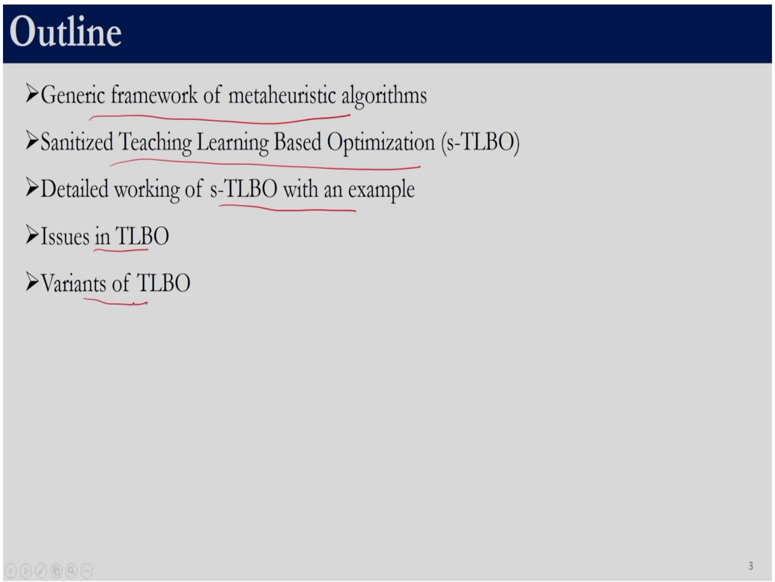
Computer Aided Applied Single Objective Optimization
Prof. Prakash Kotecha
Department of Chemical Engineering
Indian Institute of Technology, Guwahati

Lecture – 06
Teaching Learning Based Optimization

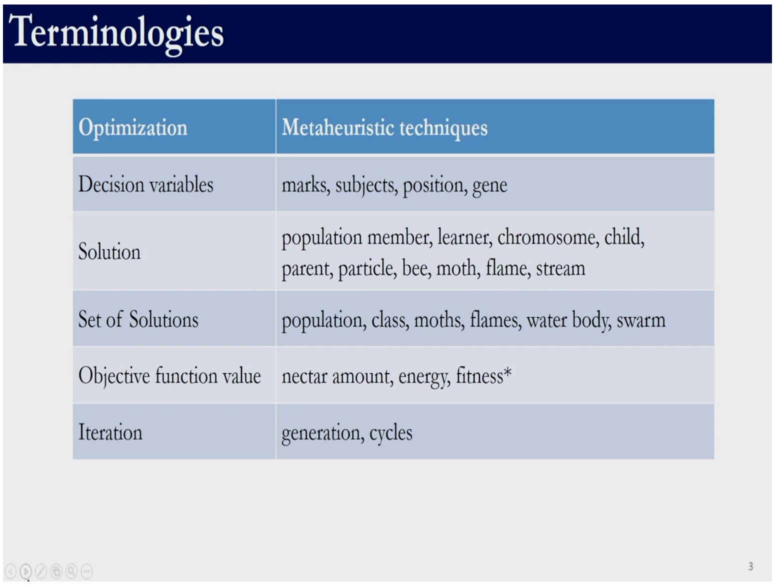
So, in this session, we will be looking at Teaching Learning Based Optimization. Teaching learning based optimization is a newly proposed technique, it was proposing 2011 right. Among the five techniques which we are going to see as part of this course T L B was the latest proposed technique. We have taken T L B as our first meta heuristic technique to learn. Because, we found that; T L B was much easier to understand as well as it does not have complexity with respect to tuning of the algorithmic parameters unlike other algorithms there are only two tuning parameters over here.

(Refer Slide Time: 01:01)



So, the outline of the session is that; first we will discuss the generic framework of metaheuristic algorithms. We will follow it; we will follow it up with the description of sanitized teaching learning based optimization. We will demonstrate the application of s TLB on a four variable problem right. Then, we will look into some of the common issues associated with TLBO in literature and we will also look at some of the variants of TLBO.

(Refer Slide Time: 01:27)

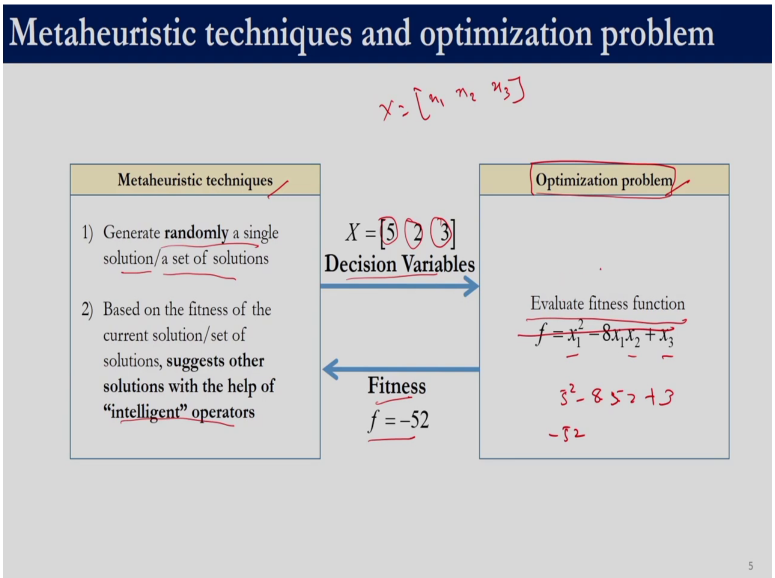


Before getting into TLBO, let us familiarize ourselves with the terminologies that; we will be using in metaheuristic techniques and how they relate to the terminologies. We have been using so far in optimization. So, for example, in optimization; we were using the decision variables.

So, decision variables are also known as marks or subjects or position or gene depending upon the metaheuristic technique that; we are working with a solution is called as a population member, a learner chromosome, a child or parent there is a set of solutions are known as population, class, moths, flames, water body or swam. The objective function value which we use in the optimization literature is also known as nectar amount or energy or fitness depending upon the metaheuristic technique which we are working with.

Thickness in particular; usually, means objective function value except for a few algorithms. So, in a b c fitness does not necessarily exactly correspond to the objective function value. We will come to it later, but most of the algorithms use the terminology fitness to indicate the objective function value. So, iterations are also known as generations or cycles in meta heuristic techniques.

(Refer Slide Time: 02:40)



This slide shows the communication between metaheuristic techniques and the optimization problem. Most metaheuristic techniques generate randomly a single solution or a set of solutions right. So, when we say solutions; it is nothing but, the value of the decision variables. So, if you have a problem with five decision variables then we have a vector of five values within the bounds of the problem.

Once we generate the solution right. So, those decision variables are sent to the optimization problem. And over here the fitness function is evaluated; that is the only communication between the metaheuristic techniques and optimization problem for every solution right. We will get the fitness function value. So, the fitness function value is returned back to the algorithm. So, the algorithm sends the decision variables and the problem sends back the fitness function.

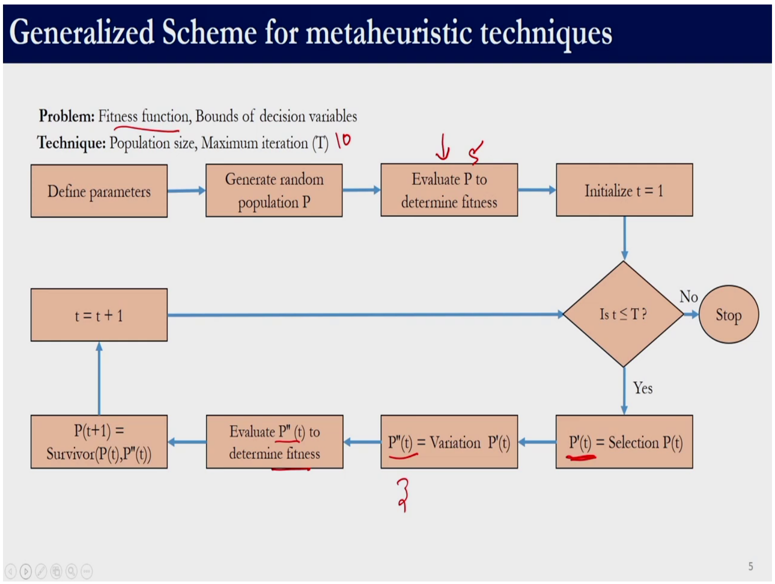
Based on this information; the fitness and the current set of decision variables which were sent to the optimization problem. The algorithm uses intelligent operator to come up with new set of decision variables which is again send to the problem and the fitness function value is received from the problem. This communication goes on repeatedly till we complete a specified termination criteria. So, if you take an example right.

So, let us assume that; we have a three variable problem x 1 x 2 x 3 right. And they are arranged in this order right, that the first variable is x 1, second variable is x 2, third variable is x 3 right. So, here 5 2 3 indicates that; a value of 5 has been assigned to decision variable 1, a value of 2 has been assigned to decision variable 2 and the value of 3 has been assigned to decision variable 3 right.

So, now, that is sent to the optimization problem. Let us assume that our optimization problem is to minimize this function f is equal to x 1 squared minus 8 x 1 x 2 plus x 3 right. So, here if we see or to calculate the fitness all that we are required to do is; 5 square minus 8 into 5 into 2 plus 3. So, if we calculate this value it should come out to be minus 52 right and that value is sent back to that metaheuristic techniques. So, this is the only communication that is going to happen between the metaheuristic technique and the optimization problem broadly if we see right.

So, the optimization problem could be anything. It can be even an experiment over here, instead of this a fitness function; it can be even an experiment right. And the result of the experiment a quantitative measure is to be sent back to the metaheuristic technique.

(Refer Slide Time: 05:10)



So, let us see the generalized scheme for metaheuristic techniques right. So, for to employ any metaheuristic techniques; we need the fitness function and bounds of the decision variable. So, that comes with the problem definition itself. So, if we have an optimization problem; we will have a way to estimate the quality of a solution in most cases it is in terms of the objective function, a mathematical expression and the number of decision variables and their respective bounds right.

So, the first step is to define the parameters. So, metaheuristic techniques come with their own set of parameters which have to be provided by the user. So, far most of the technique the two commonly required parameters are; population size and maximum number of iteration. Maximum number of iteration will tell us when to stop the procedure whereas, population size is the number of solution which the technique is going to work with.

So, the first step is to generate a random population within the domain of decision variables followed by evaluation of the fitness function of this population. So, this is the place where in we need the fitness function. Once that is done; we initialize the counter t is equal to 1 to keep account as to how many times we are repeating the cycle. So, these are iterative techniques. So, we have a counter to keep track of the number of iterations that we have completed so far.

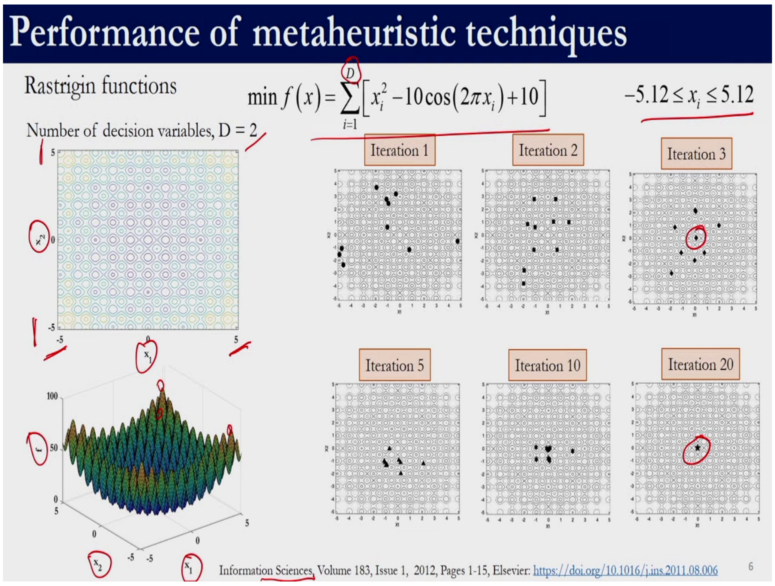
So, we initialize t is equal to 1 and then check for this condition; whether t is less than or equal to capital T right. So, in the first place let us say; if my number of iterations is 10. So, initially t is equal to 1, so it will enter this loop. This is where the technique actually starts. Now that we have a population. Few members of the populations are selected, those elected members are known as P prime like P prime of t. So, you already we already have a population; we select few members from the population.

We perform some operations to them, so as to vary them right. So, the solutions which we have; which we have selected from the initial population; we vary them. So, as to get a new set of solutions. So, once we have the new solutions; we will evaluate the fitness function of the new solutions. So, now we have let us say; we started with a population size of 5 right. So, let us say, we generated we selected a few solutions from the 5 solutions and then we employed the variation operator.

Let us say we came up with 3 new solutions right. So, now, we have 8 solutions and they are fitness function value. We will employ a survival strategy. So, that out of this 8 solutions 5 solutions are selected right, that would complete 1 iteration right. So, we increased the iteration counter and then go and check for this condition whether this condition is satisfied or not.

So, as long as this condition is met this cycle is repeated and once this condition is met; we terminate the procedure and that is the end of the algorithm. Since we have a population at any given point of at any given iteration the best solution in the last iteration is considered to be the optimal solution determined by that particular metaheuristic technique.

(Refer Slide Time: 08:05)



So, let us see how the performance of a typical metaheuristic technique will be. So, this is a benchmark function known as Rastrigin function right. It is a 2 variable function, it is a scalable function as in like by varying this value of D, you can have it as a 2 variable problem 3 variable problem and so on so forth. So, if we fix the value to be 2, we get this third space right. So, x 1 and x 2, it varies between minus 5.12 to 5.12.

So, both this decision variable have the same bounds between minus 5.12 to 5.12. This shows the contour plot. So, we have discussed in the last class; what are contour plots and this shows the figure below shows the surface plot of it right. So, this is x 1 this is x 2 and this is f right f is the objective function value. So, for a particular value of x 1 and x 2, what is the objective function value can be determined using this expression. Once we know x 1 and x 2 and that is how this plot has been made right.

So, now, if you look at this plot there are a large number of peaks. Here If you see, there are large number of peaks and there are large number of valleys. So, the job of our metaheuristic technique is to locate the minima in this complex function right. It does not know where the minima is. Initially, we are starting with a random population. So, if it is a intelligent technique; it should be able to figure out by itself as to where the minimum solution is located and should move towards it right.

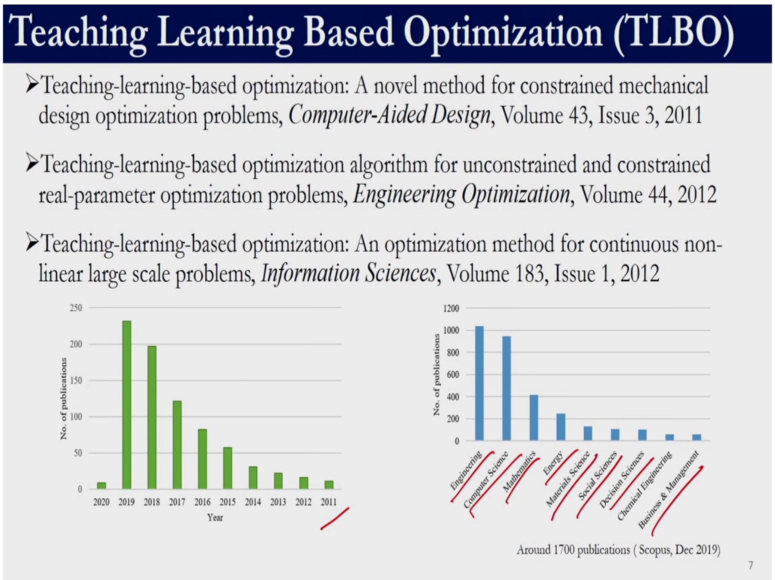
So, the as we said, as we have seen in the previous slide that; these are iterative techniques. So, let us see how this will pan out. So, initially, we said we will generate few solutions right, let us say we have 10 solutions. So, initially, those 10 solutions in the search space will look like this as shown in iteration 1 and then we perform those a selection, variation and survival strategy to come up with iteration 2. So, the solution in a iteration 1 and the solution in iteration 2, if we see they have moved from their original place and the same thing; we can see in 3 right.

So, this particular solution, if you see it has come closer to 0 0 ok. So, if we keep doing this for 20 iterations; this is what happens. So, this is a typical performance of metaheuristic technique right. The initially, the population are scattered as the iteration progresses, they are moving towards that one particular point right. In this case it happens that this point is nothing but, the global optima right it manages to reach the global optima, for this problem the optima is 0 0.

So, if you substitute 0 in this objective function; you will see that you will get a value of 0 for a f right. So, this is how the performance of where typical metaheuristic technique is going to look like. That initially, the solutions have to be generated and scattered randomly, we employed those three major steps; selection, variation, survival.

We do it repeatedly for multiple times and the expectation is that; randomly scattered solutions will converge to a particular point. It is important to remember that it is not necessary for all the points to converge right. As long as one of the solutions in the population is converging towards the optima, it is a good technique, because the rest of the solution members can still explore the search space right.

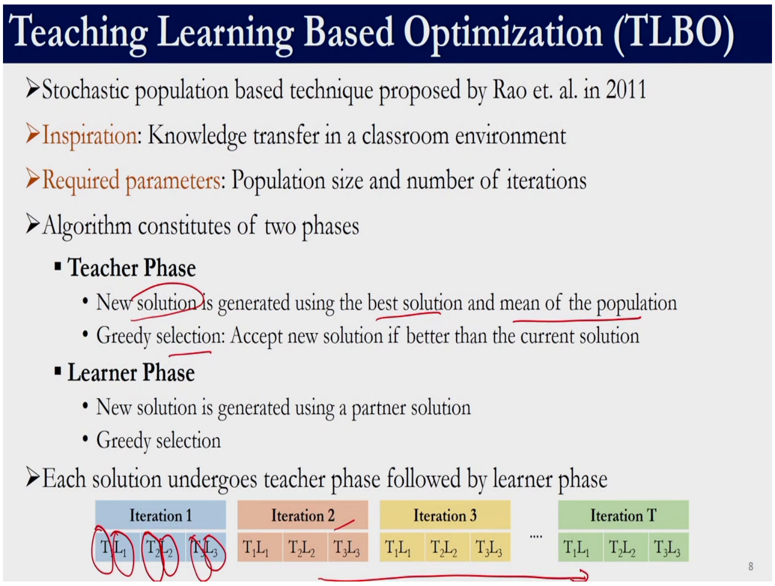
(Refer Slide Time: 11:18)



So, coming to teaching learning based optimization; teaching learning based optimization was proposed in 2011 right it is commonly not found in regular textbooks right.. So, if you want additional learning you can look into this paper or one of these two papers. So, all these three papers had proposed teaching learning based optimization. So, this plot shows the popularity of TLBO. It was proposed in 2011 and ever since the number of publications, citing or teaching learning based optimization has been increasing exponentially.

This plot shows the use of TLBO across various fields. So, for example, it has been used in engineering, social sciences, business decision, sciences, mathematics, computer science energy, material sciences and many other fields. So, TLBO even though it is a recently proposed algorithm seems to be gaining increased attention. So, as we discussed in the beginning; we have chosen TLBO, because it seems to be working on many of the problems and it is One of the simplest techniques out there. So, we thought that we will start with the simplest technique and then build on that right.

(Refer Slide Time: 12:25)



So, T L B O is a stochastic population based technique proposed by Rao et al. So, the inspiration for this algorithm is a knowledge transfer in a classroom environment. The required parameters for this metaheuristic technique is population size and number of iterations. So, if you want to use TLBO, we need to fix a population size; which is an integer right. So, basically, what we mean is that; we will be working with that number of solutions at any given point of time.

So, that is population size and then we have a number of iteration which basically tells when to stop the algorithm. So, if we look at a classroom environment; the teaching usually happens in two phases; one is the teacher phase and the learner phase. In the teacher phase; the students learn from the teacher. Whereas, in the learner phase; the students interacting among themselves and trying to increase their knowledge right.

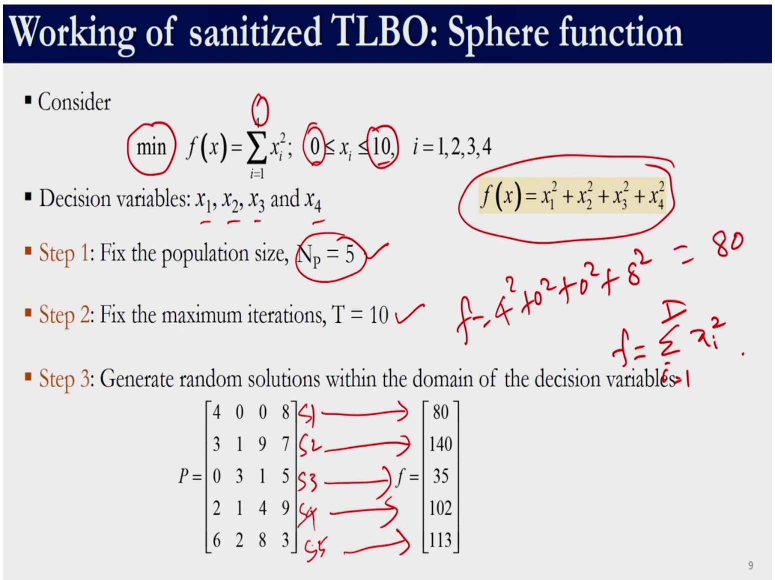
So, TLBO mimics these two phases; teacher phase and learner phase. In teacher phase; we will be generating a new solution right using the best solution available so far and the mean of the class or mean of the population. So, in TLBO, the population is also known as a class right. And then, we will employ a greedy selection strategy right. So, in greedy selection strategy we will see, if the new solution which we have generated is better. So, if this new solution is better; we will bring that solution inside the population and eliminate the solution which was used to generate this new solution.

So, that is why, we have the terminology greedy selection strategy. In learner phase also we use a different variation operator to generate a new solution right. Over here, we do not rely on the best solution; whereas, we rely on a, what is called as a partner solution. A partner solution is a randomly selected member from the class. And we will employ a greedy selection strategy. So, again the same thing that if the new solution which we generate is better than the solution which is used to generate it then we will take the new solution.

If the new solution is not better, then we will retain the current solution right the new solution is discarded in that case. So, this is a important step in teaching learning based optimization right. So, what we are going to do is; we have a set of solution. The first solution will undergo teacher phase. The first solution will undergo learner phase, only then second student will undergo the teacher phase right. So, the first student completes the teacher phase as well as the learner phase and subsequently, the second student or the second member of the class undergoes teacher and learner phase right.

So, this is the second student this is the third student right. So, if we have 5 students, we will have T 4 L 4 T 5 L 5 that would complete 1 iteration right. And we are expected to do T such iterations right. So, again in iteration 2; the first solution will undergo teacher phase, the first solution will undergo learner phase. Once that is complete; the second solution will undergo teacher phase and the second solution will undergo learner phase. So, this cycle is repeated till we complete the specified number of iterations.

(Refer Slide Time: 15:21)



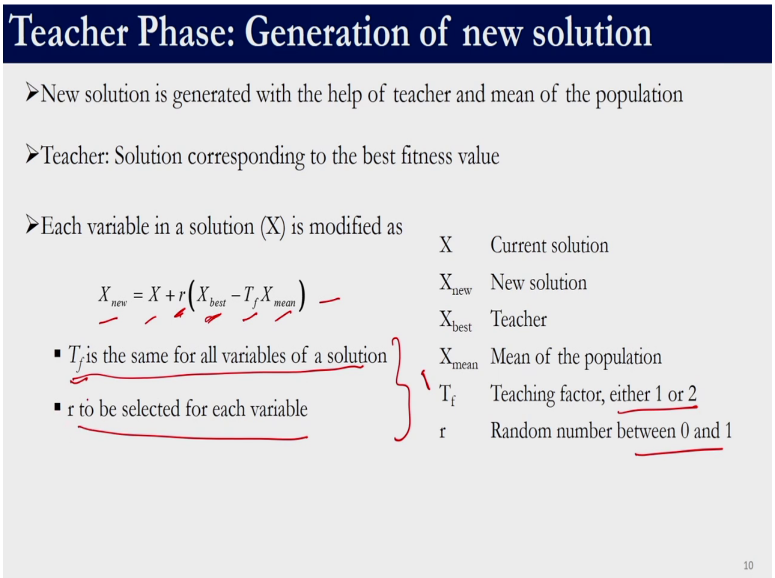
So, we will now see the working of sanitized TLBO on a sphere function. The objective function in sphere function is given by the summation of square of the decision variable. So, here we will take 4 decision variable right. So, domain of the decision variable is 0 to 10, our objective function is x 1 squared plus x 2 squared plus x 3 squared plus x 4. So, our objective is to minimize right. So, this function is actually a scalable function.

So, instead of four variable a problem, this can also be converted into a five variable problem because the objective function is actually given by sigma i equal to 1 to D x i square. So, depending upon the value of D this can be a four variable problem five variable problem. We can scale this problem as we want. The first step in a sanitized T L B O is to fix the population size.

So, we are taking a population size of N P is equal to 5. We need to fix the number of iterations because, that is what is going to tell us as to when to complete the algorithm. So, T is equal to 10. The next step is to generate a random initial population. So, the domain of the variable is given 0 to 10, we are supposed to generate five random solutions. So, this is solution 1, solution 2, solution 3, solution 4 and solution 5, since my population size is 5 right.

So, in this case, we have generated all random numbers. So, at least for the first iteration it is easier to calculate the objective function. So, the objective function of S 1 is 80, the objective function of S 2 is 40, objective function of S 3 is 35 4 is 102 5 is 113. So, the way to calculate this objective function is f is equal to in the, for the first solution it is 4 square plus 0 square plus 0 square plus 8 square. So, this will turn out to be 80 right. So, similarly, we calculate the objective function value for all the solutions and this is also known as the fitness function value.

(Refer Slide Time: 17:10)



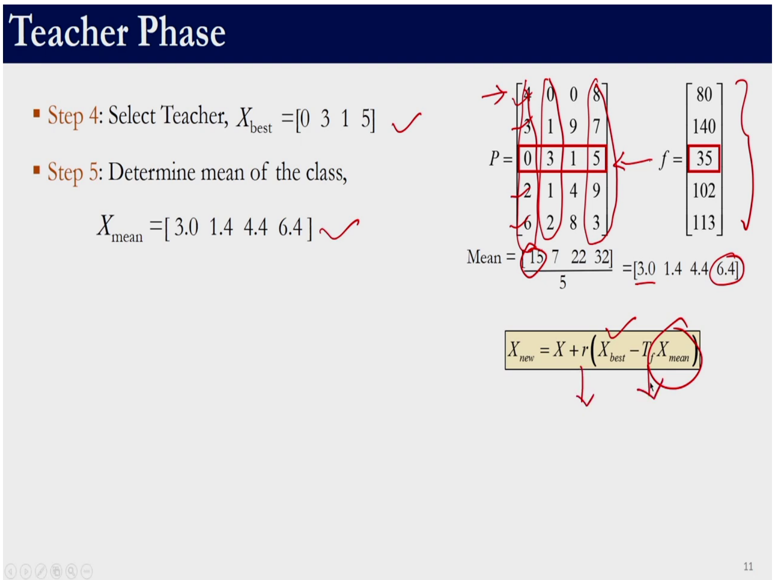
So, the next phases the teacher phase. In teacher phase; we generate a new solution with the help of teacher and mean of the population. So, the teacher is the solution which corresponds to the best fitness function value. So, now, we have the population and we also have the fitness function values of the population. So, from the fitness function value we can identify which is the least value right, because we are solving a minimization problem.

Once, we identified the least value in the fitness function; the solution corresponding to it is the a solution which has the best fitness functional value. So, that solution will act as teacher. Once we have selected the teacher uh; we need to generate a new solution with the help of this equation. So, every member of the population is to undergo the teacher phase. So, whichever a member is undergoing teacher phase; that is that will correspond to this X. r is a random number between 0 and 1, X based is the teacher right, the best solution in the class or the population.

T f is our teaching factor; T f is known as teaching factor and it has to be either 1 or 2 right. X mean is the mean of the population. So, you need to remember that T f is constant for all the variables. So, if I have a five variable problem even then I need to have only one scalar which has to be either 1 or 2 for generating one new solution. If we are generating a second solution; we will select T f again randomly right.

So, for 1 solution T f only 1 T f is needed which is not the case for r right. So, if I have three variable problem then I need to select 3 values of random number between 0 to 1 right. So, r is to be the r is to be selected for each variable whereas, T f is the same for all variables of the solution. Remember this is only for generating one solution when I want to generate another solution using teacher phase; again, I need to generate T f and r right.

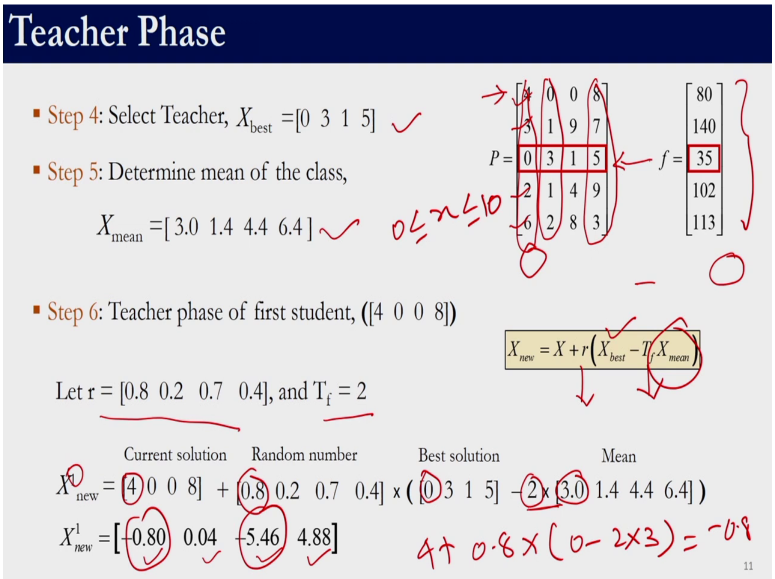
(Refer Slide Time: 19:05)



So, now let us apply this teacher phase for our first population numbers. So, this is the population member which is undergoing the teacher phase. The first population member 4 0 0 8. The best solution if we see; here it is 35 right. So, the solution corresponding to it is the teacher. So, now, we have selected teacher. So, in this equation I know this one right. Now, I need to say I need to find out the mean right. So, mean is calculated by taking the average of all the columns. So, the average of this, the average of this right.

So, in this case it is 4 plus 3 plus 2 plus 6, that comes out to be 15, 15 by 5 3. Similarly, for this column; it is the summation of this which should turn out to be 32, 32 by 5 6.4. So, now, we have mean of the class. Remember, the fitness value of the mean solution is not required and it is definitely not the mean of this right. So, we only require the mean solution, not its fitness. We will not spend a functional evaluation trying to evaluate the fitness function of this mean solution, because we do not require it right.

(Refer Slide Time: 20:14)



So, now we need this random values and we need this teaching factor. Let us take the random very random values to be 0.8, 0.2, 0.7, 0.4. I need four values, because it is a four variable problem. Whereas, I need only one teaching factor and the teaching factor has to be either 1 or 2. So, the new solution is given by this equation. So, X new 1 is equal to this 1 indicates that we are generating a new solution for the first solution with the help of the first solution.

So, this is the current solution plus the random number which we have selected multiplied by the difference of best solution and mean with the teaching factor right. So, if we calculate this it will come out to point a minus 0.80, 0.04, minus 5.46 and 4.88. So, it is element to element a operation. So, for example, the first this minus 0.8 is determined by 4 plus 0.8 into 0 minus

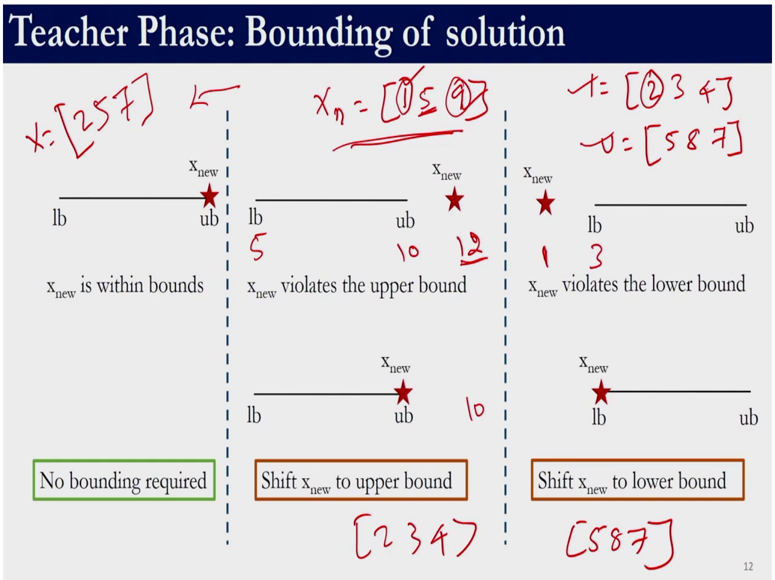
2 into 3 right. So, the first variable is 4, the random number for the first variable is 0.8, the best solution; first variable of the best solution is 0 and 3 is the first variable in the mean, this 2 is the teaching factor.

So, this should give us minus 0.8 right. So, now, we have generated a new solution. Now that, we have generated a new solution; we need to find out the fitness of this function right. Only when we have the fitness of this function; we will be able to say whether this function this solution is better than the solutions which we already have.

So, what we will do is; we will calculate the fitness function of this, but there is a small problem over here right. If you remember the domain of the decision variable was between 0 and 10 and this variable and this variable do not fall in this domain. The objective function is to be evaluated only for solutions within the bound. We can still calculate with this values we can still calculate the fitness function value, but no matter how good or bad it is useless, because I cannot use this solution.

The solution violates my bounds right. So, what we need to do is; first bound this solution. Somehow we need to bring it back into the region. Once it is in the region; we can evaluate the fitness function value right. So, let us see how to bound this solution.

(Refer Slide Time: 22:31)



So, there can be three cases for bonding of the solution; the first case is given the lower bound and the upper bound. The value of a particular variable is within its bounds. So, for example, in this case, it is somewhere within its bound. In this case, the variable is already within its bounds. So, we do not need to bound, it can happen that the variable exceeds its upper bound. So, for example, if my lower bound is 5 and my upper bound is 10; the variable has a value of let us say 12 right.

So, in this case it is violating the upper bounds. So, since it is violating the upper bound; we need to bring it back into their domain. So, what we will do is; we will push it to the boundary. So, this 12 is overwritten with a value of 10 right. So, because then it comes to within the bound right. Similarly, the violation can be in the lower bound. So, for example, if my lower bound is let us say 3, the teacher phase can give me a solution in which this particular variable has a value of 1 as its lower bound right. So, now, the solution is violating the lower bound.

So, what we will do is; we will again push it to its boundary right. So, this will become 3. So, in this example. So, this is just a single variable example; what I demonstrated. So, for example, consider a case where in the lower bound is it is a three variable problem, 2 3 4 right and the upper bound is let us say 5 8 and let us say 7. And let us say I have a solution I have generated a new solution which is actually 1, let us say 5 and let us say 9 right. So, in the solution if I, if we see this 1 actually violates the lower bound, because the lower bound for the first variable is 2.

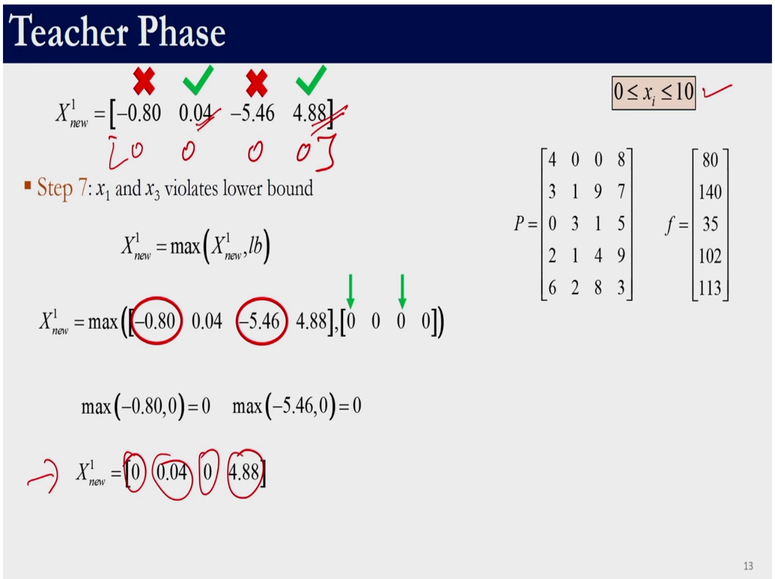
So, the corrected value would be the boundary solution would be X will be 2. For the second variable it is 5. The new solution has the variable value of 5, 5 is within the bound between 3 and 8 it lies within the bound. So, I do not need to bound it, it is already within the bound. So, 2 5 for the third variable the lower bound is 4, the upper bound is 7 right, but what the value that I have is 9. So, in this case we will push the solution to the upper bound so 7.

If this is our upper and lower bound and if this is the new solution which has been generated using teacher phase. Bounding of the solution will give us this. So, this bounding procedure is commonly known as corner bounding, because we are pushing the variable which is violating its bound to its particular bound the lower bound or the upper bound.

So, bounding is required almost in all metaheuristic techniques, because the operators designed are not necessarily designed to give values within the bounds right. There are some operators which can give you values within the bounds, but most operators do not do that. So, in that case a bounding strategy has is to be used to bound the solutions.

There are various other bounding strategies that can be used, but most of the algorithms use this corner bounding right. So, again to emphasize this one it is not the entire solution is not bounded right. So, for example, if this is 1 5 9, it is not the entire solution is pushed to one of the bound. So, the equivalent solution for this is not 2 3 4 or 5 8 7 right. Only the variable that is violating the bounds. So, 1 is violating the bound and this 9 is violating the bound. So, only those variables are to be corrected ok. So, this is the bounding.

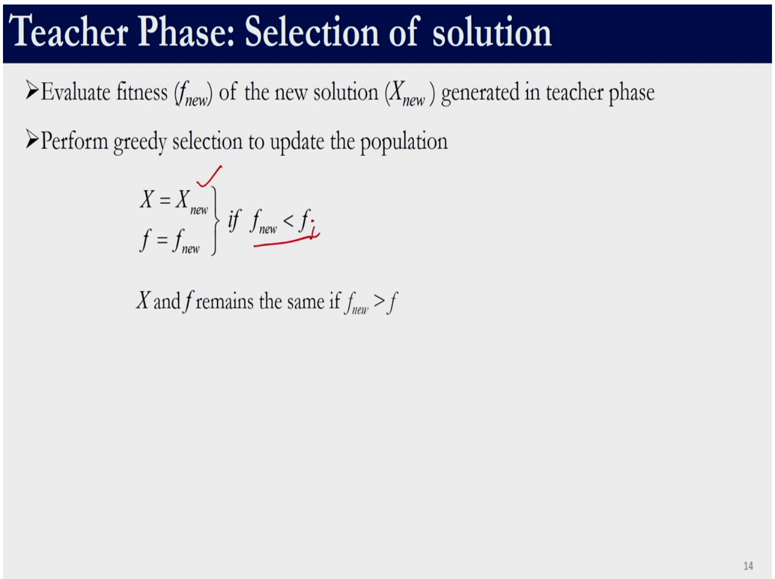
(Refer Slide Time: 26:11)



So, now we know how to bound the solution. So, the solution which we had was minus 0.8 0.04 minus 5.46 and 4.88 and the domain of the decision variable was between 0 and 10. So, now, we can see that x 1 and x 3 violates the lower bound. So, I can use this operator this max operator right. So, what we are going to do is this we have 0 0 0 0 over here that is our lower bound. So, we are going to compare element to element.

So, what is the maximum of these two. Maximum of minus 0.8 and 0 is maximum is 0. Maximum of 0.04 and 0 is 0.04 maximum of 0 and minus 4 5.46 is 0 and maximum of 4.88 and 0 is 4.88. So, this is our new solution, this solution is within the bounds. So, here we can see that these two variables did not violate the bounds. So, they are retained in the new solution as such right. Now that, we have generated the new solution. The next step is to evaluate the fitness of this solution right.

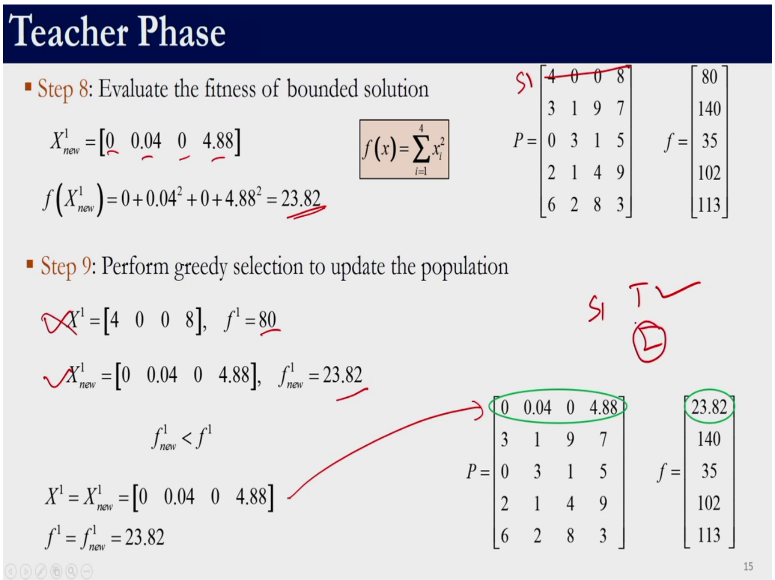
(Refer Slide Time: 27:10)



So, once we evaluate the fitness of the solution; subsequently, we will perform a greedy selection strategy to update the; to update the population. So, if the new solution is better; if the fitness function of the new solution is be ter than the fitness of the solution used to generate it. So, this is fi. So, the ith solution, then the new solution will be taken inside the population right else the new solution will be discarded right.

If this condition does not hold then the new solution would be discarded. If this condition holds; the new solution enters the population in place of the solution which was used to generate this new solution.

(Refer Slide Time: 27:50)

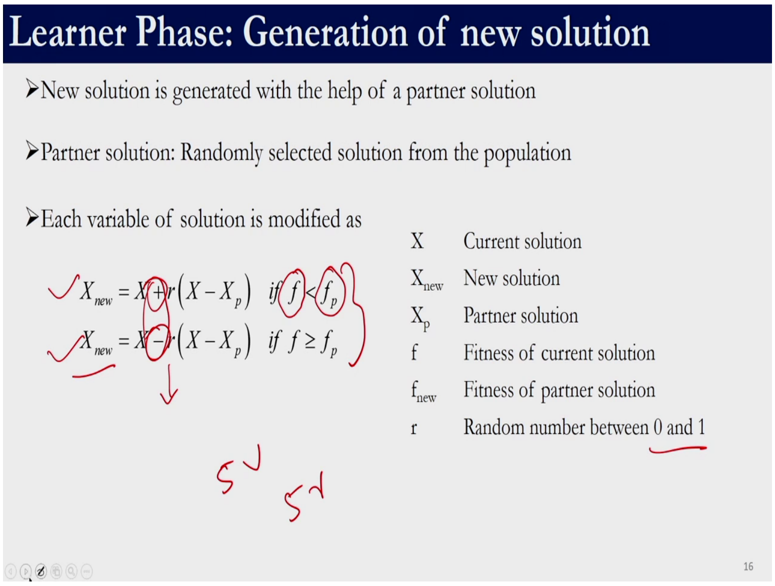


So, let us now apply this to our population right. We need to remember that; we are still at solution 1 right. Solution 1 is supposed to complete the teacher phase right. We have generated a new solution; now we can evaluate the fitness function of this new solution 0.04 0 4.88. So, if we plug that into the objective function; we get a fitness function value of 23.82 right.

So, this was the solution which was used to generate the new solution and this is the solution. The fitness of both of the solutions are given over here right. So, since 23.82 is less than 80, this new solution is actually better right. And this X 1 is to be replaced right. So, if I update the population; this solution enters it and this solution is eliminated from the population. So, this completes teacher phase of the first solution right.

The second step is learner phase of the first solution. Remember, we need to complete the teacher and learner phase of the first solution only then we are supposed to do the teacher and learner phase of the second solution right. So, that is to be taken care of right. So, first solution teacher phase is done right. Now first solution has to undergo the learner phase right.

(Refer Slide Time: 29:08)

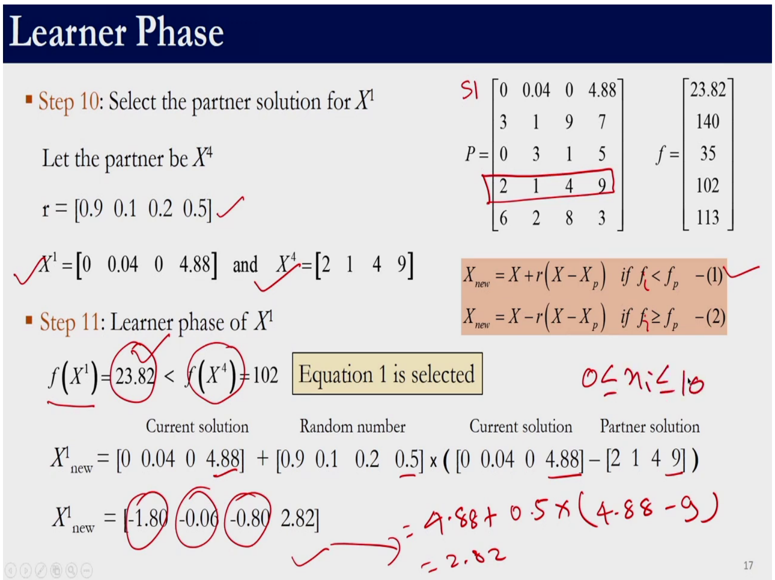


So, let us see what is learner phase. So, in learner phase a new solution is generated with the help of a partner solution. So, the partner solution is randomly selected solution from the population. So, we have a population will randomly select one member of the population who will act as partner to the solution which is undergoing that learner phase. In learner phase; there are these 2 equations which are available to generate a new solution.

So, the selection of the appropriate equation depends upon the fitness function of the member which is undergoing the learner phase and the fitness function of the partner right. So, if the fitness function of the member is better than the fitness function of the partner; then this equation is to be used, else this equation has to be used. So, the only difference between these two equations is that; in one case it is you need to add this vector and another case you need to subtract this right.

So, again here r is a random number between 0 and 1, we need to generate as many random numbers, as many the number of decision variables for generating one solution right. So, if i have a 5 variable problem I need to generate 5 random values between 0 and 1 to generate 1 new solution right. So, if we want to generate another new solution when the second solution enters the learner phase; we will have to generate again 5 new random numbers right. So, let us apply learner phase.

(Refer Slide Time: 30:31)



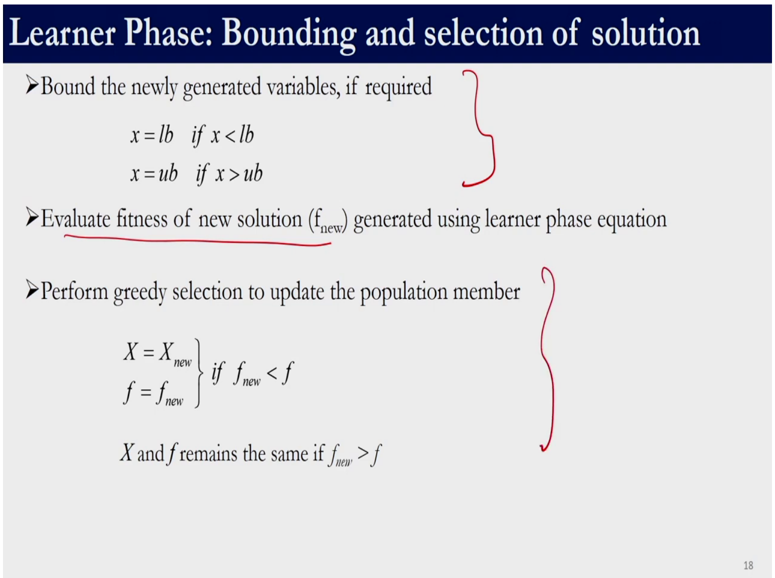
So, we are still at with the solution 1. So, solution 1 is supposed to go to the learner phase right. Let us randomly select one of this. So, let me select that solution 4 will act as the partner for solution 1 right. So, we have a solution 1 here, we have a solution 4 and we need 4 random numbers. So, let these be the random numbers. Now we will have to select which of these equations are to be applied right.

So, the fitness function of the first member is 23.82 and the fitness function of the fourth member which is the partner that is 102. So, between these two solutions; if we see 23.82 is better. So, f here there has to be a i right. So, this equation is valid right. Equation 1 is to be selected. So, if we apply equation 1 so, equation 1 specifies that the new solution is current solution plus the random number multiplied by the difference between the current solution and the partner solution.

So, here also it is the same thing. So, for example, this 4.88 has to be added with this number 0.5 into 4.88 minus 9. Because, this I am generating for the fourth variable, you can generate, you can check for the rest of the three variables. So, it is these are element to element operation. So, this value should turn out to be 2.82. That is why this 2.82 is over here.

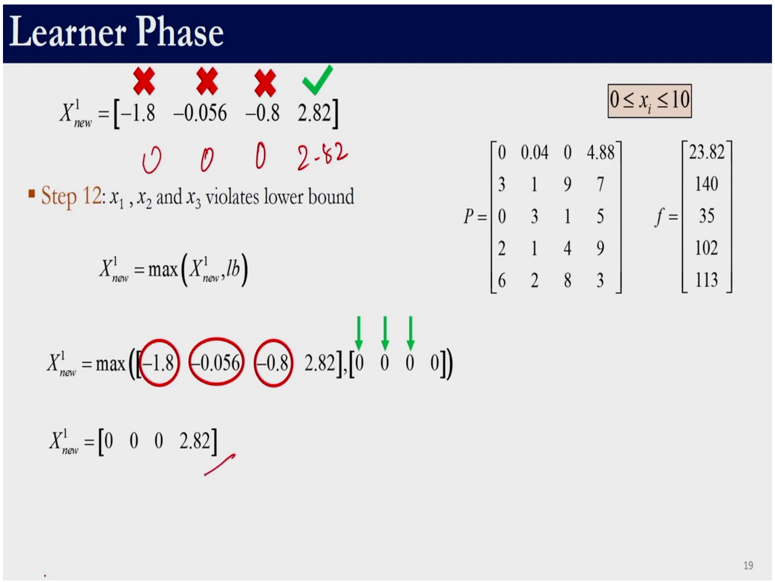
So, now, we have generated a new solution again the same problem which we encountered in teacher phase is being encountered over here that; these three variables are not within the bounds of the decision variable, the bound of th bounds of the decision variable was between 0 and 10 right.

(Refer Slide Time: 32:20)



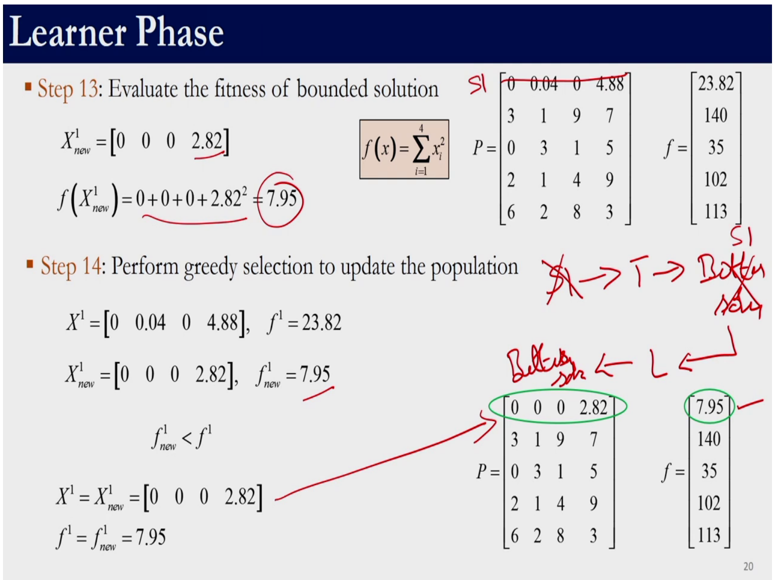
So, again we will employ a cornered bounding strategy to bound it right. So, this is what it specifies; we need to bound the solution, then evaluate the fitness of the new solution and then again perform a greedy search. This is this these tips are exactly similar to what we did in a teacher phase.

(Refer Slide Time: 32:36)



So, now let us employ that over here for our example. So, x 1 x 2 x 3 violates the lower bound right. So, if I bound them; this will become 0, this will become 0, this will remain 2.82, because it is not violating any of the bounds. So, this is our new solution right.

(Refer Slide Time: 32:53)

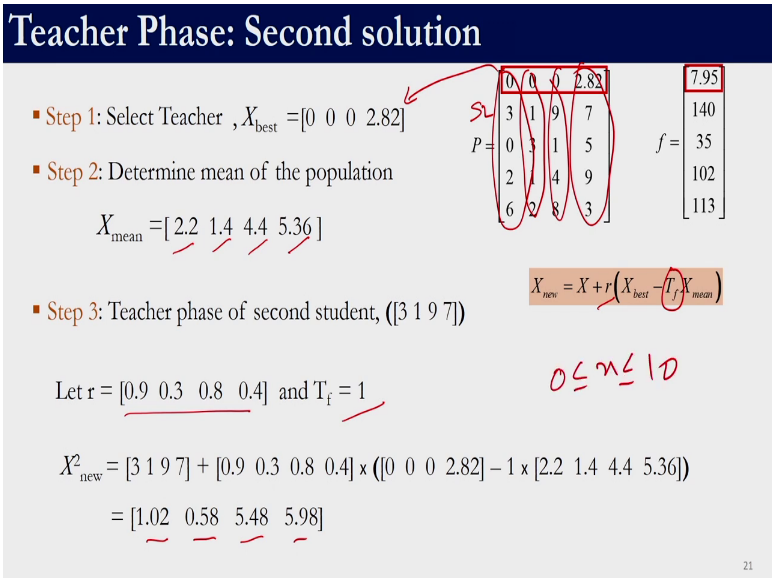


So, the next step is to evaluate the fitness of the new solution. So, it is fitness of the new solution will be 7.95. Now, we will have to perform a greedy search right. So, the solution that was undergoing that is undergoing learner phase is solution 1. So, this is our solution 1, this is the new solution generated with the help of solution 1 right. So, now, there is a competition between these two solutions.

Only 1 of them can survive the 1 that will survive is the new solution, because it has a lower fitness value then the solution which was used to generate right. Since 7.95 is less than 23.82, this solution will enter the population and this solution will be taken out right. So, to consolidate what happened for the solution 1 was; solution went solution 1 underwent teacher phase it got a better solution.

So, this solution became S 1 right and this solution underwent learner phase and it got another better solution right. So, when we obtained this better solution at the end of teacher phase 1; this was removed. Now, this will be removed because, we have a better solution right. So, by applying a teacher and learner phase for the first member; we have been able to reach 7.95. The best solution now is this 7.95. In this case, it happened that both teacher phase and learner phase gave us better solution, but we could have encountered failures also.

(Refer Slide Time: 34:34)

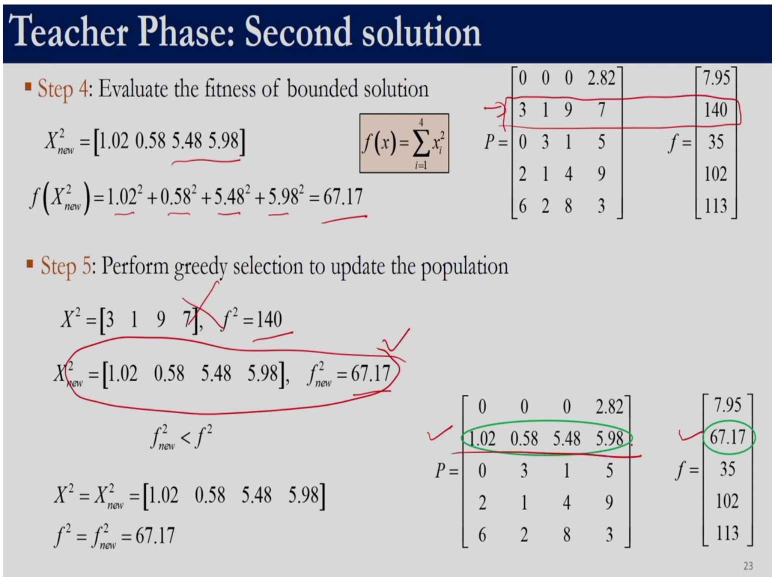


So, now let us apply a teacher phase to the second solution right. So, S 2 right, so this is S 2. So, the first step is to identify the teacher. So, the teacher is 7.95, because it has the least objective function value. So, this is the teacher solution. Now, the next step is to calculate the mean of the solution. So, the mean of the population is nothing but, the mean of this 4 columns that turns out to be 2.2, 1.4, 4.4 and 5.36.

So, now, we have the mean of the population; we have the best solution. We know the solution which is undergoing the teacher phase right. So, now, we can apply this equation to generate a new solution. Since teaching factor is required in the teacher phase; we will have to generate a random number either which has to be either 1 or 2. So, let us assume that; the random number which we generated is 1 and I need 4 random numbers because I have 4 decision variables the 4 random numbers be this.

So, using this equation we can again generate the new solution right. So, the new solution will be 1.02, 0.58, 5.48 and 5.98 right. In this case, if we see none of the variables are violating their bounds because, the bounds are between 0 and 10. All of the variables are within the bounds.

(Refer Slide Time: 35:53)

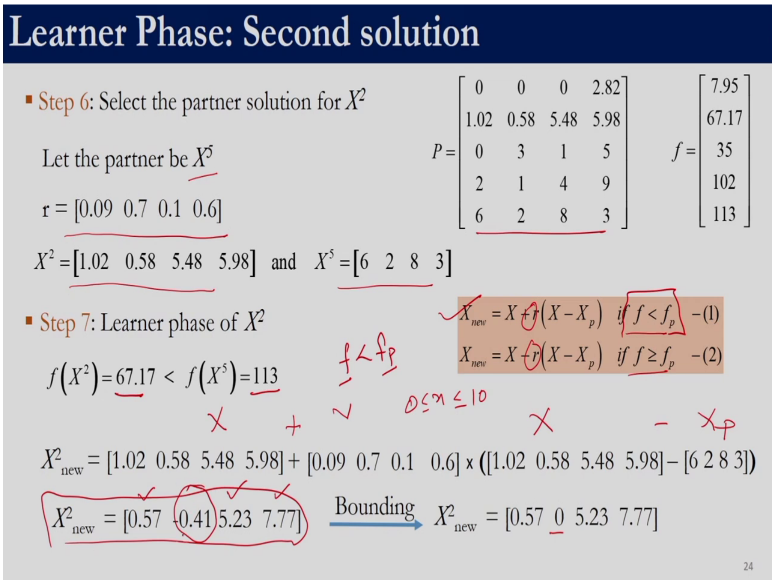


So, we can directly evaluate the fitness function of the solution. Since, the solution is already bounded; we can evaluate its fitness function right. So, fitness function is again nothing, but the objective function value. So, x 1 squared plus x 2 squared plus x 3 squared plus x 4 squared. So, that will work out to be 67.17.

We know that it was the second solution which is undergoing their teacher phase right and the new solution it has a fitness value of 1 140. So, this is the solution which is undergoing the teacher phase and the new solution which we generated has a fitness of 67.17 right. So, the new solution which we have generated is better than the old solutions. So, this is better. So, we will use this to update that population right.

So, remember when we are updating the population; we need to update the decision variables as well as the fitness function value. Now, we have completed the teacher phase of the second solution. So, the next step is to perform the learner phase for the second solution right. So, now, our second solution is this one right. So, this is the second solution.

(Refer Slide Time: 36:56)



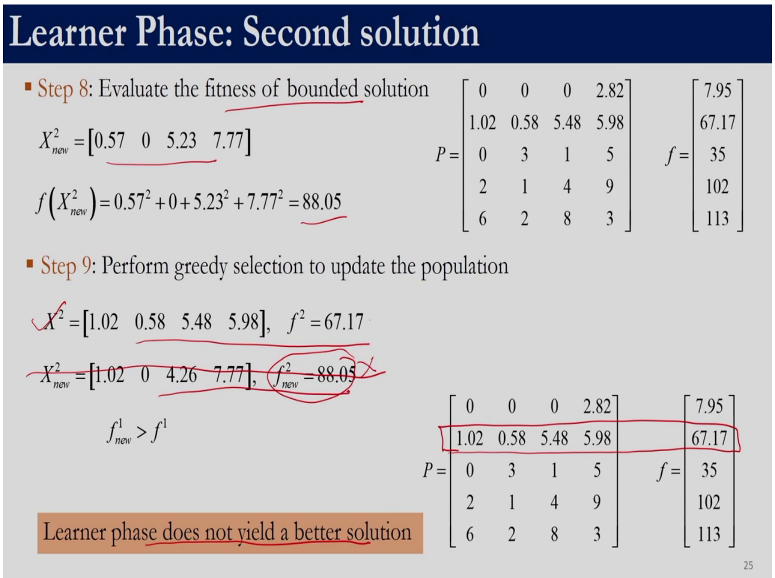
So, in learner phase, as you know; we need to randomly select a partner. So, let us select the partner to be 5 and depending upon the fitness of the current solution which is undergoing the learner phase and the fitness of the partner we will have to select one of this equation. No matter which equation we use; we will require this random numbers as many as decision variables.

So, let the 4 random numbers be 0.09, 0.7, 0.1 and 0.6. So, the solution that is undergoing learner phase is X 2 and the partner is X 5, 6 2 8 3. And we know the fitness function of both the solutions. The second solution has a fitness function value of 67.17 and the fifth solution has a fitness function value of 113 right. So, since f is less than f p right, f is the current solution which is undergoing learner phase f p is the partner which we have selected. So, 67.17 is less than 113.

So, we are supposed to use the first equation because of this condition. So, if we apply this equation. So, this is the current solution which is undergoing the learner phase plus the random number into the current solution minus the partner right. So, partner is the fifth solution; which is 6 2 8 3. We will get a new solution given over here, 0.57 minus 0.41, 5.23 and 7.77. Again if we see these three variables are between the bounds right.

So, remember the bound is X has to be less than X has to be between 0 and 10 right. So, only this variable has a value which is not within the bounds. So, what we do is we employ corner bounding strategy and move this variable to the to its lower bound. Since it is violating the lower bond; we change this value to 0, since this value is violating the lower bound.

(Refer Slide Time: 38:48)

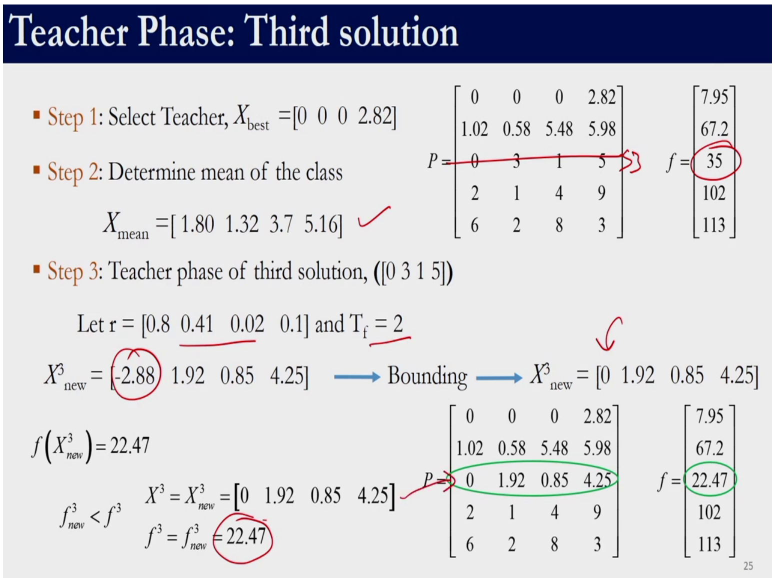


So, the next step is to evaluate the fitness of the bounded solution right. So, this solution is within the bound. So, we can use it to evaluate the fitness. So, the fitness turns out to be 88.05

right. So, the solution which underwent learner phase; that is the second solution has a fitness of 67.17 and the solution which it generated is 88.05 right. So, the new solution which we have generated is poor than the current solution right.

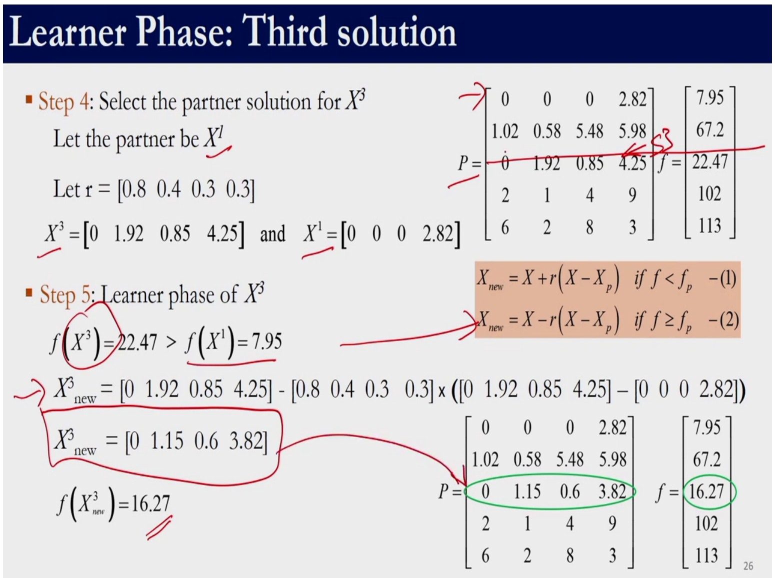
So, what we will do is; we will discard the solution and we will retain this solution in the population. So, that is what you see over here right. So, this value is not updated because, the new value 88.05 is poor than 67.17. So, in this case the learner phase does not yield a better solution. As we discussed previously, it is not necessary that the teacher phase or the learner phase definitely give us a good solution. There is no guarantee for it.

(Refer Slide Time: 39:43)



So, this slide shows for the third member right. So, this is the main of the class, these are the random numbers teaching factor this variable is violating the bounds. So, it is brought back in inside the bounds and then a greedy search is employed. So, 0 3 1 5 this solution, has a fitness function of 35. Whereas, the new solution which we generated has a fitness function of 22.47 right. So, this will be eliminated and this will be incorporated into the population right.

(Refer Slide Time: 40:16)

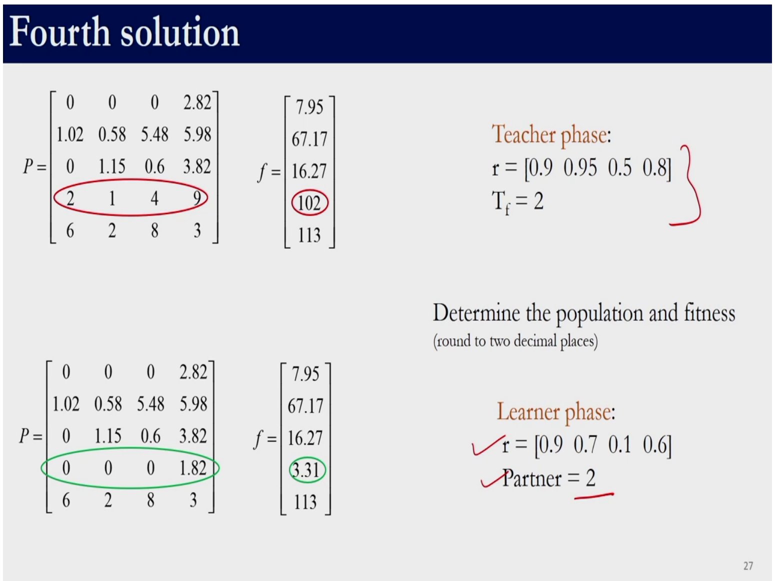


Then the solution 3 undergoes the learner phase that is also similar to what we have discussed right. So, this is the initial population. So, the solution 3; this is the solution that we are working with right. So, in this case, we have selected the partner to be 1. So, this is the partner right. So, we have this solution which is undergoing; we have the partner. We need to compare the fitness function the third solution has a fitness function of 22.47.

The first solution which is the partner solution has a fitness function of 7.95. In this case the partner is having a better fitness function value then the solution that is going undergoing learner phase. So, we need to select the second equation right. So, if we apply second equation; we get this new solution, which is already within the bound. So, we do not need to worry about bounding the solution. So, if we calculate the objective function of this solution it turns out to be 16.27 right.

So, the solution which was used to generate this has a fitness function of 22.47 and we have obtained a solution 16.27. So, this can be eliminated right because I have a better solution, this solution enters in place of the solution which is discarded right. So, this completes the learner phase of the third solution.

(Refer Slide Time: 41:25)



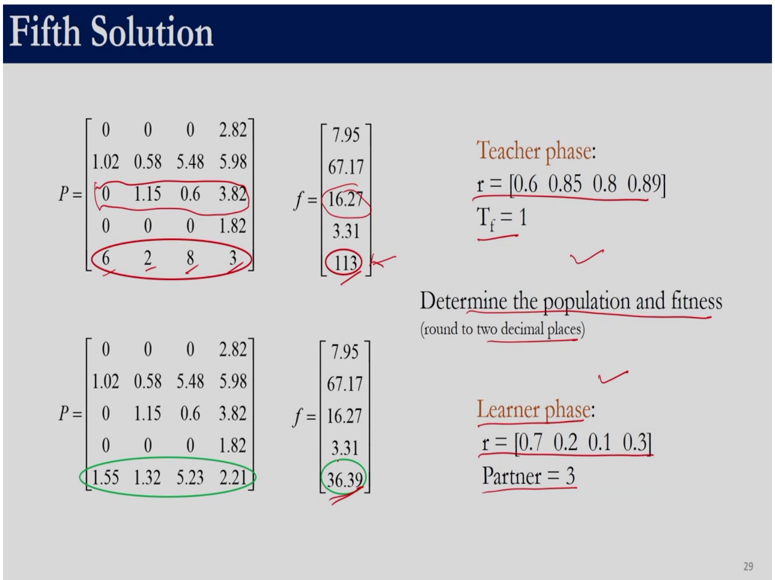
For fourth solution, we will not show you the entire procedure. It is fairly simple and straight forward. We expect you to try it out by yourself and see if the values that; we have matched with the values that you have obtained. If the fourth solution undergoes teacher phase. So, remember even teacher phase to perform teacher phase we need four random values because we have four decision variables and we need one teaching factor which has to be a 1 or 2 right.

So, if we take this random numbers and this teaching factor then you will get a new solution, you need to check whether the new solution is within the bounds or not. If the new solution is within the bounds well and good you go and determine the fitness function value. If the new solution is out of the bounds you need to bound it and then evaluate the fitness function. Once you have evaluated the fitness function; you need to compare the fitness function of the new solution which you have generated with the fourth solution.

So, whichever is better will be retained in the population and a inferior one would be discarded right. So, that would be the end of teacher phase for the forth solution. And then perform learner phase with this random number and this partner to see if you are able to get this one. Unlike the teacher phase wherein we had just one equation, here we have two equation in the learner phase. So, only one of the equation has to be selected. So, the equation that we will select depends upon the fitness function of the solution that is undergoing the learner phase and the partner right.

So, in this case the partner is two. So, it has to be compared with that fitness function value and appropriate equation has to be selected. And if you complete the calculation; see what is the solution that you are getting. Again check for the bounds whether it is in the bounds or not if it is not in the bounds; you need to bound the solution and evaluate its fitness function subsequently perform a greedy search, if the new solution is better take it inside the population or else discarded right.

(Refer Slide Time: 43:19)



So, similar to the fourth solution we expect you to perform the calculation for the fifth solution right. So, the fifth solution is 6 2 8 3 and it has a fitness function value of 113. So, remember for performing teacher phase. So, if you use this set of random number and this teaching factor perform the teacher phase for this particular student, the fifth solution right. Once you perform the so, once you generate a new solution you will have to check whether it is in bounds or not.

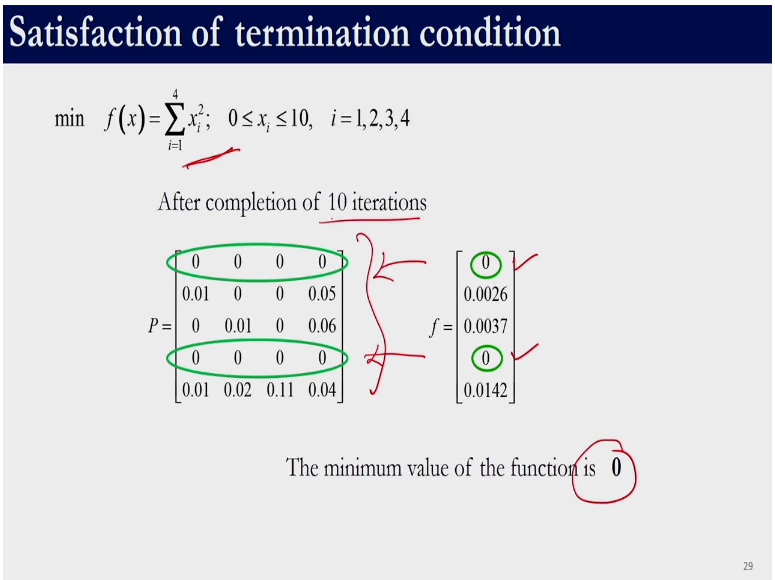
If it is not in the bounds you will have to bound it and evaluate its fitness function. If the new solution has a fitness function value better than 113, we will update the population else we will not update the population right. So, that is what we expect you to do determine the population and the fitness after the teacher phase. Once you have that; we expect you to perform the learner phase right. In learner phase again you will require a four random numbers use this random numbers and you will also require a partner. So, take the partner to be the third solution. So, this will be the partner solution.

Similarly, if you perform the learner phase; you will get a new solution bounded evaluate its objective function value check whether it is better or not. If it is better than the solution that is undergoing the learner phase update the population right. So, in this case, either in teacher phase or in the learner phase; you will come up with a solution which will have a fitness function value of 36.39 right. So, that is why it is updated in this population to consolidate we had 5 solutions right.

So, the first solution underwent teacher phase followed by the first solution undergoing the learner phase. Only when the first solution had completed teacher phase and learner phase; did the second solution undergo teacher phase. Similarly, we perform for all the 5 solutions. So, once this is complete; we have completed one iteration one iteration or one generation or one cycle of TLBO.

This has to be repeated multiple times, remember the generalised structure of metaheuristic techniques that we discuss some time ago. So, this has this procedure has to be repeated multiple times. How many times? T times, T was a user define parameter which we had fixed at the beginning of solving this example right.

(Refer Slide Time: 44:33)

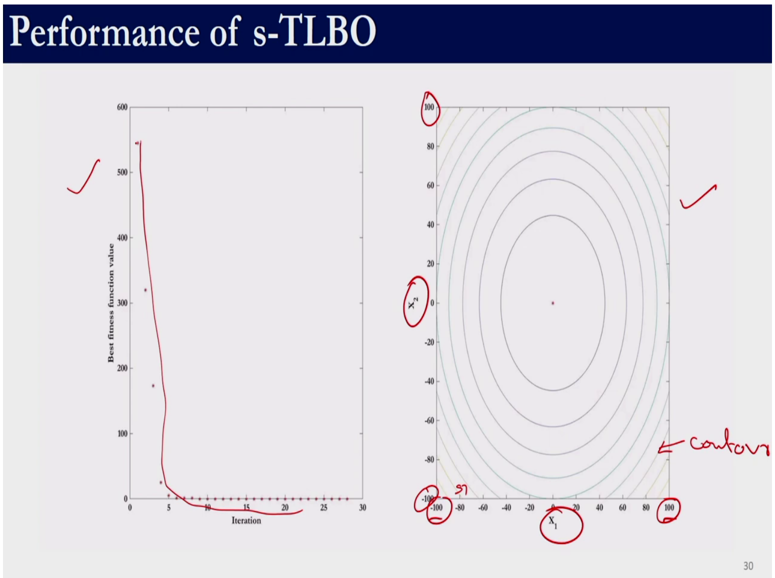


So, here if we see at the end of 10 iteration this is what the solution would look like. By now, you would have realized that no doubt whatever we were doing was very simple, basic arithmetic operations is what we were performing. But there are way too many computations right. So, usually this techniques are useful only when we use them implemented on a computer. Because, it is too difficult too laborious I would say to implement these many iterations for these many population size.

So, in this example we had taken a population size of only 5 right. If your number of decision variables are large than we would work with the larger population size and will have to perform many many iterations right. So, it is very difficult to fully realize the potential of metaheuristic techniques, if you do not use a computer. So, coming back to this problem right.

So, at the end of 10 iterations this was this is the population right. So, in this population if we see two solutions have reached an identical value identical objective function value of 0 and their decision variables are also identical right. So, now since we have performed the required number of iterations; we will consider 0 to be the minimum of this function. In this case, it happens that; 0 is indeed the minimum of this function, but TLBO does not guarantee that will always reach the global optimal solution right. The expectation is that it will reach, but it may or may not reach the globally optimal solution.

(Refer Slide Time: 47:11)



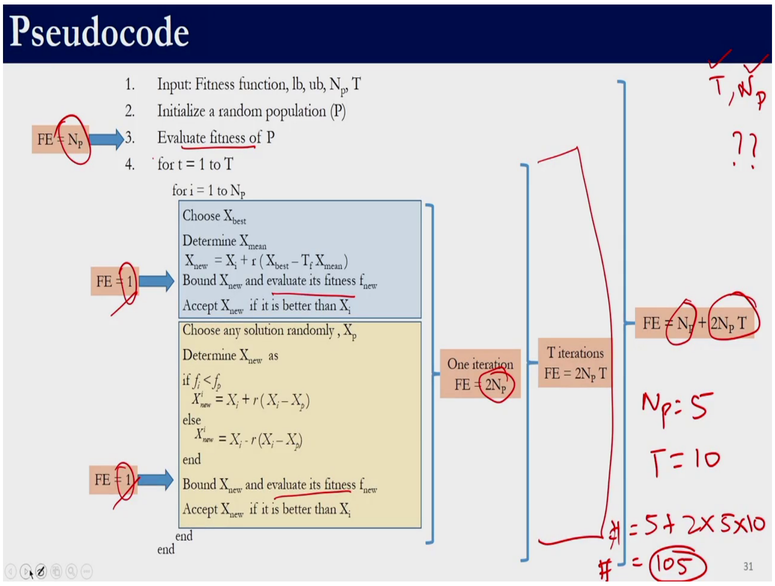
So, let us see what was actually happening during the iteration. Since, we performed only one iteration it is difficult to see how things were panning out right. So, this is a video which was generated using a MATLAB program. We have taken two variable problem X 1 and X 2 with a lower and upper bound of minus 100 to 100 that is the lower and upper bound for both the variables it is the same. This points which you see on the right hand side are the solutions right the initially we generated a random solutions right.

So, those are plotted over here. So, as iteration progresses; this solutions would move right. So, when the let us say if this is let us say solution 1. So, when solution 1 underwent teacher phase it would have changed and when it underwent learner phase it would have further it might have further changed right. So, these solutions are going to keep moving in the search space right.

So, this is the search space, these lines are our contour plots. This plot shows the search space and this plot shows the objective function value right. So, if we if I run this video right. So, here you will see the solutions are moving right. So, the solutions are trying to converge to a particular point right. So, see all the solutions are moving towards one particular point right as my iteration progresses. So, as iteration progresses; all the solutions have converged to a particular solution right.

So, same thing if you will see on this plot right. So, here we have the objective function value which is continuously decreasing. Initially, when we started we had an objective function value of more than 500 and now it has converged to 0. So, this shows the working of TLBO let us consolidate whatever we have seen it in terms of a pseudo code right.

(Refer Slide Time: 49:07)



So, pseudo code will is a simple representation of whatever we have discussed so far right. So, the first thing was; we had to provide the input fitness function, the lower and upper bounds. So, this comes from the problem definition we had to fix the population size and the termination criteria. Then we initialized a random population let us call that random population as P after generating the initial random population; we evaluated its fitness right. Fitness in the sense; we evaluated its objective function value. We decided to do something for T times.

So, this is that loop for i is equal to 1 to T and for each iteration right. So, that is the iteration loop and for each iteration; every member of the class was supposed to undergo the teacher phase as well as the learner phase. So, this far loop is going to take care of that fact within this second for an end is going to repeatedly happened for every member.

So, for every member, we determine the best solution from the population. So, that was the teacher we determined the mean of the population and then we found a new solution using this equation. The new solution may or may not be within the bounds. So, we bound the new solution and then evaluate its fitness function value. Once we evaluate the fitness function value; now we are in a position to compare the solution which was used to generate the new solutions. So, the solution which was used is X of i right.

So, we will accept the new solution if it is better than X i right. So, that would complete the teacher phase. So, this block indicates the teacher phase in the learner phase for every member we had to select a partner random partner. So, X p is the partner. So, we have we had to decide which is the equation to be used right depending upon the fitness function of the solution which is undergoing the learner phase and the partner solution right.

So, depending upon that we had to decide on this equation and then we will be able to generate a new solution. Once a new solution is generated; we bounded that new solution. So, we bound those variables which are violating the domain constraints and then we evaluate the fitness of the new solution. So, now, we have a new solution just like teacher phase; we have a new solution, we have its fitness, we have a solution which underwent learner phase and we have its signal.

So, we perform a greedy search over here and accept the solution whichever is better right. So, this complete this shows the entire procedure which we have been discussing for past sometime right. So, this is a pseudo code of T BO. Algorithms are very often evaluate on the are evaluated on the basis of the number of times we evaluate the fitness function to reach the optimal solution right.

So, if you have seen every time we generate a new solution; we go and evaluate the fitness function right. So, the question is if I am doing T iterations right with the population size of N P. How many functional evaluations am I doing right. So, that is what we are going to evaluate now right. So, if we see there are three places where in we are evaluating the fitness, here we are evaluating the fitness, here we are evaluating the fitness and here we are evaluating the fitness right.

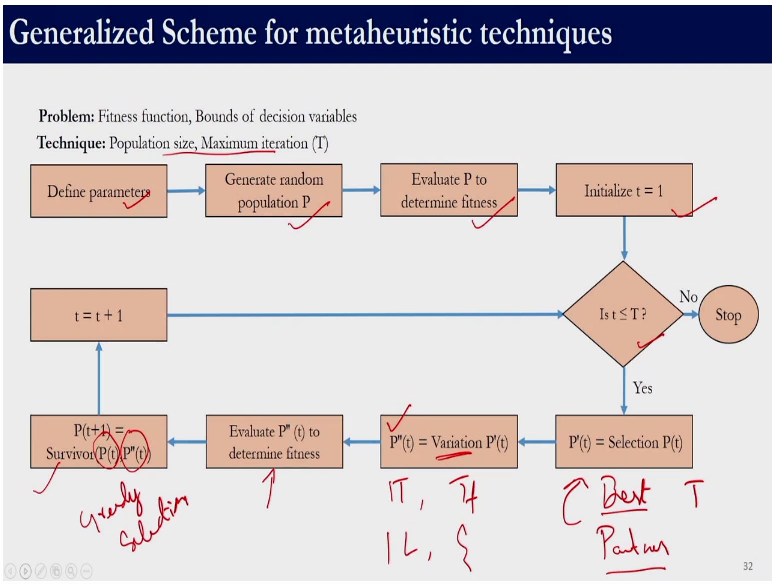
So, when we evaluate the fitness here; we are evaluating the fitness for the entire population right. So, if my population size is N P, we will be doing N P evaluations of the fitness function right. Whereas, here in both these places; we are evaluating 1 only once right. Only once for a particular member, this is inside the loop I am not estimating as of now I am not estimating the total number of fitness function evaluation right. I am just saying that for one particular solution, how many fitness function evaluations are required in teacher phase and learner phase.

So, 1 over here and 1 over here. Since, we are going to do this for N P populations it will be 2 into N P right. Because, I have 2 per member. So, if I have N P members I require 2 N P fitness function evaluations and this I am going to repeat for t generations right. So, the total number of fitness function evaluation is 2 into N P into T. So, that is inside this algorithm rule. I also spent N P functional evaluations at the beginning to estimate the fitness function of the random population.

So, that is within this iteration; we have 2 N P T functional evaluations right. And I had also spent N P functional evaluations over here. So, the total number of functional evaluation is this 2 N P T which is consumed during the iteration and this N P functional evaluations which I spent for evaluating the initial functional evaluation right.

So, this is the total number of functional evaluation. So, if you specify N P to be 5 right and if you specified T to be 10 right. So, I do not need to actually run the algorithm to find out how many functional evaluations would be required right. So, this will be N P 5 plus 2 N P T. So, 2 into 5 into 10 right. So, N P is 5 2 N P 10. So, the total number of functional evaluations is so, this is 10 100 105. So, this is how; we evaluate the total number of fitness function evaluation. So, now, that we have understood TLBO.

(Refer Slide Time: 54:16)



Let us see how well it fits into the generalized scheme which we have started with. So, remember in the starting of this session we had discuss the generalized scheme for metaheuristic techniques right. So, let us see how well T L B O fits into this. So, defining parameters, we had to do that in T L B O right we have to fix the population size and the maximum number of iterations. We generated a random population, we evaluated its fitness functions.

So, all this 3 steps which are there in the generalized scheme was also there in T L B O. So, we had to initialize the number of iterations right. So, we did something repeatedly for t iterations, though we showed it only for the first iteration right. You are supposed to do it for multiple iteration. So, this seems to be right. So, this selection operator, if you remember we did 2 types of selection one was we selected the best solution at in the teacher phase and then we had selected a partner solution right. So, that can be said to be fit into this selection phase.

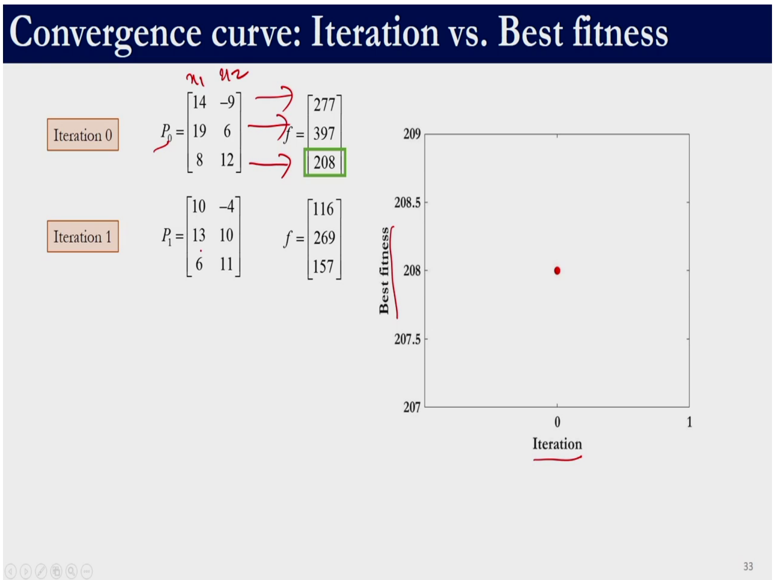
So, it is not necessary that I select specified number of solutions. So, that is the beauty of metheuristic techniques that in TLBO, we are only selecting the best solution and the partner solution right. So, in some other metaheuristc technique, we maybe selecting more than let us say 5 solutions or more than 10 solutions from the solution pool.

So, it varies from technique to technique, but basically I did have some kind of selection operator right and then if you remember we had this 3 equations 1 equation in teacher phase which involved the teaching factor right and then I had this 1 equation in my learner phase like there were 2 equations, but one of the equation is valid for every member depending upon the fitness function right.

So, I varied the solutions to get new solutions right. So, this variation also happened right. So, once we generated new solutions in T L in TLBO also we bounded them, if it is not within the bounds, we bound the solutions. So, the bounding can be considered as part of variation itself right. So, then we evaluated at the fitness function right of the new solution as and when we evaluated the fitness function of the new solution, here we employed a greedy selection strategy right.

So, we said like between the solution which is used to generate the solution and the solution which is which has been generated one of them will survive based on their fitness function. So, this also is valid in TLBO. So, we can see that TLBO fits well into this metaheuristic techniques and as we progress, we will see that list of the techniques also more or less fit into this framework. So, now, we have completed the study of teaching learning based optimization the algorithm. Let us see how do we represent results from this algorithm right.

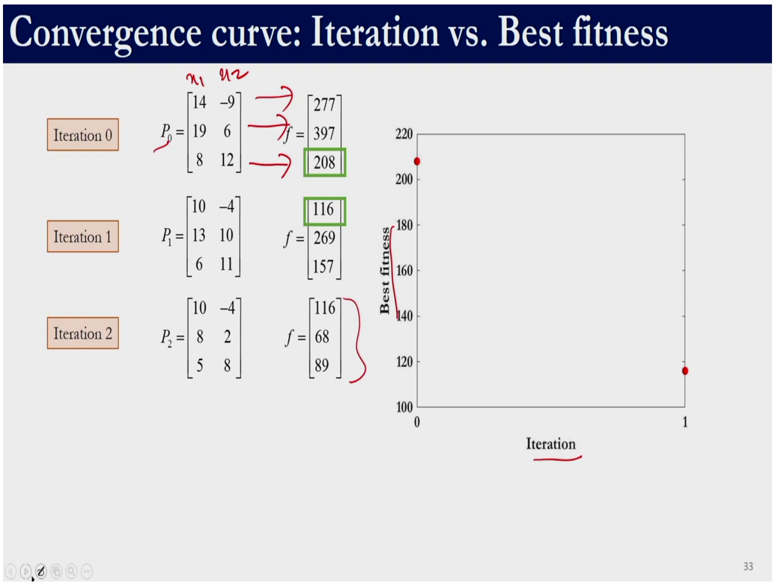
(Refer Slide Time: 57:08)



So, we have various type of conversion curve the one that is probably the most widely used is curve between the number of iteration and the best fitness function values. So, how do we plot it? Let us say at the beginning, my population was P 0 right, these were the solution this is value of decision variable x 1 and x 2 right and these are their corresponding fitness function value or objective function value.

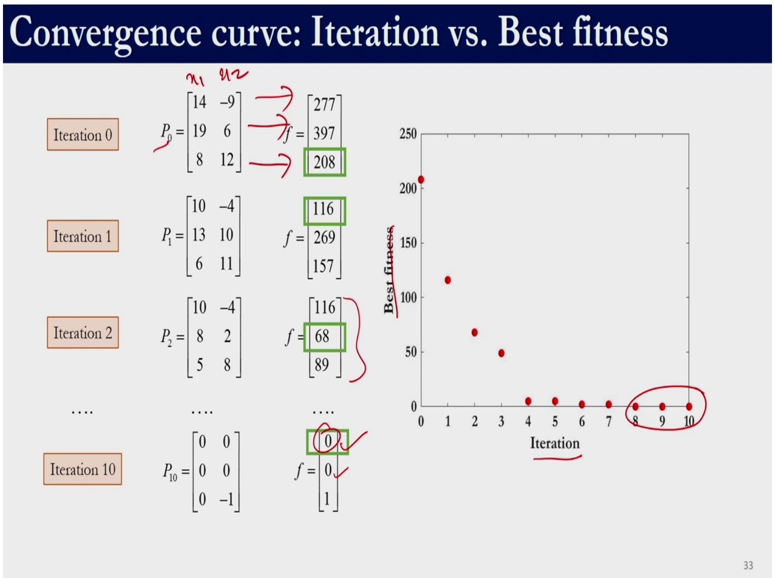
So, what we can do is, we can make a plot the x axis iteration and the y axis is the best fitness function value right. So, among these 3 solutions, if we see 208 is the best solution. So, I am going to plot that particular value alone. I am not going to plot 277 or 3 naught 397. Only the best particular best value in the fitness function is plotted. So, then subsequent to those, we perform the teacher phase the learner phase and at the end of iteration 1 let us say this is our population and these are their corresponding fitness function value right.

(Refer Slide Time: 58:07)



So, in this at the end of iteration 1 the least value is 116 in this right. So, I retain the first point because it corresponds to my iteration 0 and in at the end of iteration 1 I had 116. So, we have plotted that 116 and we continue doing so. So at the end of iteration 2 let us say this is the fitness function value.

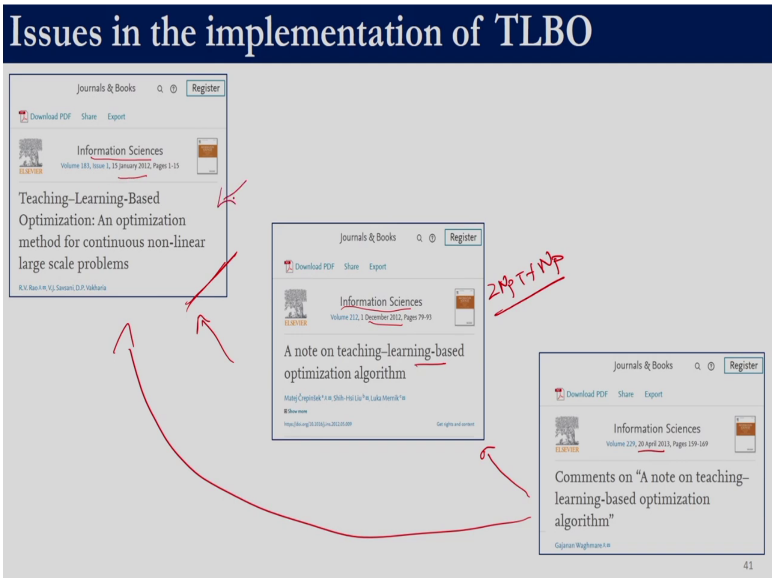
(Refer Slide Time: 58:27)



The best value here is 68. So, we plot that 68 over here and then we keep doing it and say at the end of iteration 10, we have this fitness function value right 0 0 1. In this the least 1 is 0, it does not matter whether you take this 0 or this 0 right. We are not plotting the solutions, we are plotting only the fitness function value right. So, this will look like something like this. So, this is called as the convergence curve.

So, this curve shows you that as iterations progress, how much improvement we were able to do in the objective function value in this case it seems to have converged to the global optima right. So, there are there is not much change or at least visible change in those 3 values. So, this curve tells us the performance of the algorithm as iteration proceeds, how we are improving in terms of the best solution that we have.

(Refer Slide Time: 59:18)



So, now we look into some of the issues in implementation of TLBO right. So, as we have shown you earlier TLBO was proposed in 3 different papers right. So, this one is from information science it was published in January 2012 right. So, subsequent to this a paper came in December 2012. So, the same here, in the same general information sciences. So, this paper reported some of the issues with this paper right broadly there were 2 issues, one is that the number of functional evaluation was not estimated properly.

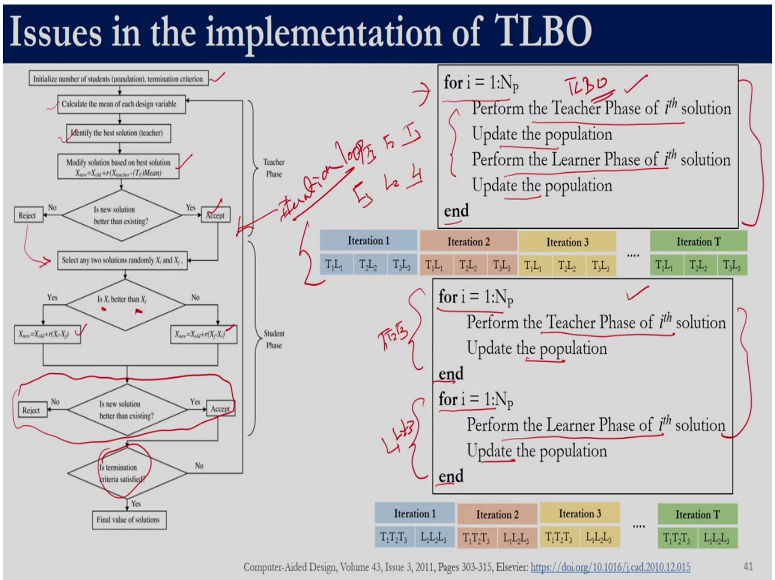
So, this paper reported that the function number of functional evaluation has to be what we discussed earlier 2 N p T plus N p right. So, before that it was not calculated as this one, but something else right. Additionally there was another step implemented in teaching learning based optimization called as duplicate removal. We will look into that then, we have duplicate removal.

We will actually substitute a new solution instead of the duplicates that also consumes functional evaluation. So, that was not considered in this paper right apart from that there was this issue as to when teacher phase and learner phase is to be implemented right. So, subsequent to this paper, another paper appeared in 2013 April 2013, which kind of criticize this paper right. Stating that the conclusions drawn in this paper are correct.

What we are trying to show you over here is that, these techniques are extremely simple, but it is necessary to exact to give a exact description of whatever has been proposed right. So, as we will see the teaching learning based optimization in this paper does not clearly specify how the teacher and learner phase are to be implemented. Very often 1 way to resolve this is to look into the implementation.

Many authors have started to provide their course right. So, it is always a good idea to have a go to go and have a look at the course because it is the course which ultimately would have generated the results. Ideally, there should not be any mismatch between what has been described in the paper and what has been implemented in the code.

(Refer Slide Time: 61:29)



So, this is the flowchart that is given in one of the paper right, the one of the original paper which proposed TLBO. So, here if we see the first step is to initialize the number of students and the termination criteria which is fine enough and then it says calculate the mean of each design variable. So, each design variable over here means, each decision variable identify the best solution which is the teacher generate a new solution as per those equation.

If the new solution is better than existing then, we have to accept it right and continue. But if the new solution is bad then, we have to reject it over here and then there is no connect how to if we reject it how do we proceed forward right. So, assuming that it is even if we reject it is to be, we need to go over here right. Then we need to select any 2 solutions randomly. So, here is where the learner phase is being implemented.

So, we compare the fitness of the current member and the partner right, depending upon which one is better we select either this equation or this equation to generate a new solution and then again we perform a greedy selection over here right. So, whatever is the better solution, we accept that solution and then we check for the satisfaction of the termination criteria. If the termination criteria is satisfied, we stop the algorithm else we go back over here right.

So, over here if you see I can interpret this flowchart in 2 different ways right. First one is this one, that for every iteration right. So, for i is equal to 1 to N p we will perform the teacher phase of ith solution. We will update the population, perform the learner phase of the ith population and update the population.

So, this is one way to interpret this flowchart because, it does not exactly specify this far loop right this loop which you see is the iteration loop right. Because, it talks about the satisfaction of the termination criteria. We check for this criteria at the end of the termination right. I can also interpret this flowchart in this way, for i is equal to 1 to N p perform the teacher phase of the ith solution update the population end.

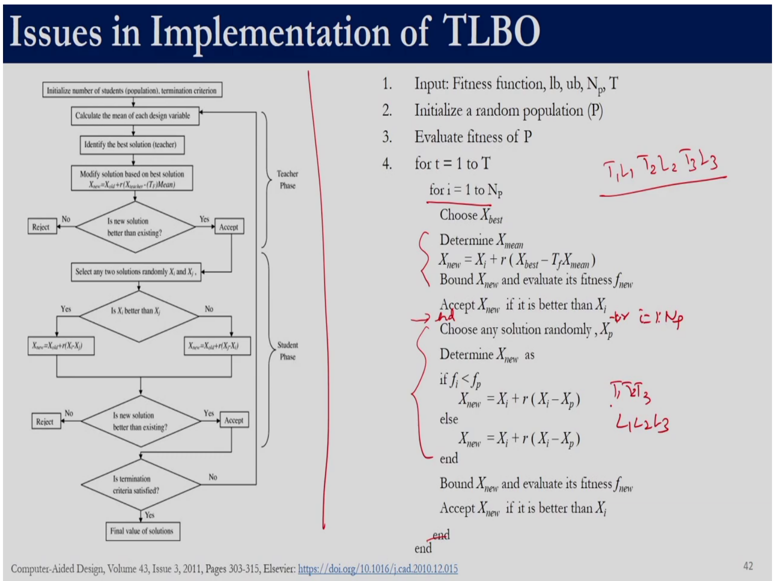
So, in this case all the members first complete the teacher phase and then I have 4 for i is equal to 1 to N p. We perform the learner phase of the ith solution update the population and end. So, all the learner phase is over here right. So, here it is T 1 L 1 right. So, here all the teacher phase are completed. If I have 3 members, the teacher phase of all the 3 are completed and the learner phase of all the 3 members are implemented over here.

Whereas, here the first member will complete the teacher phase, the first member will complete the learner phase. Then the second member will undergo teacher phase, the second member will undergo the learner phase, the third member will undergo the teacher phase and the third member will undergo the learner phase. So, this is what we have discussed previously. But since this flowchart is not clearly given right. It does not explicitly state where to start the teacher phase and where to end it. So, this anomaly arises over here right.

So, whatever we have seen is this one, that in a iteration 1 a member will complete teacher phase will complete learner phase only then the second member will undergo teacher and learner phase. This is how the code of TLBO is right. So, in one of the latter papers they have also given the code of teaching learning based optimization. So, in that case this is how it is implemented, where as if you look at the example given in the paper right, it employees this strategy right.

So, if I trust the code then I need to call this as the true TLBO. If I look into merely into the paper, this is the implementation of TLBO right. So, that is because it is not clearly described as to when to begin the teacher phase and when to complete it and similarly when to begin the learner phase and when to complete it.

(Refer Slide Time: 65:33)

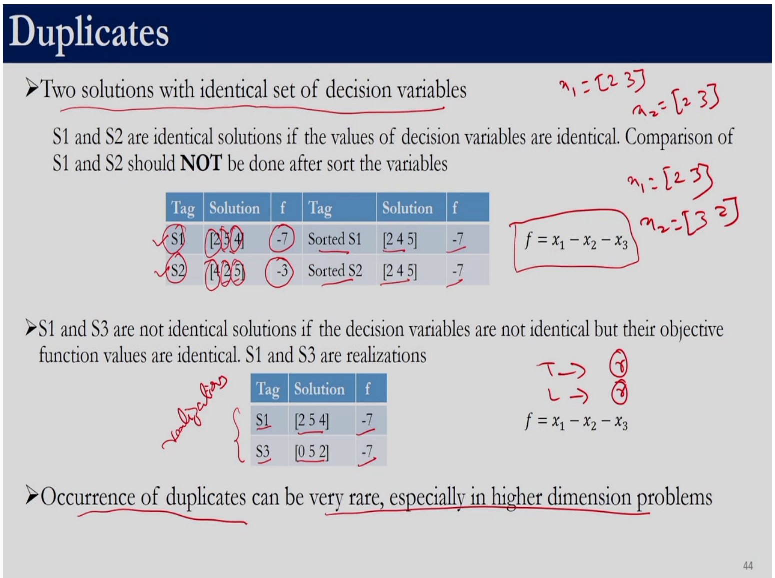


So, this is exactly the same thing what we have discussed previously. This is the flowchart and this is the pseudo code which we have discussed right. So, here if we see we start the population loop over here right and here we complete the teacher phase right. And we implement the learner phase over here. So, the second member can undergo teacher phase, only when the first member has completed the teacher phase as well as the learner phase.

But it is not difficult to implement the other version right all we need to do is over here we need to write a end right and then we need to begin another loop for i is equal to 1 to N p. So, if you look at the codes you will get both types of codes right one is where in the teacher phase learner phase of every member is completed before the next member begins right this is what we have discussed, but if we implement, if you just put a end over here and begin another for over here. It becomes T 1, T 2, T 3, L 1, L 2, L 3 right in every iteration. So, this is one of the issues in this is one of the issues in implementation of TLBO right.

The other one is the removal of duplicates right. So, the description of the TLBO algorithm does not mention the removal of duplicates. First let us try to understand what are duplicate solutions. So, two solutions with identical set of decision variables are said to be duplicates right.

(Refer Slide Time: 66:52)



So, if i have a 2 variable problem let us say x 1, the solution x 1 is 2 3 and the solution x 2 is also 2 3 right. Then x 1 and x 2 are duplicates. Obviously, the fitness function value of x 1 and x 2 would be identical. So, these are duplicate solutions; however, the solution x 1 is equal to 2 3 and a solution x 2 is equal to 3 2 are not identical right. Depending upon the fitness function they may or may not have identical fitness function value right.

So, here we have given 2 solutions, let us say solution 1 solution 2. The values of the 3 decision variables are 2 comma 5 comma 4 and here it is 4 comma 2 comma 5. And the fitness function is x 1 minus x 2 minus x 3. So, if we calculate the fitness function of this first solution it will turn out to be minus 7 and for the second solution it will turn out to be minus 3.

Since the fitness function values are different, we cannot call S 1 and S 2 as duplicates; however, if we sort both the solution. So, if we sort the solution, it will become 2 comma 4 comma 5 and the solution s 2 will also become 2 comma 4 comma 5 and their fitness function would be identical right. So, when we are comparing the solutions right. The solution should not be sorted and then compared.

So, the value of decision variable 1 over here should be compared with the value of the decision variable over here similarly the second variable has to be compared and the third variable has to be compared we should not sort. However, the TLBO code which has been given by the authors actually sorts the solution which is incorrect.

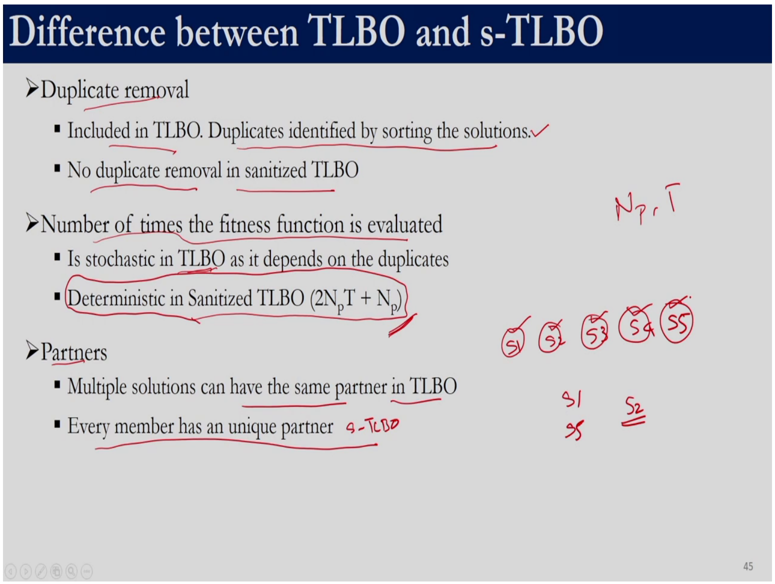
So, here S 1 and S 2 are not duplicates; however, if S 1 had been 2, 5, 4 and if S 3 another solution had been 0, 5, 2 right. So, both of them have fitness function value of minus 7 minus 7. So, in this case I cannot call both of this is duplicates these are realizations. So, duplicates are when 2 set of solutions are identical. The decision variables are identical again without sorting right those are duplicate solutions.

So, now, if you look at the equations right. So, the new solution which we are generating teacher phase or in the learner phase we have this random numbers which we are generating and since we will be generating them on a computer usually those random numbers would be definitely be different right. So, it is very rare for us to come across duplicates, specially if our problem dimension is high right. So, now, the question is should we remove the duplicates right.

So, to remove the duplicates we will have to first identify the duplicates and then have a procedure to remove them right. So, if you remove one particular solution, we again need to substitute it with some other solution right. So, one is the identification of the duplicate solution and the second is the computational time that is required for this purpose right. But if the frequency of duplicates occurring itself is low then, we need not remove this duplicates.

So, most algorithms do not employ this duplicate removal procedure right, because it is extremely rare for 2 solutions to be identical given that we will be working with some 16 decimal.

(Refer Slide Time: 70:13)



So, to consolidate the difference between TLBO and s TLBO is that duplicate removal is included in TLBO right. And the problem is that the duplicates are identified by sorting the solutions which is not the definition of duplicate right. So, that is one problem with TLBO. So, in sanitized TLBO, we do not have any duplicate removal procedure. This is consistent with the other algorithms which will be discussing as part of this course right.

So, the second thing is the number of times the fitness function is evaluated right. So, in sanitized TLBO, we have seen this expression that this is unique right as long as we fix N p and T, the number of fitness function evaluation is deterministic and it can be determined.

Whereas, in TLBO, that is not the case because, it then depends upon how many duplicates we encountered for every duplicate solution that we will encounter, will generate a new solution.

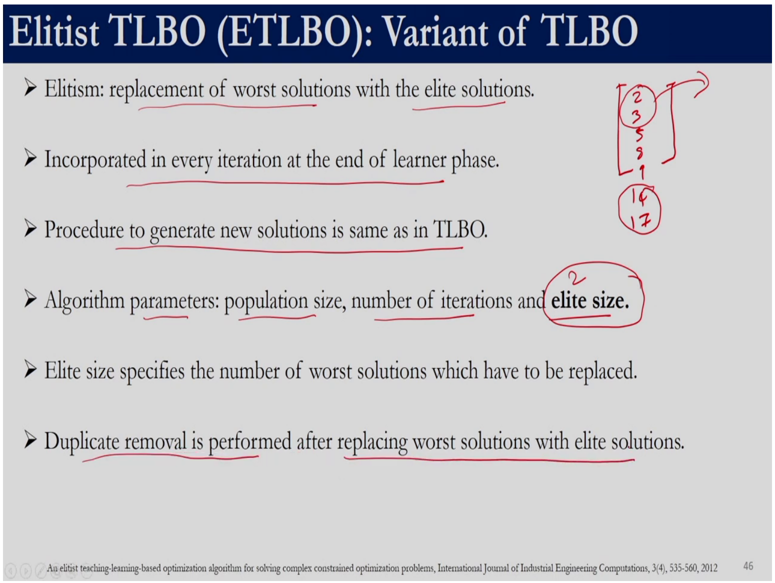
So, if we generate a new solution we will also have to evaluate its fitness value right. So, a priori it is not possible to say as to how many duplicates will be generated right and the number of fitness function evaluations will depend on how many duplicate solutions we generate. So, another difference between TLBO and s TLBO is that, how do we identify the partners right. So, in TLBO it is perfectly possible that more than one solution have the same partner right.

So, if we have let us say 5 solutions S 1, S 2, S 3, S 4 and S 5 right then S 1 and S 5 can have the same partner S 2. So, every time it is randomly selected in TLBO, whereas, in s TLBO we ensure that every member has an unique partner. So, each member S 1, S 2, S 3, S 4, S 5 will be partner for one solution or the other solution. So, each solution will get an opportunity to be to be partner to one another solution. So, that is what is s TLBO.

So, we do not employ duplicate removal since we do not employ duplicate removal the number of fitness function evaluation is deterministic given the size of the population and the number of iterations we can determine a priori as to how many functional values will be required and in sanitized teaching learning based optimization every member gets to be a partner to some other solution right. Now let us look at couple of variance of T LBO.

So, TLBO as you know was proposed in 2011. Subsequent to that there have been large number of variance which have been proposed for TLBO.

(Refer Slide Time: 72:51)

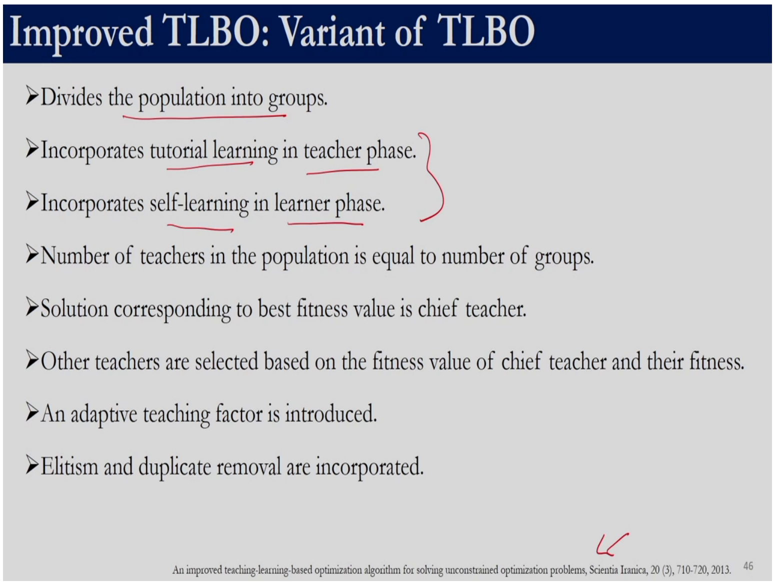


We will briefly touch upon 2 variants of TLBO, one is elitist TLBO known as E TL BO. So, what is done over here is? In every iteration, we will identify the worst solutions and we will replace it with elite solutions. So, in every iteration, what we will do is, we will sort the fitness function value 2, 3, 5 8, 9, 14, 17 something like this right and we will fix an elite size all right.

So, what we will do is, we will save this 2 and 3 in some other variable before we begin that iteration right and at the end of the iteration, we will replace the 2 worst solution right. So, if the elite size is 2 then we will replace the 2 worst solutions with the 2 best solutions. So, that is what is done in E TLBO. So, it is incorporated in every iteration at the end of learner phase otherwise the procedure to generate new solution is the same as in TLBO.

So, in this case the algorithmic parameters are population size number of iterations and elite size. So, the user is also supposed to give the elite size right. So, even in E TLBO duplicate removal is performed, but after replacing worst solution with elite solutions.

(Refer Slide Time: 74:01)



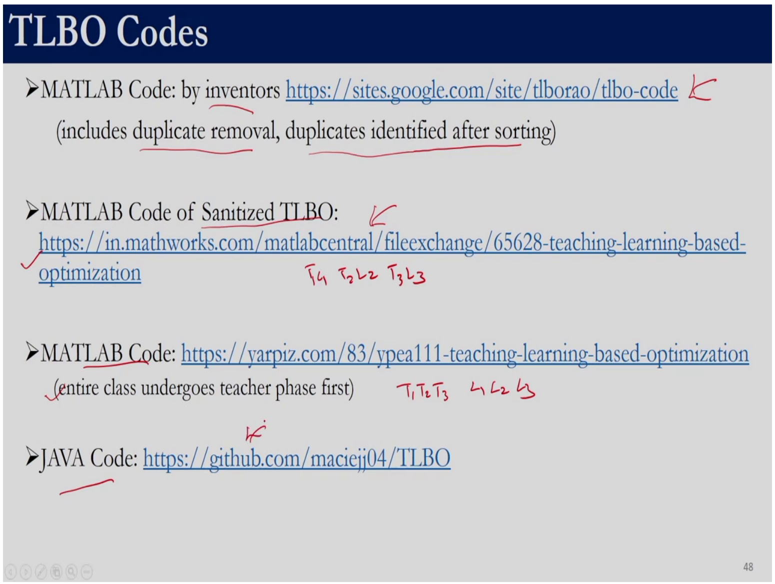
So, in improved teaching learning based optimization the population is divided into subgroups right it incorporates tutorial learning and self learning in teacher phase and learner phase respectively right. So, this basically means that whatever equation we had, there are couple of more terms to those equations like.

So, if you are interested you can go and have a look at this paper it gives the details about this. Here for this course, we are actually restricting with only teaching learning based optimization. Just to give you an overview about the developments, we are discussing these 2 variance, but our objective was to learn only teaching learning based optimization. That is why we are not doing this improve teaching learning based optimization in greater detail all right.

Coming back to improved TLBO right. So, here we do not have one teacher, but we have a number of teachers which equal to the number of groups. So, if you have a population size of 100 we may decide to have 5 groups right. So, there will be 5 teacher corresponding to each group and there will be a chief teacher. So, chief teacher is the best solution in among all the 100 solutions right.

So, the remaining teachers are selected based on their fitness value and the fitness value of the chief teacher all right. In addition to that an adaptive teaching factories introduced in improve teaching learning based optimization. Additionally, the concept of elitism and duplicate removal is also incorporated in improved TLBO.

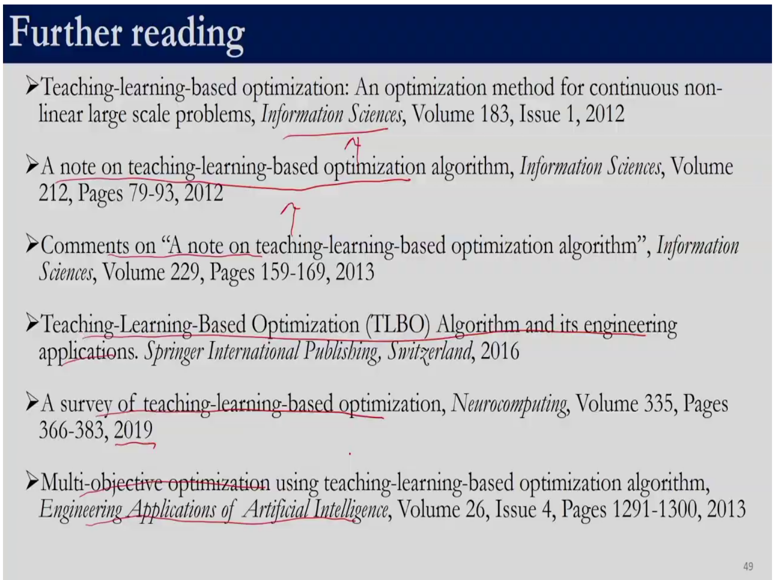
(Refer Slide Time: 75:23)



So, the codes of TLBO are freely available right in the subsequent in the subsequent session, we will also implement TLBO on MATLAB right. So, as invented by the inventors is available over here right. So, this is the website of Professor Rao right. You can go and have a look, you will be able to see that it includes duplicate removals you can also see how duplicates are actually identified after sorting which is not correct right. So, you cannot download the sanitize TLBO from this link right. So, over here, we employ the strategy that T 1, L 1, T 2, L 2, T 3, L 3 and so on.

So, the first member completes the teacher and learner phase only then the second number undergoes teacher and learner phase. Whereas, the MATLAB code over here in this case, all the members complete their teacher phase first right. And then all the members undergo learner phase. So, for an arbitrary problem it may not be possible to say which one will work, which 1 will give better results right. So, but both versions are available and the Java code is also available at github right. So, you can download any of these resources and use it.

(Refer Slide Time: 76:32)



So, some additional reading is given over here right. So, this is the information science paper by in which TLBO was proposed right. So, this is again a note on teaching learning based optimization which criticized this paper. This is comments on a note. So, which criticized this paper there is a book on TLBO by Professor Rao right. So, you can have a look at it teaching learning based algorithm and its engineering applications.

A survey of teaching learning based optimization is also available it was published in 2019 right. So, teaching learning based optimization has also been extended for solving multi objective optimization problem right. So, you can go to this journal engineering applications of artificial intelligence and have a look at it if you are interested. In the session, we saw the generic framework of metaheuristic algorithms.

We then learnt what is sanitized teaching learning based optimization we showed you the working of sanitize TLBO on a 4 variable optimization problem with then discuss some of the issues which are there in TLBO right. So, for example, duplicate removal how do we identify duplicates right all those things we discuss with then looked into the 2 variants of TLBO elitist TLBO and improved teaching learning based optimization with that we will end the session. In the next session, we will implement teaching learning based optimization using MATLAB.

Thank you.