Outline

Issues with Binary-Coded Genetic Algorithm

Binary GA in Continuous Search Space

- Binary GA makes the search space discrete.
- Hamming cliffs: The decoded values of string (10000) and string (01111) are 16 and 15. However, each bit of these string is different.

Issues with Binary-Coded Genetic Algorithm

Binary GA in Continuous Search Space

- Binary GA makes the search space discrete.
- Hamming cliffs: The decoded values of string (10000) and string (01111) are 16 and 15. However, each bit of these string is different.
- Arbitrary precision $((x_i^{(U)} - x_i^{(L)})/(2^l - 1))$ impossible due to fixed-length coding 10^{-8}

Issues with Binary-Coded Genetic Algorithm

Binary GA in Continuous Search Space

- Binary GA makes the search space discrete.
- Hamming cliffs: The decoded values of string (10000) and string (01111) are 16 and 15. However, each bit of these string is different.
- Arbitrary precision $((x_i^{(U)} - x_i^{(L)})/(2^l - 1))$ impossible due to fixed-length coding

Remedy

- We should code decision variables as real numbers directly.

- However, crossover and mutation operators need structural changes.
- Important to note that selection operator remains the same because it requires fitness value.

Generalized Framework of EC Techniques

Algorithm 1 Generalized Framework

Solution representation (1) %real number

Input: $t := 1$ (Generation counter), Maximum allowed generation $= T$

Initialize random population $(P(t))$; %Parent population

Evaluate $(P(t))$; %Evaluate objective, constraints and assign fitness

while $t \leq T$ **do**

  $M(t) :=$ Selection$(P(t))$; %Survival of the fittest

  $Q(t) :=$ Variation$(M(t))$; (2) %Crossover and mutation

  Evaluate $Q(t)$; %Offspring population

  $P(t+1) :=$ Survivor$(P(t), Q(t))$; $(\mu+\lambda)$- strategy %Survival of the fittest

  $t := t + 1$;

end while

Real-Parameter Optimization Using EC Techniques

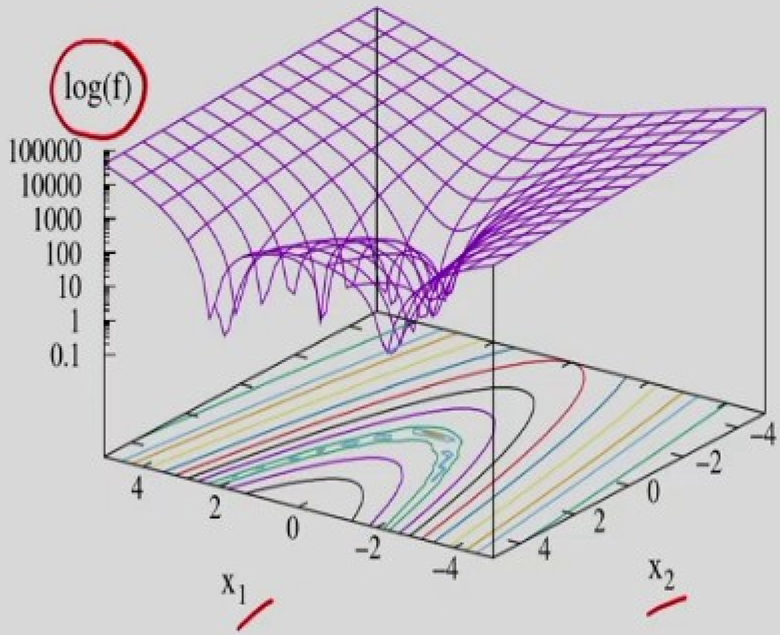
Real-Coded EC Techniques

- Real-coded genetic algorithm (RGA)
  - Similar to binary-coded genetic algorithm
- Evolutionary strategies (ES)
  - One of the oldest EC techniques
- Differential evolution (DE)
- Particle swarm optimization (PSO)
  - Motivated from flocking of birds
- Artificial bee colony (ABC), etc.
  - Motivated from foraging behavior of swarm of bees

Working Principles Through An Example

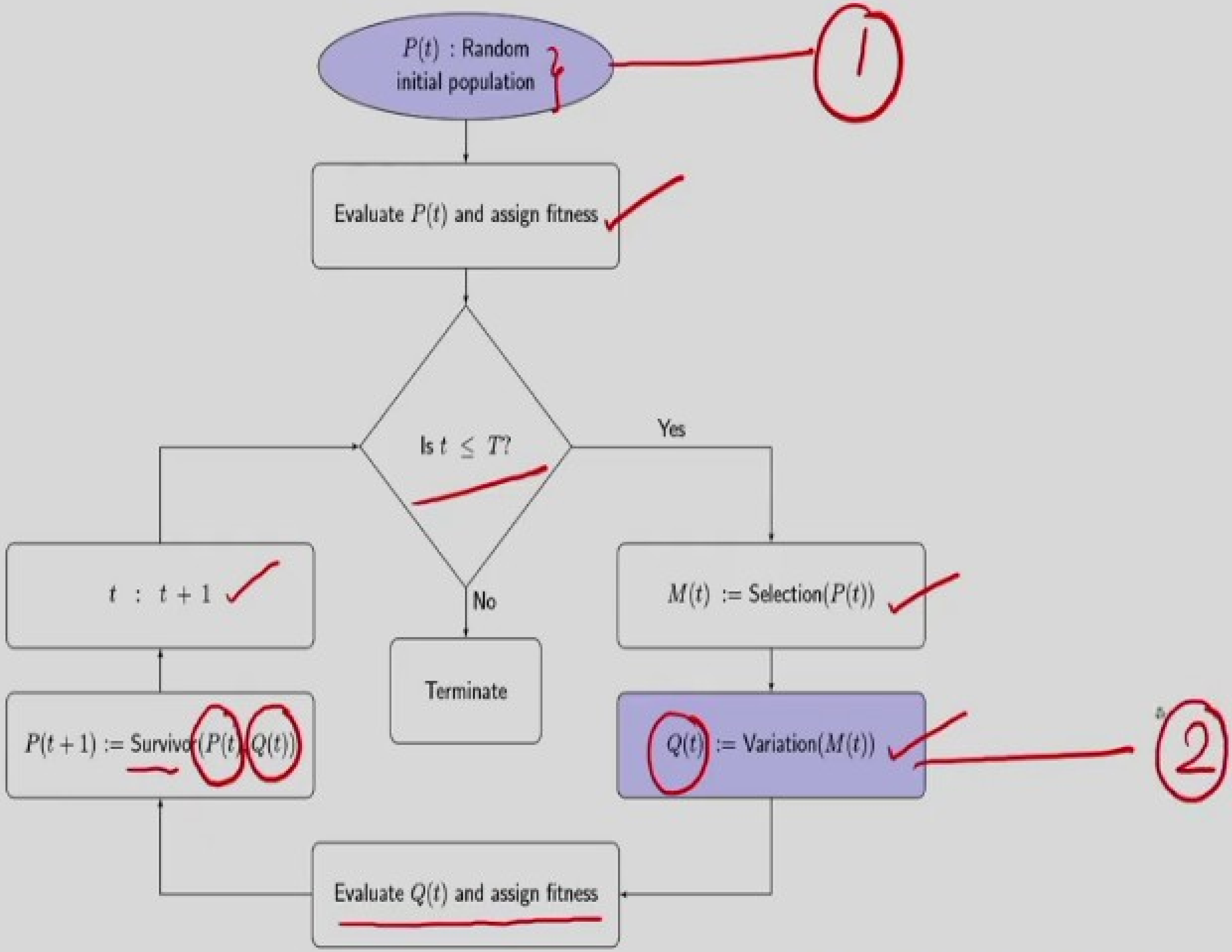
Rosenbrock Function

Minimize $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$,
bounds $-5 \leq x_1 \leq 5$ and $-5 \leq x_2 \leq 5$.



- Optimum solution is $x^* = (1, 1)^T$ and $f(x^*) = 0$

Flowchart of RGA



Initial Population

- Let the population size is $N = 8$.

Initial population		
Index	x_1	x_2
1	2.212	3.009
2	−2.289	−2.396
3	−2.393	−4.790
4	−0.639	1.692
5	−3.168	0.706
6	0.215	−2.350
7	−0.742	1.934
8	−4.563	4.791

$-5 \leq x_1 \leq 5$

$-5 \leq x_2 \leq 5$

The decision variables are coded as real numbers.

Evaluate Population

- Now, we calculate objective function $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ for each solution.
- For Solution 1: $x^{(1)} = (2.212, 3.009)^T$ and $f(x^{(1)}) = 357.154$.
  - Let us take the fitness value as the function value.
- For Solution 2: $x^{(2)} = (-2.289, -2.396)^T$ and $f(x^{(2)}) = 5843.569$.

Initial population

Index	x_1	x_2	$f(x_1, x_2)$
1	2.212	3.009	357.154
2	−2.289	−2.396	5843.569
3	−2.393	−4.790	11066.800
4	−0.639	1.692	167.414
5	−3.168	0.706	8718.166
6	0.215	−2.350	574.796
7	−0.742	1.934	194.618
8	−4.563	4.791	25731.235

Termination Condition

- Since $t < T$, we proceed to selection operator.

Binary Tournament Selection Operator

- The purpose is to identify good (usually above-average) solutions in the population.
- We eliminate bad solutions and make multiple copies of good solutions.

Index	$f(x_1, x_2)$	Index	$f(x_1, x_2)$
1	357.154	5	8718.166
2	5843.569	6	574.796
3	11066.800	7	194.618
4	167.414	8	25731.235

4

N=8

Index	$f(x_1, x_2)$	Winner
7	194.618	Index 7
6	574.796	
4	167.414	Index 4
5	8718.166	
8	25731.235	Index 3
3	11066.800	
1	357.154	Index 1 ✓
2	5843.569	

Binary Tournament Selection Operator

- The purpose is to identify good (usually above-average) solutions in the population.
- We eliminate bad solutions and make multiple copies of good solutions.

Index	$f(x_1, x_2)$	Index	$f(x_1, x_2)$
1	357.154	5	8718.166
2	5843.569	6	574.796
3	11066.800	7	194.618
$\boxed{4}$	167.414	8	25731.235

Index	$f(x_1, x_2)$	Winner
7	194.618	Index 7
6	574.796	
4	167.414	Index 4
5	8718.166	
8	25731.235	Index 3
3	11066.800	
1	357.154	Index 1
2	5843.569	

Index	$f(x_1, x_2)$	Winner
6	574.796	Index 1
1	357.154	
3	11066.800	Index 2
2	5843.569	
8	25731.235	Index 7
7	194.618	
4	167.414	Index 4
5	8718.166	

Crossover

- Crossover operator is responsible for creating new solutions. These new solutions explore the search space.
- Crossover is performed with probability (p_c). Generally, the value of p_c is kept high that supports exploration of search space.

Mating Pool

Old index	New index	x_1	x_2	$f(x_1, x_2)$
7	1	-0.742	1.934	194.618
4	2	-0.639	1.692	167.414
3	3	-2.393	-4.790	11066.800
1	4	2.212	3.009	357.154
1	5	2.212	3.009	357.154
2	6	-2.289	-2.396	5843.569
7	7	-0.742	1.934	194.618
4	8	-0.639	1.692	167.414

Single-Point Crossover in Binary GA

Property 1

The average of decoded values of binary strings before and after crossover are the same.

Parents	Offspring
10|1001	10|1010
01|1010	01|1001

2nd

Single-Point Crossover in Binary GA

Property 1

The average of decoded values of binary strings before and after crossover are the same.

Parents	**Offspring**
10|1001	10|1010
01|1010	01|1001

- Decoded values are: 41 and 26
- Average is 33.5

- Decoded values are: 42 and 25
- Average value is 33.5

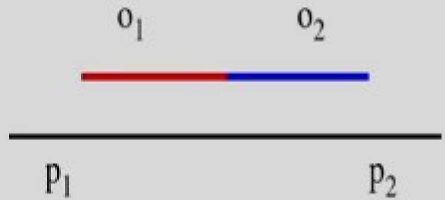
Single-Point Crossover in Binary GA

Property 2

Spread factor β is defined as the ratio of the spread of offspring solutions to that of parent solutions.

$$\beta = \left| \frac{o_2 - o_1}{p_2 - p_1} \right| \tag{1}$$

- Contracting crossover, $\beta < 1$
  - The offspring solutions are enclosed by the parent solutions.



- Expanding crossover, $\beta > 1$
  - The offspring solutions enclose the parent solutions.

- Stationary crossover, $\beta = 1$
  - The offspring solutions are the same as the parent solutions.

Crossover Operator for Real Numbers

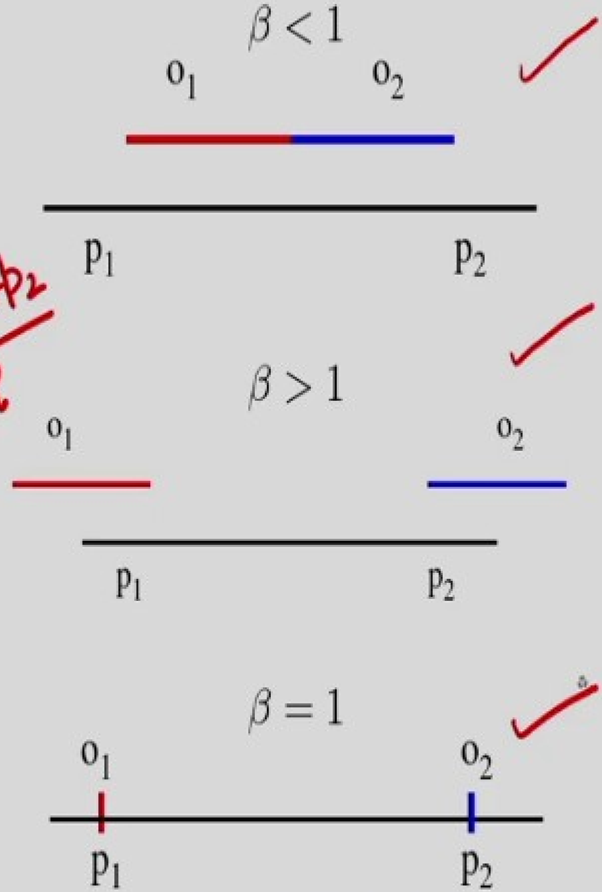
- We need structural change in crossover operator for dealing with real parameters.
- K. Deb and R. B. Agrawal. Simulated binary crossover for continuous search space. Complex Systems, 9(2):115–148, 1995
- SBX crossover operator: It was designed with respect to one-point crossover properties in binary coded GA.
  - **Average property**: The average of decoded values of binary strings before and after crossover are the same.
  - **Spread factor property**: Spread factor β is defined as the ratio of the spread of offspring solutions to that of parent solutions.
- The offspring solutions are calculated by following the average property.

SBX Crossover Operator

- Offspring:

$$
\left.
\begin{aligned}
o_1 &= \bar{x} - \tfrac{1}{2}\beta(p_2 - p_1) \\
o_2 &= \bar{x} + \tfrac{1}{2}\beta(p_2 - p_1)
\end{aligned}
\right\} \quad \frac{o_1+o_2}{2} = \frac{p_1+p_2}{2}
$$

- where, $\bar{x} = \frac{1}{2}(p_1 + p_2)$ and $p_2 > p_1$.
- It ensures that the average values of offspring and parent solutions are $\boxed{\bar{o} = \bar{p}}$.



Crossover Operator

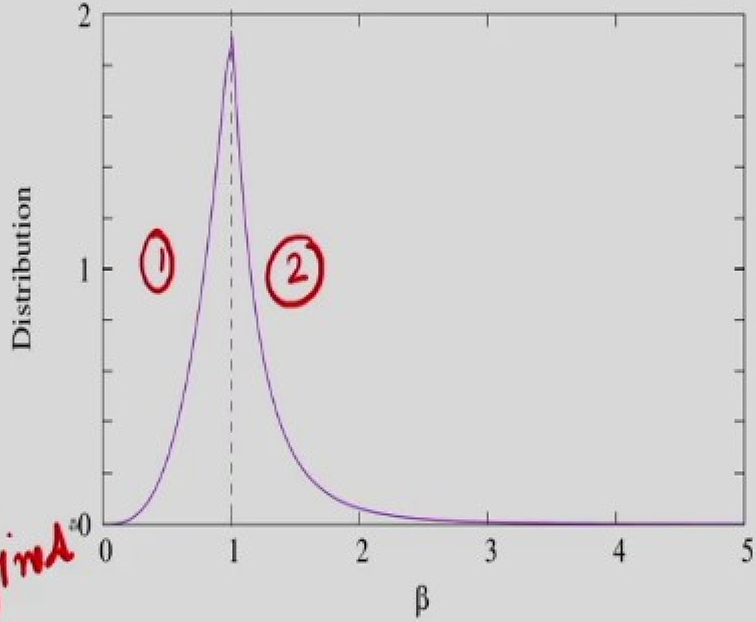
- Probability distribution of β in SBX should be similar to the probability distribution of β in binary-coded GA.

- Probability distribution function:

$$p(\beta_i) = \begin{cases} 0.5(\eta_c + 1)\beta_i^{\eta_c}, & \text{if } \beta_i \leq 1 \\ 0.5(\eta_c + 1)\frac{1}{\beta_i^{\eta_c+2}}, & \text{otherwise.} \end{cases}$$

- η_c is the SBX crossover operator distribution factor that is set by us.

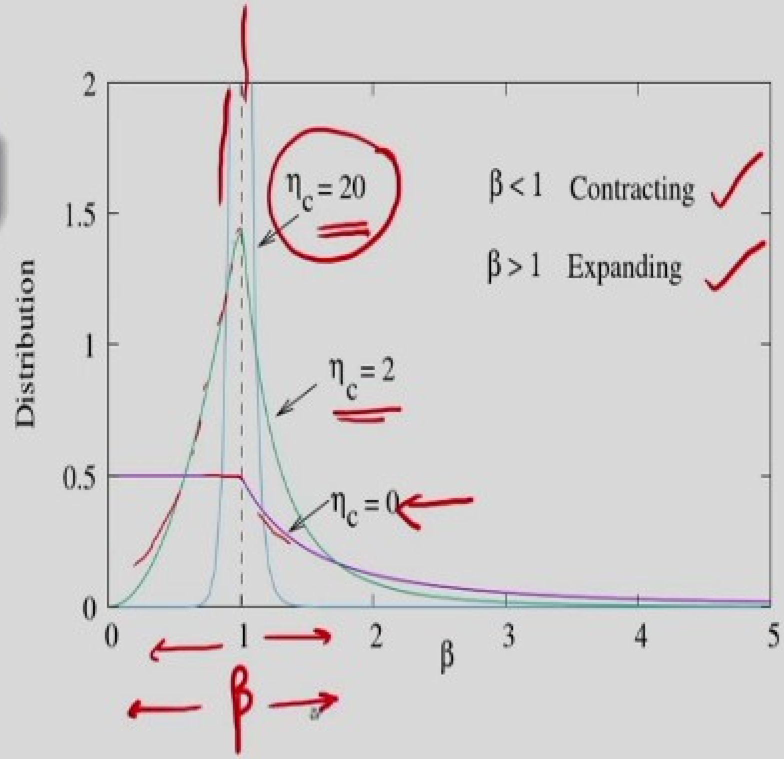
user-defined



Crossover Operator

Effect of η_c

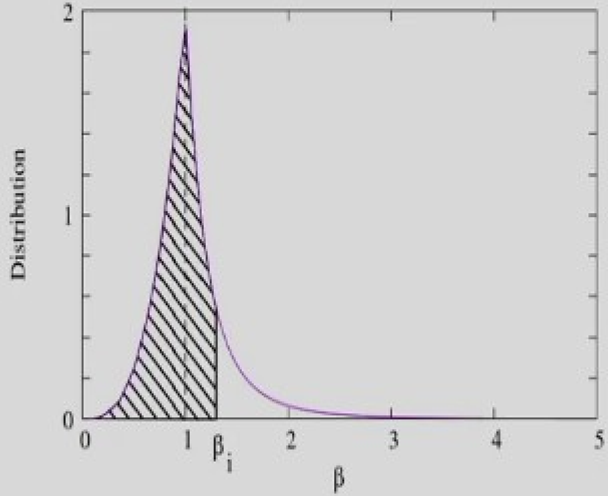
- For a larger value of η_c, there is higher probability that offspring solutions will be created near to their parent solutions.
- Contracting and expanding on the sides of $\beta = 1$ value for different values of η_c.



Crossover Operator

- Probability distribution function:

$$p(\beta_i) = \begin{cases} 0.5(\eta_c+1)\beta_i^{\eta_c}, & \text{if } \beta_i \le 1 \\ 0.5(\eta_c+1)\frac{1}{\beta_i^{\eta_c+2}}, & \text{otherwise.} \end{cases}$$



- Calculate β_i by equating area under the probability curve equal to u_i (a random number $\in [0,1]$)

$$\beta_i = \begin{cases} (2u_i)^{\frac{1}{\eta_c+1}} & \text{if } u_i \le 0.5 \\ \left(\frac{1}{2(1-u_i)}\right)^{\frac{1}{\eta_c+1}}, & \text{otherwise.} \end{cases}$$

- Offspring are:

$$x_i^{(1,t+1)} = 0.5\left[(x_i^{(1,t)} + x_i^{(2,t)}) - \beta_i(x_i^{(2,t)} - x_i^{(1,t)})\right]$$
$$x_i^{(2,t+1)} = 0.5\left[(x_i^{(1,t)} + x_i^{(2,t)}) + \beta_i(x_i^{(2,t)} - x_i^{(1,t)})\right]$$

- Here, $x_i^{(1,t+1)}$ represents $i-$th decision variable of the first parent solution at $(t+1)-$th generation. Also, $x_i^{(1,t)} < x_i^{(2,t)}$.

$p_1 < p_2$

SBX Crossover Operator: Hand Calculations

Mating Pool

New Index	x_1	x_2	$f(x_1, x_2)$
1	-0.742	1.934	194.618
2	-0.639	1.692	167.414
3	-2.393	-4.790	11066.800
4	2.212	3.009	357.154
5	2.212	3.009	357.154
6	-2.289	-2.396	5843.569
7	-0.742	1.934	194.618
8	-0.639	1.692	167.414

- Let us assume that the probability of crossover is $p_c = 0.9$, and user-defined parameter of SBX crossover operator is $\eta_c = 15$.

SBX Crossover Operator: Hand Calculations

- We pick pairs of two solutions randomly from the mating pool for performing the crossover.
- We then generate random numbers sequentially for each of the pair as given below.

Pair	Random number (r)	Pair	Random number (r)
{3, 7}	0.63	{8, 4}	0.98
{5, 2}	0.13	{6, 1}	0.57

- Offspring equations

$$\left.\begin{aligned} x_i^{(1,t+1)} &= 0.5\left[(x_i^{(1,t)} + x_i^{(2,t)}) - \beta_i(x_i^{(2,t)} - x_i^{(1,t)})\right] \\ x_i^{(2,t+1)} &= 0.5\left[(x_i^{(1,t)} + x_i^{(2,t)}) + \beta_i(x_i^{(2,t)} - x_i^{(1,t)})\right]. \end{aligned}\right\}$$

$p_1 < p_2$

- Offspring equations can also be written as

$$\begin{aligned} x_i^{(1,t+1)} &= 0.5\left[(1+\beta_i)x_i^{(1,t)} + (1-\beta_i)x_i^{(2,t)}\right] \\ x_i^{(2,t+1)} &= 0.5\left[(1-\beta_i)x_i^{(1,t)} + (1+\beta_i)x_i^{(2,t)}\right]. \end{aligned} \Longrightarrow$$

SBX Crossover Operator: Hand Calculations

- For the first pair $\{3, 7\}$, since $r = 0.63 < p_c = 0.9$, we perform crossover.
  - Let us perform crossover on x_1 variable of both solutions, that is, $x_1^{(3)} = -2.393$ and $x_1^{(7)} = -0.742$.
  - It is important to note that when we perform crossover between two solutions p_1 and p_2, $p_1 < p_2$. Otherwise, interchange the values.
  - For this pair, $p_1 = x_1^{(3)} = -2.393$ and $p_2 = x_1^{(7)} = -0.742$.
  - The random number for SBX crossover operator is $u_1 = 0.236$. The value of $\beta_1 = 0.954$.
  - The new values are $0.5\left[(1+\beta_1)p_1 + (1-\beta_1)p_2\right]$ and $0.5\left[(1-\beta_1)p_1 + (1+\beta_1)p_2\right]$, that are, -2.355 and -0.780.
  - Let us perform crossover on x_2 variable of both solutions, that is, $x_2^{(3)} = -4.790$ and $x_2^{(7)} = 1.934$.
  - For this pair, $p_1 = x_2^{(3)} = -4.790$ and $p_2 = x_2^{(7)} = 1.934$.
  - The random number for SBX crossover operator is $u_2 = 0.461$. The value of $\beta_2 = 0.995$.
  - The new values are $0.5\left[(1+\beta_2)p_1 + (1-\beta_2)p_2\right]$ and $0.5\left[(1-\beta_2)p_1 + (1+\beta_2)p_2\right]$, that are, -4.773 and 1.917.
- Two new solutions are: $(-2.355, -4.773)^T$ and $(-0.780, 1.917)^T$

SBX Crossover Operator: Hand Calculations

- For the second pair of solutions $\{5, 2\}$, since $r = 0.13 < p_c$, we perform mutation.
  - Let us perform crossover on x_1 variable of both solutions, that is, $x_1^{(5)} = 2.212$ and $x_1^{(2)} = -0.639$.

$p_1 < p_2$

SBX Crossover Operator: Hand Calculations

- For the second pair of solutions $\{5, 2\}$, since $r = 0.13 < p_c$, we perform mutation.
  - Let us perform crossover on x_1 variable of both solutions, that is, $x_1^{(5)} = 2.212$ and $x_1^{(2)} = -0.639$.
  - For this pair, $p_1 = x_1^{(2)} = -0.639$ and $p_2 = x_1^{(5)} = 2.212$.
  - The random number for SBX crossover operator is $u_1 = 0.896$. The value of $\beta_1 = 1.103$.
  - The new values are $0.5\left[(1+\beta_1)p_1 + (1-\beta_1)p_2\right]$ and $0.5\left[(1-\beta_1)p_1 + (1+\beta_1)p_2\right]$, that are, -0.785 and 2.359.

  - Let us perform crossover on x_2 variable of both solutions, that is, $x_2^{(5)} = 3.009$ and $x_2^{(2)} = 1.692$.
  - For this pair, $p_1 = x_2^{(2)} = 1.692$ and $p_2 = x_2^{(5)} = 3.009$.

$p_1 < p_2$

SBX Crossover Operator: Hand Calculations

- For the second pair of solutions $\{5, 2\}$, since $r = 0.13 < p_c$, we perform mutation.
  - Let us perform crossover on x_1 variable of both solutions, that is, $x_1^{(5)} = 2.212$ and $x_1^{(2)} = -0.639$.
  - For this pair, $p_1 = x_1^{(2)} = -0.639$ and $p_2 = x_1^{(5)} = 2.212$.
  - The random number for SBX crossover operator is $u_1 = 0.896$. The value of $\beta_1 = 1.103$.
  - The new values are $0.5\left[(1+\beta_1)p_1 + (1-\beta_1)p_2\right]$ and $0.5\left[(1-\beta_1)p_1 + (1+\beta_1)p_2\right]$, that are, -0.785 and 2.359.
  - Let us perform crossover on x_2 variable of both solutions, that is, $x_2^{(5)} = 3.009$ and $x_2^{(2)} = 1.692$.
  - For this pair, $p_1 = x_2^{(2)} = 1.692$ and $p_2 = x_2^{(5)} = 3.009$.
  - The random number for SBX crossover operator is $u_2 = 0.511$. The value of $\beta_2 = 1.001$.
  - The new values are $0.5\left[(1+\beta_2)p_1 + (1-\beta_2)p_2\right]$ and $0.5\left[(1-\beta_2)p_1 + (1+\beta_2)p_2\right]$, that are, 1.691 and 3.010.
- Two new solutions are: $(-0.785, 1.691)^T$ and $(2.359, 3.010)^T$.

SBX Crossover Operator: Hand Calculations

Crossover

- For the third pair of solutions $\{8, 4\}$, since $r = 0.98 > p_c$, we do not perform ~~mutation~~.
  - Copy the same solutions: $(-0.639, 1.692)^T$ and $(2.212, 3.009)^T$.

SBX Crossover Operator: Hand Calculations

- For the third pair of solutions $\{8, 4\}$, since $r = 0.98 > p_c$, we do not perform ~~mutation~~.
  - Copy the same solutions: $(-0.639, 1.692)^T$ and $(2.212, 3.009)^T$.
- For the last pair of solutions $\{6, 1\}$, we perform crossover because $r = 0.57 < p_c$. The relevant data is: $u_1 = 0.118$ and $\beta_1 = 0.914$, and $u_2 = 0.335$ and $\beta_2 = 0.975$.
  - Two new solutions are: $(-2.223, -2.342)^T$ and $(-0.809, 1.881)^T$.

SBX Crossover Operator: Hand Calculations

Solutions after crossover

Index	x_1	x_2	$f(x_1, x_2)$
1	-0.809	1.881	153.874
2	-0.785	1.722	125.261
3	-2.355	-4.773	10655.925
4	2.212	3.009	357.154
5	2.359	2.978	670.843
6	-2.223	-2.342	5313.910
7	-0.780	1.917	174.606
8	-0.639	1.692	167.414

Observations

- The best fitness among solutions before crossover was 167.414.
- Now, we have two solutions having better fitness than 167.414 in the table.

- Better solutions will be emphasized and bad solutions will be eliminated by selection operator in further generations.

Mutation Operator

Purpose

- Create new solution in a population with a low probability (p_m): exploitation

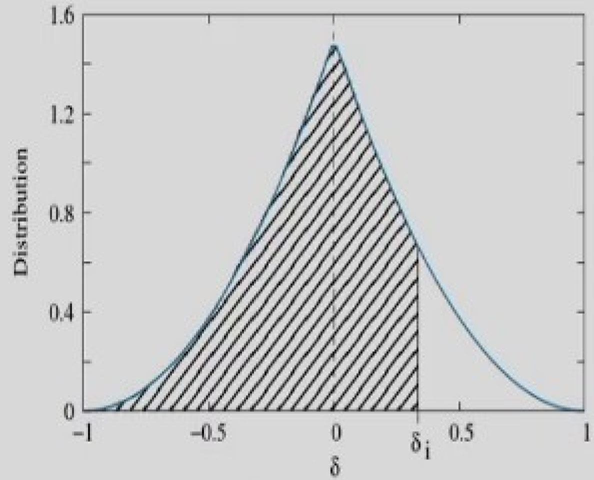
Polynomial Mutation operator

- Use polynomial distribution for perturbing a solution

$$y_i^{(1,t+1)} = x_i^{(1,t+1)} + (x_i^{(U)} - x_i^{(L)})\bar{\delta}_i$$

- $\bar{\delta}_i$ is calculated from polynomial probability distribution

$$\int P(\delta) = 0.5(\eta_m + 1)(1 - |\delta|)^{\eta_m} :$$



- Calculate $\bar{\delta}_i$ by equating area under the probability curve equal to r_i (a random number $\in [0,1]$)

$$\bar{\delta}_i = \begin{cases} (2r_i)^{1/(\eta_m+1)} - 1, & \text{if } r_i < 0.5, \\ 1 - [2(1-r_i)]^{1/(\eta_m+1)}, & \text{if } r_i \geq 0.5. \end{cases}$$

η_m : user-defined

Polynomial Mutation Operator: Hand Calculations

$n=2$

- Let the probability of mutation $p_m = 1/n = 0.5$ and the user-defined parameter for polynomial mutation is $\eta_m = 20$.

Polynomial Mutation Operator: Hand Calculations

- Let the probability of mutation $p_m = 1/n = 0.5$ and the user-defined parameter for polynomial mutation is $\eta_m = 20$.
- We create random number between $[0, 1]$ for each solution to perform mutation. Let the random numbers are: 0.24, 0.62, 0.98, 0.79, 0.31, 0.71, 0.49, 0.83.
- For solution 1, since $r = 0.24 < p_m$, we perform mutation.
  - Let the random number $r_1 = 0.956$ for $x_1^{(1)} = -0.809$. We calculate $\bar{\delta_1} = 0.109$.
  - New value is $x_1^{(1)} + (x_1^{(U)} - x_1^{(L)})\bar{\delta_1} = -0.809 + (10)(0.109) = 0.284$.
  - Let the random number $r_2 = 0.635$ for $x_2^{(1)} = 1.881$. We calculate $\bar{\delta_2} = -0.003$.
  - New value is $x_2^{(1)} + (x_2^{(U)} - x_2^{(L)})\bar{\delta_2} = 1.881 + (10)(-0.003) = 1.856$.
  - New solution is $(-1.283, 1.856)^T$
- For solutions '2', '3' and '4', we do not perform mutation because $r > p_m$.
  - Copy them.

Polynomial Mutation Operator: Hand Calculations

- For solution 5, since $r = 0.31 < p_m$, we perform mutation.
  - Let the random number $r_1 = 0.217$ for $x_1^{(5)} = 2.359$. We calculate $\bar{\delta_1} = -0.039$.
  - New value is $x_1^{(5)} + (x_1^{(U)} - x_1^{(L)})\bar{\delta_1} = 2.359 + (10)(-0.039) = 1.969$.
  - Let the random number $r_2 = 0.617$ for $x_2^{(5)} = 2.978$. We calculate $\bar{\delta_2} = 0.013$.
  - New value is $x_2^{(5)} + (x_2^{(U)} - x_2^{(L)})\bar{\delta_2} = 2.978 + (10)(0.013) = 3.104$.
  - New solution is $(1.969, 3.104)^T$
- For solution '6', we do not perform mutation because $r > p_m$.
  - Copy it.
- We perform mutation on solution '7' because $r = 0.49 < p_m$. The other details are as follows: $r_1 = 0.217$ and $\bar{\delta_1} = -0.039$, and $r_2 = 0.617$ and $\bar{\delta_2} = 0.013$. New solution is $(-0.528, 0.564)^T$.
- For solution '8', we do not perform mutation because $r > p_m$.
  - Copy it.

Polynomial Mutation Operator: Hand Calculations

Solutions after mutation

Index	x_1	x_2	$f(x_1, x_2)$
1	0.284	1.856	315.568
2	-0.785	1.722	125.261
3	-2.355	-4.773	10655.925
4	2.212	3.009	357.154
5	1.969	3.104	60.744
6	-2.223	-2.342	5313.910
7	-0.528	0.564	10.515
8	-0.639	1.692	167.414

- Best solution in the initial population: $(-0.639, 1.692)^T$ with $f(x_1, x_2) = 167.414$.
- Best solution after crossover: $(-0.785, 1.691)^T$ with $f(x_1, x_2) = 118.471$.
- Best solution after mutation: $(-0.528, 0.564)^T$ with $f(x_1, x_2) = 10.515$.
- However, solution 1 got worse after mutation.

Observations on Mutation

- Mutation can create better or worse solution than parent solution.
- In selection, good solutions will be emphasized and bad solutions may get deleted.

Survivor or Elimination

- We choose better solutions for the next generation.
- Applying $(\mu + \lambda)-$strategy, meaning, combine parent and offspring populations and choose the best $N = 8$ solutions.

Parent population

Index	x_1	x_2	$f(x_1, x_2)$
1	2.212	3.009	357.154
2	−2.289	−2.396	5843.569
3	−2.393	−4.790	11066.800
4	−0.639	1.692	167.414
5	−3.168	0.706	8718.166
6	0.215	−2.350	574.796
7	−0.742	1.934	194.618
8	−4.563	4.791	25731.235

+

Sort $f(x_1,x_2)$

Offspring population

Index	x_1	x_2	$f(x_1, x_2)$
1	0.284	1.856	315.568
2	−0.785	1.722	125.261
3	−2.355	−4.773	10655.925
4	2.212	3.009	357.154
5	1.969	3.104	60.744
6	−2.223	−2.342	5313.910
7	−0.528	0.564	10.515
8	−0.639	1.692	167.414

Survivor or Elimination

- We choose better solutions for the next generation.
- Applying $(\mu + \lambda)-$strategy, meaning, combine parent and offspring populations and choose the best $N = 8$ solutions.

Parent population

Index	x_1	x_2	$f(x_1, x_2)$
1	2.212	3.009	357.154
2	−2.289	−2.396	5843.569
3	−2.393	−4.790	11066.800
× 4	−0.639	1.692	167.414
5	−3.168	0.706	8718.166
6	0.215	−2.350	574.796
7	−0.742	1.934	194.618
8	−4.563	4.791	25731.235

Same

Offspring population

Index	x_1	x_2	$f(x_1, x_2)$
1	0.284	1.856	315.568
2	−0.785	1.722	125.261
3	−2.355	−4.773	10655.925
4	2.212	3.009	357.154
5	1.969	3.104	60.744
6	−2.223	−2.342	5313.910
7	−0.528	0.564	10.515
8	−0.639	1.692	167.414 ✓

- Offspring 7, Offspring 5, Offspring 2, Parent 4, Offspring 8, Parent 7, Offspring 1

Survivor or Elimination

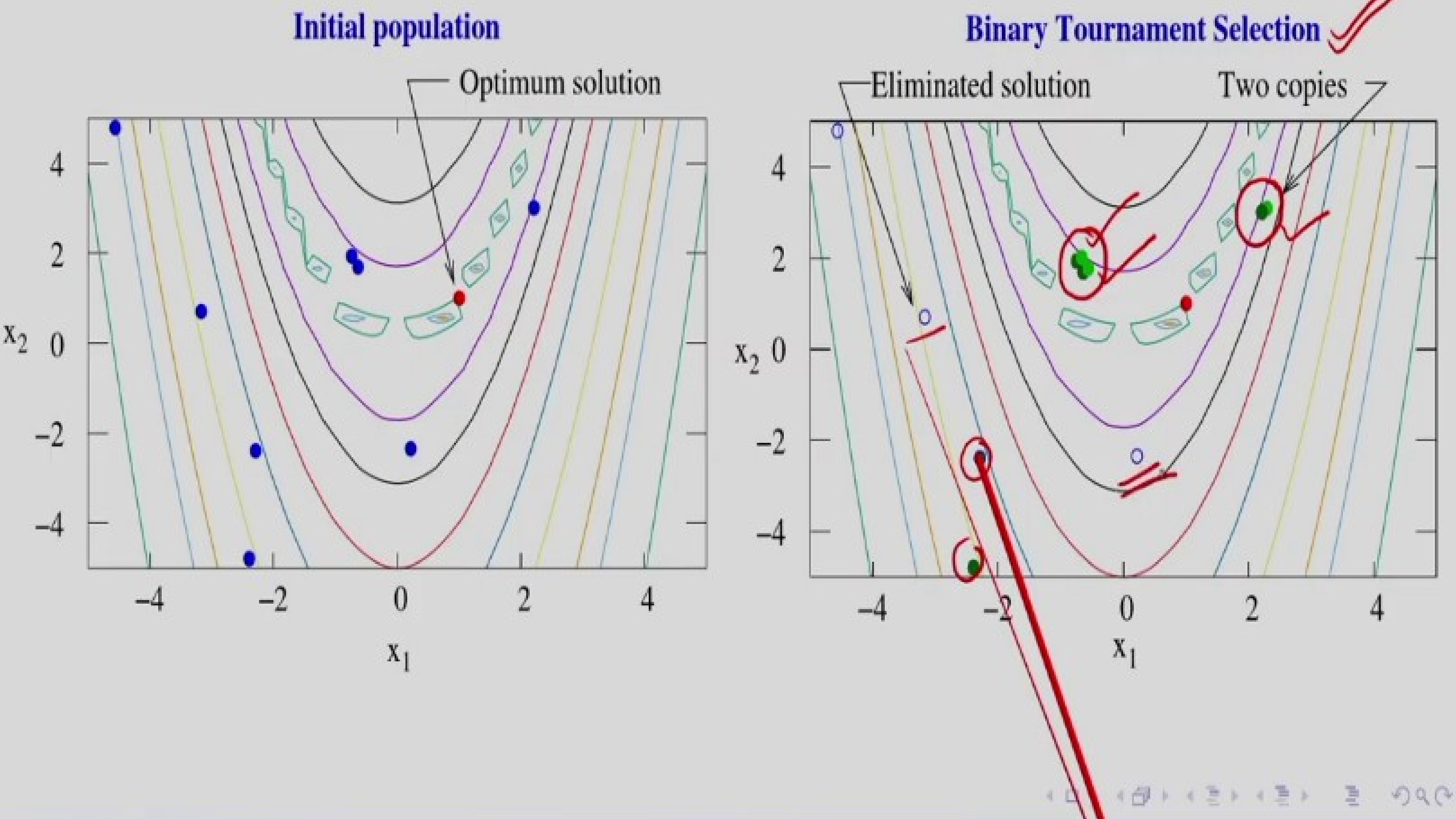
Next generation parent population

New index	x_1	x_2	$f(x_1, x_2)$
1	-0.528	0.564	10.515
2	1.969	3.104	60.744
3	-0.785	1.722	125.261
4	-0.639	1.692	167.414
5	-0.639	1.692	167.414
6	-0.742	1.934	194.618
7	0.284	1.856	315.568
8	2.212	3.009	357.154

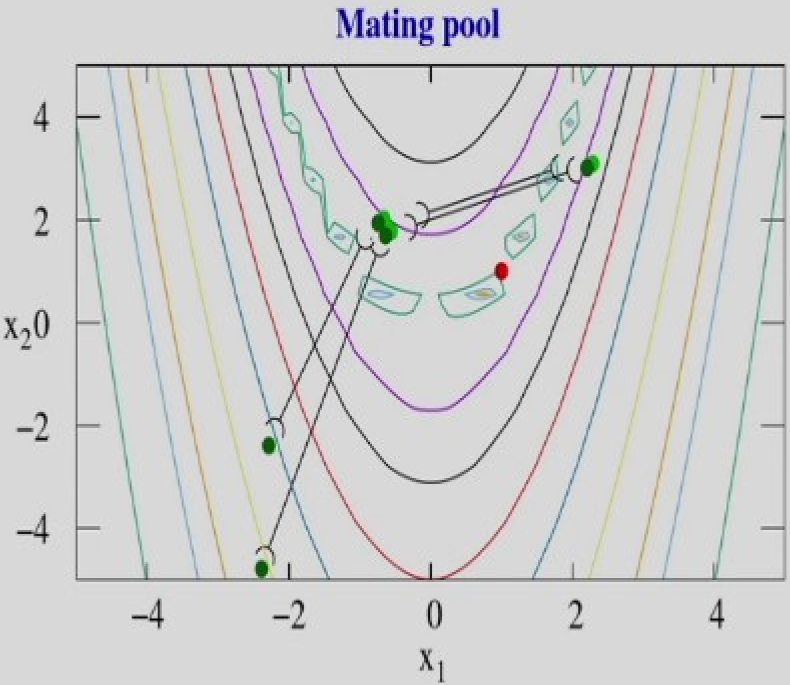
Generation

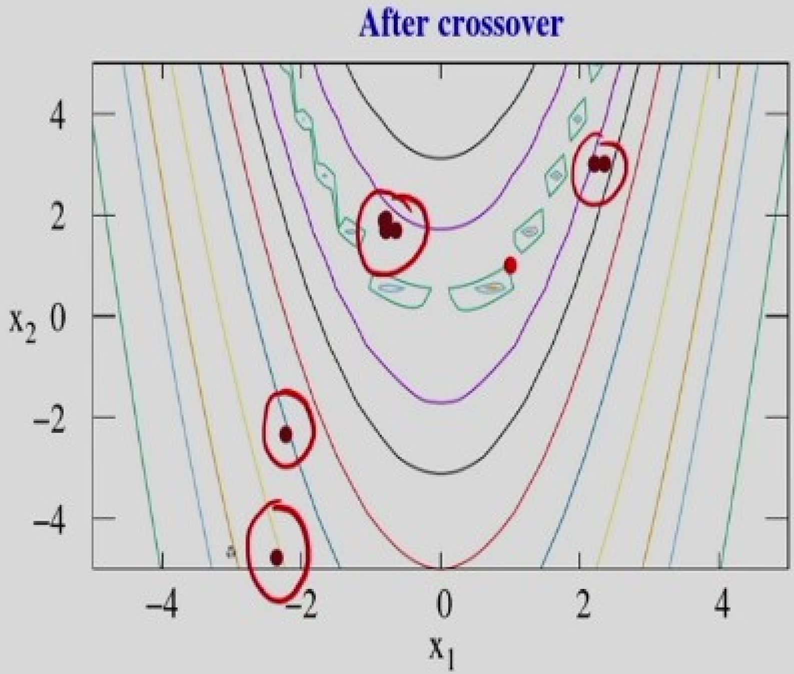
- First generation is over. Increase the counter by one, i.e., $t = t + 1 = 2$. Check termination condition.
- If the condition not met, start from the binary tournament selection operator followed by crossover and mutation operators. Finally, select best N solutions using survivor stage.

Graphical Example



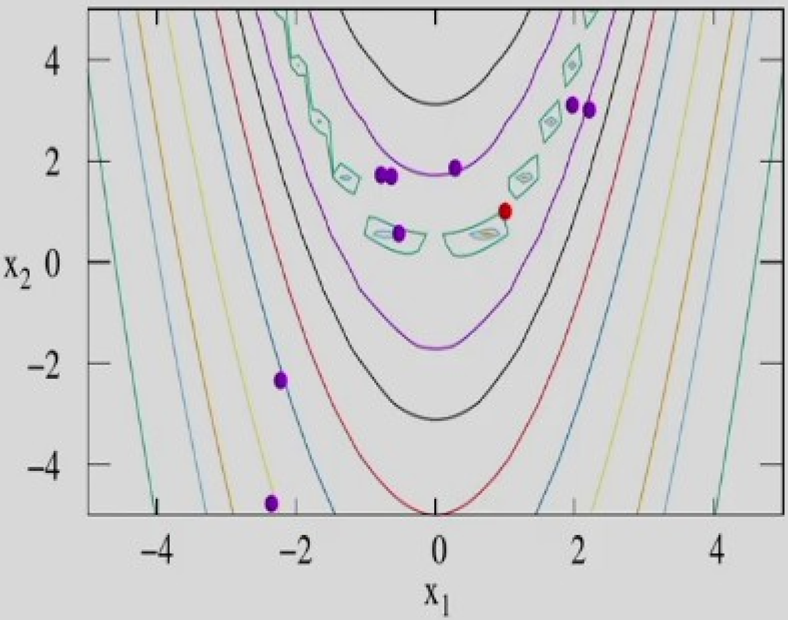
Graphical Example



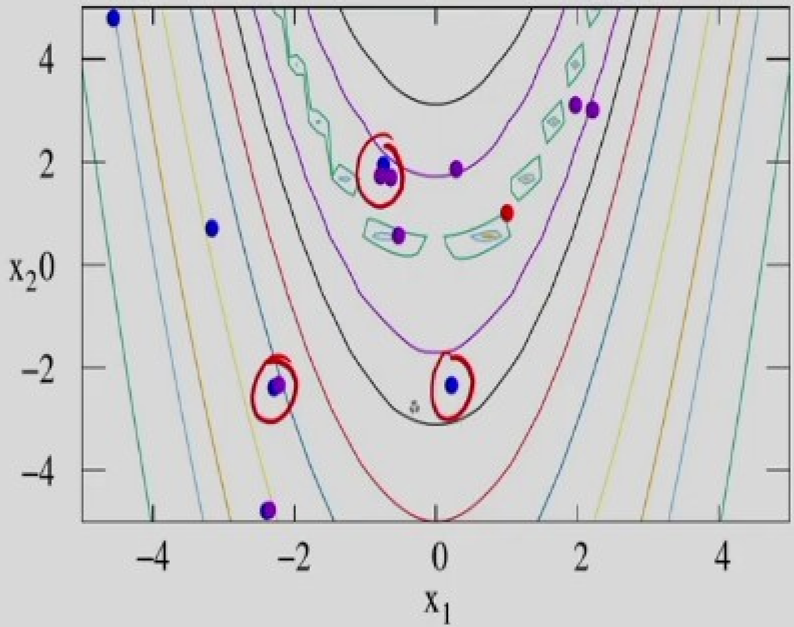


Graphical Example

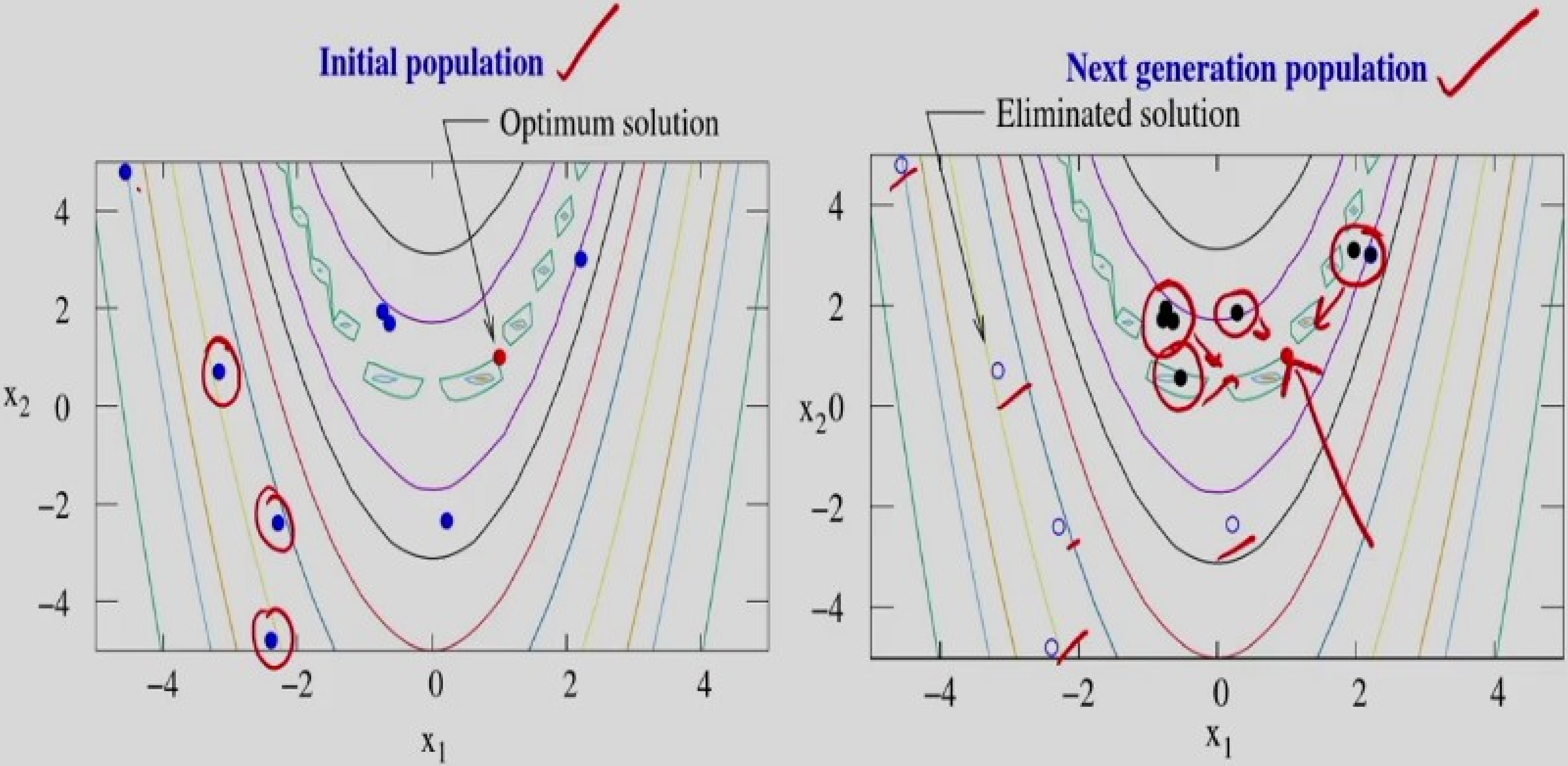
After mutation



Parent population + Offspring population



Graphical Example



Closure

- Limitations of BGA for solving problems with continuous search space
- EC Techniques for real-parameter optimization
- Generalized framework for RGA
- Working principles of RGA through an example
  - Initial population in which parameters are coded as real numbers
  - Selection operator: No change in its functioning was observed with respect to BGA because it used the fitness values.
  - Crossover operator: Properties of single-point crossover operator, SBX crossover operator

β — u_i

p_1, p_2

Closure

- Limitations of BGA for solving problems with continuous search space
- EC Techniques for real-parameter optimization
- Generalized framework for RGA
- Working principles of RGA through an example
  - Initial population in which parameters are coded as real numbers
  - Selection operator: No change in its functioning was observed with respect to BGA because it used the fitness values.
  - Crossover operator: Properties of single-point crossover operator, SBX crossover operator
  - Polynomial mutation operator
  - $(\mu + \lambda)$–strategy for survival stage
- Hand calculations for one generation
- Graphical illustration of RGA for one generation