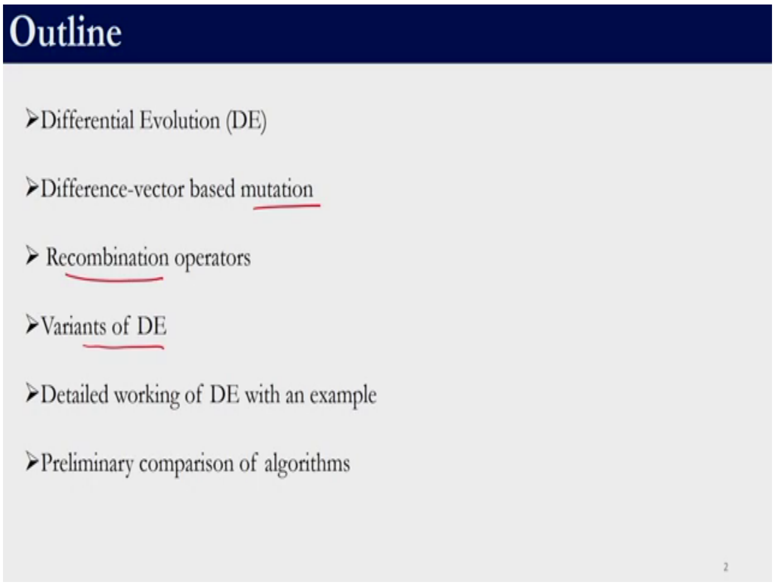
# Computer Aided Applied Single Objective Optimization

Dr. Prakash Kotecha
Department of Chemical Engineering
Indian Institute of Technology, Guwahati

## Lecture – 12
## Differential Evolution

Welcome back in this session we will be looking at Differential Evolution. Previously we have studied 2 other metaheuristic techniques teaching learning base optimization and particles swarm optimization. So, in this session we will be looking into differential evolution.

(Refer Slide Time: 00:46)



Outline

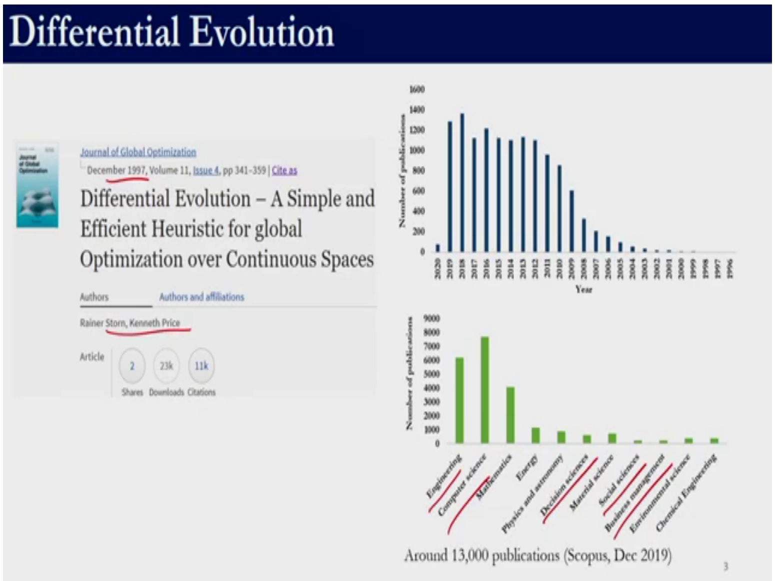
- Differential Evolution (DE)
- Difference-vector based mutation
- Recombination operators
- Variants of DE
- Detailed working of DE with an example
- Preliminary comparison of algorithms

2

So, this is going to be the outline of this session. First we will see what is differential evolution? Differential evolution, primarily consist of two operations; one is called as mutation and another one is called as recombination. So, we will be looking to those operator's mutation and recombinant operators. Depending upon minor variation we have few variants of

DE we look into that and as usual will take the example of sphere function and we will look into the detailed working of DE and then we will do a preliminary comparison of the 3 metaheuristic techniques that we have studied so far.

(Refer Slide Time: 01:21)



## Differential Evolution

Journal of Global Optimization

December 1997, Volume 11, Issue 4, pp 341–359 | Cite as

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces

Authors Authors and affiliations

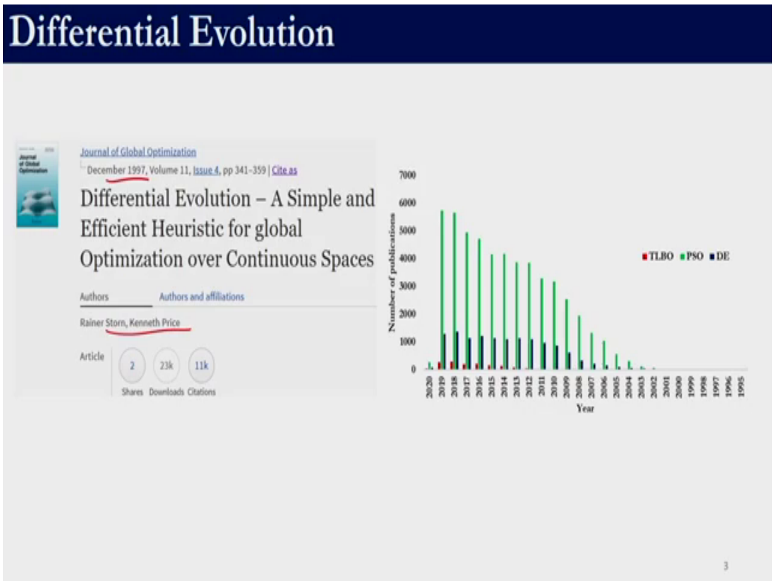
Rainer Storn, Kenneth Price

Article 2 Shares 23k Downloads 11k Citations

Around 13,000 publications (Scopus, Dec 2019)

So, Differential Evolution was proposed in 1197 by Storn and Price. Ever since then it has been widely used by researchers. So, here as we can see as time has progressed differential evolution has been used by a large number of research community. Since metaheuristic techniques are independent of the domain of the problem, it has been used in engineering, computer science, decision sciences, social sciences, business management and various. Here what we have showing is just a sample of few selected areas there are various other areas in which differential evolution has been applied.

(Refer Slide Time: 01:56)



Differential Evolution

Journal of Global Optimization

December 1997, Volume 11, Issue 4, pp 341-359 | Cite as

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces

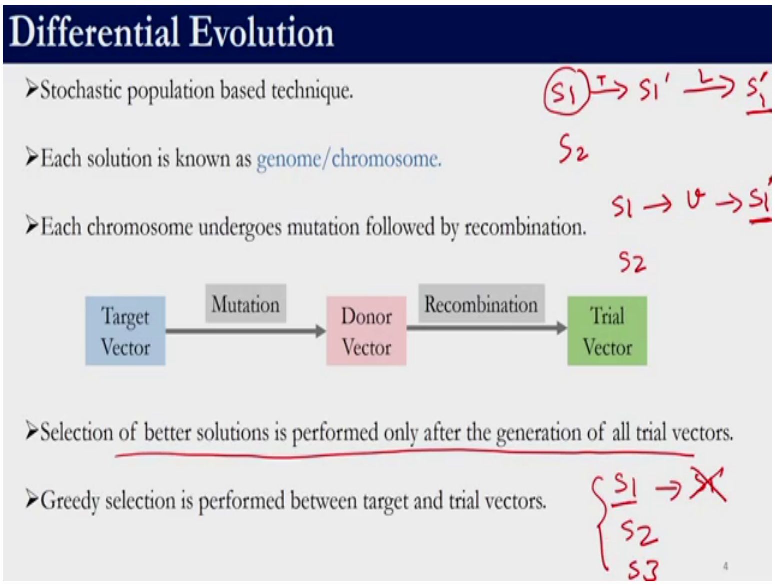
Authors | Authors and affiliations

Rainer Storn, Kenneth Price

Article 2 Shares 23k Downloads 11k Citations

So, this shows the use of the three techniques that, we have come across what is teaching learning based optimization. The second is particle swarm optimization and the third one is differential evolution. So, TLBO as you know was a came up into 2011. So, obviously, it has been not as widely used as particle swarm optimization and differential evolution, but between particles swarm optimization and differential evolution particle swarm optimization seems to have attracted more attention than differential evolution. Just like particle swarm optimization and teaching learning based optimization differential evolution is say stochastic population technique.

(Refer Slide Time: 02:37)



## Differential Evolution

- Stochastic population based technique.
- Each solution is known as genome/chromosome.
- Each chromosome undergoes mutation followed by recombination.

Target Vector → Mutation → Donor Vector → Recombination → Trial Vector

- Selection of better solutions is performed only after the generation of all trial vectors.
- Greedy selection is performed between target and trial vectors.

$\boxed{S_1} \xrightarrow{T} S_1' \xrightarrow{L} S_1'$

$S_2$

$S_1 \rightarrow U \rightarrow S_1'$

$S_2$

$\left\{ \begin{array}{l} S_1 \rightarrow \cancel{X} \\ S_2 \\ S_3 \end{array} \right.$

4

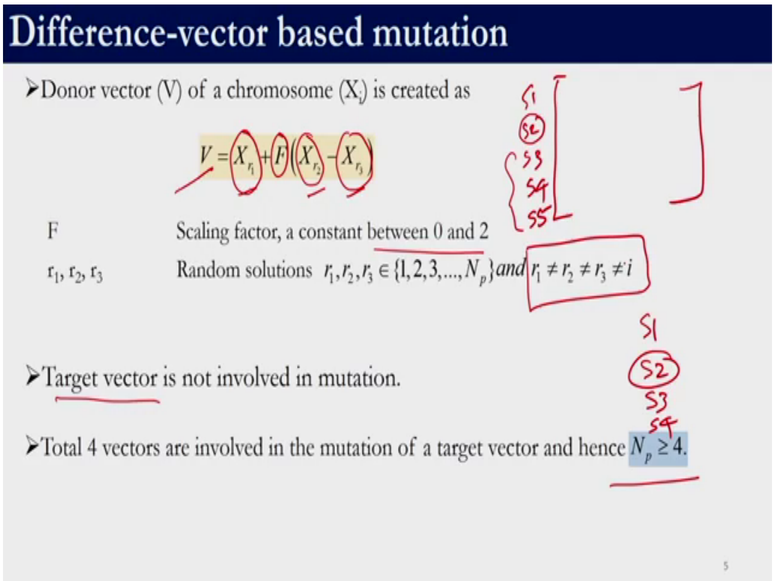
In differential evolution each solution is known as genome or chromosome. So, each chromosome undergoes mutation followed by recombination. So, those are the two operator's mutation and recombinant operators. So, there are few terminologies that we need to keep in mind. So, one is target vector. So, target vector is the solution which is undergoing evolution. So, the target vector is used in mutation to generate a donor vector and the donor vector undergoes recombination to obtain that trial vector.

So, between the trial vector and target vector a greedy selection is employed right. So, whichever solution is better will survive for the next generation right. So, here one thing we need to keep in mind is that selection of better solution is performed only after the generation of all trial vectors. So, for example, in TLBO if we had solution one let us say it underwent teacher phase and we obtain solution S 1 prime and it underwent learner phase. Let us say in learner phase we did not obtain a good solution. So, we retained  S 1 prime.

So, when we are working with solution 2, if we randomly selected solution 1 then we will be working with this updated member right not with this initial member. That is what was happening in teaching learning based optimization. Similarly, in particle swarm optimization, we had a solution 1 we generated its velocity, updated velocity then we updated its position let say  S 1 prime and when we were working with solution 2.

If we had to randomly select a solution, it is possible that we could have we would have picked up S 1 prime because that is updated. As soon as we get a solution we update a population that is not the case in differential evolution. So, here we will have if we have solution 1 we may generate  S 1 prime right, but when we are generating S 2 or  S 3. We if required we will be using S 1 right and not S 1 prime right. This updation will happen after all the solutions have undergone the mutation and recombination.

(Refer Slide Time: 04:45)



## Difference-vector based mutation

- Donor vector (V) of a chromosome ($X_i$) is created as

$$V = X_{r_1} + F(X_{r_2} - X_{r_3})$$

| | |
|---|---|
| F | Scaling factor, a constant between 0 and 2 |
| $r_1, r_2, r_3$ | Random solutions $r_1, r_2, r_3 \in \{1, 2, 3, \dots, N_p\}$ and $r_1 \neq r_2 \neq r_3 \neq i$ |

- Target vector is not involved in mutation.
- Total 4 vectors are involved in the mutation of a target vector and hence $N_p \geq 4$.

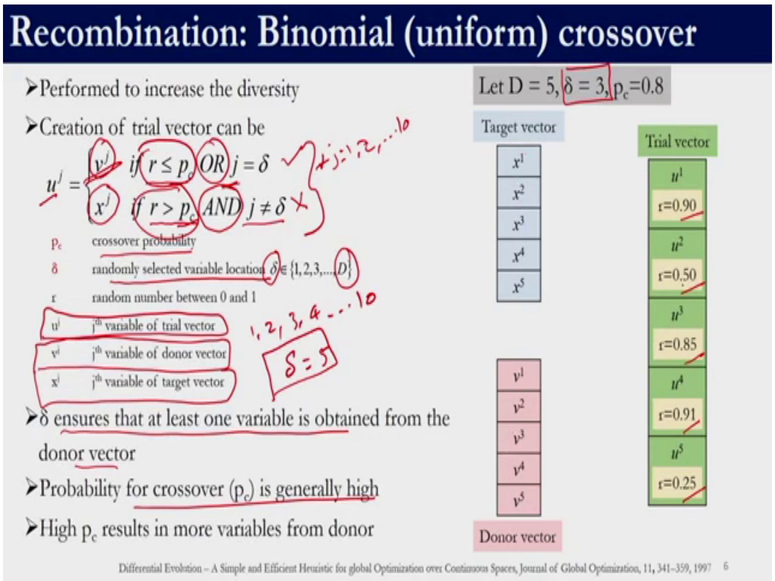
S1
S2
S3
S4
S5
S1
S2
S3
S4
5

Mutation operator is a very simple operation here. In this mutation operation we need to select three random solutions right. So, again we have a population of solutions. Let us say S 1 S 2 S 3 S 4 and so on. So, we are supposed to select 3 solutions randomly and they should not be identical. So, for example, if let say we select S 3 S 4 S 5 right. So, we then we need to apply this equation, that the first solution plus a scaling factor which is a user defined parameter. So, it is usually between 0 and 2 the second solution minus the third solution. So, that will give us our donor vector right.

So, in this equation if we see that the solution for which we are generating the donor vector is not at all involved. So, the target vector is not involved. So, when we were generating donor vector for S 1 3 other random solutions help us to generate the donor vector for S 1. So, when we are generating a donor vector for S 2 3 other random solution help us to generate donor vector, for S 2 S 1 and S 2 are not involved when their donor vectors are generated.

So, here if we see that our population size has to be at least greater than or equal to 4. Because let us say we are working with solution 1 right, then we need 3 random solutions. So, that has to be 3 different solutions. So, at least this, so, in this case when we are working with S 2 the 3 other random solutions would be S 1 S 3 S 4. So, here we need to keep in mind that we need 3 random solutions and they should not be equal right and the target vector itself does not participate in mutation.

(Refer Slide Time: 06:29)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \quad ✓ \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \quad ✗ \end{cases} \quad \forall j : 1, 2, \ldots 10$$

| | |
|---|---|
| p<sub></sub>$p_c$ | crossover probability |
| δ | randomly selected variable location $\delta \in \{1,2,3,\ldots D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 . . . 10

$\delta = 5$

- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, δ = 3, $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

Once we have completed mutation the next step is to apply recombination. So, recombination is applied to increase the diversity of the population right. So, here let us first understand the terminologies right. So, here u j indicates the j th variable of the trial vector right, v j indicates the j th variable of the donor vector and x j indicates the j th variable of the target vector. This crossover which we are studying is known as binomial crossover or uniform crossover right.

So, in this the target vector takes part in the recombination operator right, whereas that was not the case in the mutation operator. So, here if we see that the trial vector is going to have variables which are coming either from donor vector or target vector. We are not going to calculate anything as such right. It is just that we need to pick a variable either from the donor vector or from the target vector. These are the 2 conditions which will help us to decide if a variable has to be picked from the target vector or from the trial vector right.

So, we have this user defined parameter known as crossover probability p c. So, in addition to crossover probability we will have to randomly select a variable within the number of decision variable. So, if capital D denotes the total number of decision variable. We will have to select an integer between 1 and D. So, let say if we have 10 variables 1 2 3 4 and so on up to 10, then delta is to be generated randomly for every solution. So, let us say delta is 5 right.

So, this j will run from 1 to 10 and delta. So, it is a randomly generated integer for every solution that is undergoing this crossover operator right. So, if this condition satisfies that if r is greater than the crossover probability and j is not equal to delta right. Then in that case the target vectors value is taken into the trial vector, else if this condition is satisfied that if the random number is less than equal to p c. Then the mutant vectors value is taken into the trail vector right.

So, here if we see here there is an OR condition and here there is an AND condition right. So, if you see for one variable j which is equal to delta. So, here we are going to modify each and every variable right. So, for the j th variable right. So, for example, del is equal to 5. So, this j is going to vary from this equation holds for j equal to 1 to all the way up to the number of decision variables right.

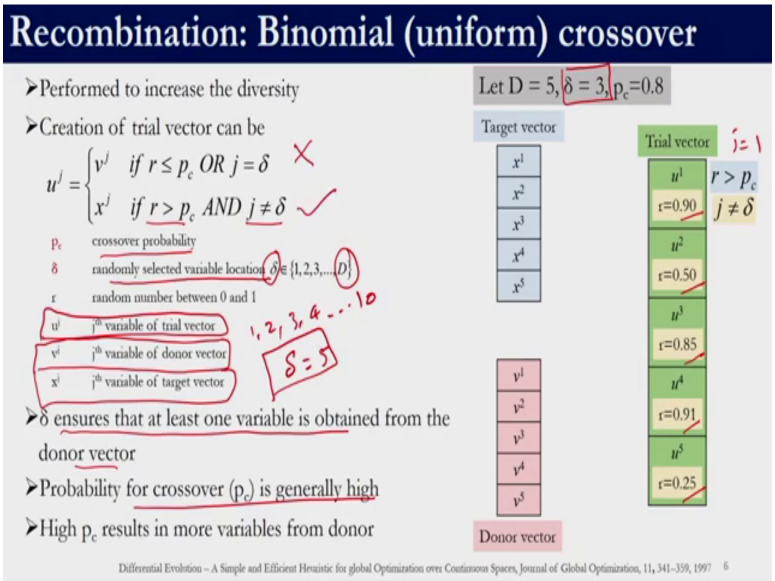
So, in one case we are going to meet this condition where j is going to be equal to delta right. So, in that condition no matter what is the crossover probability this equation would not be used, but this equation would be used right over here we have an OR condition. So, irrespective of the crossover probability the value is going to come from the mutant vector right for the variable j is equal to delta.

So, that is what we discussed that the del ensures that at least one variable is obtained from the donor vector and probability for crossover is generally high. If it is high value then most of the random numbers which you generate will be less than p c. Most of the values for the various decision variables will be coming from this mutant vector and not from the target vector.

So, let us see an example right. So, let say there are 5 decision variables and we have generated a random number delta is equal to 3 and let the crossover probability be 0.8. So, let us assume this is our target vector x 1 x 2 x 3 x 4 x 5 is our target vector. Whereas, our donor vector is given by v 1 v 2 v 3 v 4 and v 5 right. So, now, the question is how do we generate our trial vector right? So, since there are 5 variables, we need to generate 5 random numbers for the first variable the random number is 0.9, second variable it is 0.5, third variable it is 0.85, fourth variable it is 0.91 and fifth variable it is 0.25.

So, we have generated 5 random numbers corresponding to 5 decision variable delta value was randomly chosen right between 1 to the number of decision variables. So, in this case it is 3. So, this u 1 to u 5 right, these values are not known x 1 x 2 x 3 x 4 x 5 are known because that is the target vector we applied the mutation operator. So, we also know v 1 v 2 v 3 v 4 v 5. Now the question is which of these values from the target vector or the donor vector would come into the trial vector. So, that is again governed by this equation which we have given over here.

(Refer Slide Time: 11:22)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \quad ✗ \\ x^j & \text{if } \underline{r > p_c} \text{ AND } \underline{j \ne \delta} \quad ✓ \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\ldots,D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 ... 10

$\boxed{\delta = 5}$

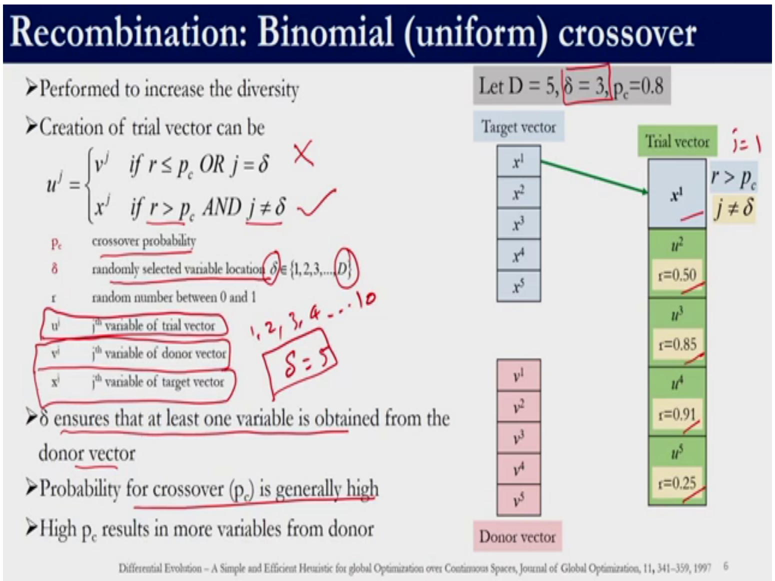
- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, $\boxed{\delta = 3,}$ $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

So, let us see for the first variable what would happen. So, for the first variable the crossover probability is 0.8 right, where as the random number is 0.9. So, 0.9 is greater than 0.8. So, this condition is satisfied right. And j is not equal to delta. So, j for this variable is 1 because it is the first variable. So, the second condition is satisfied right and the first condition is not satisfied, 1 is not equal to 3 right. So, this OR condition is not satisfied and the random number is also not less than the crossover probability right. So, this condition fails and this condition is satisfied for the first variable.

(Refer Slide Time: 12:02)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & if\ r \le p_c\ OR\ j = \delta \quad \times \\ x^j & if\ r > p_c\ AND\ j \ne \delta \quad ✓ \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| δ | randomly selected variable location $\delta \in \{1,2,3,\ldots D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 ... 10

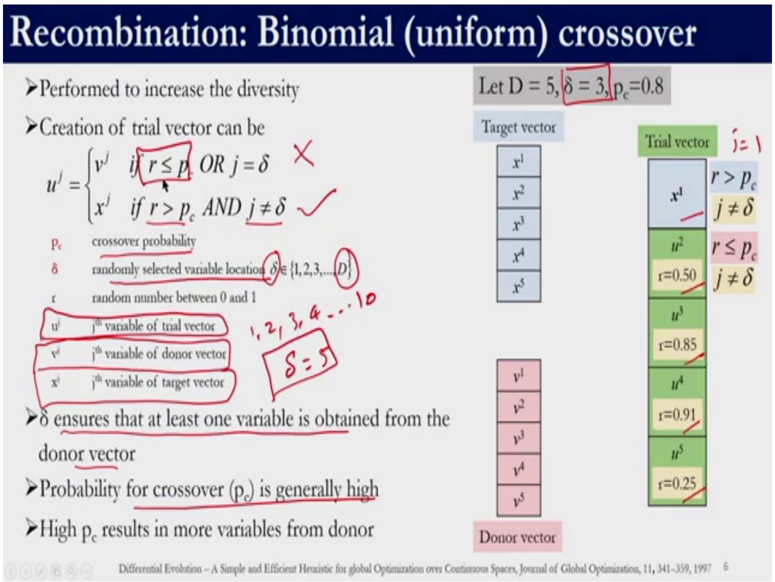
$\delta = 5$

- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

So, x 1 value is to be taken as the first decision variable in the trial vector right.

(Refer Slide Time: 12:09)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } \boxed{r \le p_c} \text{ OR } j = \delta \quad \times \\ x^j & \text{if } \underline{r > p_c} \text{ AND } \underline{j \ne \delta} \quad \checkmark \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\ldots,D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 . . . 10

$\boxed{\delta = 5}$

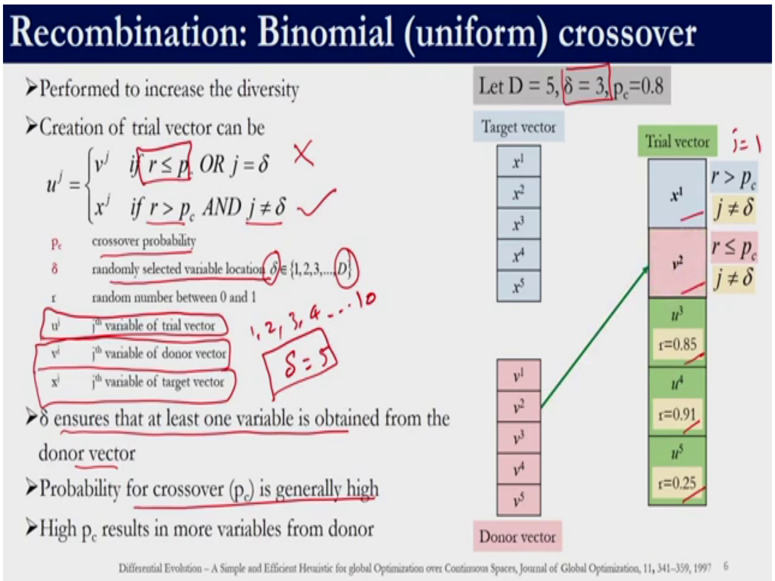
- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, $\boxed{\delta = 3,}$ $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

In the second case random number is 0.5. Our crossover probability is 0.8 right. So, in this case what will happen is r is less than p c right. So, for the second case r is less than p c this condition is satisfied right. So, it will come from the donor vector.

(Refer Slide Time: 12:26)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } \boxed{r \le p_c} \text{ OR } j = \delta \quad \times \\ x^j & \text{if } \underline{r > p_c} \text{ AND } \underline{j \ne \delta} \quad \checkmark \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1, 2, 3, \ldots, D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 . . . 10

$\boxed{\delta = 5}$

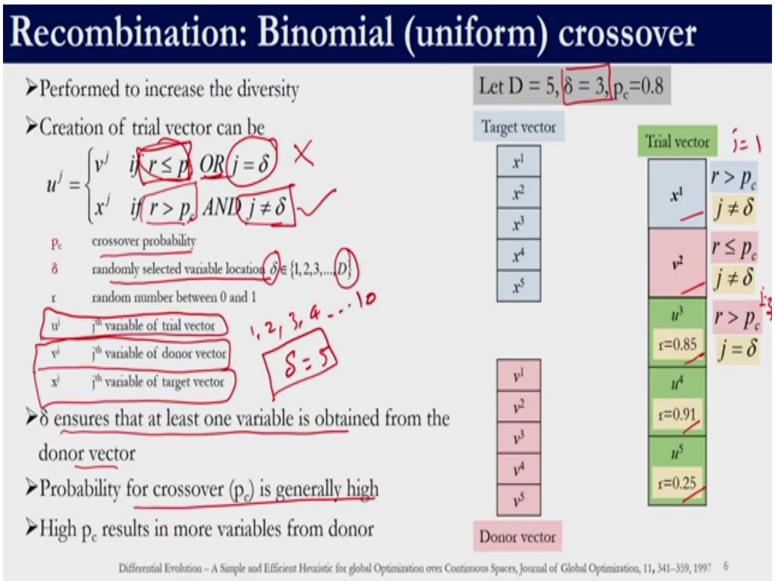
- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, $\boxed{\delta = 3,}$ $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

So, the value is going to come from the donor vector.

(Refer Slide Time: 12:29)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } \boxed{r \le p_c} \text{ OR } (j = \delta) \quad \times \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } \boxed{j \ne \delta} \quad ✓ \end{cases}$$

$p_c$ crossover probability

$\delta$ randomly selected variable location $\delta \in \{1, 2, 3, \ldots, D\}$

r random number between 0 and 1

$u^j$ $j^{th}$ variable of trial vector

$v^j$ $j^{th}$ variable of donor vector

$x^j$ $j^{th}$ variable of target vector

1, 2, 3, 4 . . . 10

$\boxed{\delta = 5}$

- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

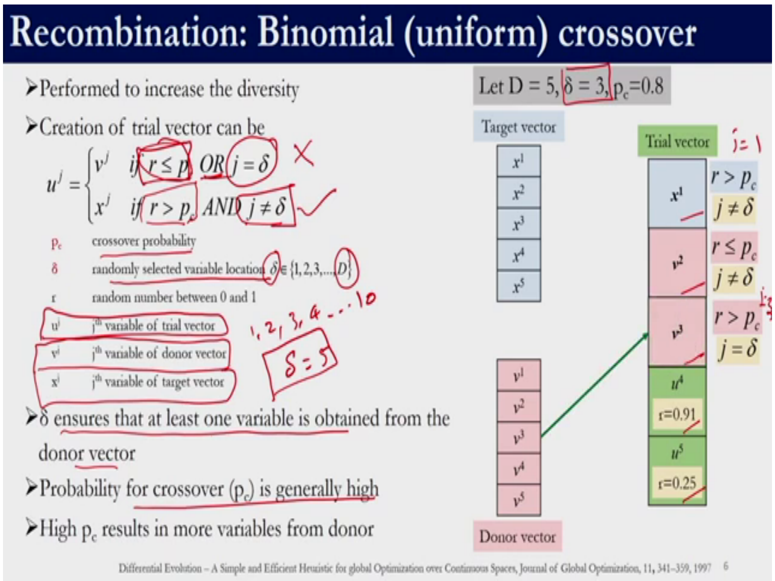
Let D = 5, δ = 3, $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

So, for the third case the random number is 0.85, where as the crossover probability is 0.8. So, 0.85 is greater than 0.8 first part of the third condition is satisfied, 0.85 is greater than 0.8 right that is satisfied. However, j for this is 3 for this variable j is 3 because it is the third variable. So, this second condition fails right that j should not be equal to delta whereas, now delta is also 3 which we had randomly selected initially and the value of j is also 3. Second condition fails despite the random number greater than being p c right.

So, in that case the first equation is to be used in the first equation remember. This part is again not satisfied r is not less than equal to p c right, but this is a OR condition right. So, since this is a OR condition this part of the equation is satisfied that j is equal to delta. So, this value comes from the mutant right.

(Refer Slide Time: 13:27)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

×

✓

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\dots D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 . . . 10

$\delta = 5$

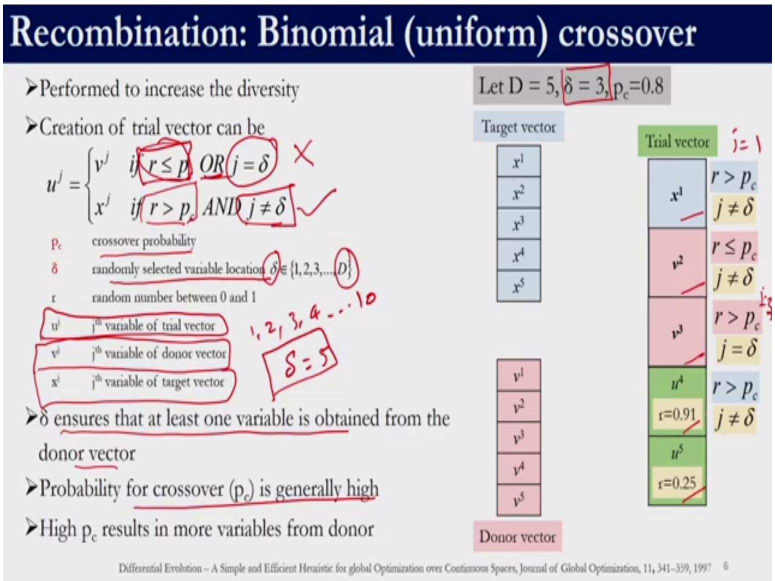
- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, δ = 3, $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

Similarly, we can see for the fourth variable.

(Refer Slide Time: 13:32)



### Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } \boxed{r \le p_c} \text{ OR } j = \delta \quad \times \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } \boxed{j \ne \delta} \quad \checkmark \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\ldots,D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | jth variable of trial vector |
| $v^j$ | jth variable of donor vector |
| $x^j$ | jth variable of target vector |

1, 2, 3, 4 . . . 10

$\boxed{\delta = 5}$

- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, $\boxed{\delta = 3,}$ $p_c$=0.8

Target vector

Trial vector

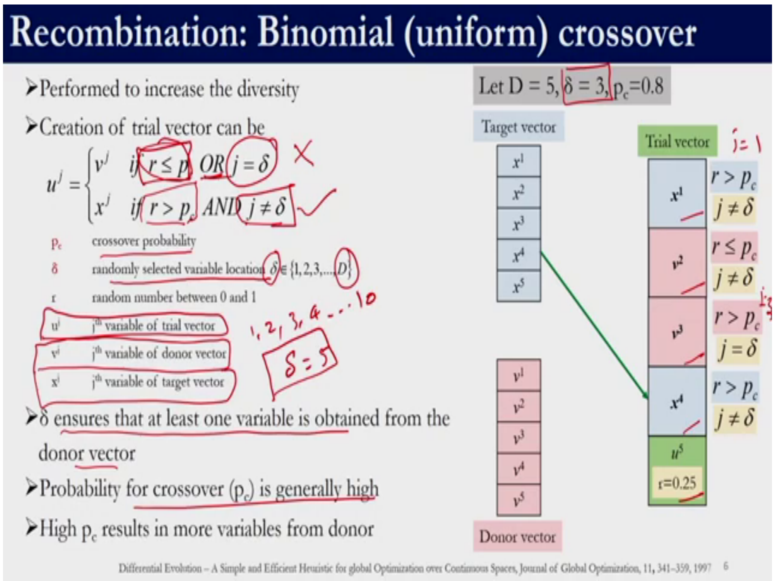
$j = 1$

Donor vector

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

So, j is not equal to delta and 0.91 is greater than 0.8.

(Refer Slide Time: 13:39)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } \boxed{r \le p_c} \text{ OR } j = \delta \quad \times \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } \boxed{j \ne \delta} \quad \checkmark \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\ldots D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

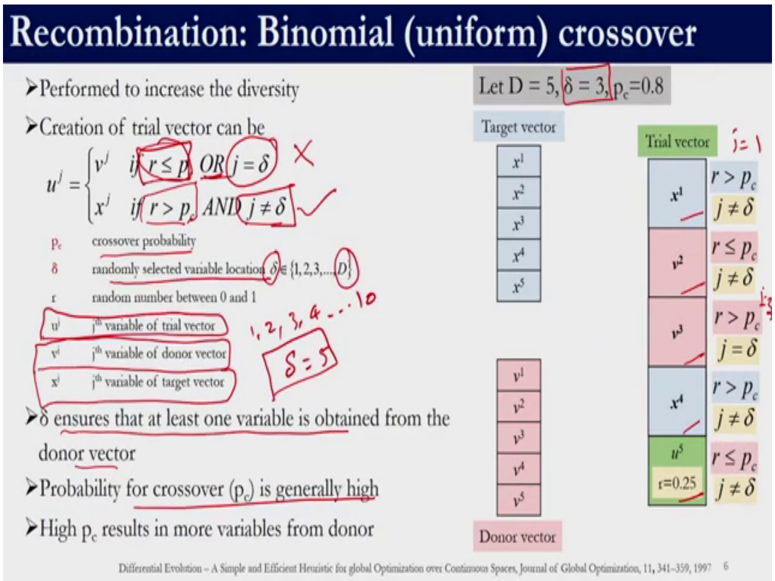
- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, $\delta = 3$, $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

So, it is going to come from the target vector. So, the random number for the fifth variable is 0.25 right. So, it satisfies the first equation right 0.25 is less than 0.8 right.

(Refer Slide Time: 13:51)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & if\ \boxed{r \le p_c}\ OR\ (j = \delta) \quad \times \\ x^j & if\ (r > p_c)\ AND\ \boxed{j \ne \delta} \quad \checkmark \end{cases}$$

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\ldots,D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

1, 2, 3, 4 . . . 10

$\boxed{\delta = 5}$

- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

Let D = 5, $\boxed{\delta = 3,}$ $p_c$=0.8

Target vector: $x^1$, $x^2$, $x^3$, $x^4$, $x^5$

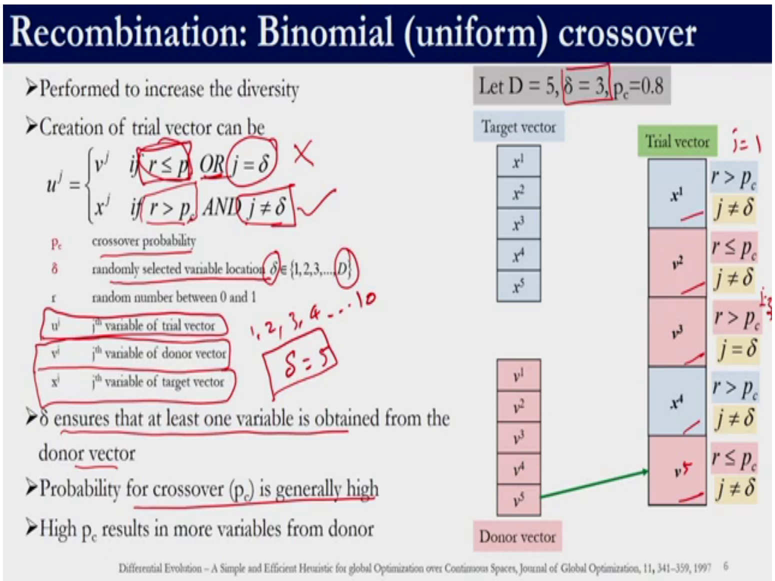
Donor vector: $v^1$, $v^2$, $v^3$, $v^4$, $v^5$

Trial vector j = 1

| | |
|---|---|
| $x^1$ | $r > p_c$<br>$j \ne \delta$ |
| $v^2$ | $r \le p_c$<br>$j \ne \delta$ |
| $v^3$ | $r > p_c$<br>$j = \delta$ |
| $x^4$ | $r > p_c$<br>$j \ne \delta$ |
| $u^5$<br>r=0.25 | $r \le p_c$<br>$j \ne \delta$ |

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

(Refer Slide Time: 13:52)



## Recombination: Binomial (uniform) crossover

- Performed to increase the diversity
- Creation of trial vector can be

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

p

| | |
|---|---|
| $p_c$ | crossover probability |
| $\delta$ | randomly selected variable location $\delta \in \{1,2,3,\ldots,D\}$ |
| r | random number between 0 and 1 |
| $u^j$ | $j^{th}$ variable of trial vector |
| $v^j$ | $j^{th}$ variable of donor vector |
| $x^j$ | $j^{th}$ variable of target vector |

- δ ensures that at least one variable is obtained from the donor vector
- Probability for crossover ($p_c$) is generally high
- High $p_c$ results in more variables from donor

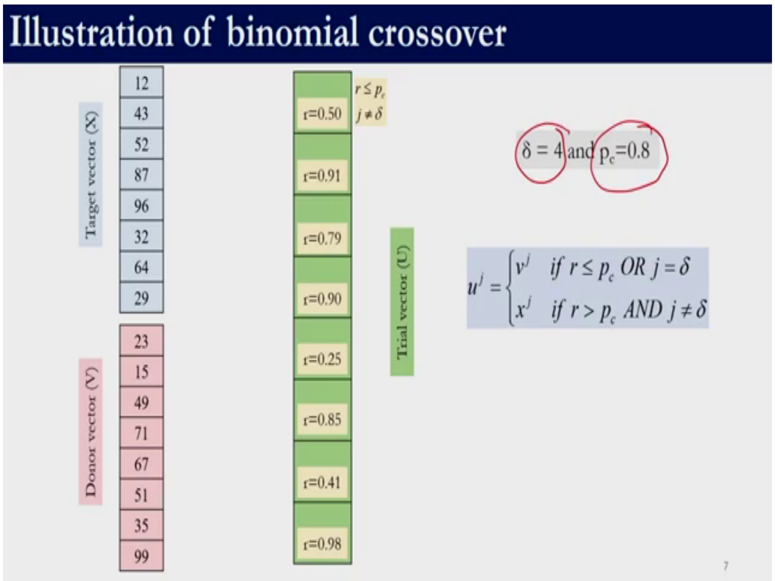
Let D = 5, δ = 3, $p_c$=0.8

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341–359, 1997 6

X
✓
1, 2, 3, 4 . . . 10
δ = 5
Target vector
$x^1$
$x^2$
$x^3$
$x^4$
$x^5$
Donor vector
$v^1$
$v^2$
$v^3$
$v^4$
$v^5$
Trial vector
j = 1
$x^1$ $r > p_c$ $j \ne \delta$
$v^2$ $r \le p_c$ $j \ne \delta$
$v^3$ $r > p_c$ $j = \delta$
$x^4$ $r > p_c$ $j \ne \delta$
$v^5$ $r \le p_c$ $j \ne \delta$

So, it will come from the mutant vector. So, this should be v 5 right. So, this is how that trial vector is generated right for the trial vector, the values we are going to take either from the target vector or from the donor vector. Depending upon the crossover probability and the delta value which we have randomly chosen we will be deciding whether the value comes from the trial vector or from the target vector.

(Refer Slide Time: 14:18)



## Illustration of binomial crossover

Target vector (X)

12
43
52
87
96
32
64
29

Donor vector (V)

23
15
49
71
67
51
35
99

Trial vector (U)

r=0.50
r=0.91
r=0.79
r=0.90
r=0.25
r=0.85
r=0.41
r=0.98

$r \le p_c$
$j \ne \delta$

$\delta = 4$ and $p_c=0.8$

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

7

And so, this is another example right. So, here in this case instead of x 1 x 2 x 3 x 4 x 5 we have taken the value. So, this is what is going to happen when we are working with a problem right. So, this is the target vector and this is the donor vector right. So, the target vector, donor vector is given the trial vector is what we are trying to find out and we have generated random number between 0 and 1 for all the variables right and the delta value which you have generated randomly is 4 and the crossover probability is 0.8. So, if we apply this equation. So, in the first case r is less than equal to p c and j is not equal to delta j for the first variable is 1 right.

(Refer Slide Time: 15:01)

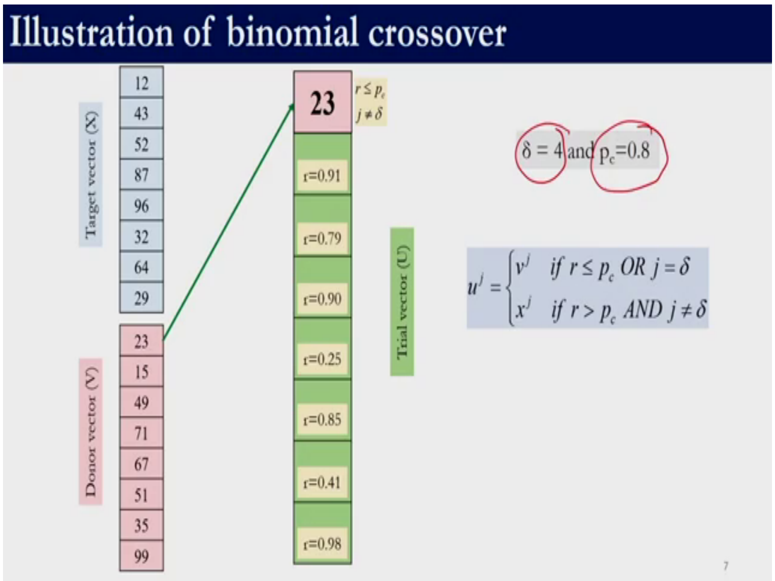
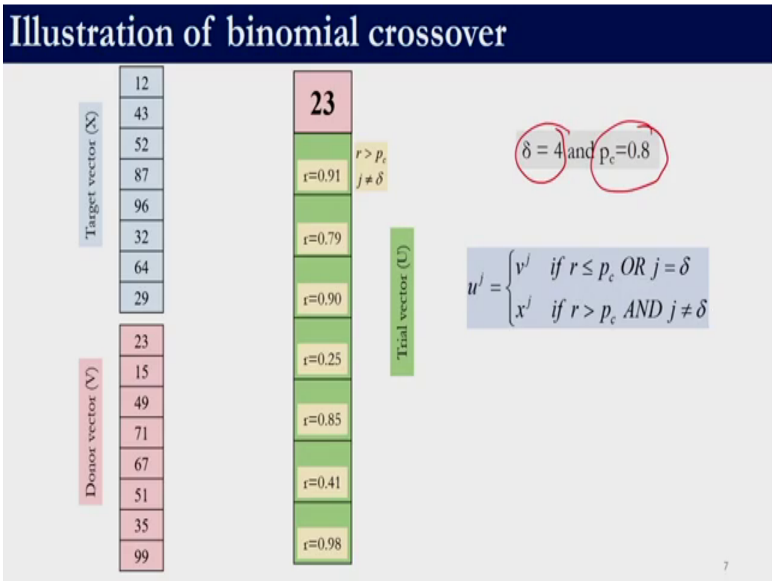


Illustration of binomial crossover

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

So, the value is going to come from mutant vector or the donor vector right. So, the donor vector has a value of 23. So, that value 23 is to be taken over here.

(Refer Slide Time: 15:07)



## Illustration of binomial crossover

Target vector (X): 12, 43, 52, 87, 96, 32, 64, 29

Donor vector (V): 23, 15, 49, 71, 67, 51, 35, 99

Trial vector (U): 23, r=0.91, r=0.79, r=0.90, r=0.25, r=0.85, r=0.41, r=0.98

$r > p_c$
$j \neq \delta$

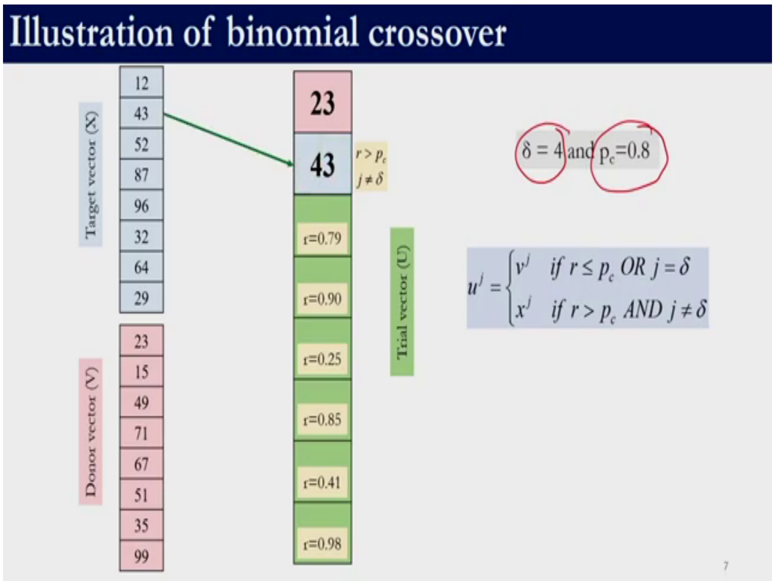
$\delta = 4$ and $p_c=0.8$

$$u^j = \begin{cases} v^j & if\ r \le p_c\ OR\ j = \delta \\ x^j & if\ r > p_c\ AND\ j \neq \delta \end{cases}$$

7

In the second case r is greater than p c and j is not equal to delta because p c is 0.8, r value which you have taken is 0.91. So, in this case the value the second condition is satisfied right.

(Refer Slide Time: 15:25)



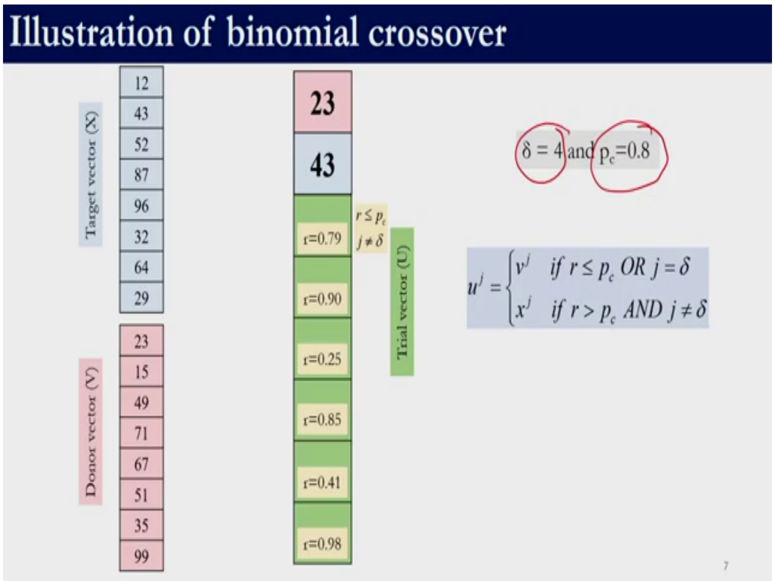
## Illustration of binomial crossover

δ = 4 and $p_c$=0.8

$$u^j = \begin{cases} v^j & if\ r \le p_c\ OR\ j = \delta \\ x^j & if\ r > p_c\ AND\ j \neq \delta \end{cases}$$

So, if the second condition is satisfied the value is to be taken from the target vector. The value of the second decision variable is 43.

(Refer Slide Time: 15:29)



## Illustration of binomial crossover

Target vector (X)

| Target vector (X) | Donor vector (V) | Trial vector (U) |
|---|---|---|
| 12 | 23 | 23 |
| 43 | 15 | 43 |
| 52 | 49 | r=0.79 |
| 87 | 71 | r=0.90 |
| 96 | 67 | r=0.25 |
| 32 | 51 | r=0.85 |
| 64 | 35 | r=0.41 |
| 29 | 99 | r=0.98 |

$r \le p_c$
$j \ne \delta$

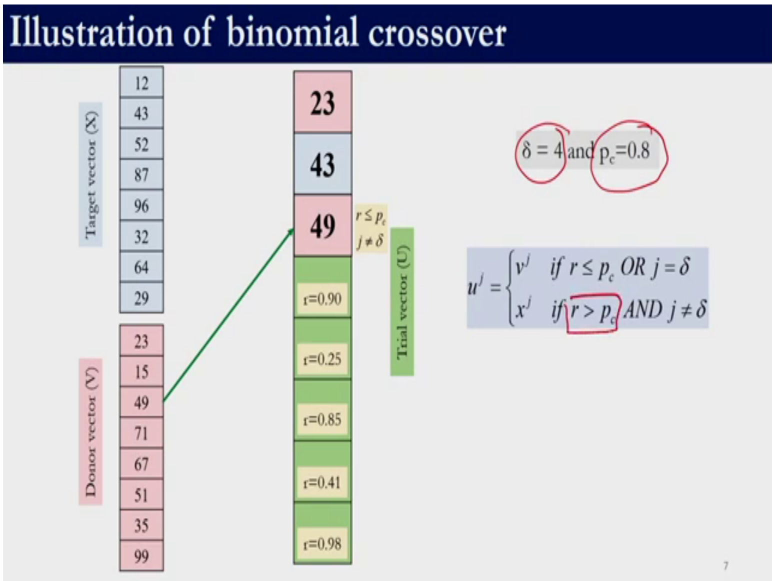
$\delta = 4$ and $p_c = 0.8$

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

7

So, for the third variable r is less than equal to p c because 0.79 is less than 0.8. So, the third variable is going to come from the donor vector.

(Refer Slide Time: 15:38)



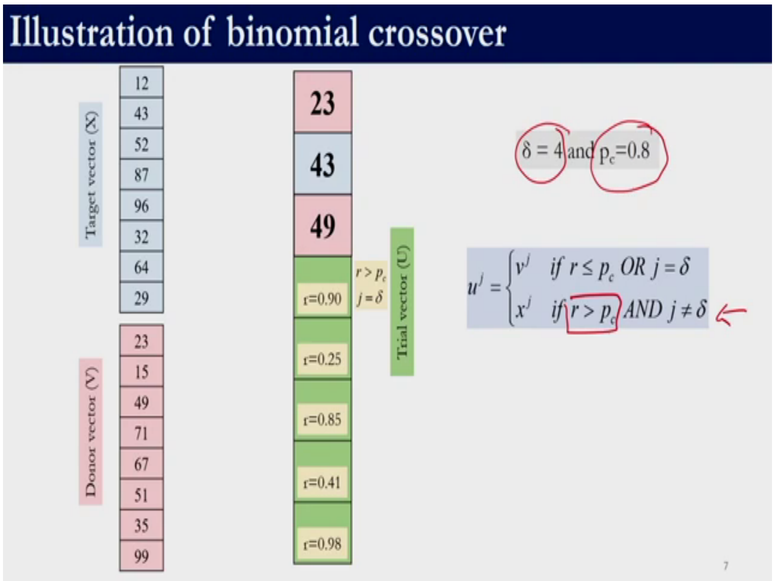
## Illustration of binomial crossover

$\delta = 4$ and $p_c = 0.8$

$$u^j = \begin{cases} v^j & if\ r \le p_c\ OR\ j = \delta \\ x^j & if\ \boxed{r > p_c}\ AND\ j \ne \delta \end{cases}$$

For the fourth variable we have 0.9 right. So, this condition is satisfied 0.9 is greater than 0.8, but this condition is not satisfied that j is equal to delta.

(Refer Slide Time: 15:50)



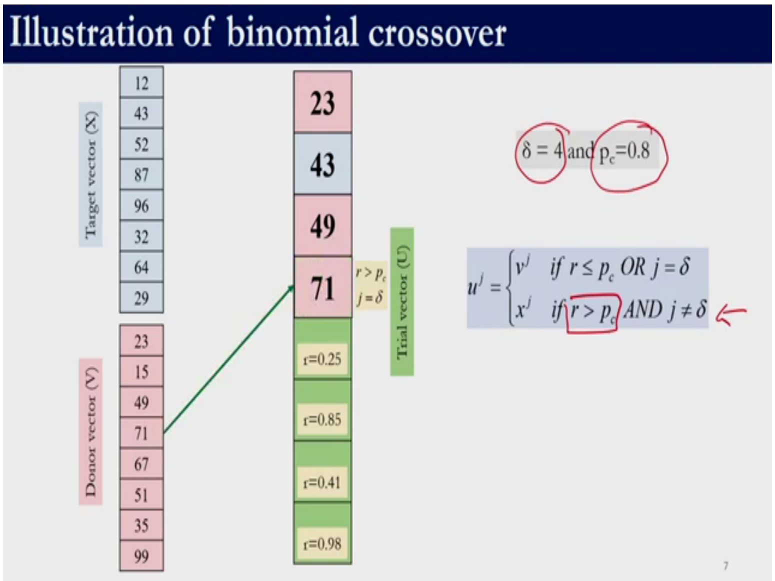
## Illustration of binomial crossover

$\delta = 4$ and $p_c=0.8$

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } j \ne \delta \end{cases}$$

This we can use only if j not equal to delta right. So, this equation we will not be using it, the value will not come from the target vector.

(Refer Slide Time: 16:03)



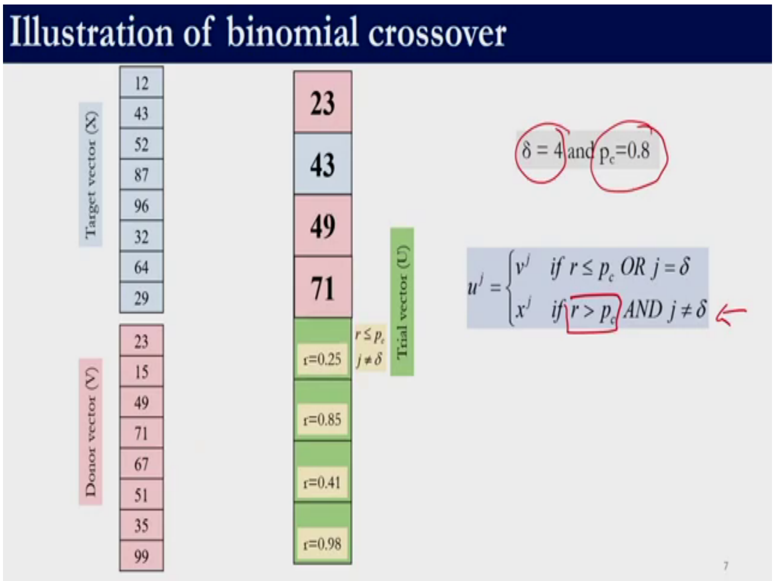
## Illustration of binomial crossover

$\delta = 4$ and $p_c = 0.8$

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } j \ne \delta \end{cases}$$

But it will come from the donor vector right.

(Refer Slide Time: 16:04)



## Illustration of binomial crossover

Target vector (X): 12, 43, 52, 87, 96, 32, 64, 29

Donor vector (V): 23, 15, 49, 71, 67, 51, 35, 99

Trial vector (U): 23, 43, 49, 71, r=0.25, r=0.85, r=0.41, r=0.98

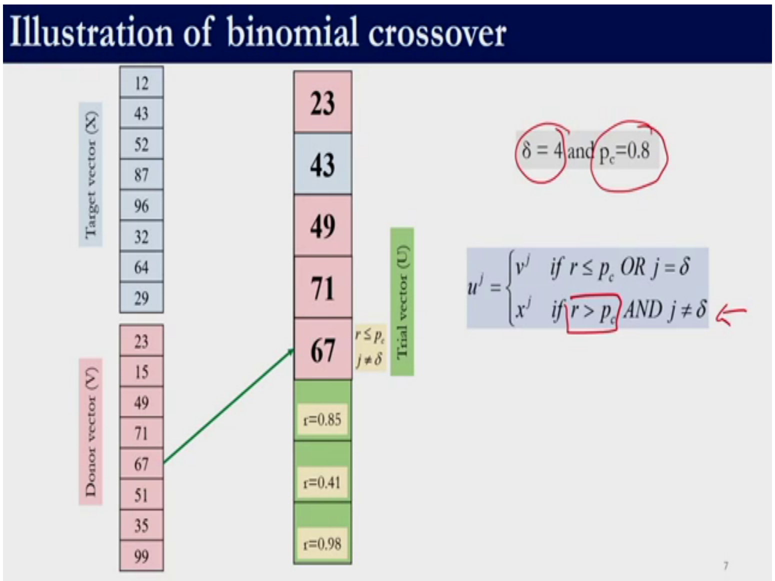
$r \leq p_c$
$j \neq \delta$

δ = 4 and $p_c$=0.8

$$u^j = \begin{cases} v^j & if\ r \leq p_c\ OR\ j = \delta \\ x^j & if\ \boxed{r > p_c}\ AND\ j \neq \delta \end{cases} \leftarrow$$

So, similarly we can find out for the rest of the variables. So, for this fifth variable r is less than p c right ah because r was 0.25, so 0.25 is less than equal to p c then it is going to come from the donor vector.

(Refer Slide Time: 16:14)

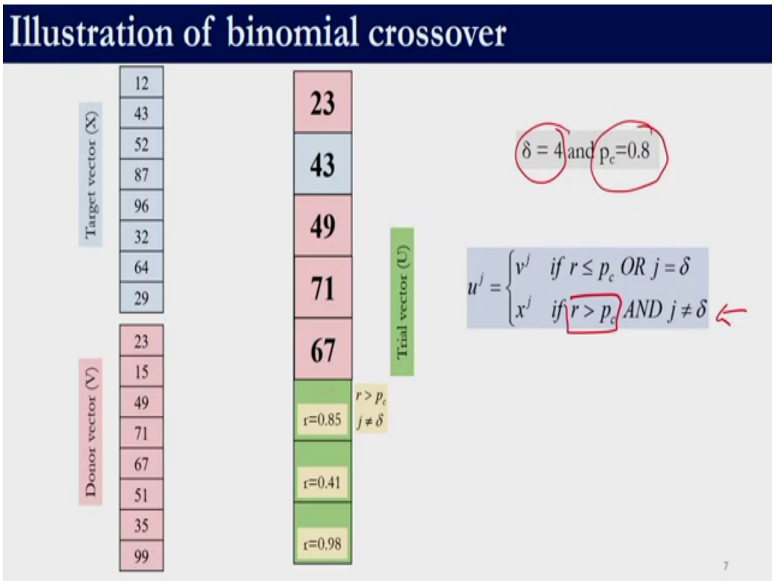


## Illustration of binomial crossover

δ = 4 and $p_c$=0.8

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } j \ne \delta \end{cases}$$

(Refer Slide Time: 16:19)



## Illustration of binomial crossover

Target vector (X)

12, 43, 52, 87, 96, 32, 64, 29

Donor vector (V)

23, 15, 49, 71, 67, 51, 35, 99

Trial vector (U)

23, 43, 49, 71, 67, r=0.85, r=0.41, r=0.98

$r > p_c$ $j \neq \delta$

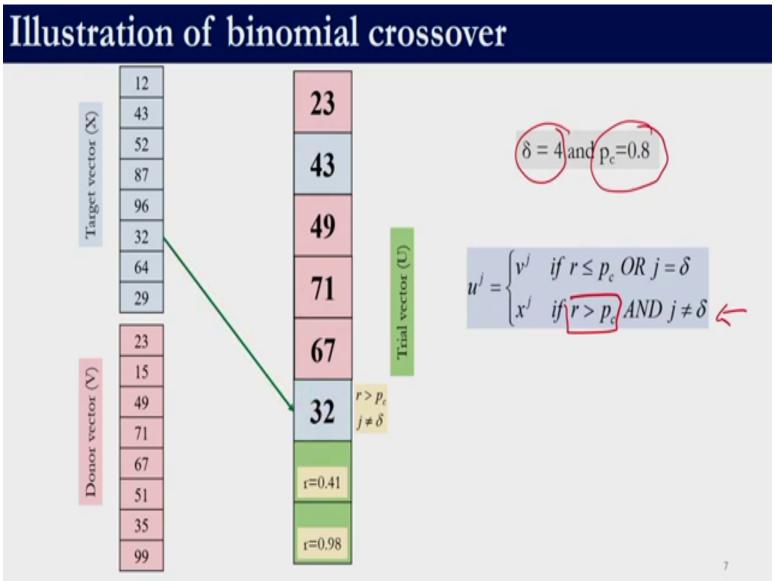
$\delta = 4$ and $p_c$=0.8

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } j \neq \delta \end{cases}$$

7

For the sixth case 0.85 is greater than 0.8 and j is not equal to delta. Because j for this is 6 and the delta value which we have taken is 4 right.

(Refer Slide Time: 16:32)



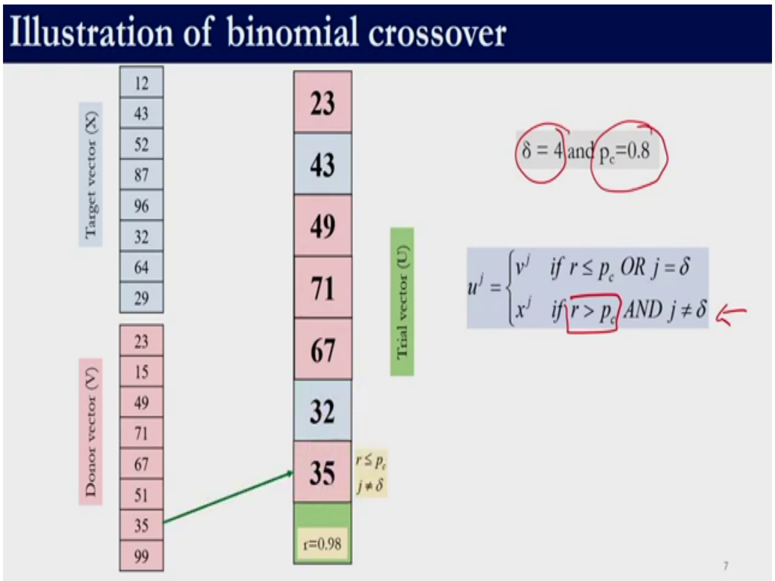
## Illustration of binomial crossover

$\delta = 4$ and $p_c=0.8$

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } j \ne \delta \end{cases}$$

So, it is going to come from the target vector.

(Refer Slide Time: 16:35)



## Illustration of binomial crossover

Target vector (X)

12 43 52 87 96 32 64 29

Donor vector (V)

23 15 49 71 67 51 35 99

Trial vector (U)

23 43 49 71 67 32 35

$r \leq p_c$

$j \neq \delta$

r=0.98

δ = 4 and $p_c$=0.8

$$u^j = \begin{cases} v^j & \text{if } r \leq p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \neq \delta \end{cases}$$

7

(Refer Slide Time: 16:36)

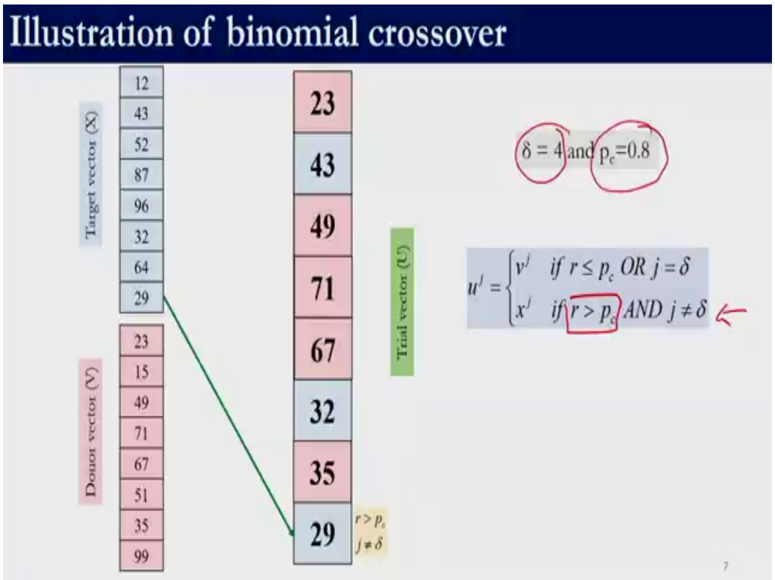


Illustration of binomial crossover

δ = 4 and $p_c$=0.8

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } \boxed{r > p_c} \text{ AND } j \ne \delta \end{cases}$$

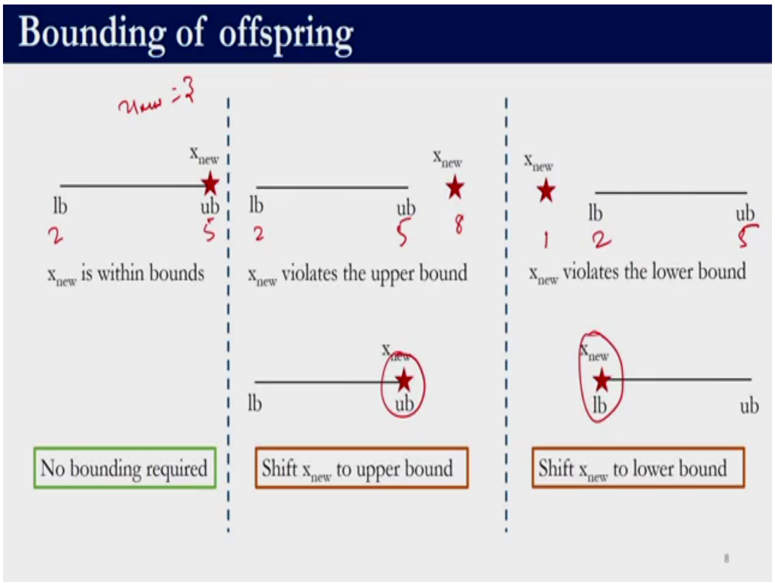
Similarly, you can perform for the rest of the 2 variables and we can generate the entire trial vector right. So, this is the binomial or the uniform crossover. So, we have looked at a mutation strategy. We have looked at a crossover strategy. There are 2 types of crossover; one is the binomial cross over and the other one is the exponential crossover. In a little while we will look at the exponential crossover. So, one of this crossover is to be used. So, the first step in differential evolution is to take a solution generates it donor vector. So, for generating its donor vector we will require 3 other random solutions which should not be identical right.

So, that way we generate the donor vector. Once we have generated the donor vector we need to apply the crossover to generate the trial vector. So, after generating the trial vector, we need to check whether the trial vector is it within the bounds of the decision variable or not. So, this is something similar to particle swarm optimization. In particle swarm optimization we generated the velocity, but we did not check whether the velocity is within the bounds right after generating the velocity. We generated the position and it was the position for which we checked whether it is in the bounds of the decision variables or not.

So, similarly over here the donor vector may or may not be in the bounds right. It is not to be bounded because the donor vector is subsequently going to be used to generate the trial vector. So, it is the trial vector whose bounds have to be checked right. So, the first step is to employ mutation to generate the donor vector, then we need to apply the uniform crossover or the exponential crossover to generate the trial vector and then we need to check whether the trial vector is within the bounds are not.

So, the bounding strategy which we have discussed in particle swarm optimization and teaching learning based optimization is the corner bounding strategy. So, the same strategy is employed in differential evolution right.

(Refer Slide Time: 18:35)



Bounding of offspring

$x_{new}$ is within bounds

$x_{new}$ violates the upper bound

$x_{new}$ violates the lower bound

No bounding required

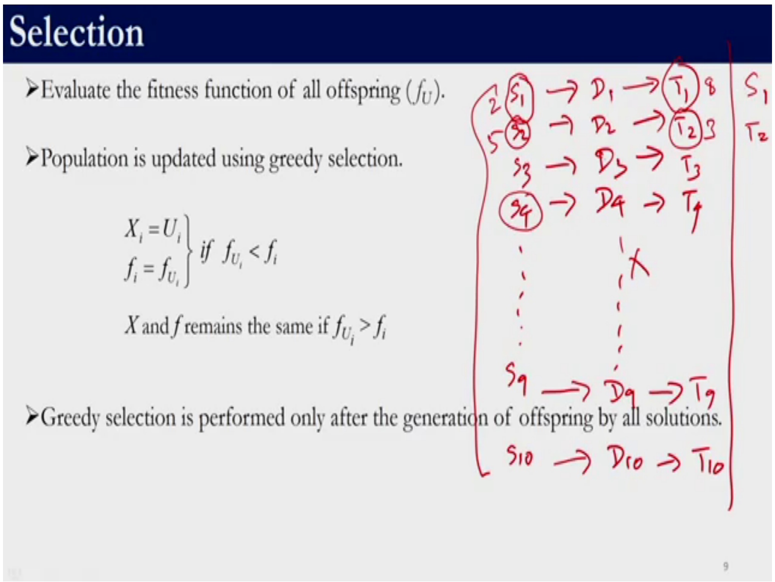
Shift $x_{new}$ to upper bound

Shift $x_{new}$ to lower bound

So, here if we see if the variable is within the bounds, let us say the lower bound is 2, the upper bound is 5 and the variable is if the variable value is 3 right. So, we do not need to bound it because it is already in the bounds. If the lower bound is 2 and the upper bound is 5 and if we get a value of 8 then it has to be pushed back to the upper bound right.

So, x new is being push to the upper bound similarly, if the lower bound is to the upper bound is 5 and we get a value of 1 then the decision variable will be pushed back to the lower bound right. So, this is the bounding strategy which we are employing so far. So, the same bounding strategy will be employed in differential evolution to generate the trial vector which is in the bounds.

(Refer Slide Time: 19:16)



## Selection

- Evaluate the fitness function of all offspring ($f_U$).
- Population is updated using greedy selection.

$$\left.\begin{array}{l} X_i = U_i \\ f_i = f_{U_i} \end{array}\right\} \text{if } f_{U_i} < f_i$$

$X$ and $f$ remains the same if $f_{U_i} > f_i$

- Greedy selection is performed only after the generation of offspring by all solutions.

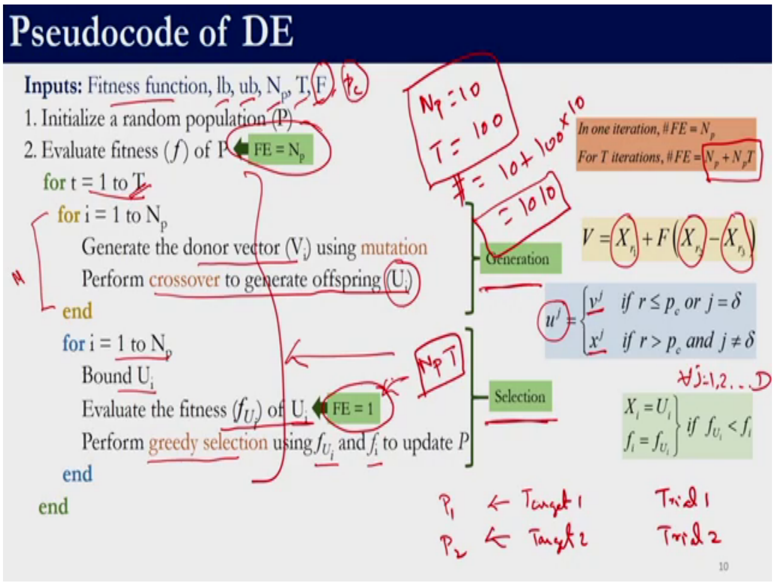
So, once we have generated the trial vector, the next step is to evaluate the fitness function value. So, in this as we explained earlier right, so, if we have 10 solutions S 1, S 2, S 3, S 4 all the way up to S 10 right. So, for each solution donor vector is to be generated and for each donor vector trial vector is generated, trial 1, trial 2, trial 3 and so on right. So, here we will have a different this thing. So, here donor 3, trial 3 again donor 4 trial 4 and all the way up to, all the trial vectors have to be first generated and they should not be updated right. The updation happens only after we have generated trial vector for all the solutions.

So, for example, when we are generating for S 4 if we had to pick solutions randomly from the population, we are suppose to pick from this S 1 to S 10 and not from T 1 to T 10 or D 1 to D 10. The solution update will happen after generating all the trial vectors once we have generated all the trial vectors right then we need to do a greedy selection between S 1 and T 1 whichever vector is better or whichever solution is better will survive.

Let us say between S 1, S 1 has an objective function of 2 and T 1 has an objective function value of 8 and let us say we are the solving a minimization problem. So, between this 2 and 8, 2 would be selected for the next population S 1 will be the first member right. So, for example, S 2 is let us say 5 and T 2 let us say it is 3 right. So, in this case T 2 will survive right.

So, similarly we will find out between these 10 comparisons, we will get 10 solutions that solution will form the population for the subsequent iteration that completes the discussion of differential evolution, consolidate whatever we have seen so far to generate a pseudo code for differential evolution.

(Refer Slide Time: 21:31)



## Pseudocode of DE

```
Inputs: Fitness function, lb, ub, $N_p$, T, F, $p_c$
1. Initialize a random population (P)
2. Evaluate fitness ($f$) of P   ← FE = $N_p$
   for t = 1 to T
     for i = 1 to $N_p$
        Generate the donor vector ($V_i$) using mutation
        Perform crossover to generate offspring ($U_i$)
     end
     for i = 1 to $N_p$
        Bound $U_i$
        Evaluate the fitness ($f_{U_i}$) of $U_i$   ← FE = 1
        Perform greedy selection using $f_{U_i}$ and $f_i$ to update $P$
     end
   end
```

Generation

Selection

*In one iteration, #FE = $N_p$*

*For T iterations, #FE = $N_p + N_p T$*

$V = X_{r_1} + F\left(X_{r_2} - X_{r_3}\right)$

$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ or } j = \delta \\ x^j & \text{if } r > p_c \text{ and } j \neq \delta \end{cases}$

$\left.\begin{matrix} X_i = U_i \\ f_i = f_{U_i} \end{matrix}\right\} \text{if } f_{U_i} < f_i$

10

So, for differential evolution we need to provide the fitness function. The lower bound, the upper bound, the population size, the termination criteria, the scaling factor which is to be used in mutation and the crossover probability which is to be used in the recombinant operator right. So, scaling factor has to be between 0 and 2 and crossover probability as the name indicates, it is a probability. So, it has to be between 0 and 1 right. The first step is to initialize a random population and evaluate its fitness.

So, this is common to all the 2 other techniques which we have studied right, then we begin the iteration. Set of steps have to be repeated for T times. So, again here we are taking that the number of iterations is a termination criteria, but it can be any other termination criteria also right.

So, then the next step is to a perform mutation for all the solutions. So, we have this for loop right. So, this is our mutation loop right. So, for i is equal to 1 to N p, so, each member is going to undergo mutation. We need to generate the donor vector right. So, the donor vector is going to be generated using 3 random solutions X r 1, X r 2 and X r 3. This V i indicates that for each solution in the population, we generate a donor vector. Once the donor vector is generated we will perform crossover to generate the offspring right.

So, this is for all the variables. We need to decide whether the value of the j th decision variable in the trial vector will come from the donor vector V j or will it come from the target vector X j. Once that is done then we need to bound the solutions, each of this U i is to be bounded right and we need to evaluate the fitness function of it f of U i and f of i. So, the first target vector will be compared with the first trial vector whichever wins will going to the next iteration or next generation right. So, that is what whichever wins will be updated as population number 1 right.

So, between target 2 and trial 2 whichever wins will be updated as the second member of the population. This we have discussed multiple times in this presentation. So, here if we see, this is the generation phase and this is the selection as well as the survivor stage. So, we if we see this also fits into the generalized team of metaheuristic techniques, which we have discussed earlier that we have a selection operator, we have a variation operator and then we have a survivor operator. Here selection was done randomly variation was done using crossover as well as mutation and the selection was done using a greedy selection strategy.

So, here if we see for every member we do one functional evaluation right. In this phase and this happens for N p time's right. So, for every member we will be evaluating the fitness function value. So, this will be N p T times right because, this one single functional evaluation is inside this termination loop right. So, the total number of functional evaluation required over in this for loop section is N p into T and we require initial N p evaluation of the functions in the initial stage before we begin the iterations.

So, the total number of fitness function values is given by N p plus N p into t. So, this if we compare with particle swarm optimization, this is similar to particle swarm optimization. In particle swarm optimization also for every member, in one particular iteration. We had to do one functional evaluation. So, for N p members we need to do N p functional evaluation, for T iteration we will have to do N p into T times and then we have the initial N p evaluations right.

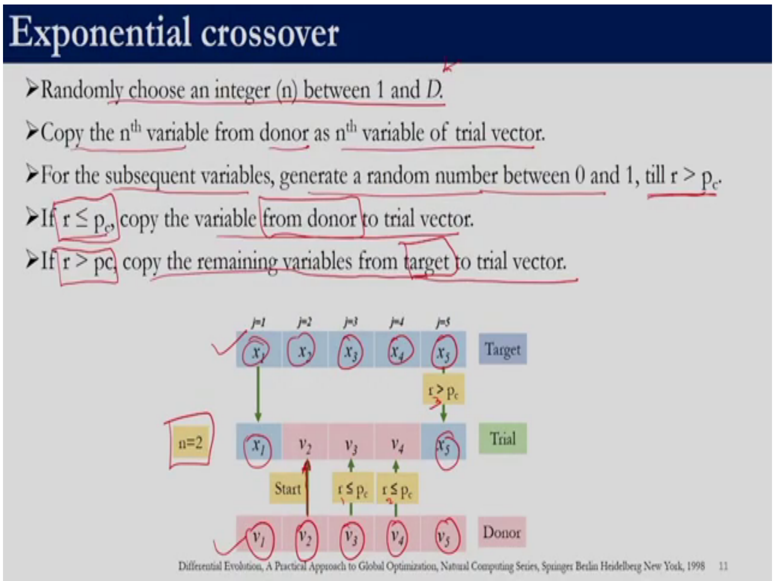
So, this number of functional evaluation in particle swarm optimization as well as in differential evolution is same. So, what it means is that if we take a population size of 10 and if we take a number of iteration as 100 right then the total number of functional evaluation can be given by 10 plus 100 into 10. So, this will work out to 1010. So, if we take a population size of 10 and if we are to perform 100 iterations, we will have to evaluate the objective function 1010 times. So, that completes the pseudo code of differential evolution right.

So, it is a very fairly simple technique if we see right first is we need to perform mutation and then crossover. So, this is to be performed for each solution in mutation. We do not require the target vector to participate right. So, mutation is done with the help of randomly selected solutions from the population. So, in crossover we generate the trial vector. So, the values of the trial vector will come either from the donor vector which we had obtained at the end of mutation or it will come from the target vector right.

So, once we have generated the trial vector. We need to bound the trial vectors. So, that it is in the bounds of the decision variables we need to evaluate their fitness function values. Once we have evaluated their fitness function value we are in a position to do a greedy selection right. So, whichever solution is better between the target 1 and the trial 1 survives to the next generations.

So, this is done for all population members. So, 1 crucial difference which we are repeating time and again is that the update will happen only after all the trial vectors are generated.

(Refer Slide Time: 27:25)



## Exponential crossover

- ➢Randomly choose an integer (n) between 1 and D.
- ➢Copy the n$^{th}$ variable from donor as n$^{th}$ variable of trial vector.
- ➢For the subsequent variables, generate a random number between 0 and 1, till $r > p_c$.
- ➢If $r \leq p_c$, copy the variable from donor to trial vector.
- ➢If r > pc, copy the remaining variables from target to trial vector.

Differential Evolution, A Practical Approach to Global Optimization, Natural Computing Series, Springer Berlin Heidelberg New York, 1998 11

In exponential crossover we will have to first randomly choose an integer between 1 and D, where D is the total number of decision variables in our problem right. So, once we have generated n we need to copy the n th variable from the donor as n th variable of trial vector. So, if there are 10 variables and let us say n we have taken it as 7. So, the 7 th variable of donor will be directly copied as 7 th variable of trial vector right.

For the subsequent variables we need to generate a random number between 0 and 1 right. So, we need to keep doing this till we encounter this condition r is greater than p c. So, technically we do not need to generate random number for every decision variable. We need to generate only till this condition is met that random number is greater than crossover probability right. So, if the random number that we generate is less than equal to crossover probability, we will copy the variable from the donor to the trial vector right.

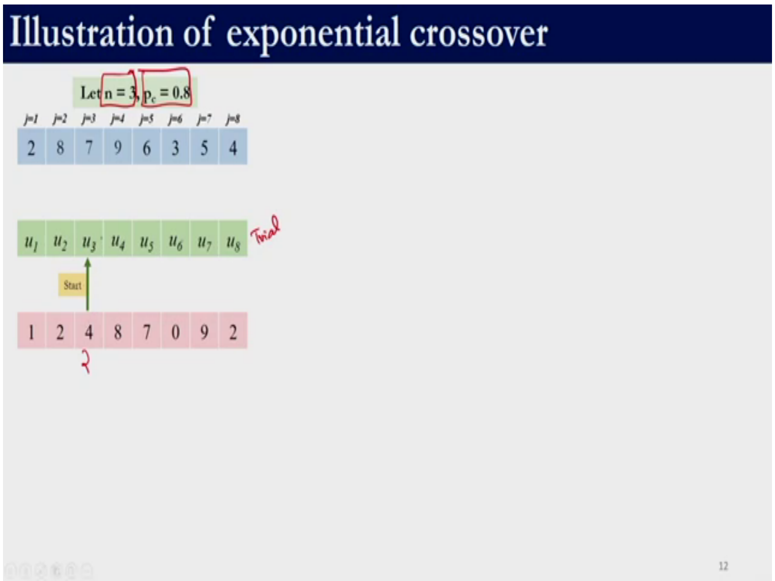
So, here it is from donor, if the random number is less than or equal to p c we will copy it from the donor. If it is greater than p c right not only that variable, but all the remaining variables will be copied from the target to the trial right. So, first time we encounter this condition that the random number is greater than p c we do not need to subsequently generate any random numbers, for generating the trial vector although variables from the target vector will be copied to the trial vector.

So, let us see an example right. So, this is our target vector. So, this is known for this target vector a donor vector was generated. So, this is known. So, the values over here is are known x 1, x 2, x 3, x 4, x 5 are values which are known, similarly, v 1 to v 5 are also known right. We do not know what will be the trial vectors value right. So, to determine the trial vector value will have to first generate an integer between 1 and 5, because we have 5 decision variables.

Let us assume that the random number is 2 right. So, for the second variable, so, we need to start from the second variable right is directly copied from the donor vector. For the third variable we need to generate a random number, if the random number is less than equal to p c, we need to take the value from the donor vector to the trial vector. Subsequent to that we need to generate another random number right if that happens to be less than or equal to the crossover probability. We need to copy from the donor vector to the trial vector right.

Then we need to move on to the next variable. So, in this for the fifth variable let us assume that the random number which we are generating is actually greater than the crossover probability. So, in this case what we need to do is we need to start copying from the target for all the variables which are yet not filled. So, this variable is yet not filled and this variable is yet not filled right so, we need to copy x 5 from the target vector and x 1 also from the target vector right.

(Refer Slide Time: 30:29)



## Illustration of exponential crossover

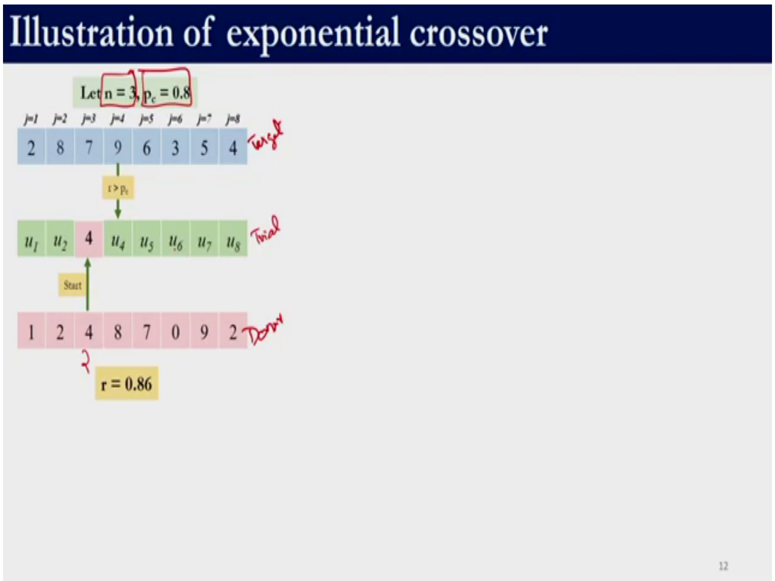
Let $n = 3$, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 |
|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 |
| $u_1$ | $u_2$ | $u_3$ | $u_4$ | $u_5$ | $u_6$ | $u_7$ | $u_8$ |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 |

Now, let us look at an example for exponential crossover. So, let us taken 8 variable problem right and we need to generate a random integer between 1 and 8. So, let us say we generate a random number between 1 and 8 as 3 and the let the crossover probability set by the user be 0.8 right. So, this is our trial vector we need to decide on what are the values that would come to that trial vector either from the donor vector or from the target vector right.

So, if n is equal to 3 that particular variable is to be directly copied from the donor vector right. So, in this case n is equal to 3. So, we are going to start over here. So, for this variable, for the third variable no random number is to be generated it is to be directly copied from the donor vector right.

(Refer Slide Time: 31:19)



## Illustration of exponential crossover

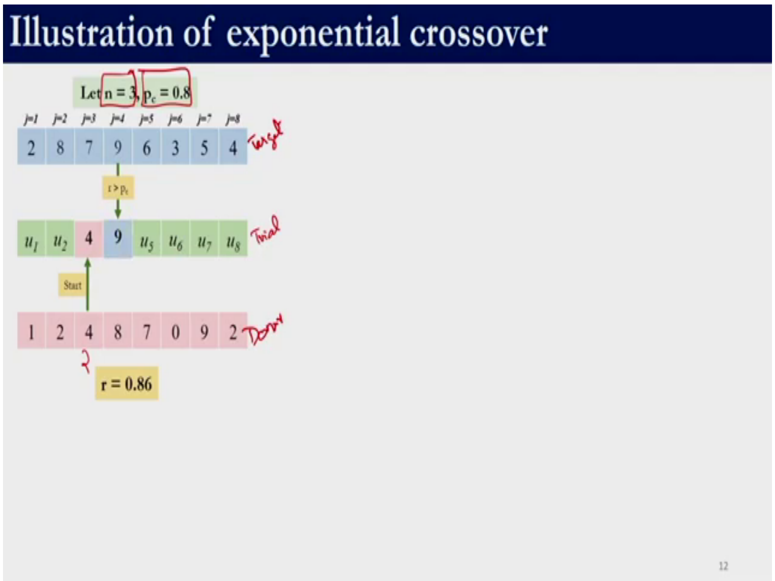
Let n = 3, $p_c$ = 0.8

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| $u_1$ | $u_2$ | 4 | $u_4$ | $u_5$ | $u_6$ | $u_7$ | $u_8$ | Trial |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

r = 0.86

So, this is our donor vector and this is our target vector right. So, 4 is directly copied and then we need to generate a random number right. So, let the let us say that the random number is 0.86 whereas; our crossover probability is 0.8 right. So, this condition fails right that r is less than or equal to p c fails right. So, since it is failing all the rest of the values. So, u 4, u 5, u 6, u 7, u eight, u 1, u 2 which we have not determined so far are to be directly copied from the target vector.

(Refer Slide Time: 31:54)



# Illustration of exponential crossover

Let n = 3, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| $u_1$ | $u_2$ | 4 | 9 | $u_5$ | $u_6$ | $u_7$ | $u_8$ | Trial |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

r = 0.86

12

(Refer Slide Time: 31:56)

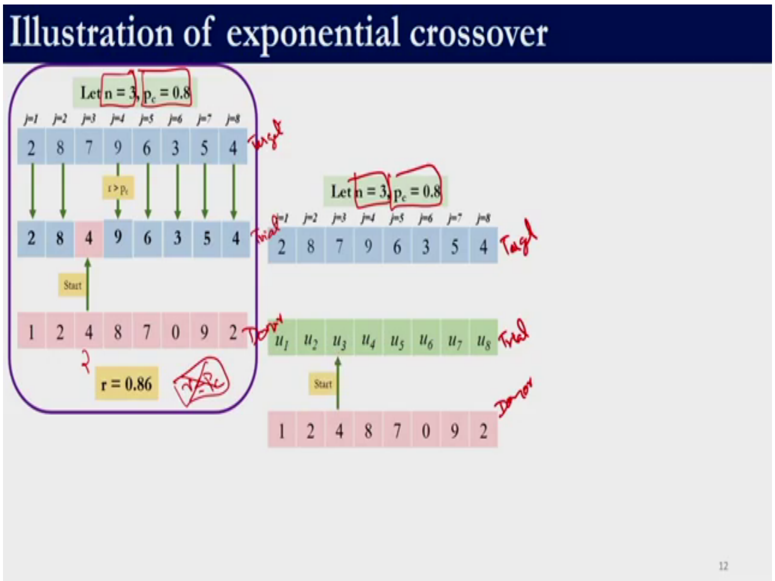


Illustration of exponential crossover

Let n = 3, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 |
|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 |
| 2 | 8 | 4 | 9 | 6 | 3 | 5 | 4 |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 |

$r > p_c$

Start

r = 0.86

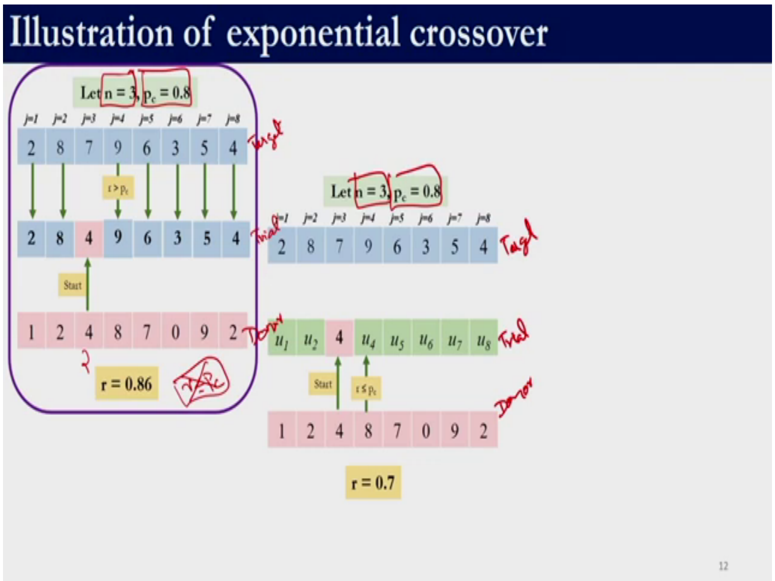
Let n = 3, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 |
|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 |
| $u_1$ | $u_2$ | $u_3$ | $u_4$ | $u_5$ | $u_6$ | $u_7$ | $u_8$ |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 |

Start

9 6 3 5 4 2 8 all of them are directly copied from the target vector because we fail to satisfy this condition r is less than equal to p c, for this variable for the fourth variable this condition failed right. So, all the variables are to be copied from the target vector right. So, this is one case which can occur. So, for the second case this would be our target vector, this is our trial vector and this is our donor vector right. So, in this case again let us assume that we start with 3 and our crossover probability is 0.8 right. So, in this case 4 would be directly copied no random number is to be generated right.

(Refer Slide Time: 32:42)



## Illustration of exponential crossover

Let $n = 3$, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| 2 | 8 | 4 | 9 | 6 | 3 | 5 | 4 | Trial |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

$r > p_c$

Start

$r = 0.86$

Let $n = 3$, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| $u_1$ | $u_2$ | 4 | $u_4$ | $u_5$ | $u_6$ | $u_7$ | $u_8$ | Trial |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

Start

$r \leq p_c$

$r = 0.7$

For the second variable let us say we generate 0.7. So, since we generate 0.7 that is less than equal to p c. So, it is to be copied from the donor vector.

(Refer Slide Time: 32:53)

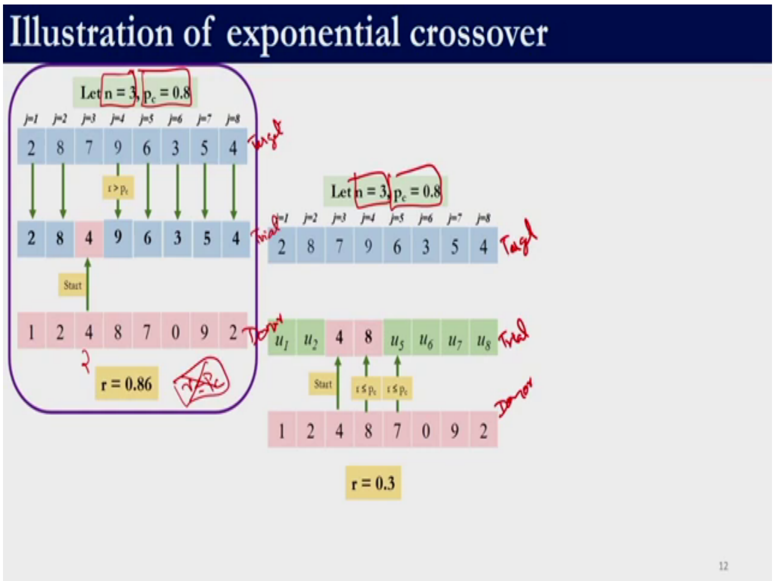


Illustration of exponential crossover

Let $n = 3$, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| | | | $r > p_c$ | | | | | |
| 2 | 8 | 4 | 9 | 6 | 3 | 5 | 4 | Trial |
| | | Start | | | | | | |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

$r = 0.86$

Let $n = 3$, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| $u_1$ | $u_2$ | 4 | 8 | $u_5$ | $u_6$ | $u_7$ | $u_8$ | Trial |
| | | Start | $r \le p_c$ | $r \le p_c$ | | | | |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

$r = 0.3$

12

For the next variable let us say we generate 0.3 as the random number. So, it is also less than p c because our crossover probability is 0.8. So, 7 is to be directly copied here again for this sixth variable we need to generate a random number right.

(Refer Slide Time: 33:08)

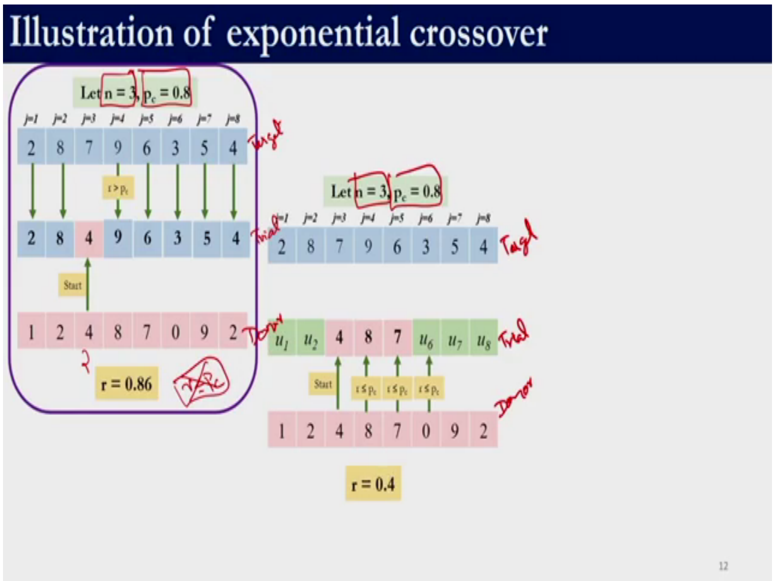
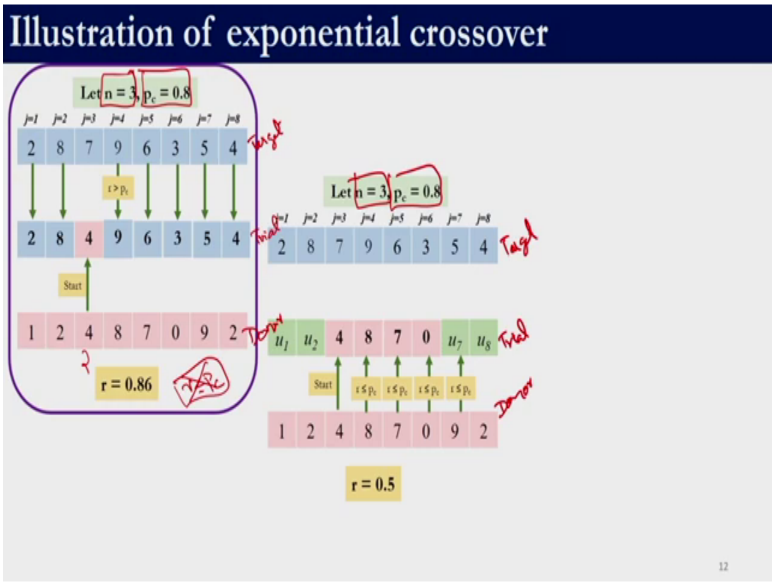


Illustration of exponential crossover

Let us say it is 0.4. So, again we need to copy it from the donor vector.

(Refer Slide Time: 33:15)



## Illustration of exponential crossover

Let n = 3, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| 2 | 8 | 4 | 9 | 6 | 3 | 5 | 4 | Trial |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

r = 0.86

$r > p_c$

Let n = 3, $p_c = 0.8$

| j=1 | j=2 | j=3 | j=4 | j=5 | j=6 | j=7 | j=8 | |
|---|---|---|---|---|---|---|---|---|
| 2 | 8 | 7 | 9 | 6 | 3 | 5 | 4 | Target |
| $u_1$ | $u_2$ | 4 | 8 | 7 | 0 | $u_7$ | $u_8$ | Trial |
| 1 | 2 | 4 | 8 | 7 | 0 | 9 | 2 | Donor |

r = 0.5

Let us say for the fifth variable the crossover probability is 0.5. So, it is again to be copied from the donor vector because r is less than p c is satisfied right.

(Refer Slide Time: 33:24)

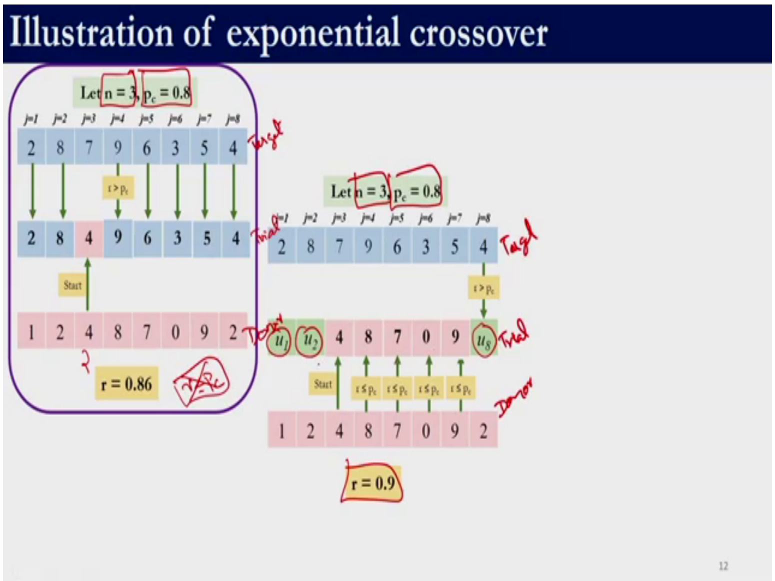
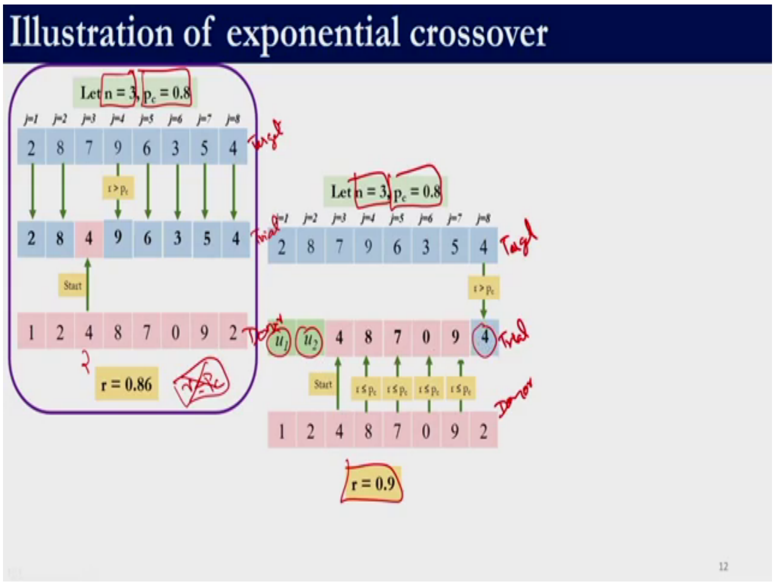


Illustration of exponential crossover

For the next variable let us say the random number is 0.9 and our crossover probability is 0.8 right. So, here the condition fails. The condition fails and we still are to fill 3 values u 1, u 2 and u 8. So, all these three variables are to be copied from the target vector right.

(Refer Slide Time: 33:43)



# Illustration of exponential crossover

(Refer Slide Time: 33:45)

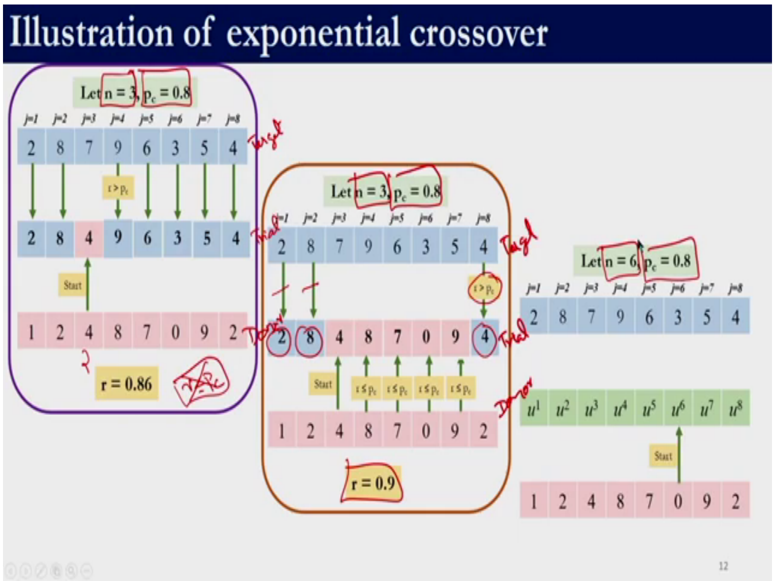
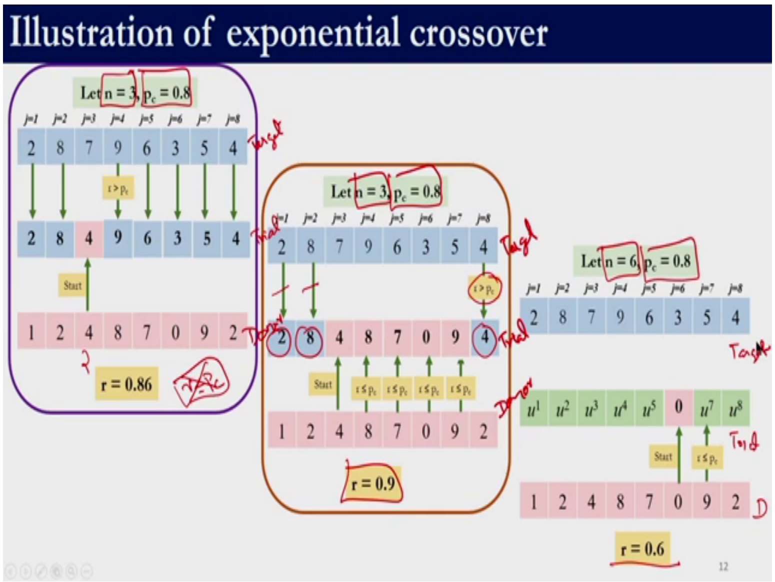


Illustration of exponential crossover

So, 4 2 8 are directly copied from the target vector. So, what we need to remember is once we hit a failure, that once the r is greater than the crossover probability whatever values are yet to be filled are to be copied from the target vector. So, we do not need to generate random numbers after we meet the first failure of r greater than p c right, after that no random numbers are generated.

So, here if we see we did not generate a random number for 2 and for this 8 or the first variable and the second variable. We did not generate because once this condition occurs all the variables are to be copied from the target vector right. So, this is another example in which n is 6 and the crossover probability is 0.8 right.

(Refer Slide Time: 34:33)



## Illustration of exponential crossover

So, the sixth variable is to be directly copied right for the seventh variable, let us say what we have random number is 0.6 and crossover is probability is 0.8. So, this again will come from the donor vector right. So, this is donor, this is trial and this is target right.

(Refer Slide Time: 34:54)

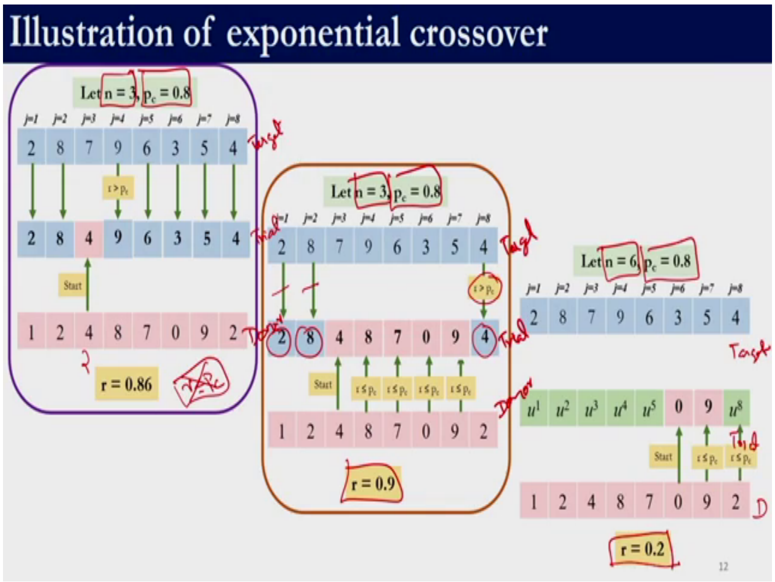


Illustration of exponential crossover

So, this condition satisfied for the next variable random number is 0.2. So, that is also less than p c. So, it will come from the donor vector.

(Refer Slide Time: 35:02)

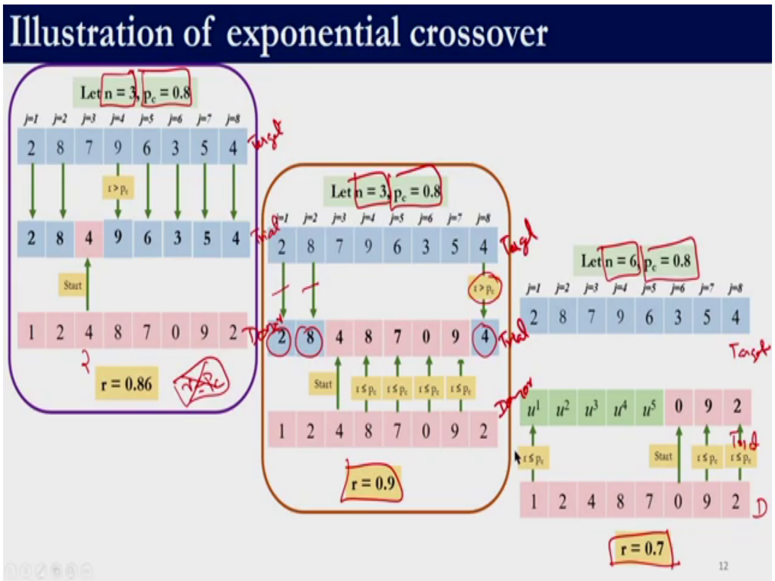


Illustration of exponential crossover

Let n = 3, $p_c$ = 0.8

j=1 j=2 j=3 j=4 j=5 j=6 j=7 j=8

2 8 7 9 6 3 5 4

$r > p_c$

2 8 4 9 6 3 5 4

Start

1 2 4 8 7 0 9 2

r = 0.86

Let n = 3, $p_c$ = 0.8

j=1 j=2 j=3 j=4 j=5 j=6 j=7 j=8

2 8 7 9 6 3 5 4

$r > p_c$

2 8 4 8 7 0 9 4

Start $r \le p_c$ $r \le p_c$ $r \le p_c$ $r \le p_c$

1 2 4 8 7 0 9 2

r = 0.9

Let n = 6, $p_c$ = 0.8

j=1 j=2 j=3 j=4 j=5 j=6 j=7 j=8

2 8 7 9 6 3 5 4

$u^1$ $u^2$ $u^3$ $u^4$ $u^5$ 0 9 2

$r \le p_c$ Start $r \le p_c$ $r \le p_c$

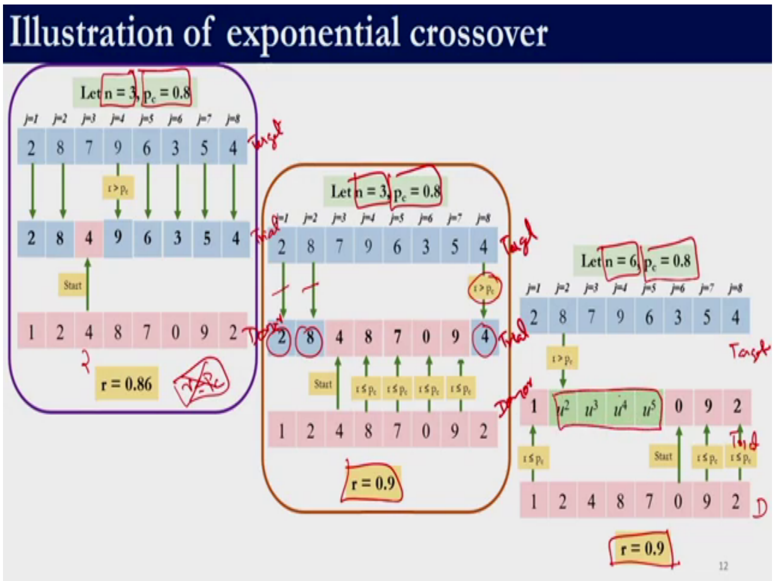
1 2 4 8 7 0 9 2

r = 0.7

Now, we have come to the end of all the decision variable. So, again we need to start from 1 right and generate a random number. So, let us say we generate 0.7 which is less than 0.8. So, we will copy from the mutant or the donor vector right.

(Refer Slide Time: 35:17)

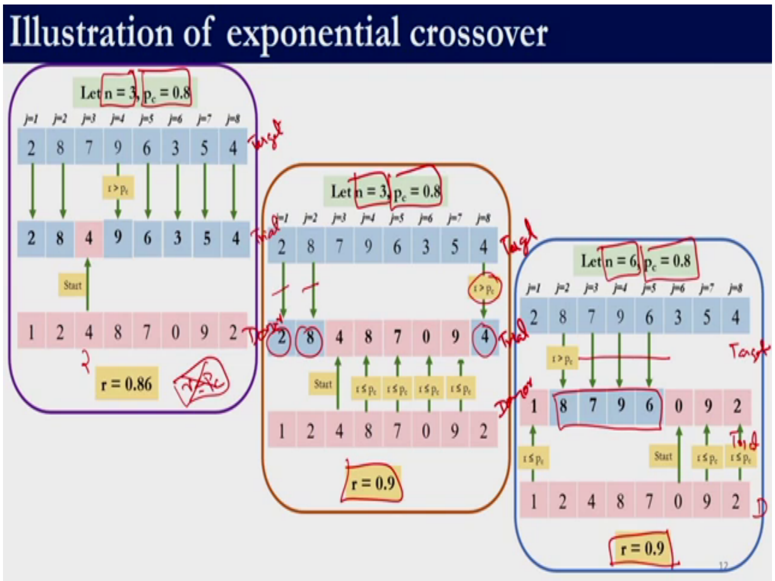
## Illustration of exponential crossover



So, here let us say the random number generated for the second variable is 0.9. So, here we fail right. So, now, we have encountered a failure. So, all these 4 values u 2, u 3, u 4, u 5 will have to come from the target vector right.

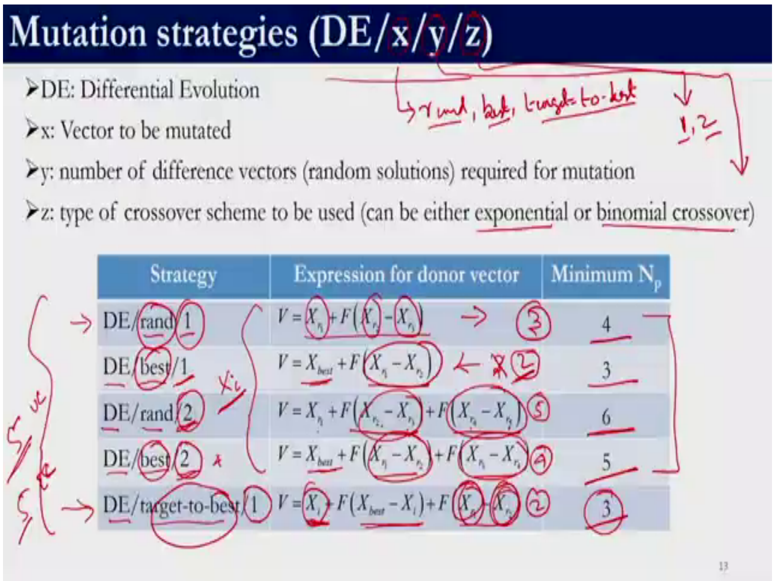
(Refer Slide Time: 35:35)

## Illustration of exponential crossover



So, here we are not again generating random numbers right. So, this completes our discussion on exponential crossover. If you carefully analyze either it is binomial crossover or it is exponential crossover one variable will always come from the donor vector.

(Refer Slide Time: 35:54)



## Mutation strategies (DE/x/y/z)

- ➢DE: Differential Evolution
- ➢x: Vector to be mutated
- ➢y: number of difference vectors (random solutions) required for mutation
- ➢z: type of crossover scheme to be used (can be either exponential or binomial crossover)

| Strategy | Expression for donor vector | Minimum $N_p$ |
|---|---|---|
| DE/rand/1 | $V = X_{r_1} + F(X_{r_2} - X_{r_3})$ | 4 |
| DE/best/1 | $V = X_{best} + F(X_{r_1} - X_{r_2})$ | 3 |
| DE/rand/2 | $V = X_{r_1} + F(X_{r_2} - X_{r_3}) + F(X_{r_4} - X_{r_5})$ | 6 |
| DE/best/2 | $V = X_{best} + F(X_{r_1} - X_{r_2}) + F(X_{r_3} - X_{r_4})$ | 5 |
| DE/target-to-best/1 | $V = X_i + F(X_{best} - X_i) + F(X_{r_1} - X_{r_2})$ | 3 |

There are few variance of DE right. So, those variance are generally represented by this common format DE slash x slash y slash z. This x indicates something, this y indicates something and this z indicates something right. So, depending upon what mutation strategy we are using, what crossover strategy, we are using this x y z are to be filled right. So, here let us say look at this one. So DE, so, that stands for differential evolution right slash rand right. So, this rand indicates the solutions which we are selecting for the donor vector.

So, for the donor vector if you see we are selecting these 3 solutions randomly right. We are not saying that it is the best solution or the worst solution or it is the mean solution or anything right. So, it is selected randomly. So, this rand indicates that thing right and this 1 indicates the difference right. So, here we have one difference between 2 randomly selected solutions. So, this indicates 1 right.

So, let us look at the second one. So DE again stands for the differential evolution here it is best. So, here 3 solutions are not selected randomly, but only 2 solutions are selected randomly right and for the third one we use the best solution. So, that is why it is called as DE slash best right and there is only one difference between 2 randomly selected solutions. So, this one stands over here right.

And then we have DE rand so obviously, rand means that whatever is selected is to be selected randomly for the mutation operator right and here we have 2. So, 2 indicate there are 2 differences. So, we have 1 difference over here. So, this is a difference between a randomly selected solution and this is another difference randomly selected solution. So, if we are to apply this strategy right DE slash rand slash 2 then we need 5 randomly selected solutions 1, 2, 3, 4, 5 to generate a donor vector right.

So, the next variant is DE again stands for differential evolution. We are using best right. So, it is just same as the previous one except that we are using the best over here and there are 2 differences. So, that is 2 indicates the number of differences. So, we have 1 difference over here and 1 difference over here. So, again this is a difference between 2 randomly selected solutions this is a difference between 2 randomly selected solutions right.

Here the donor vector is generated using the best right and 2 differences of randomly selected solutions. So, when we say 2 differences; obviously, we need 4 randomly selected solutions right. So, the last one is DE target to best right and 1 difference between randomly selection solution. So, in all this 4 strategies, if we see we have not selected the solution which is undergoing the mutation right. So, we did not select the X i solution right.

So, in differential evolution, if we are going to employ this strategy for mutation DE slash target to best slash 1 right. So, this indicates the solution which is undergoing the mutation phase right. So, it did not appear in any of the 4. This difference right is not between randomly selected solutions, but between best as and the i th member which is undergoing the mutation phase right. So, that is why this target to best right and then we have 1 random difference. So, it is 1 right.

So, x can take either rand best or target to best right. Rand means all the solutions required are selected randomly best. You would indicate that the best solution is involved right and target to best would indicate that we have 1 difference which is between the best and the target right. So, that is what x would stand, this y would indicate for how many differences are there are there 1 difference or are there 2 differences. When we say differences remember it is the differences between randomly selected solutions.

So, for example, in this case we have a difference between best and the target, but still we write 1 over here there are 2 subtractions agreed, but only 1 of this is between randomly selected solutions. So, this y can take 1 or 2 and this z indicates the types of crossover we are using are we using exponential crossover or binomial crossover right. So, we have this 5 strategies let say with uniform crossover right.

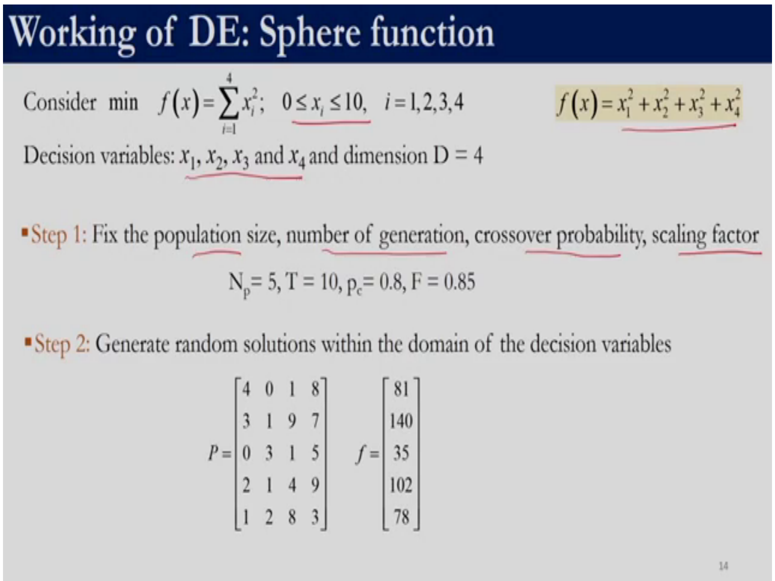
Similarly, we can have 5 more strategies with exponential crossover right. So, basically if we talk in terms of number, we have looked at 10 variance10 variance which depends upon whether we select random solution, best solution or target to best. How many differences do we take do we take? 1 difference or 2 differences and whether we take exponential crossover or binomial crossover?

So, if you analyze this right. So, in this case we need 3 random solutions right. So, obviously, if we need 3 random solutions, we require the minimum population size to be 4. In this case we require 2 random solutions. So, in that case the minimum number of solutions that we require in the population is 3 here we have 5 randomly selected solutions right. So, including the population number which is to undergo mutation we will require 6 members in the population. Here again we have 4 randomly selected solution.

So, the population size has to be at least 5 right. In this case we required only 2 random solutions. So, the population size has to be at least 3 right. So, two random solutions, so the population size is 3 right. So, number of random solutions plus 1 gives them minimum population size right, because the member which is undergoing the mutation phase is not participating in the mutation right. So, for particularly in this 4 it does not participate. So, X i is not there only in this case X i is there, but even in that case we need X i X r 1 X r 2 right.

So, we still need 3 solutions over here. So, those are the mutations strategies in differential evolution. Now, that we have looked into differential evolution let us take the same example which we have used for teaching learning based and particle swarm optimization right. We will take this sphere function with 4 decision variables and see the working of differential evolution. So, that will help you to clear some of the concepts which we have discussed so far.

(Refer Slide Time: 43:01)



Working of DE: Sphere function

Consider min $f(x)=\sum_{i=1}^{4} x_i^2;\quad 0\le x_i\le 10,\quad i=1,2,3,4$ $\quad f(x)=x_1^2+x_2^2+x_3^2+x_4^2$

Decision variables: $x_1, x_2, x_3$ and $x_4$ and dimension D = 4

▪Step 1: Fix the population size, number of generation, crossover probability, scaling factor

$N_p = 5,\ T = 10,\ p_c = 0.8,\ F = 0.85$

▪Step 2: Generate random solutions within the domain of the decision variables

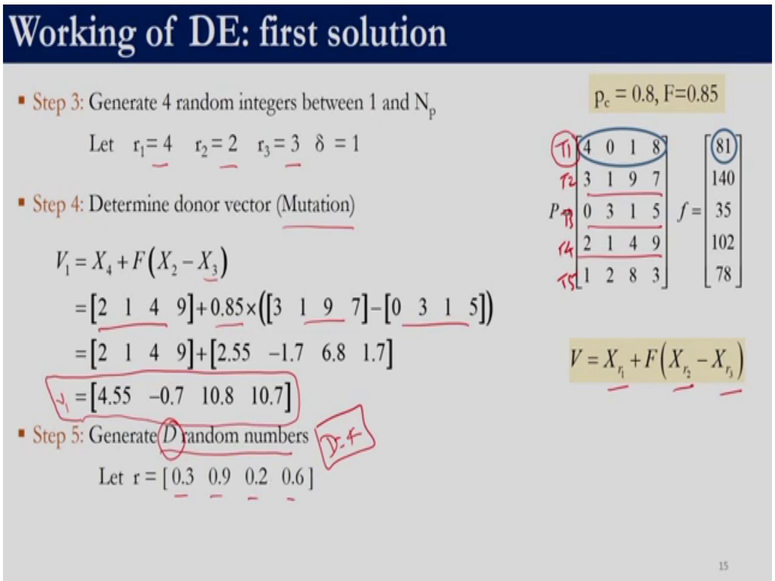
$$P=\begin{bmatrix}4&0&1&8\\3&1&9&7\\0&3&1&5\\2&1&4&9\\1&2&8&3\end{bmatrix}\qquad f=\begin{bmatrix}81\\140\\35\\102\\78\end{bmatrix}$$

14

So, here we have 4 decision variables x 1, x 2, x 3, x 4. The decision the domain of the decision variable is between 0 and 10 right. The objective function is the sphere function which is sum of square of all the decision variables right. So, of the first step is to fix the population size, it is a population based technique. So, we need to fix the population size. We will take the number of generation has termination criteria and then we need to fix the crossover probability and the scaling factor. So, these 4 are use a defined parameters in addition to this we also need to specify the type of technique we are going to use crossover right.

So, here we will be using binomial uniform crossover. So, the next step is to generate a random population. So, in this case we have generated 5 random solutions within the domains and evaluated its fitness function right.

(Refer Slide Time: 43:49)



## Working of DE: first solution

- Step 3: Generate 4 random integers between 1 and $N_p$

  Let $r_1 = 4$ $r_2 = 2$ $r_3 = 3$ $\delta = 1$

- Step 4: Determine donor vector (Mutation)

$$V_1 = X_4 + F(X_2 - X_3)$$
$$= [2\ \ 1\ \ 4\ \ 9] + 0.85 \times ([3\ \ 1\ \ 9\ \ 7] - [0\ \ 3\ \ 1\ \ 5])$$
$$= [2\ \ 1\ \ 4\ \ 9] + [2.55\ \ -1.7\ \ 6.8\ \ 1.7]$$
$$\boxed{V_1 = [4.55\ \ -0.7\ \ 10.8\ \ 10.7]}$$

- Step 5: Generate $D$ random numbers $\boxed{D\text{-}4}$

  Let r = [0.3 0.9 0.2 0.6]

$p_c = 0.8$, F=0.85

$$P = \begin{array}{c|cccc|} T1 & 4 & 0 & 1 & 8 \\ T2 & 3 & 1 & 9 & 7 \\ T3 & 0 & 3 & 1 & 5 \\ T4 & 2 & 1 & 4 & 9 \\ T5 & 1 & 2 & 8 & 3 \end{array} \quad f = \begin{bmatrix} 81 \\ 140 \\ 35 \\ 102 \\ 78 \end{bmatrix}$$

$$V = X_{r_1} + F(X_{r_2} - X_{r_3})$$

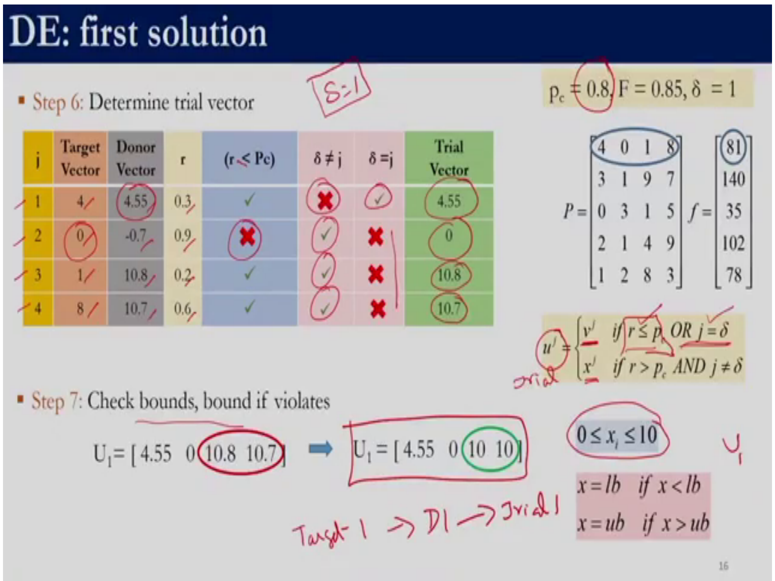
15

So, the next step is to determine the donor vector for the first solution right. So, this is our target 1, target 2, target 3, target 4 and target 5 right. So, we need to determine the donor vector for this. Once we have the donor vector will be able to determine the trial vector from target vector to determine the donor vector. We need 3 random solutions right. So, 3 random solutions we need to pick in this case. Let us assume that the 3 random solutions are fourth solution, second solution and the third solution right.

So, if we pick that we can apply this equation right the fourth solution is 2 1 4 9 and the crossover factor we have taken it as 0.85, the second solution is 3 1 9 7 and the third solution is 0 3 1 5. So, if we apply this equation we will be able to determine the donor vector. So, this is the donor vector corresponding to the first solution right, remember this solution is not to be bound.

So, what we have now done is we have only completed the mutation of the target vector mutation for the target vector we have completed. So, we have this donor vector to apply that uniform crossover. We need to generate D random numbers where D is the number of decision variable in our case D is equal to 4. We have 4 decision variables. So, we need to generate 4 random numbers between 0 and 1. So, let us say these are our random numbers.

(Refer Slide Time: 45:16)



## DE: first solution

- Step 6: Determine trial vector

$\boxed{\delta=1}$

$p_c = 0.8, F = 0.85, \delta = 1$

| j | Target Vector | Donor Vector | r | (r < Pc) | δ ≠ j | δ = j | Trial Vector |
|---|---|---|---|---|---|---|---|
| 1 | 4 | 4.55 | 0.3 | ✓ | ✗ | ✓ | 4.55 |
| 2 | 0 | -0.7 | 0.9 | ✗ | ✓ | ✗ | 0 |
| 3 | 1 | 10.8 | 0.2 | ✓ | ✓ | ✗ | 10.8 |
| 4 | 8 | 10.7 | 0.6 | ✓ | ✓ | ✗ | 10.7 |

$$P = \begin{bmatrix} 4 & 0 & 1 & 8 \\ 3 & 1 & 9 & 7 \\ 0 & 3 & 1 & 5 \\ 2 & 1 & 4 & 9 \\ 1 & 2 & 8 & 3 \end{bmatrix} \quad f = \begin{bmatrix} 81 \\ 140 \\ 35 \\ 102 \\ 78 \end{bmatrix}$$

$$\underset{\text{Trial}}{u^j} = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

- Step 7: Check bounds, bound if violates

$U_1 = [4.55 \;\; 0 \;\; 10.8 \;\; 10.7]$ ➡ $\boxed{U_1 = [4.55 \;\; 0 \;\; 10 \;\; 10]}$

$0 \le x_i \le 10$

$x = lb \quad if \; x < lb$

$x = ub \quad if \; x > ub$

$U_1$

Target 1 → D1 → Trial 1

16

Let us see the application of uniform crossover on the first target vector right. So, this is the target vector that we are currently working it. So, that is what is written over here 4 0 1 8 and these are the four decision variables, j is equal to 1, j is equal to 2, j is equal to 3 and j is equal to 4. To apply this crossover we need to have this delta which is a random number between 1 to the population size right.

So, right now we have taken delta to be 1 right and for every target vector we need to have a random number because that is what we are going to compare with the crossover probability. So, let these 4 be our random number which we have previously seen and this be the donor vector which we have obtained from the mutation operation right. So, now, we have our target vector, we have the donor vector, we have the random numbers which will be used to compare with the crossover probability and we also have this del value, we have everything in place to apply the binomial crossover to determine the trial vector right.

So, we need to see whether the first value of trial vector is going to be 4 or 4.55, the second value in the trial vector is going to be 0 or 0.7, the third value is 1 or 10.8 and the fourth value is 8 or 10.7 right. So, that is what we are going to determine. So, first let us check this condition right. So, since this condition is r is less than equal to p c. So, let us say which of this actually satisfy it right.

So, expect for the second one all the three random 3 other random numbers 0.3, 0.2 and 0.6 are actually less than our crossover probability which is 0.8 right. So, that is why we have put this 3 tick marks. So, now, that is done right. So, obviously, if r is less than equal to p c right. So, for these 3 variables, we can directly say that the value is going to come from the donor vector 4.55, 10.8 and 10.7 without further checking anything. We can actually conclude that we are going to get 4.55 10.8 and 10.7 right, it because for all of them this condition is satisfied right.

So, to check this condition we need to see whether j is equal to delta or not right. Let us do that. So, now, let us see if delta is equal to j or delta is not equal to j right. In our case delta is equal to 1 this wholes because delta is also 1, j is also 1 right. In other cases j is 2 over here, j is 3 over here and j is 4 over here right. So, for the sake of completion we have given this entire thing otherwise you do not need to checking this condition for the first variable, third variable and fourth variable because they are directly satisfying r is less than pc right.

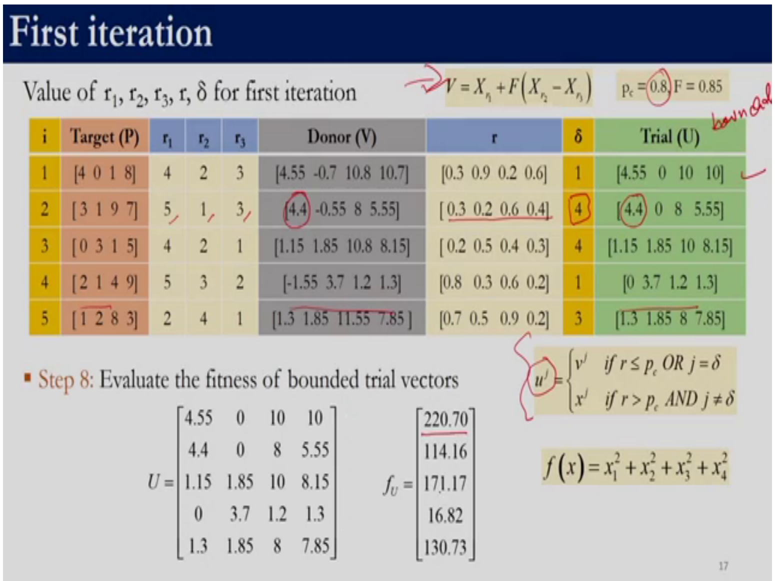
So, it is satisfied. So, here it is the opposite condition right. So, delta is 1 j is 1 right. So, this is delta not equal to j. So, delta is actually equal to j. So, if just for this variable we have this cross mark otherwise, we have this 3 tick marks right. So, as we said earlier 4.55 10.8 and 10.7 are directly going to come from the donor vector. For this case the value of r is 0.9 right. So, r is greater than p c right, r is 0.9 and p c is 0.8 right.

So, in that case we will have to take the value from the target vector 0 right and it also satisfies this condition that j is not equal to delta, j is equal to 2 for this row and delta was 1.

So, this is how we generate the trial vector right. Once the trial vector is generated we need to evaluate its fitness, but before evaluating its fitness, we need to check for its bounds. In this case the bounds for the decision variables are 0 to 10 right. So, we will have to check which of these variables violates. So, the third and fourth variable actually violates the bounds right.

So, using the corner bounding strategy will bring it back to the upper bound. So, our new trial vector is 4.55 0 10 10 right. Remember we should not bound the donor vector because donor vector is not going to be used to calculate the fitness. We will be only using the trial vector to calculate the fitness right. So, it is not necessary to bound the donor vector.

(Refer Slide Time: 49:43)



## First iteration

Value of $r_1$, $r_2$, $r_3$, $r$, $\delta$ for first iteration

$V = X_{r_1} + F(X_{r_2} - X_{r_3})$

$p_c = 0.8$, $F = 0.85$

bounded

| i | Target (P) | $r_1$ | $r_2$ | $r_3$ | Donor (V) | r | δ | Trial (U) |
|---|---|---|---|---|---|---|---|---|
| 1 | [4 0 1 8] | 4 | 2 | 3 | [4.55 -0.7 10.8 10.7] | [0.3 0.9 0.2 0.6] | 1 | [4.55 0 10 10] |
| 2 | [3 1 9 7] | 5 | 1 | 3 | [4.4 -0.55 8 5.55] | [0.3 0.2 0.6 0.4] | 4 | [4.4 0 8 5.55] |
| 3 | [0 3 1 5] | 4 | 2 | 1 | [1.15 1.85 10.8 8.15] | [0.2 0.5 0.4 0.3] | 4 | [1.15 1.85 10 8.15] |
| 4 | [2 1 4 9] | 5 | 3 | 2 | [-1.55 3.7 1.2 1.3] | [0.8 0.3 0.6 0.2] | 1 | [0 3.7 1.2 1.3] |
| 5 | [1 2 8 3] | 2 | 4 | 1 | [1.3 1.85 11.55 7.85] | [0.7 0.5 0.9 0.2] | 3 | [1.3 1.85 8 7.85] |

$$u^j = \begin{cases} v^j & \text{if } r \le p_c \text{ OR } j = \delta \\ x^j & \text{if } r > p_c \text{ AND } j \ne \delta \end{cases}$$

- Step 8: Evaluate the fitness of bounded trial vectors

$$U = \begin{bmatrix} 4.55 & 0 & 10 & 10 \\ 4.4 & 0 & 8 & 5.55 \\ 1.15 & 1.85 & 10 & 8.15 \\ 0 & 3.7 & 1.2 & 1.3 \\ 1.3 & 1.85 & 8 & 7.85 \end{bmatrix} \qquad f_U = \begin{bmatrix} 220.70 \\ 114.16 \\ 171.17 \\ 16.82 \\ 130.73 \end{bmatrix}$$

$$f(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2$$

17

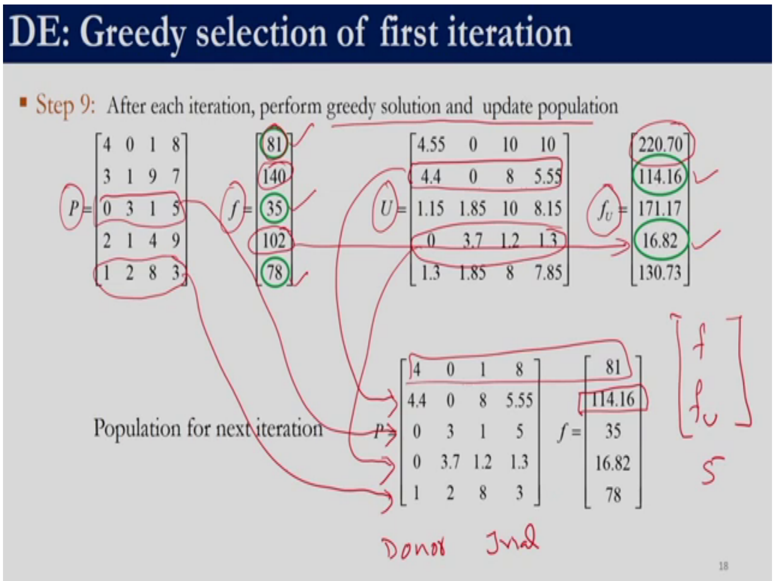
So, after generating trial vector 1 right. So, here we are not showing you the detailed calculation you can calculate it. So, this is what we have got for the first target vector right, for the second one the target vector is 3 1 9 7 and we need to chose three random variables between 1 and D right for mutation. So, we have taken the fifth solution the first solution and the third solution and we have applied this formula to calculate the donor vector. Once the donor vector is determined we need to generate 4 random numbers because we have 4 decision variables.

So, in this case we have taken 0.3 0.2 0.6 0.4 and also we need to generate a value of delta which should be between 1 and random number. So, in this case we have taken it to be 4 right and then we need to apply this condition to determine the trial vector. So, in this case if we see. So, 4.4 is coming from the donor vector because this is 0.3 and our crossover probability is 0.8. So, we are taking it from there.

Similarly, we need to find out all the other variables right and then bound it. So, this solution which is shown over here is bounded trial vector right. So, if you do the calculations we will be able to generate the trial vector after generating the trial vector you should check for the bounds right. So, similarly if we do for the rest of the 2 solution this is what we get right. So, we started with target vector we generated the donor vector with the help of this equation right and then again we generated the trial vector using this equation.

So, now, this trial vector is bounded. We can evaluate the fitness for each of this. So, the fitness is nothing, but for the first solution it is 4.55 square plus 0 square plus 10 square plus 10 square that will give you 220.70 right. Similarly the fitness for all the solution is to be determined right.

(Refer Slide Time: 51:34)



## DE: Greedy selection of first iteration

- Step 9: After each iteration, perform greedy solution and update population

$$P = \begin{bmatrix} 4 & 0 & 1 & 8 \\ 3 & 1 & 9 & 7 \\ 0 & 3 & 1 & 5 \\ 2 & 1 & 4 & 9 \\ 1 & 2 & 8 & 3 \end{bmatrix} \quad f = \begin{bmatrix} 81 \\ 140 \\ 35 \\ 102 \\ 78 \end{bmatrix} \quad U = \begin{bmatrix} 4.55 & 0 & 10 & 10 \\ 4.4 & 0 & 8 & 5.55 \\ 1.15 & 1.85 & 10 & 8.15 \\ 0 & 3.7 & 1.2 & 1.3 \\ 1.3 & 1.85 & 8 & 7.85 \end{bmatrix} \quad f_U = \begin{bmatrix} 220.70 \\ 114.16 \\ 171.17 \\ 16.82 \\ 130.73 \end{bmatrix}$$

Population for next iteration

$$P = \begin{bmatrix} 4 & 0 & 1 & 8 \\ 4.4 & 0 & 8 & 5.55 \\ 0 & 3 & 1 & 5 \\ 0 & 3.7 & 1.2 & 1.3 \\ 1 & 2 & 8 & 3 \end{bmatrix} \quad f = \begin{bmatrix} 81 \\ 114.16 \\ 35 \\ 16.82 \\ 78 \end{bmatrix}$$

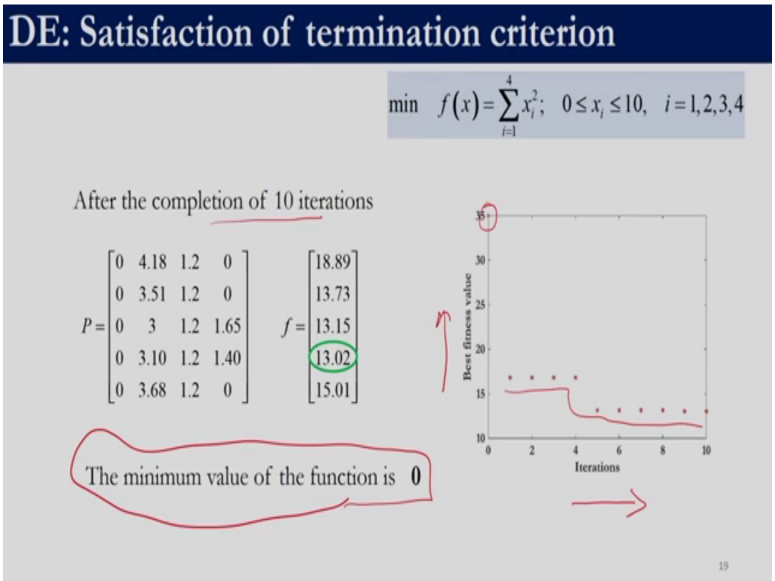
So, what we have is initially we started with this population p and this was the fitness. What we now have is u which is the set of trial vectors and its corresponding fitness function value right. So, now, what we are supposed to do is we are supposed to perform a greedy selection and update the population. So, the new population is going to be we are going to compare this 81 with 220.70. So, whichever solution is better will go to the next iteration. So, this is the first solution between 140 and 114.16, 114.16 survives right. So, this solution is taken right.

This solution becomes the second solution over here between 35 and 171.17 35 the solution with the fitness function 35 survives. So, this 0 3 1 5 will be taken as our third solution, for the fourth solution this is 102 this is 16.82 right. So, 16.82 is better. So, this will be our fourth solution. Similarly over here for the fifth solution 78 is better than 130.7.

So, this 1 2 8 3 will come over here. So, this is our new population. So, this is the population for which we will generate the donor vectors again and the trial vectors again. Similarly will bound at this trial vectors, evaluate its fitness function and then do a greedy selection. So, here when we applied greedy selection, if we see the solutions over here this solution 114.16 is actually bad than this 102 yet 114.16 has survived because of greedy selection right. 114.16. We compared with 140, we did not compare even with 102 right.

So, that is why 114.16 is able to survive where as 102 is not able to survive because 102 was compared with this 16.82. So, remember it is not like stacking all the fitness function and selecting the best 5 solutions. We are not selecting the best 5 solutions. We are performing a greedy selection right. So, greedy selection will help us to determine which is better among 2 population members. So, here in this case we are not combining f and f u and selecting the best 5 solutions right. We are doing 1 to 1 selection. So, that is how we update our population.

(Refer Slide Time: 54:02)



## DE: Satisfaction of termination criterion

$\min \; f(x)=\sum_{i=1}^{4} x_i^2; \quad 0 \le x_i \le 10, \quad i=1,2,3,4$

After the completion of 10 iterations

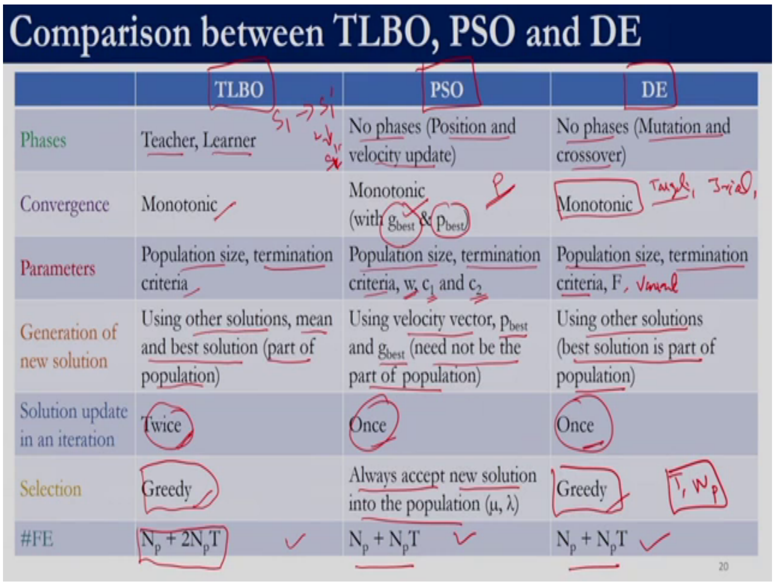
$$P=\begin{bmatrix} 0 & 4.18 & 1.2 & 0 \\ 0 & 3.51 & 1.2 & 0 \\ 0 & 3 & 1.2 & 1.65 \\ 0 & 3.10 & 1.2 & 1.40 \\ 0 & 3.68 & 1.2 & 0 \end{bmatrix} \qquad f=\begin{bmatrix} 18.89 \\ 13.73 \\ 13.15 \\ 13.02 \\ 15.01 \end{bmatrix}$$

The minimum value of the function is **0**

So, if we continued performing that at the end of 10 iterations. This is what we will get right. So, here if we say iteration versus best fitness function value, in each iteration if we see it is monotonically decreasing. So, it is started with a value of 35 and then it started decreasing right. So, if we perform few more iterations we might be able to get this globally optimal solution of 0 at the end of 10 iterations with these settings differential evolution was unable to get the globally optimal solution.

So, this completes our study of differential evolution. Now, we have studied 3 metaheuristic techniques right. So, let us make a comparison between those 3 metaheuristic techniques right.

(Refer Slide Time: 54:46)



## Comparison between TLBO, PSO and DE

| | TLBO | PSO | DE |
|---|---|---|---|
| Phases | Teacher, Learner $S_1 \to S_1'$ $\downarrow$ $S_1''$ | No phases (Position and velocity update) | No phases (Mutation and crossover) |
| Convergence | Monotonic | Monotonic (with $g_{best}$ & $p_{best}$) | Monotonic Target, Trial, |
| Parameters | Population size, termination criteria | Population size, termination criteria, w, $c_1$ and $c_2$ | Population size, termination criteria, F, Variant |
| Generation of new solution | Using other solutions, mean and best solution (part of population) | Using velocity vector, $p_{best}$ and $g_{best}$ (need not be the part of population) | Using other solutions (best solution is part of population) |
| Solution update in an iteration | Twice | Once | Once |
| Selection | Greedy | Always accept new solution into the population (μ, λ) | Greedy T, $N_p$ |
| #FE | $N_p + 2N_pT$ ✓ | $N_p + N_pT$ ✓ | $N_p + N_pT$ ✓ |

20

So, there are 3 techniques which we have learnt is teaching learning based optimization, particle swarm optimization and differential evolution right. So, teaching learning based optimization if you remember it was 2 phase algorithm. There was teacher phase and a learner phase right. So, here we call it as 2 phase because a solution S 1 was converted to S 1 prime like let say we generate S 1 prime, if it is better we retain S 1 prime and then S 1 prime was used to generate S 1 double prime using the learner phase right.

So, there were two phases in which we were updating one particular solution right. Solution can potentially get updated twice in the same iteration whereas, in particle swarm optimization and differential evolution we say that there are no phases right, but in particle swarm optimization we have a position and velocity update whereas, in differential evolution we have mutation and crossover. In all the 3 techniques the convergences monotonic right in TLBO if we see it is a greedy selection strategy that solution is discarded only if a better solution is coming in to the population.

So, that ensures this monotonic convergence, in particle swarm optimization whether the solution is better or not it enters the population right, but we also keep track of the personal best and the global best. Since we always keep track of the global best even if the best solution is getting discarded from the population, a copy of it will be in the g best right. So, that ensures monotonic convergence and differential evolution also.

We employed a greedy strategy where in we compared the target 1 with trial 1 right. So, this trial 1 enters into the population if and only if it is better than this target 1 right. So, that way we do not lose track of the best solution which we have so far. So, parameters if we see us in TLBO we had only 2 parameters to be it set like the user had to set only 2 parameters, one was population size and the other one is termination criteria.

In all the 3 cases we have considered the termination criteria to be the number of iterations whereas, in particle swarm optimization we need to provide the population size, the termination criteria, inertia weight and the acceleration coefficient c 1 and c 2. In differential evolution we need to provide population size, termination criteria, f right and also we need to select what variant we are supposed to use right. So, the various variants that we had seen was one is with respect to the crossover right.

We can either use uniform crossover or we can use exponential crossover and there were 9 mutation strategies we had seen right. So, we need to select one of those mutation strategies. So, in a way that is also to be defined by the user which a mutation variant is to be selected and which crossover is to be selected that is to be given by the user. So, while generating a new solution in teacher, teaching learning based optimization in the partner phase we randomly selected one partner in the teacher phase. We used the mean best solution right.

So, that is how we generate a new solution. In PSO, we generated a new solution using the velocity vector, the personal best and the g best whereas, in DE we took except for that one variant where we take best the other variants randomly take solutions from the population to generate a new solution right. In teaching learning based optimization as well as in differential evolution the best solution is part of the population right. The best solution is inside the population that need not be the case in particle swarm optimization. So, that is one crucial difference between TLBO, DE and PSO.

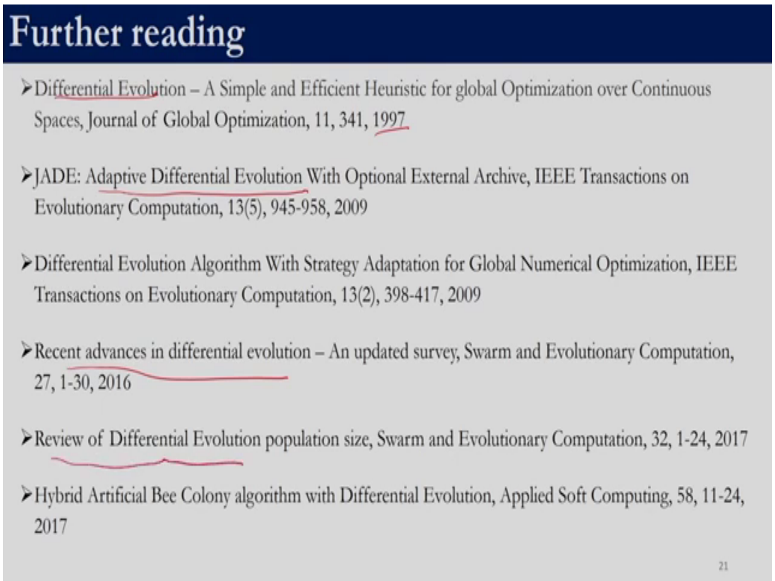
So, in TLBO there are 2 phases where in solution can get updated the teacher phase and the learner phase, whereas in particle swarm optimization we only update the position once for one population member in one iteration and similarly in DE we generate one donor vector and one trial vector for every member right. So, that is why we have 2 updates in TLBO and 1 update in DE and particle swarm optimization.

So, the selection strategy in TLBO was greedy right same thing in differential evolution. The selection strategy is a greedy selection strategy whereas; in particle swarm optimization the new solution was always accepted into the population. So, number of functional evaluation in all the 3 cases, it is deterministic the technique is stochastic, but the number of functional evaluation is deterministic. Deterministic because once we fix T and N p we can uniquely determine the number of functional evaluation.

So, in TLBO since there are 2 solution updates, we will have to evaluate the function 2 N p times in one iteration so, the total number of functional evaluation which we have previously stated is N p plus 2 N p T where as in particle swarm optimization and differential evolution the number of functional evaluation for a particular member in a particular iteration is only once right. So, if we have N p members, we will have N p evaluation in one iteration, and since we performed T iterations, the total number of functional evaluation is N p T.

We have another N p functional evaluation for the initial population. So, the expression is N p plus N p T right. So, this gives a quick comparison between the 3 metaheuristic techniques that we have studied so far. For differential evolution you can look in to these papers right.

(Refer Slide Time: 60:35)



## Further reading

- Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341, 1997
- JADE: Adaptive Differential Evolution With Optional External Archive, IEEE Transactions on Evolutionary Computation, 13(5), 945-958, 2009
- Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization, IEEE Transactions on Evolutionary Computation, 13(2), 398-417, 2009
- Recent advances in differential evolution – An updated survey, Swarm and Evolutionary Computation, 27, 1-30, 2016
- Review of Differential Evolution population size, Swarm and Evolutionary Computation, 32, 1-24, 2017
- Hybrid Artificial Bee Colony algorithm with Differential Evolution, Applied Soft Computing, 58, 11-24, 2017

21

So, this is the paper which proposed Differential Evolution subsequently there have been various development this is also Adaptive Differential Evolution right. So, here them modified the basic differential evolution a new variant was propose based on differential evolution right and then this is recent advances in differential evolution you can obtain from this paper. So, this article talks about review of differential evolution with that we will conclude this session on differential evolution.

Thank you.