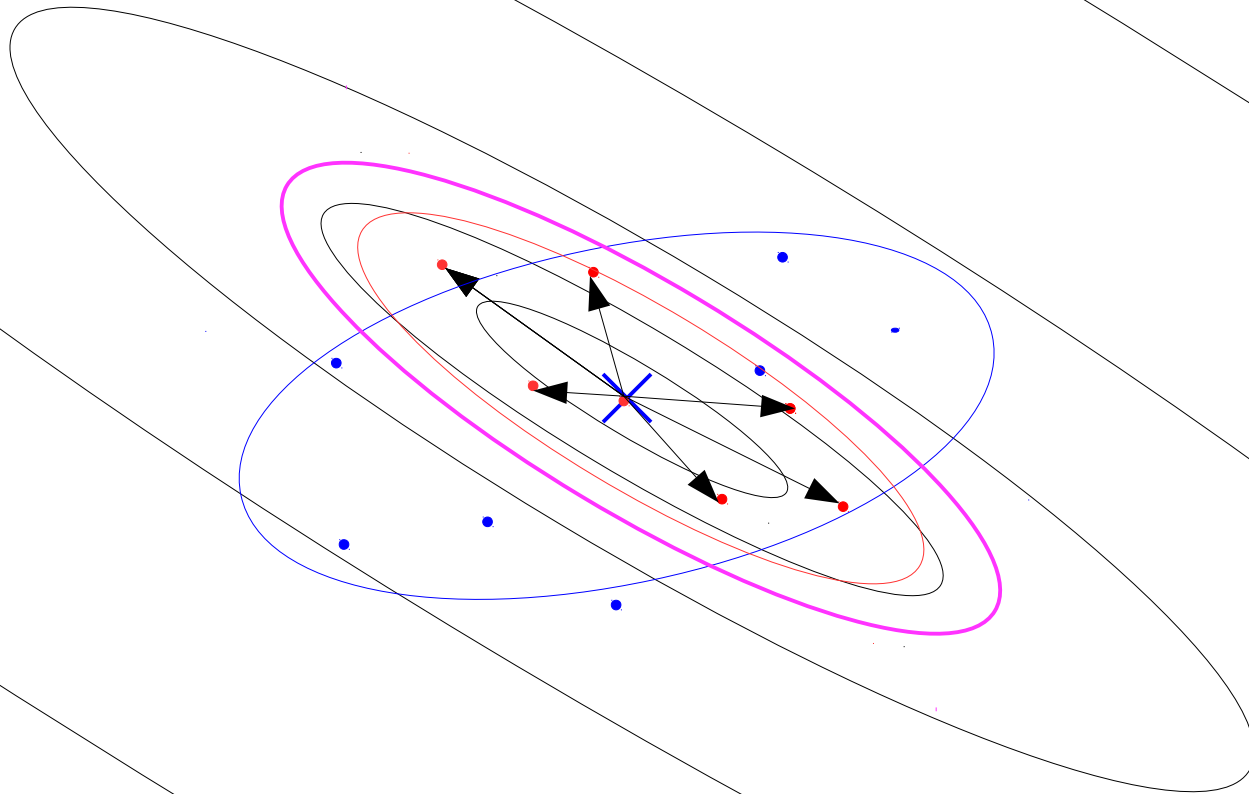
EDA



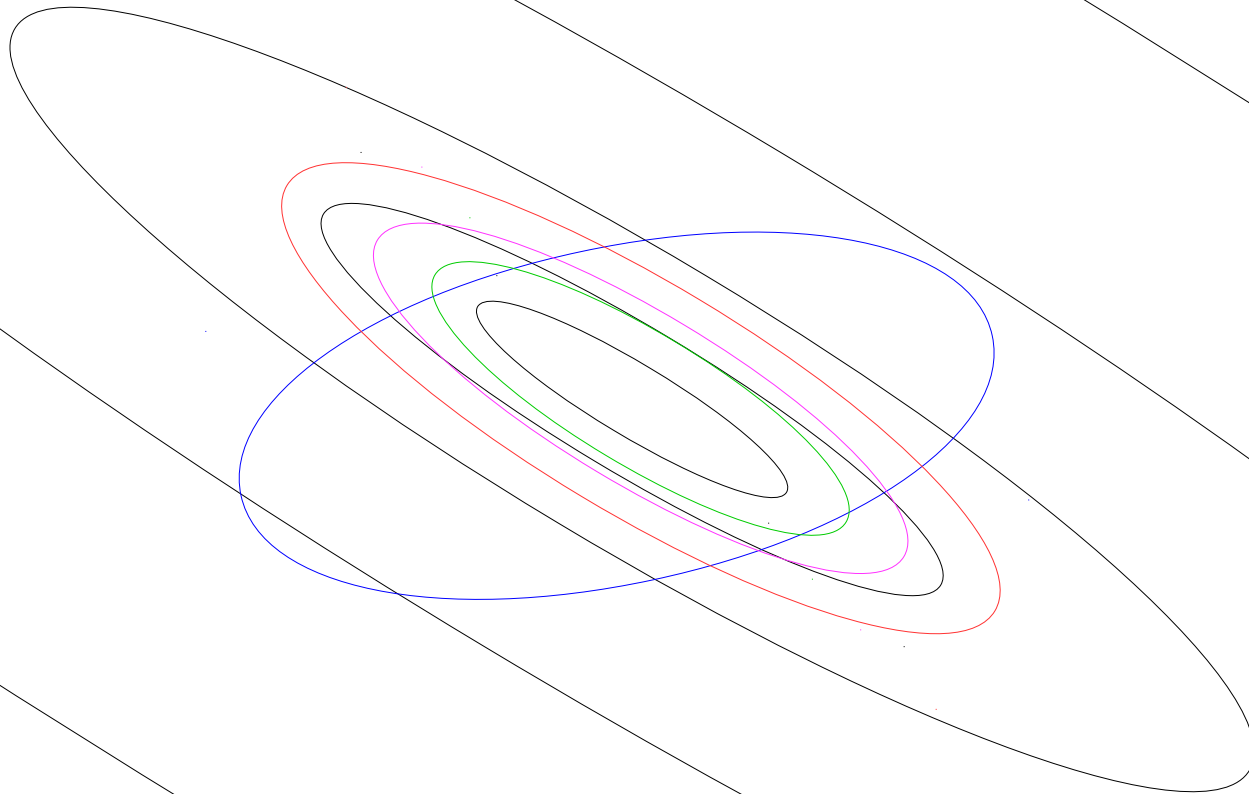
EDA na zboczu



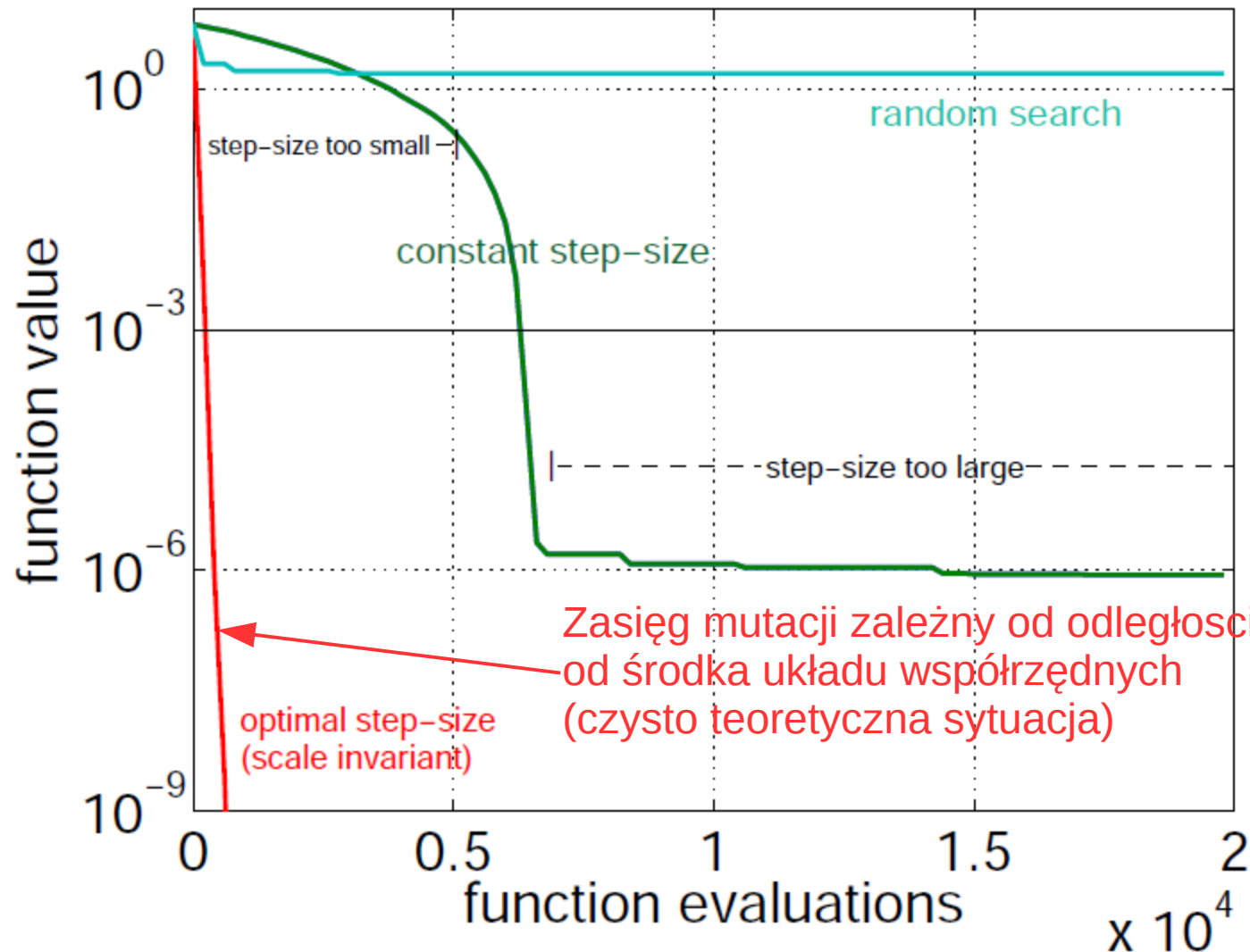
EDA na wzgórzu



EDA na wzgórzu



Zależność tempa zbieżności od zasięgu mutacji (alg. wspinaczkowy)



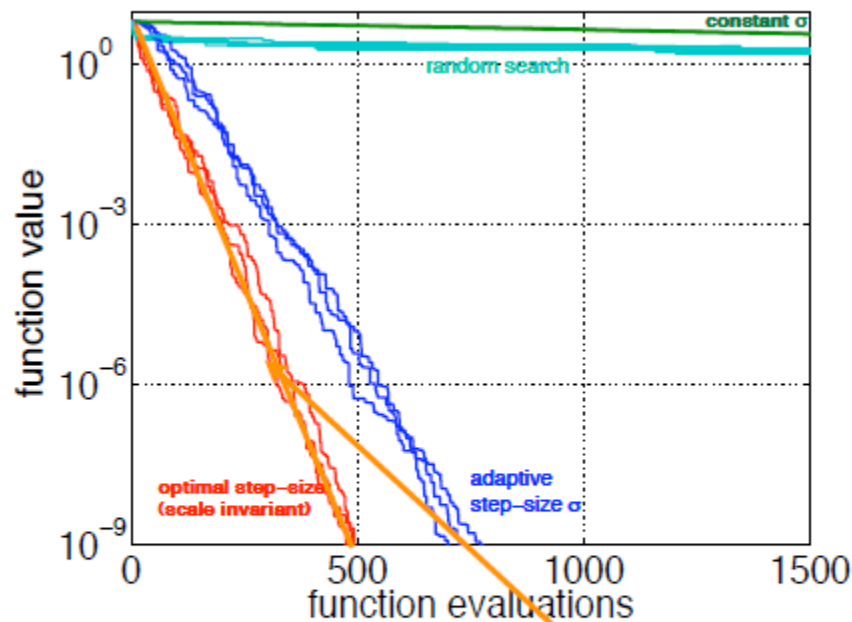
$$f(\mathbf{x}) = \sum_{i=1}^n x_i^2$$

in $[-0.2, 0.8]^n$
for $n = 10$

Zasięg mutacji zależy od odległości od środka układu współrzędnych (czysto teoretyczna sytuacja)

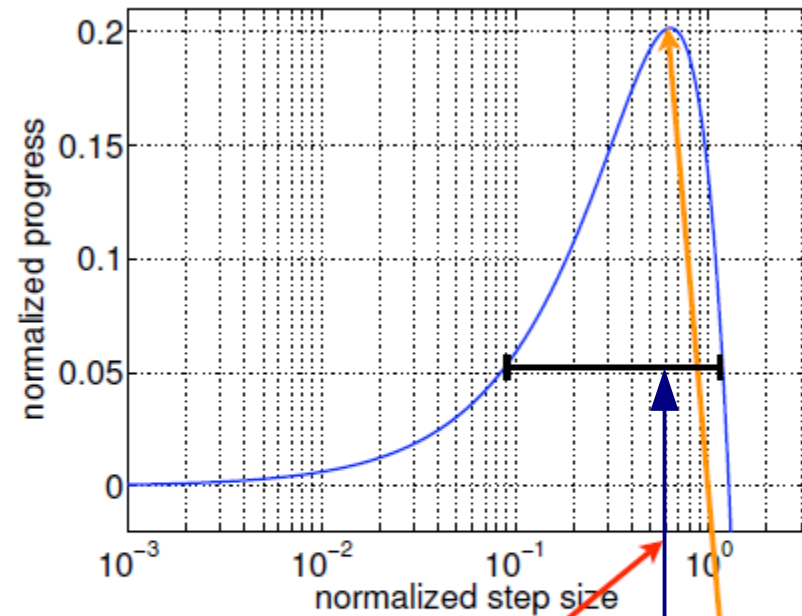
optimal step-size (scale invariant)

Zależność tempa zbieżności od zasięgu mutacji (alg. wspinaczkowy)



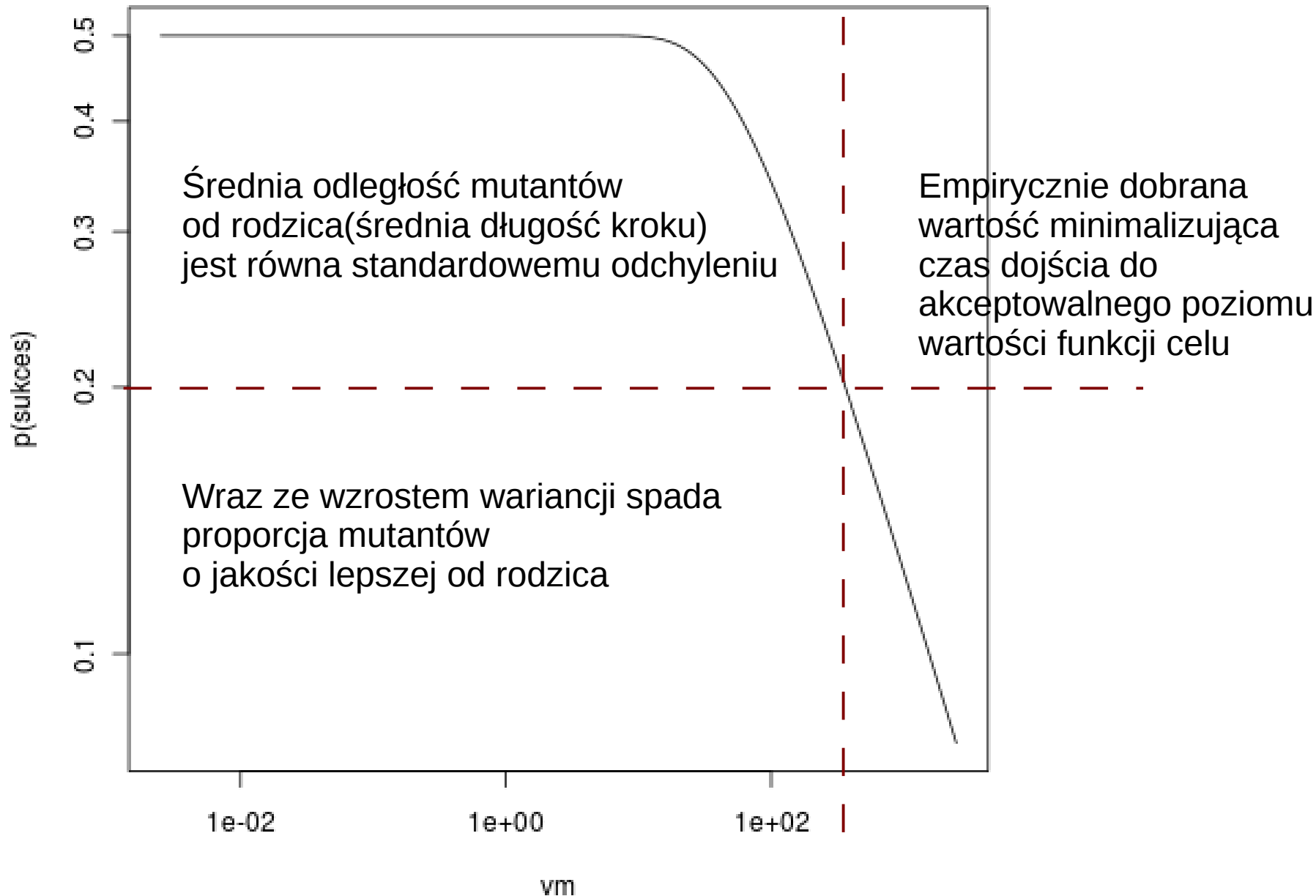
$$\sigma \leftarrow \sigma_{\text{opt}}^* \|\text{parent}\|$$

$$\frac{\varphi^*}{n}$$



okno ewolucji

Reguła 1/5 liczby sukcesów



Reguła 1/5 liczby sukcesów

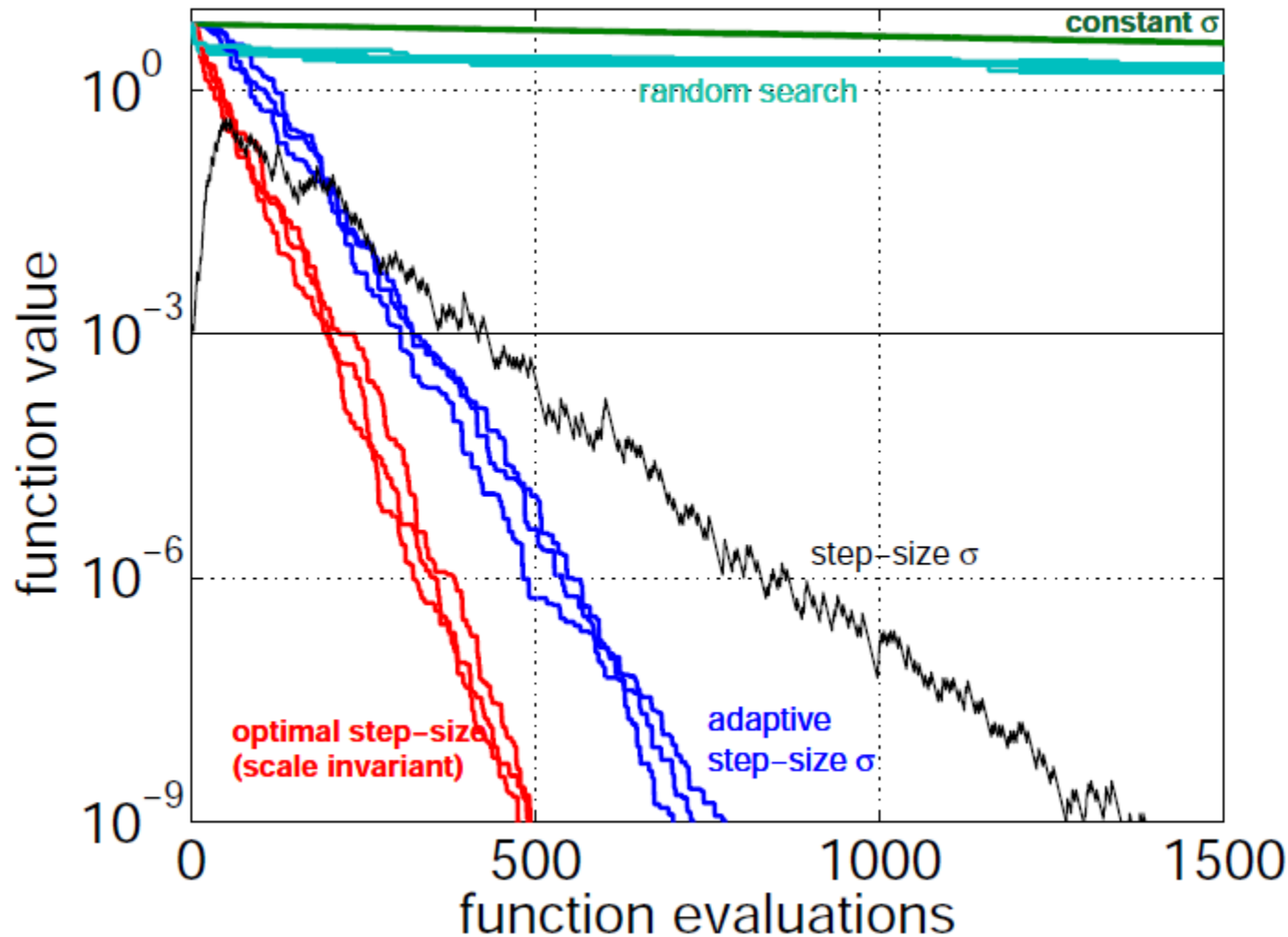
- Liczba mutacji w wyniku których mutant jest lepszy od rodzica powinna wynosić 1/5
- Sterowanie wariancją mutacji
- Pierwotnie stosowana do (1,lambda)-ES, czyli algorytmu wspinaczkowego

$$\sigma(t+k) = a \sigma(t) \quad \text{gdy } p(\text{sukces}) < 0.2$$

$$\sigma(t+k) = \frac{1}{a} \sigma(t) \quad \text{gdy } p(\text{sukces}) > 0.2$$

$$0.817 \leq a \leq 1$$

Reguła 1/5 liczby sukcesów



$$f(\mathbf{x}) = \sum_{i=1}^n x_i^2$$

in $[-0.2, 0.8]^n$
for $n = 10$

Mutacja wariantowa z wyborem zależnym od poprawy

procedure mutation(x)

$j^* = \text{select } j$

$$\text{where } p_{\text{sel}}(j) = \frac{n_s(j)}{\sum_{k=1}^K n_s(k)}$$

$y = \text{mutation}(x, j^*)$

if (*success*(j^*))

$$n_s(j^*) = n_s(j^*) + \alpha$$

$$n_s(j) = \beta \cdot n_s(j)$$

return(y)

n_s – skumulowana liczba sukcesów dla każdego wariantu mutacji

Algorytm CMA-ES

$$C(1)=I, \quad p_c(1)=0, \quad p_\sigma(1)=0$$

while ! stop

generuj $d_i(t) \sim N(0, C(t)), i=1 \dots \lambda$

Przestrzeń x

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

sortuj według $q_i(t)$

Przestrzeń skojarzona

$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

$$m(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$p_\sigma(t+1) = (1 - c_\sigma) p_\sigma(t) + C^{-1/2} \sqrt{1 - (1 - c_\sigma)^2} \sqrt{\mu} \Delta(t)$$

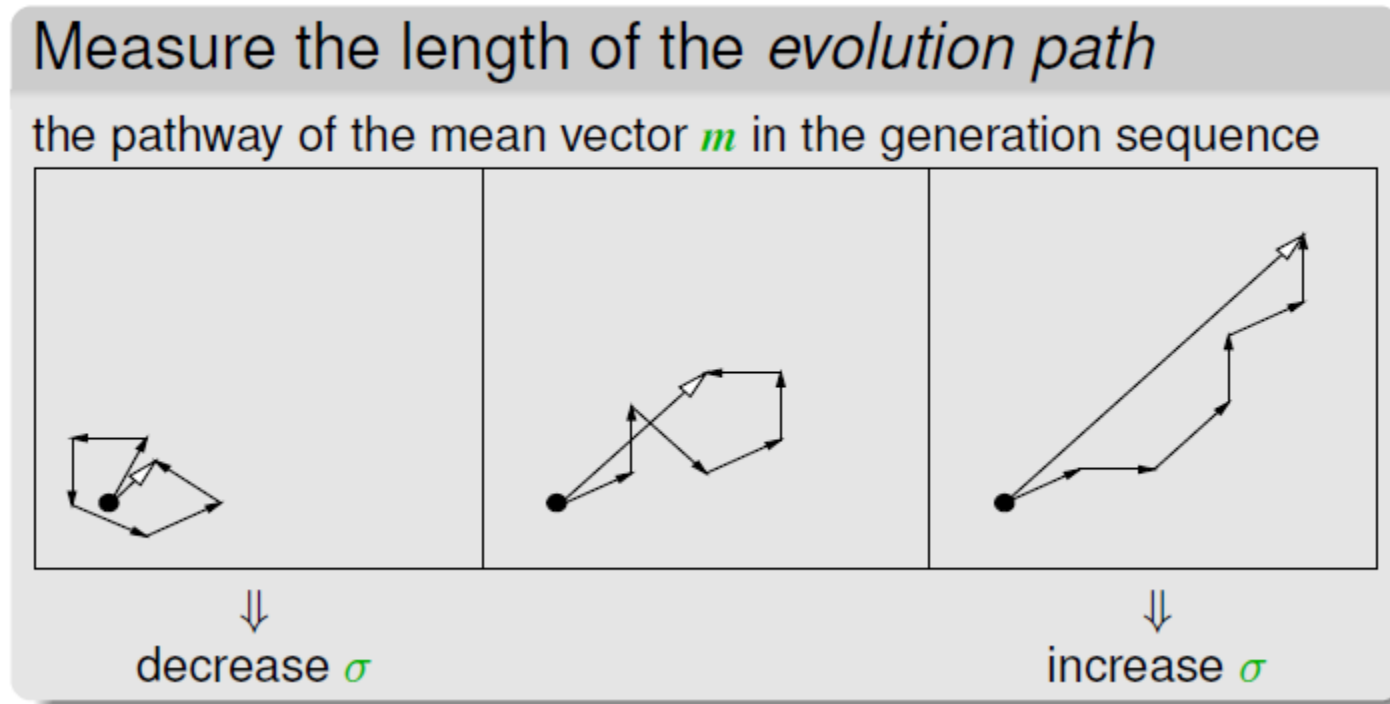
$$p_c(t+1) = (1 - c_c) p_c(t) + \sqrt{1 - (1 - c_c)^2} \sqrt{\mu} \Delta(t)$$

$$\sigma(t+1) = \sigma(t) \cdot \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma\|}{E \|N(0, I)\|} - 1\right)\right)$$

$$C(t+1) = (1 - c_1 - c_\mu) C(t) + c_1 p_c(t+1) p_c(t+1)^T + c_\mu \sum_{i=1}^{\mu} d_i(t) d_i(t)^T$$

$t \leftarrow t+1$

Adaptacja skumulowanego kroku algorytm CSA-ES



Rysunek z: A. Auger, N. Hansen:

CMA-ES — Evolution Strategies and Covariance Matrix Adaptation, tutorial, GECCO'2011

Adaptacja skumulowanego kroku algorytm CSA-ES

$$c_\sigma \approx 4/n, \quad d_\sigma \approx 1, \quad p_\sigma = 0$$

Przestrzeń x

while ! stop

generuj $d_i(t) \sim N(0, I), i=1 \dots \lambda$

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

Przestrzeń skojarzona

sortuj według $q_i(t)$

$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

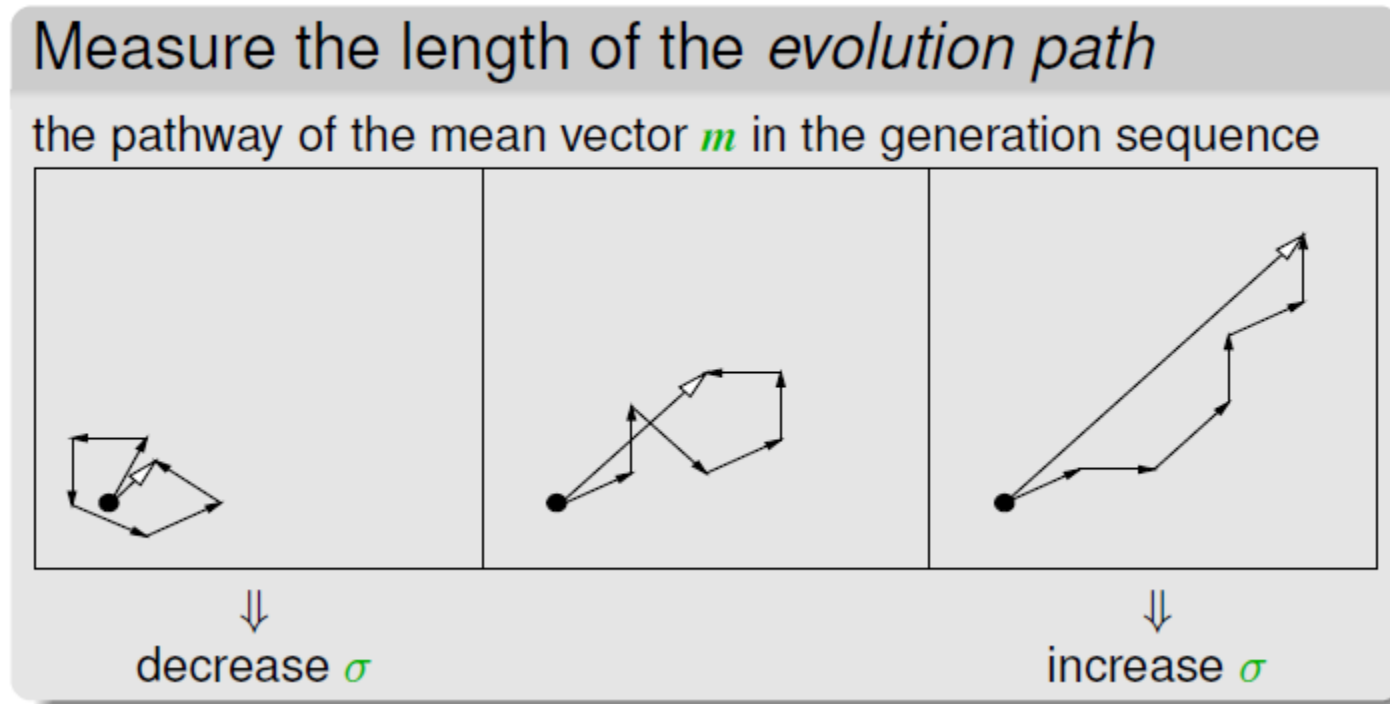
$$m(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$p_\sigma(t+1) = (1 - c_\sigma) p_\sigma(t) + \sqrt{1 - (1 - c_\sigma)^2} \sqrt{\mu} \Delta(t)$$

$$\sigma(t+1) = \sigma(t) \cdot \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma\|}{E \|N(0, I)\|} - 1\right)\right)$$

$t \leftarrow t + 1$

Adaptacja skumulowanego kroku algorytm CSA-ES



Trajektoria punktu środkowego
w przestrzeni skojarzonej

Modyfikacja macierzy kowariancji algorytm CMA-ES (wersja 0)

$$C(1) = I$$

Przestrzeń x

while ! stop

generuj $d_i(t) \sim N(0, C(t))$, $i = 1 \dots \lambda$

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

sortuj według $q_i(t)$

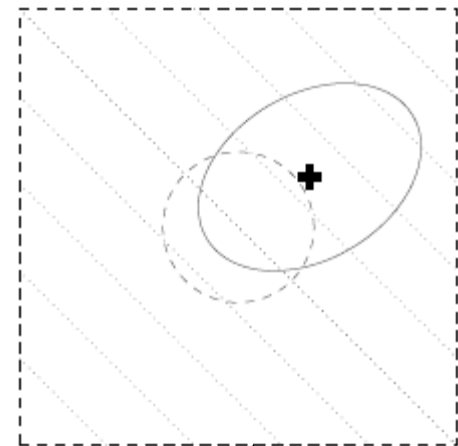
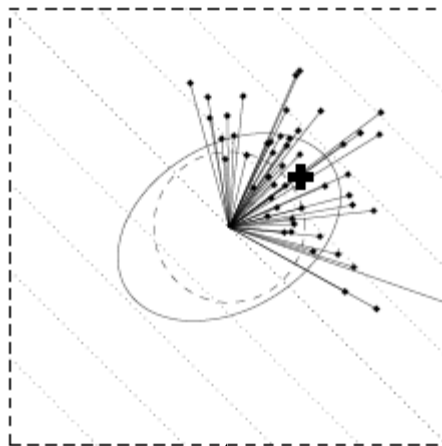
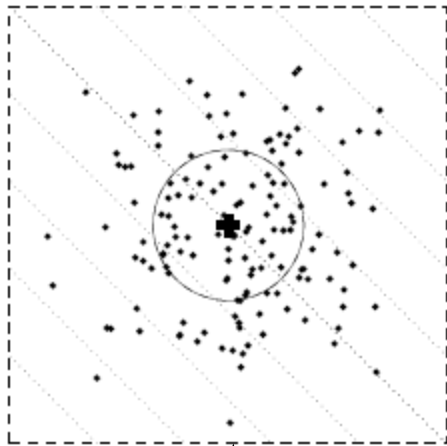
$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

$$\boxed{m}(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$\boxed{C}(t+1) = (1 - c_{cov})C(t) + c_{cov} \mu \Delta(t) \Delta(t)^T$$

$t \leftarrow t+1$

Adaptacja macierzy kowariancji algorytm CMA-ES

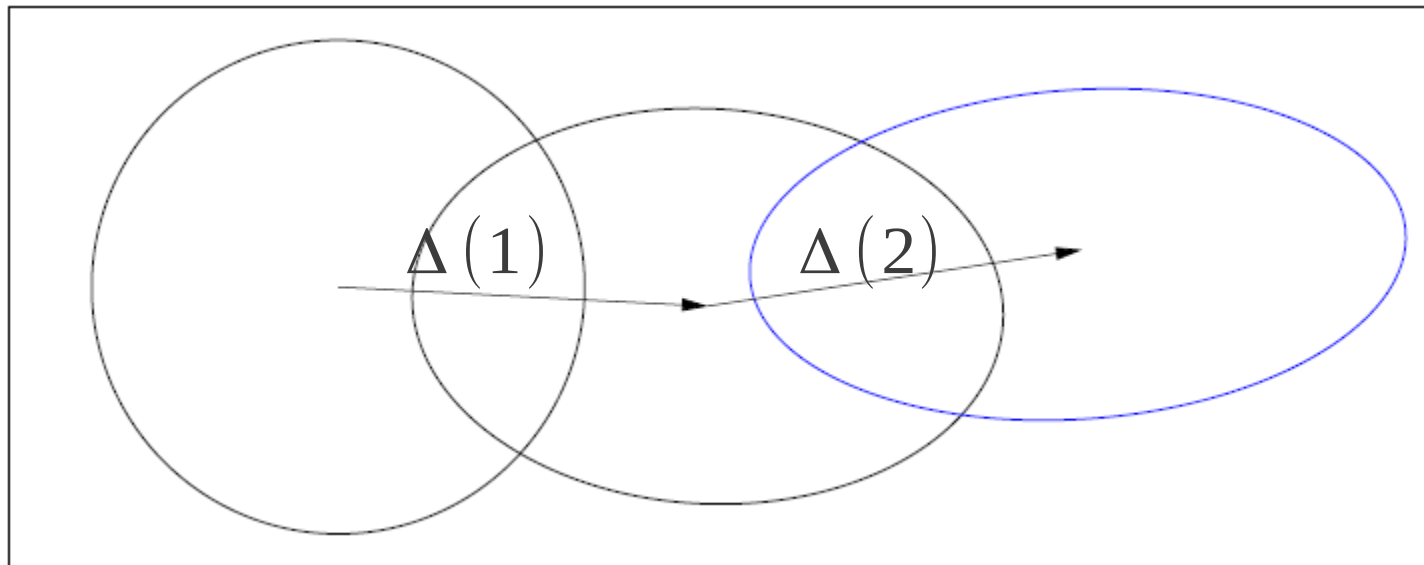


$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

$$d_i(t) \sim \sigma(t) \cdot N(0, C(t)), i=1 \dots \lambda$$

$$C(t+1) = (1 - c_{cov})C(t) + c_{cov} \mu \Delta(t) \Delta(t)^T$$

Adaptacja macierzy kowariancji algorytm CMA-ES



Modyfikacja macierzy kowariancji algorytm CMA-ES (wersja 1)

$$C(1)=I, \quad p_c(1)=0$$

Przestrzeń x

while ! stop

generuj $d_i(t) \sim N(0, C(t)), i=1 \dots \lambda$

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

sortuj według $q_i(t)$

$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

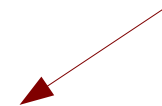
$$\boxed{m}(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$\boxed{p_c}(t+1) = (1 - c_c) p_c(t) + \sqrt{1 - (1 - c_c)^2} \sqrt{\mu} \Delta(t)$$

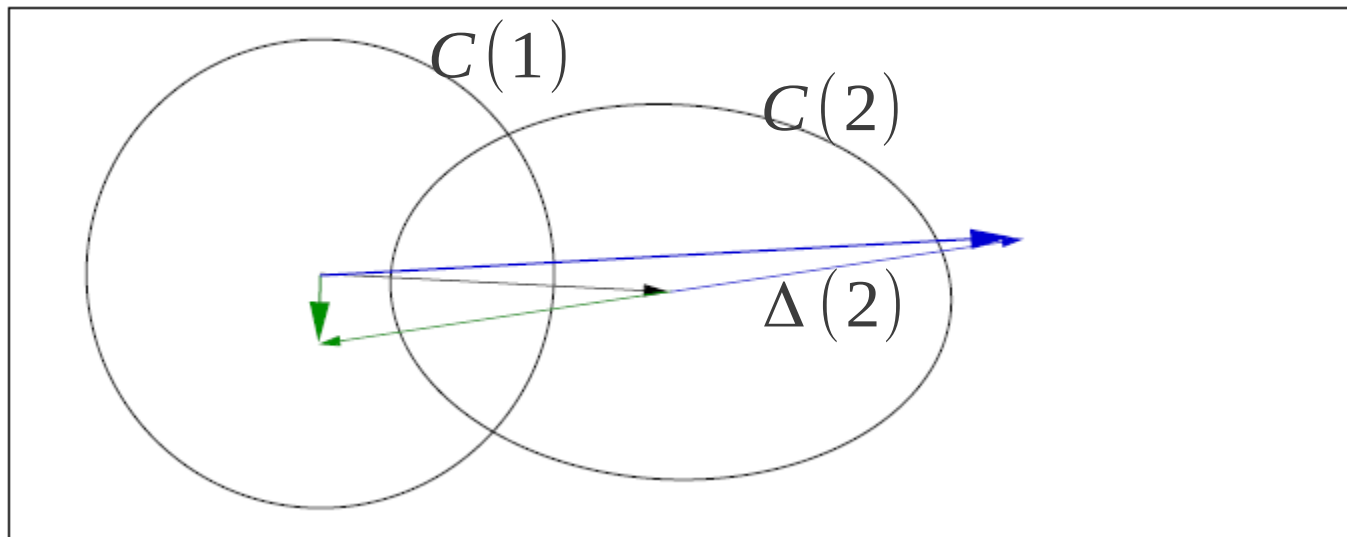
$$\boxed{C}(t+1) = (1 - c_{cov}) C(t) + c_{cov} p_c(t+1) p_c(t+1)^T$$

$$t \leftarrow t+1$$

bezwładność



Adaptacja skumulowanego kroku algorytm CMA-ES



$$p_c(t+1) = (1-c_c)p_c(t) + \sqrt{1-(1-c_c)^2} \sqrt{\mu} \Delta(t)$$
$$C(t+1) = (1-c_{cov})C(t) + c_{cov} p_c(t+1)p_c(t+1)^T$$

Rysunek z: A. Auger, N. Hansen:

CMA-ES — Evolution Strategies and Covariance Matrix Adaptation, tutorial, GECCO'2011

Modyfikacja macierzy kowariancji algorytm CMA-ES (wersja 2)

$$C(1) = I$$

Przestrzeń x

while ! stop

generuj $d_i(t) \sim N(0, C(t)), i = 1 \dots \lambda$

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

sortuj według $q_i(t)$

$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

$$m(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$C(t+1) = (1 - c_{cov}) C(t) + c_{cov} \sum_{i=1}^{\mu} d_i(t) d_i(t)^T$$

$t \leftarrow t+1$

uwzględnienie wielu
wektorów różnic



Wszystkie operacje w przestrzeni x

Pełny CMA-ES

$$C(1)=I, \quad p_c(1)=0, \quad p_\sigma(1)=0$$

while ! stop

generuj $d_i(t) \sim N(0, C(t)), i=1 \dots \lambda$

Przestrzeń x

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

sortuj według $q_i(t)$

Przestrzeń skojarzona

$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

$$m(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$p_\sigma(t+1) = (1 - c_\sigma) p_\sigma(t) + C^{-1/2} \sqrt{1 - (1 - c_\sigma)^2} \sqrt{\mu} \Delta(t)$$

$$p_c(t+1) = (1 - c_c) p_c(t) + \sqrt{1 - (1 - c_c)^2} \sqrt{\mu} \Delta(t)$$

$$\sigma(t+1) = \sigma(t) \cdot \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma\|}{E \|N(0, I)\|} - 1\right)\right)$$

$$C(t+1) = (1 - c_1 - c_\mu) C(t) + c_1 p_c(t+1) p_c(t+1)^T + c_\mu \sum_{i=1}^{\mu} d_i(t) d_i(t)^T$$

$t \leftarrow t+1$

Pełny CMA-ES

$$C(1)=I, \quad p_c(1)=0, \quad p_\sigma(1)=0$$

while ! stop

generuj $d_i(t) \sim N(0, C(t)), i=1 \dots \lambda$

oblicz $q_i(t) = q(m(t) + \sigma(t) \cdot d_i(t))$

sortuj według $q_i(t)$

$$\Delta(t) = \frac{1}{\mu} \sum_{i=1}^{\mu} d_i(t)$$

$$m(t+1) = m(t) + \sigma(t) \cdot \Delta(t)$$

$$p_\sigma(t+1) = (1 - c_\sigma) p_\sigma(t) + \sqrt{1 - (1 - c_\sigma)^2} \sqrt{\mu} C^{-1/2} \Delta(t)$$

$$\sigma(t+1) = \sigma(t) \cdot \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma\|}{E \|N(0, I)\|} - 1\right)\right)$$

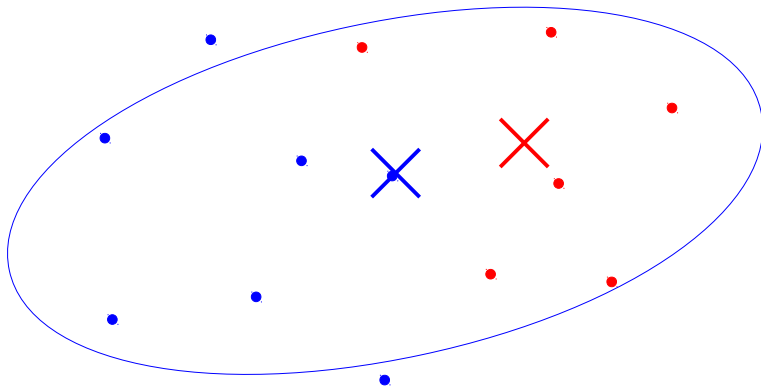
$$p_c(t+1) = (1 - c_c) p_c(t) + \sqrt{1 - (1 - c_c)^2} \sqrt{\mu} \Delta(t)$$

$$C(t+1) = (1 - c_1 - c_\mu) C(t) + c_1 p_c(t+1) p_c(t+1)^T + c_\mu \sum_{i=1}^{\mu} d_i(t) d_i(t)^T$$

$t \leftarrow t+1$

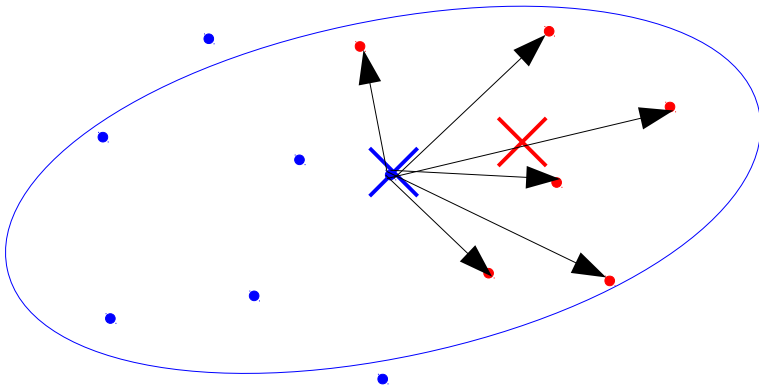
CMAES

Covariance Matrix Adaptation Evolution Strategy



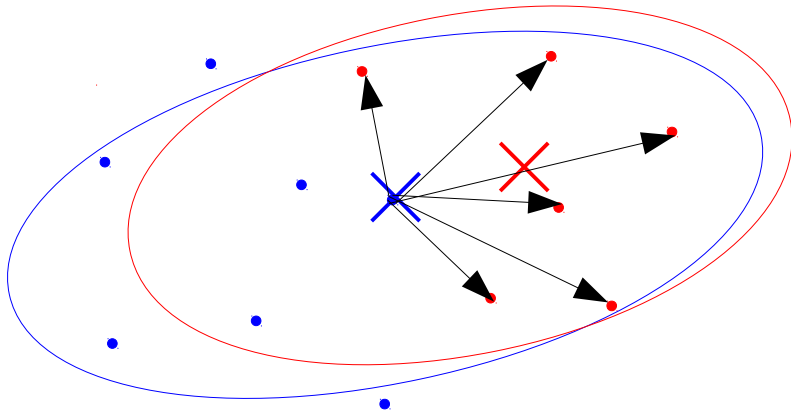
CMAES

Covariance Matrix Adaptation Evolution Strategy

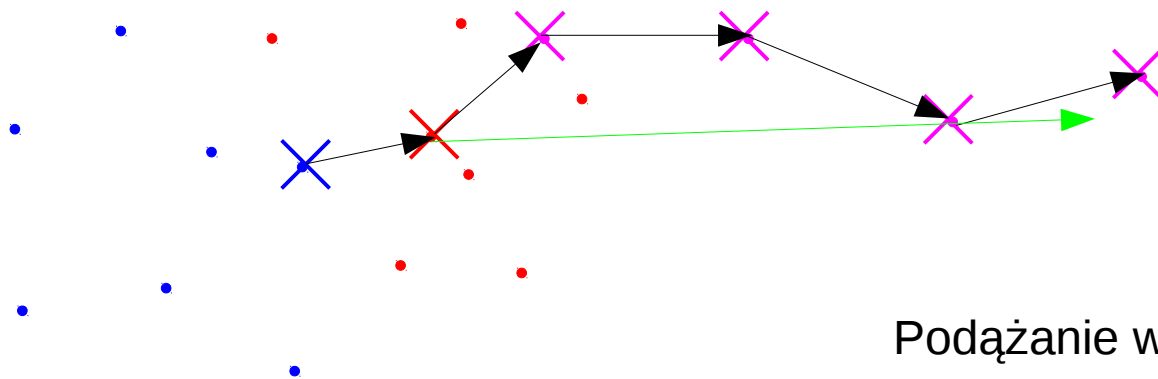


CMAES

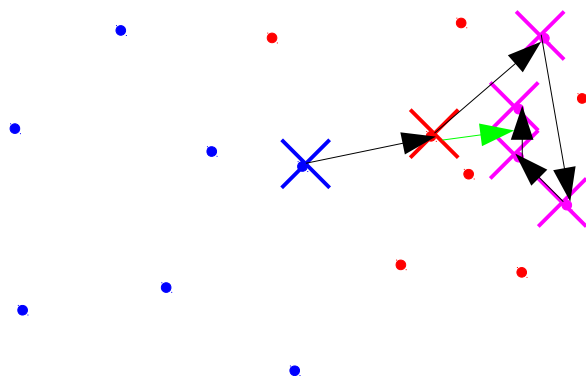
Covariance Matrix Adaptation Evolution Strategy



CMAES evolution path



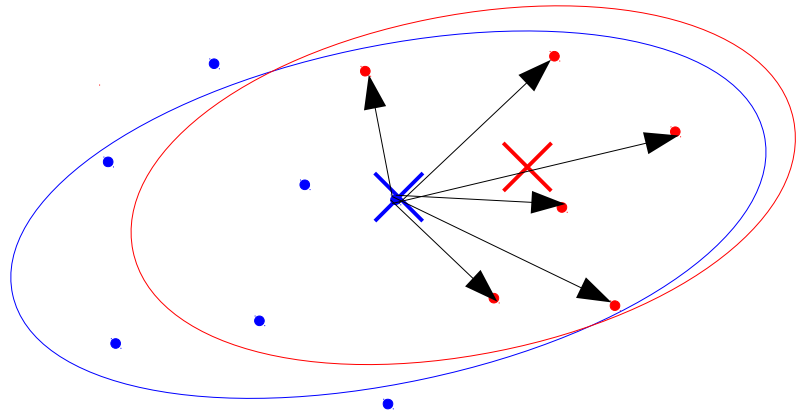
Podążanie w +/- zgodnych kierunkach



Populacja fluktuuje w jednym obszarze

CMAES

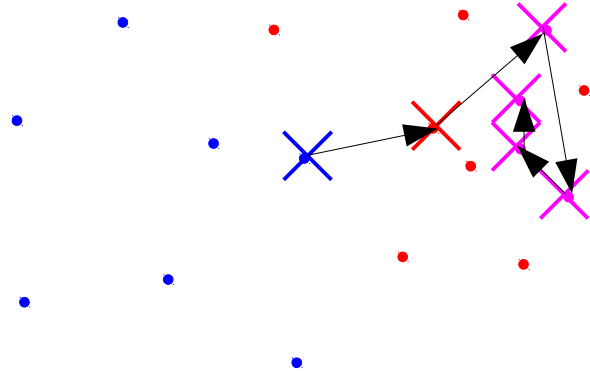
Covariance Matrix Adaptation Evolution Strategy



Na podstawie selekcji adaptuje się kształt macierzy kowariancji

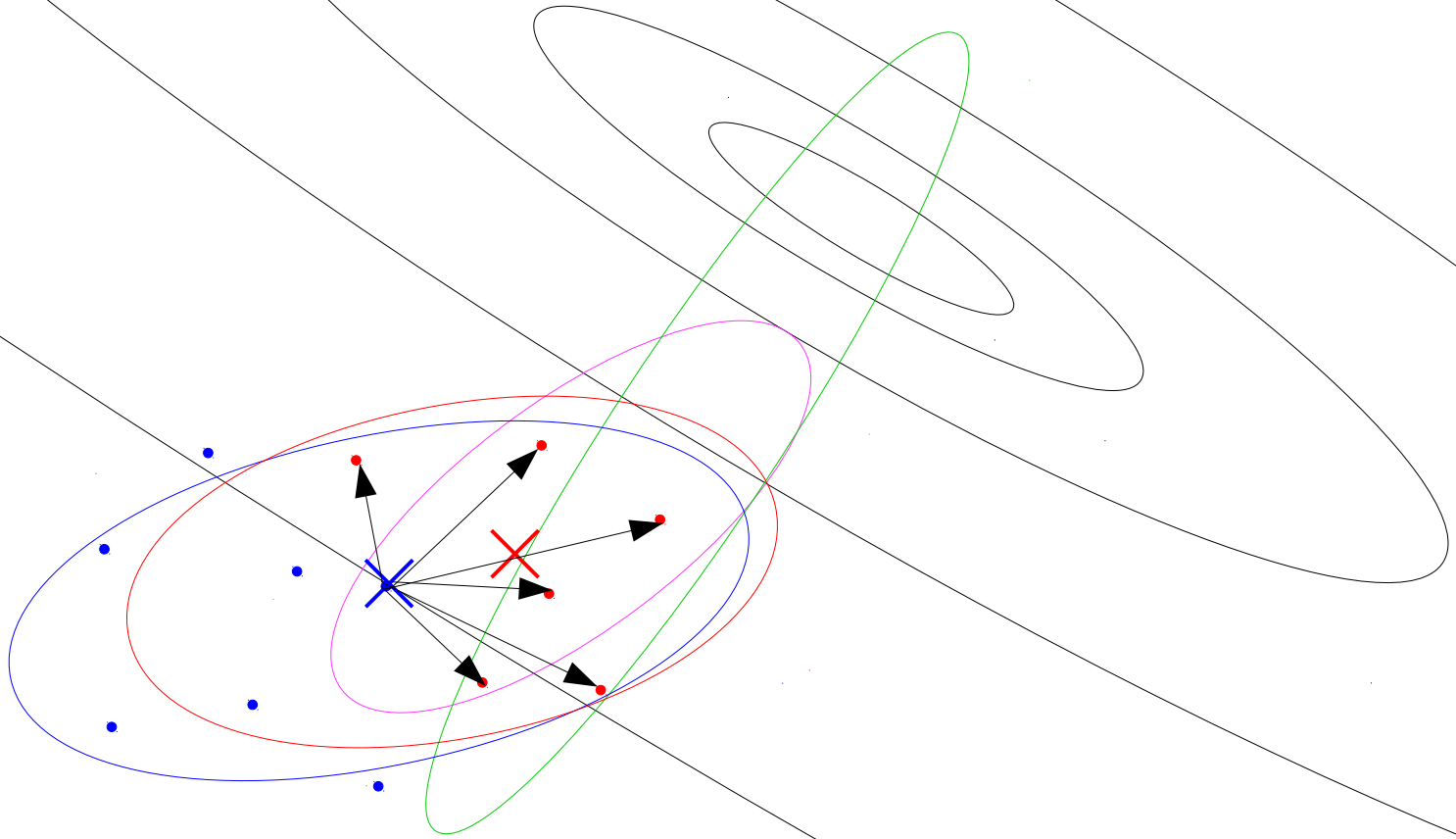
Jej skala zależy od ścieżki ewolucji

+

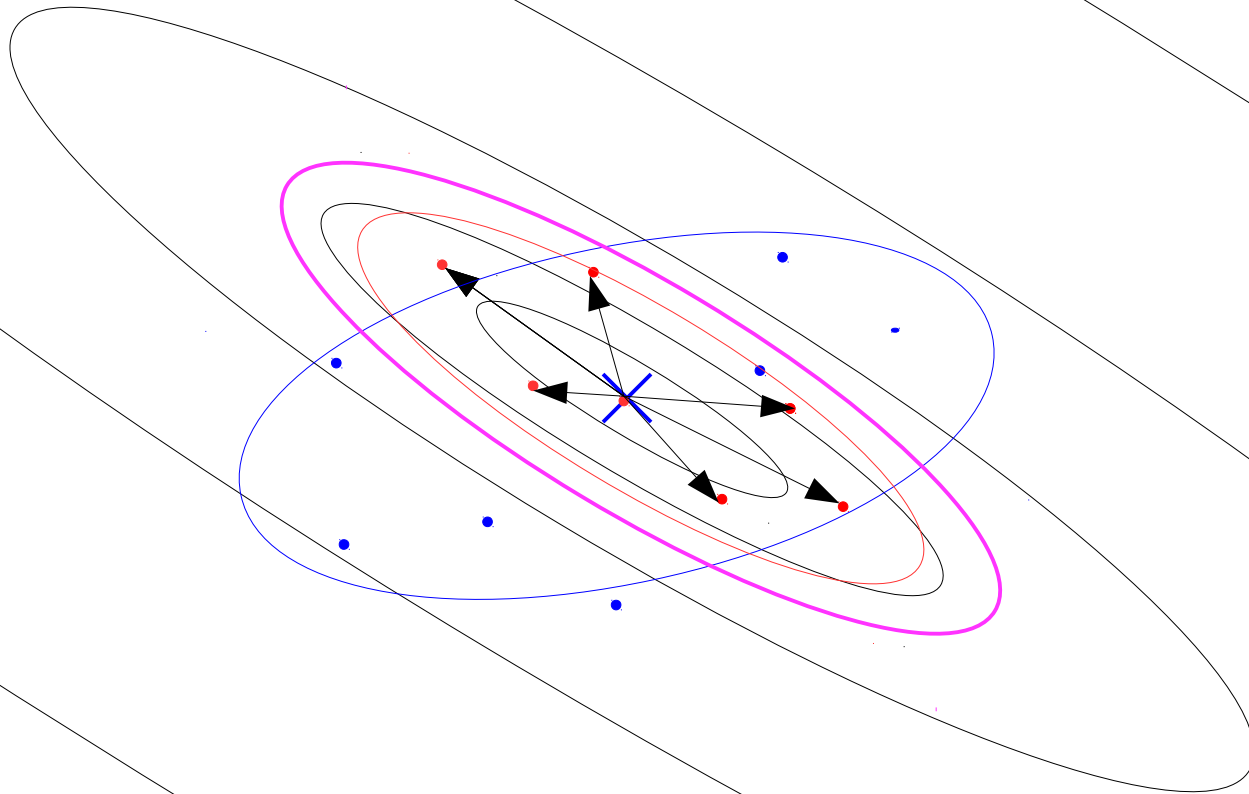


= CMAES

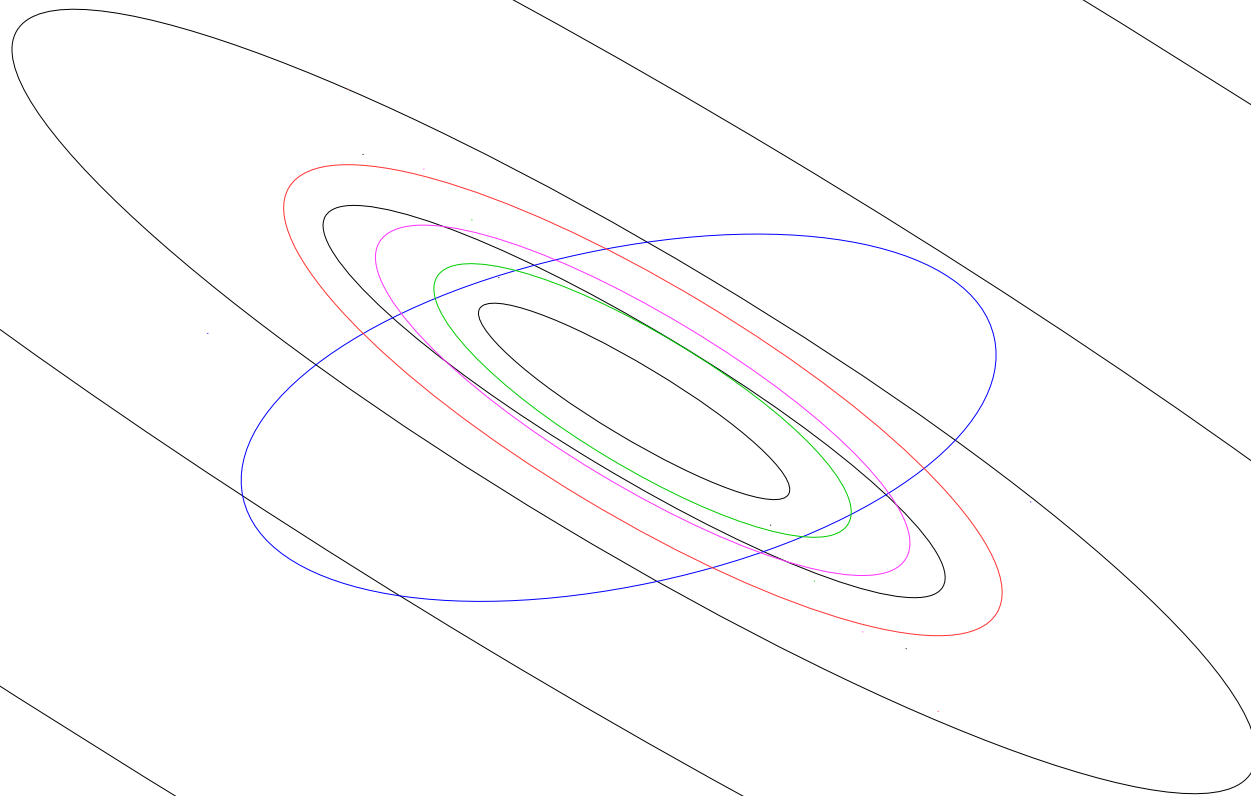
CMAES na zboczcu



CMAES na wzgórzu



CMAES na wzgórzu



Przeszukiwanie rojem cząstek

algorytm *particle swarm*

inicjuj $P^0 \leftarrow \{P_1^0, P_2^0 \dots P_\mu^0\}$

inicjuj $V^0 \leftarrow \{V_1^0, V_2^0 \dots V_\mu^0\}$

$H \leftarrow P^0$

$t \leftarrow 0$

while ! stop

$\mathbf{g}(t) \leftarrow \arg \max_{i,t} q(P_i^t)$

for ($i \in 1:\mu$)

$\mathbf{b}_i(t) \leftarrow \arg \max_t q(P_i^t)$

$V_i^{t+1} \leftarrow a(V_i^t + c(r_g(\mathbf{g}(t) - P_i^t) + r_l(\mathbf{b}_i(t) - P_i^t)))$

$P_i^{t+1} \leftarrow P_i^t + V_i^{t+1}$

$H \leftarrow H \cup P^{t+1}$

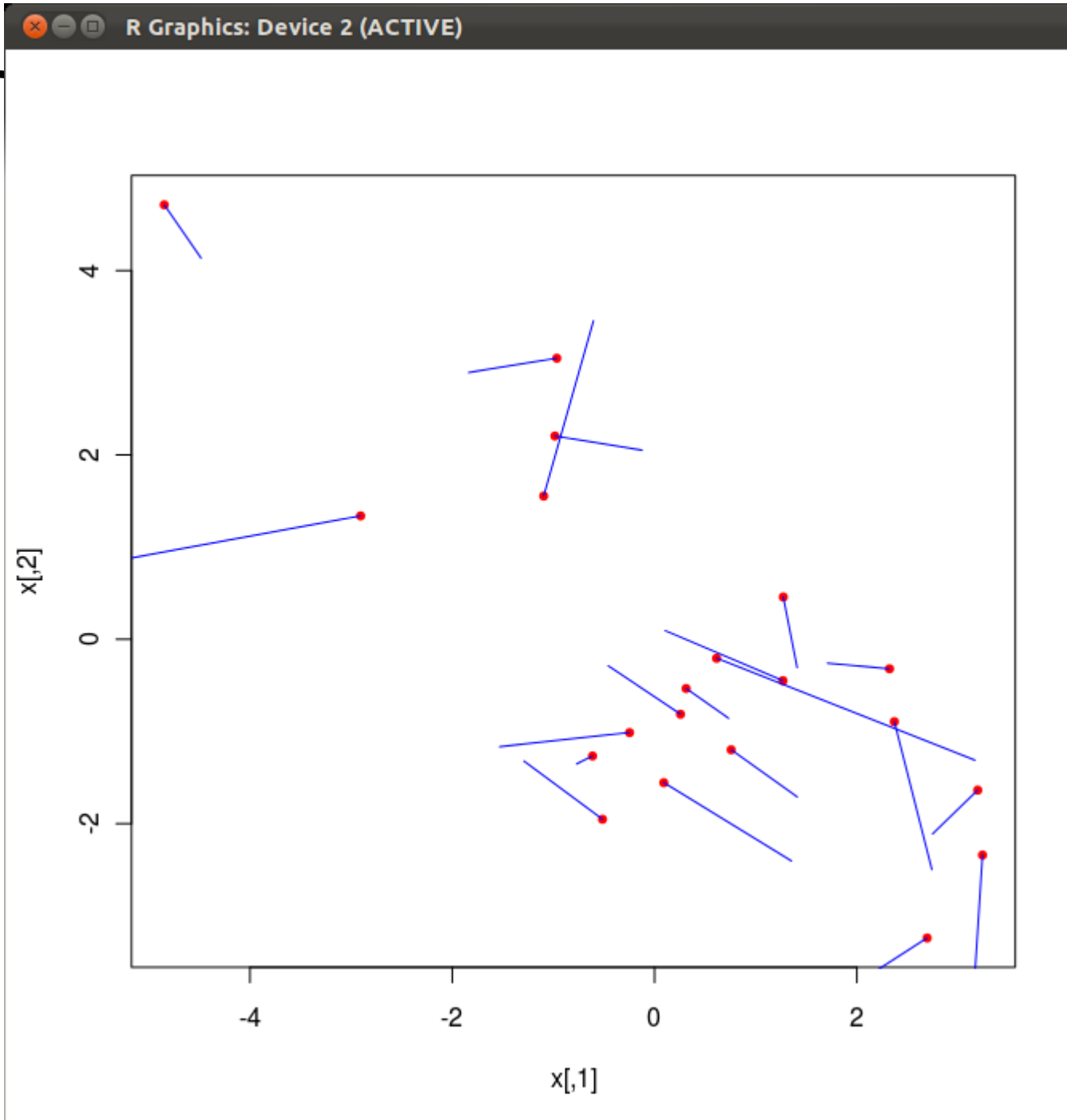
$t \leftarrow t + 1$

a, c są parametrami
typowo a=0.73, c=2.05

$r_g, r_l \sim U(0,1)$

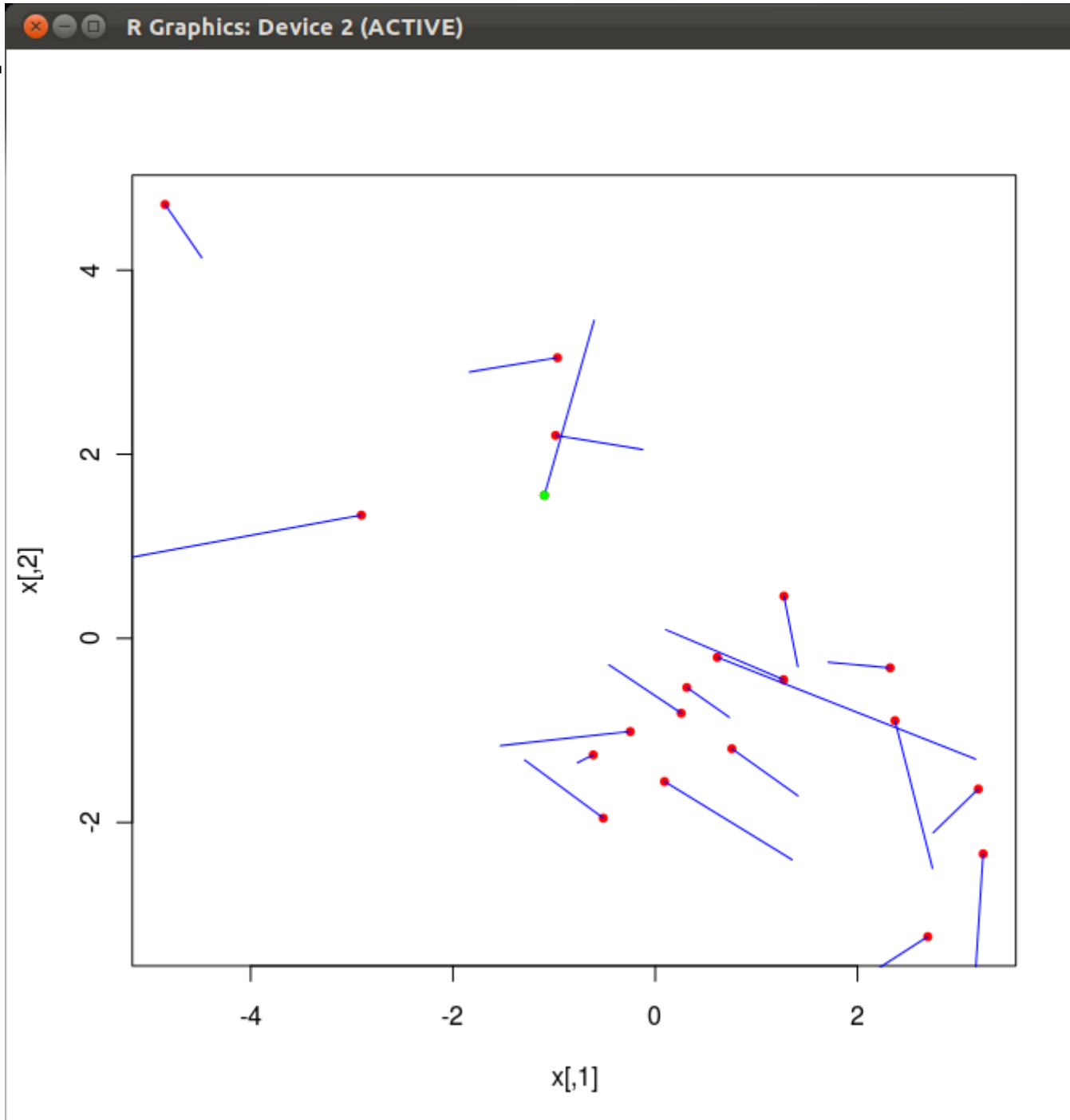
Pr

ek



Pr

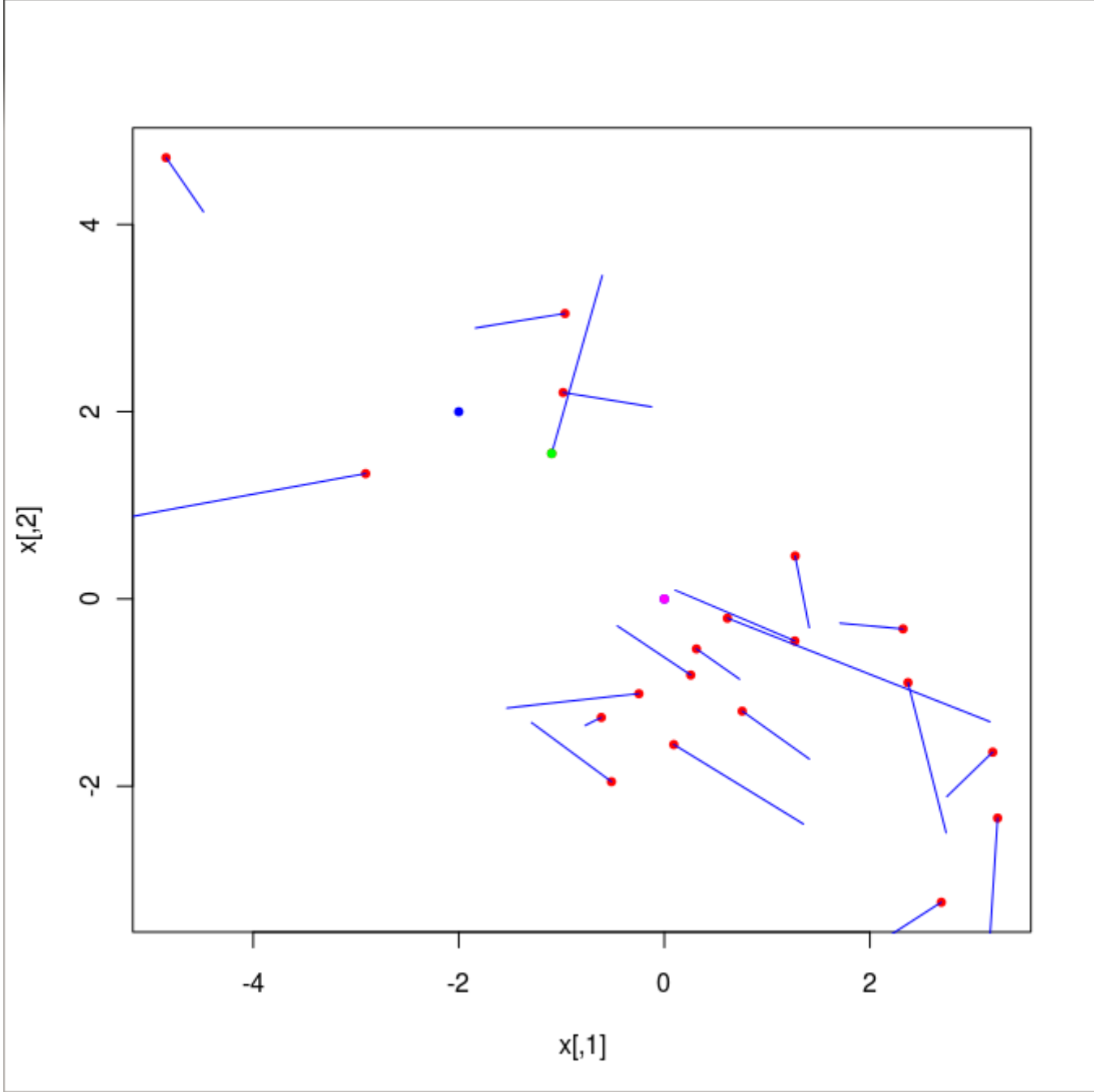
ek



Pr

R Graphics: Device 2 (ACTIVE)

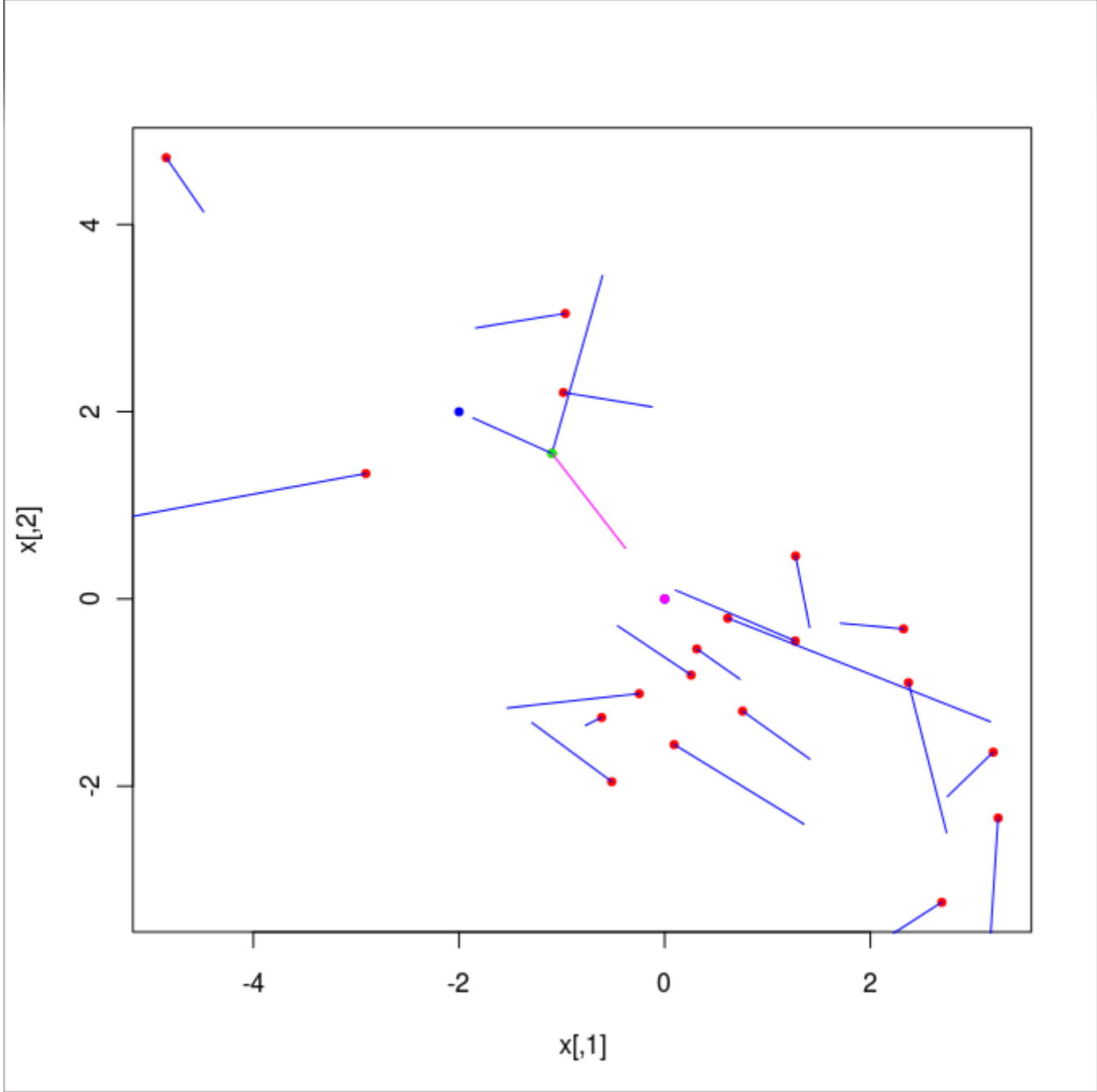
ek



Pr

R Graphics: Device 2 (ACTIVE)

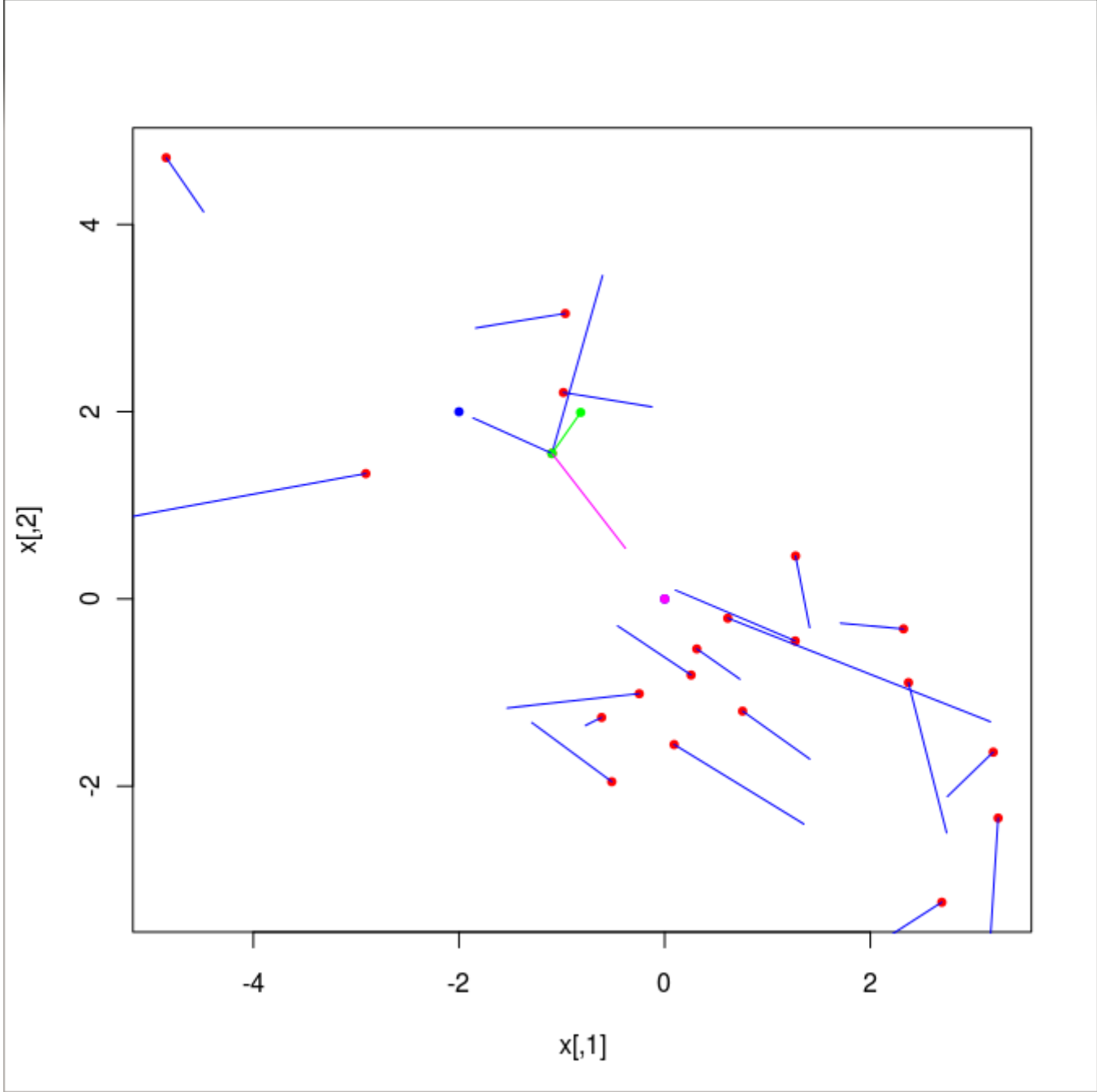
ek



Pr

R Graphics: Device 2 (ACTIVE)

ek



Binarna ewolucja różnicowa

- Jak obliczyć różnicę między wektorami binarnymi?

$x_1 = 01000011110101$

$x_2 = 11001100110011$

$y = x_2 - x_1$?

$y = 10001111000110$

Binarny rój cząstek

- Jak wyrazić prędkość dla wektora binarnego?

$x=01000011110101$

$v=11001100110011$

$y=x+v$

$y=10001111000110$

Ewolucja różnicowa i rój cząstek w przestrzeni ścieżek w grafie

