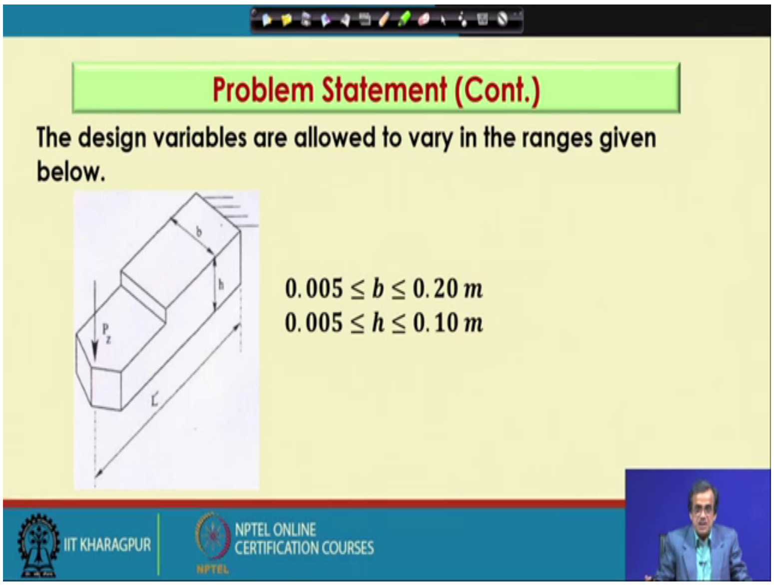
Traditional and Non-Traditional Optimization Tools
Prof. D. K. Pratihar
Department of Mechanical Engineering
Indian Institute of Technology, Kharagpur

Lecture - 35
A Practical Optimization Problem (Contd.)

(Refer Slide Time: 00:25)



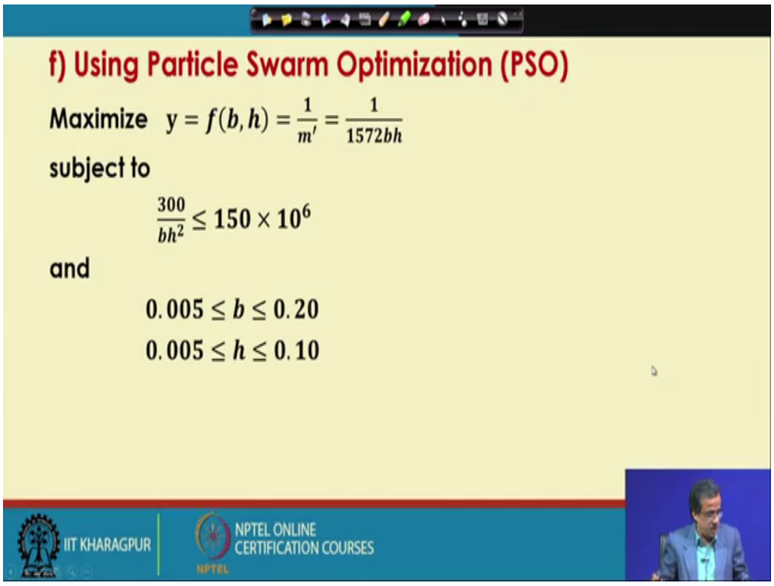
Our aims aim is to determine the optimal design of this single point cutting tool. Now this is the gripping end and this is the cutting point and it is subjected to some amount of concentrated load denoted by p z.

Now I have already discussed like how to find out? The optimal design of this single point cutting tool using the method like, the traditional methods for example, steepest descent algorithm random lock method and we also used a few non-traditional tool for optimization; like binary coded genetic algorithm, then comes real coded genetic algorithm, simulated annealing. And now I am just going to discuss how to tackle the same optimization problem using the principle of the particle swarm optimization PSO.

Now here the variable bounds like b is lying between 0.0 0 5 to 0.2 0 meter h is lying between 0.0 0 5 to 0.1 0 meter and our aim is to find out the optimal values, for this particular the b and h and the length of the cutting tool that is L prime is taken as constant.

Now let us see how to tackle? So, this particular the optimization problem using the principle of your the PSO that is particle swarm optimization. Now this particular the minimization problem we will have to solve like our aim is to minimize, the weight or the mass of this particular single point cutting tool, subject to the condition that is the develop stress is lying within the permissible limit; that means, there should not be any mechanical breakage of this particular the cutting tool.

(Refer Slide Time: 02:34)

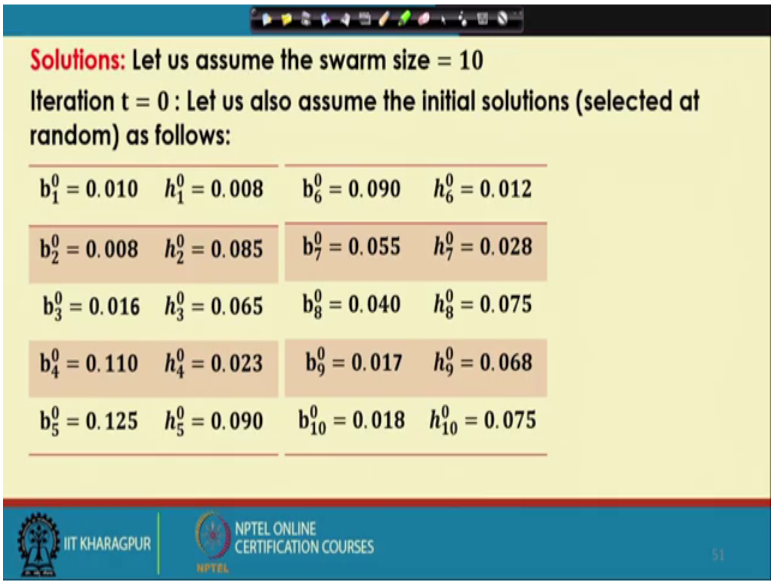


Now, mathematically this particular problem can be stated as follows, like minimize the mass of this particular cutting tool that is m prime is nothing, but 1 5 7 to b h subject to the develop stress that is 300 divided by b h square is less than equals to 150 multiplied by 10 raised to the power 6 and b and h lying within their respective ranges.

Now, this is nothing, but a minimization problem, but on principle the particle swarm optimization can solve the maximization problem and that is why. So, this particular problem has been rewritten in the form of a maximization problem like this, maximize y equals to a function of b and h and that is nothing, but 1 divided by m prime. And that is nothing, but 1 divided by 1 5 7 2 multiplied by b h. Subject to the develop stress is less than equals to 150 multiplied by 10 raised to the power 6 and b and h are lying within their respective ranges.

So, let us see how to tackle this maximization problem using the principle of this particular the particle swarm optimization.

(Refer Slide Time: 04:09)



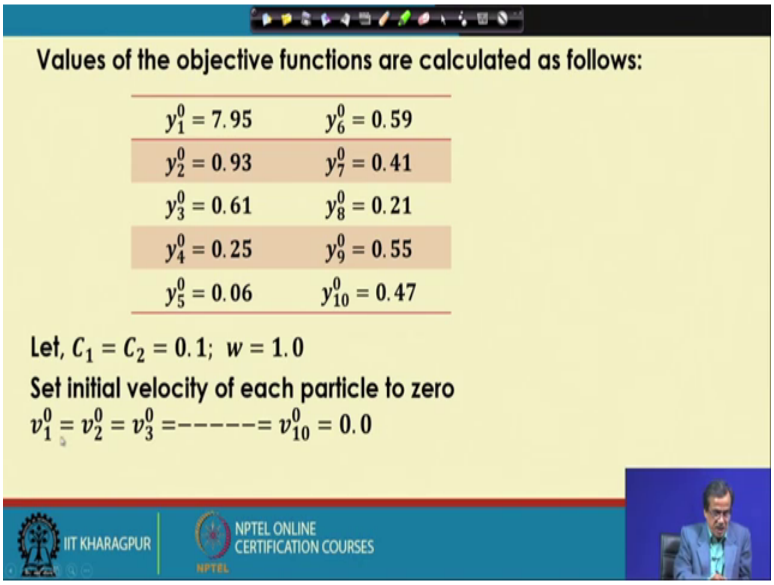
Now, here we start with a population of solution that is the swarm of particles and that is why this is known as the PSO that is particle swarm optimization. Now as I told that here we start with a swarm of particles, now here for simplicity I am just going to consider that is the swarm size is equal to 10.

So, there are only 10 solutions and let us see how to find out the optimal solution. Now let us start with iteration that is iteration is equals to 0 and the initial solutions are generated at random using the random number generator. And while determining this random solution we will have to be careful that all the solutions should lie within their respective ranges. Now here the b and h that is the width of this single point cutting tool and the height of this particular single point cutting tool.

So, those things are generated at random using the random number generator. For example, say v 1 0 is 0.0 1 0 h 1 0 is 0.0 0 8, similarly this b 2 0 is nothing, but 0.0 0 8 and h to 0 is 0.8 5 and similarly the other solutions are also selected at random using the random number generator, but we will have to be careful all such solutions should lie within their respective ranges. For example, say b 3 0 and b h 3 0 are determined at random like this, similarly b 4 0 h 4 0 b 5 0 h 5 0, then b 6 0 h 6 0 b 7 0 h 7 0 b 8 0 h 8 0 b 9 0 b h 9 0, then b h b 10 0 is 0.0 1 8 and h 10 0 is nothing, but 0.0 7 5.

So, all such solutions are selected at random and these initial solutions are selected and these solutions are lying within their respective ranges.

(Refer Slide Time: 06:55)



Now if this is the situation let us see how to proceed further to determine the optimal solution? Now once we have got the values of the design variables like b and h. Now if I substitute in the expression of the objective function; that means, if I just look into this the expression of the objective function, that is y equals to 1 divided by 1 5 7 to be h. So, if you substitute the values for this b and h then I will be able to calculate what should be the values for the objective function.

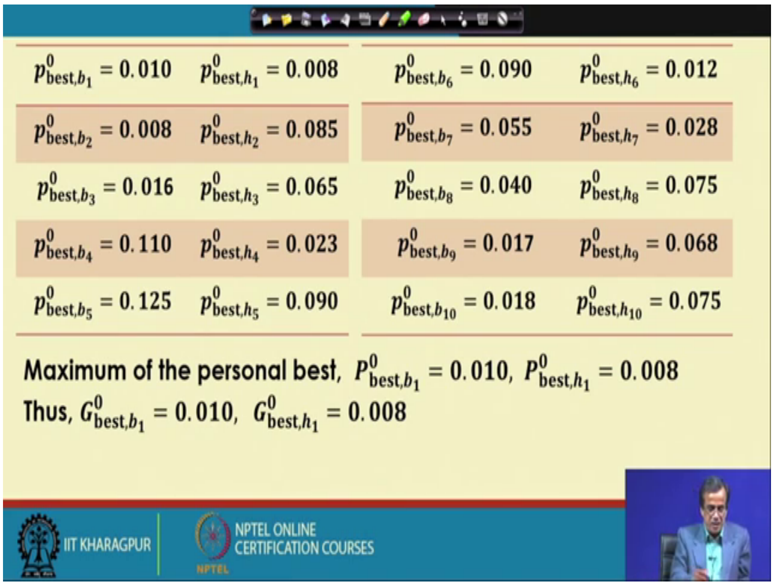
For example say y 1 0 is calculated 7.9 5, then y 2 0 is calculated as 0.9 3, then y 3 0 0.6 1 y 4 0 0.2 5 then y 5 0 is 0.0 6, then comes y 6 0 is 0.5 9 y 7 0 is 0.4 1, then y 8 0 is 0.2 1 y 9 0 is 0.5 5 and y 10 0 is 0.4 7. So, using this I can find out what should be the values for this particular the objective function. And once I have got the values of the objective function now we can proceed further and let us assume that C 1 equals to C 2 is nothing, but 0.1.

Now, the C 1 is nothing, but the coefficient for the cognitive parameter and C 2 is nothing, but the social parameter and w is nothing, but the inertia weight that is 1.0. Now these values are selected by the user at random. Now once you have got this particular thing, now we are in a position to find out what should be the velocity and initially we said the values of the velocities are equal to 0; that means, your v 1 0 that is equals to v 2 0, because to v 3 0 dot dot dot the last 1 is v 10 0 and that is equals to 0.0.

So, these initial values of the velocities those are assigned those are selected at random and then we go for like, how to update the values of these particular the velocities.

Now before I go for that.

(Refer Slide Time: 09:37)



Now, what we do is we try to assign, the best value, the best present value for each of these particular, the variables. For example, say the values which we are assigned at random and initially we try to assume that those values are nothing, but the best solution the best values corresponding to each of the design variables.

For example, say p based comma v 1 with respect to 0; that means, your so, this is nothing, but corresponding to the first iteration what should be the best value for this particular v 1 and that is nothing, but 0.0 1 0. Similarly the best value of h 1 corresponding to the 0th iteration that is P best comma h 1 0 is nothing, but 0.0 0 8 and these are nothing, but the values which are selected at random.

Similarly, we can assign that P best comma 2 0 is nothing, but 0.0 0 8, similarly P best comma h 2 with respect to 0 is nothing, but 0.0 8 5. Now similarly I can actually assigned the other values that is your P best comma b 3 with respect to 0 is nothing, but 0.0 1 6 then P best comma h 3 with respect to 0 0.0 6 5. Similarly P best comma b 4 with respect to 0 is nothing, but 0.1 1 0, then P best comma h 4 with respect to 0 is nothing, but 0.0 2 3.

And similarly I assign the other values like P best comma b 5 0 P best comma h 5 0 P best comma b 6 0 P best comma h 6 0, then comes P best 7 with respect to 0, then P best h 7 with respect to 0, then P best comma b 8 with respect to 0, then P best comma h 8 with respect to 0, then P best comma b 9 with respect to 0 P best comma h 9 with respect to 0, then comes P best b 10 with respect to 0 and P best h 10 with respect to 0; that means.

So, what we do is whatever the random values we assigned in the first iteration, now what you do we try to assign the best values for each of these particular, the design variables and that is why we can assign the personal best for each of this particular design variable corresponding to the first iteration, and let us see how to proceed further with this particular the values.

Now once you have assigned these particular P best values for each of the design variables for the whole swarm. Now we are in a position to find out like which sets up this particular the design variables is going to give us the best fitness in terms of the objective function. Now if you see in terms of the objective function, now what you can see is like let us try to find out what should be the best value for this particular the objective function. Now these shows the values of the objective function corresponding to the 10 combination of the objective functions the design variables.

Now, corresponding to this y 1 0 the value of the objective function is nothing, but 7.9 5 then corresponding to this particular y 2 0 that is 0.9 3 y 3 0 0.6 1, then y 4 0 0.2 5 y 5 0 is 0.0 6. Similarly I can find out the other y values. Now if I compare these y values and this is a maximization problem.

So, corresponding to the first combination of this particular the design variables I am going to get the maximum value for this objective function and that is nothing, but 7.9 5 and corresponding to this particular this y 1 0 is 7.9 5. So, I can find out what should be the combination of the design variables. Now the combination of the design variables, which is going to give the highest value or the maximum value of the objective function is nothing, but this.

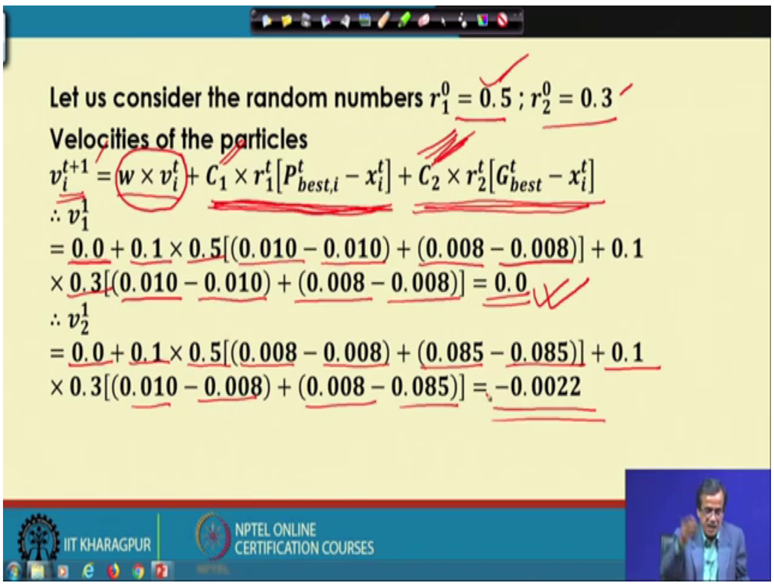
So, corresponding to these I am getting the maximum value for this particular the objective function. And once you have got the maximum value for this particular objective function. So, I can assign that this is nothing, but your the globally best solution.

So, the globally best solution is nothing, but this that is nothing, but these corresponding to which I am getting the maximum value for this particular the objective function now let we let me repeat whatever I told. So, once we have generated the initial values for the design variables at iteration 1, we try to find out the value of the objective function for each of this particular 10 combination of the input parameters. Now I am comparing the values of the objective function and corresponding to this the combination of the input parameters.

So, I am going to get the maximum value of the objective function; that means, this is nothing, but the globally best solution. So, the globally based solution is found to be something like this like g best comma b 1 with respect to 0 is nothing, but 0.0 1 0 0.0 1 0. Similarly this g based comma h 1 with respect to 0 is nothing, but so, this particular thing 0 point 0 0 8.

So, till now we have got what is the globally based solution and at the same time we have also identified and we have already assigned what is the personal best for each of these particular the design variables. So, once we have got the personal best for each of the design variables and the globally best solution.

(Refer Slide Time: 17:04)



Now, we are in a position to find out like, what should be the updated value for this particular the weight or the velocity. Now let us consider the random numbers are 1 with respect to 0 is nothing, but 0.0 5 and r 2 with respect to 0 is nothing, but 0.3. Now these values are selected at random using the random number generator.

Now using this now I can find out what should be the updated values of the variables of the different particles. Now this v I with respect to t plus 1, now this t plus 1 indicates the next iteration is nothing, but w multiplied by v I with respect to t plus c 1 multiplied by r 1 t multiplied by p best comma I with respect to t minus x i t, then comes plus C 2 multiplied by r 2 with respect to t multiplied by G best with respect to t minus x i t.

Now, here if I just concentrate now this v I with respect to t plus 1; that means, what will happen to the ith the velocity of this particular i th particle at t plus 1th iteration and that is nothing, but w multiplied by v i t. Here w is nothing, but the inertia the weight and v I with respect to t is the velocity of the i th particle at t th iteration plus C 1 is nothing, but this is nothing, but the cognitive the coefficient multiplied by this r 1 with respect to t multiplied by p best I with respect to t that is nothing, but the personal best corresponding to the i th solution at t th iteration minus x I t is nothing, but the design variable.

Now, next is c 2 multiplied by r 2 with respect to t into G best t minus x i t here C 2 is nothing, but the cognitive the coefficient sorry this is the social coefficient C 1 is the cognitive coefficient and C 2 is the social coefficient multiplied by r 2 with respect to t is the random number and G best with respect to t is the globally best, minus x i t is the value of the variable.

Now here actually what we will have to do is. So, I will have to find out what should be the updated velocity of the different particles. Now before I proceed further let me tell that this particular component is going to give some sort of the initial momentum to the solution.

Then this particular part this is the cognitive part and this is going to actually indicate or this is going to ensure the local search of these particular design variables. And the last component is nothing but, the social component which is going to ensure the global search of these particular the design variables.

Now let us see using these 3 components like how to determine the updated values for the variables or updated values for the variables this velocity. So, v 1 with respect to 1 is nothing, but w multiplied by v i t. Now here the initial velocity is equal to 0 it is assumed to be equal to 0. So, this component becomes equal to 0.0 the next is C 1 is nothing, but 0.1 then r 1 with respect to t is 0.5 as you have assigned here.

Now, the P best I corresponding to the past particle. So, this is nothing, but 0.0 1 0 this have already discussed multiplied by x i with respect to t; that means, the values of the design variable corresponding to the first particle that is nothing, but 0.0 1 0 plus 0.0 0 8 0.0 0 8 actually this is nothing, but the this P best i with respect to t corresponding to the second design variable and that is nothing, but h that is the height of the cutting tool minus this x i t is nothing, but the value of the design variable that is h corresponding to the first particle, next come C 2 is 0.1 multiplied by r 2 with respect to 2 there is 0.3 multiplied by G best with respect to t and that is nothing, but 0.0 1 0 minus x i t.

That is 0 point is 0 1 1 corresponding to b further the first the particle then comes 0.0 0 8 is nothing, but the globally best corresponding to h. Now 0.0 0 8 is nothing, but the numerical value of h corresponding to the first particle and if we just calculate. So, ultimately I will be getting 0 point is 0; that means, the updated velocity of the first particle at the next iteration is equal to 0.0. So, there is there will be no change of the velocity; that means, there will be no change of the your the position for this particular the particle.

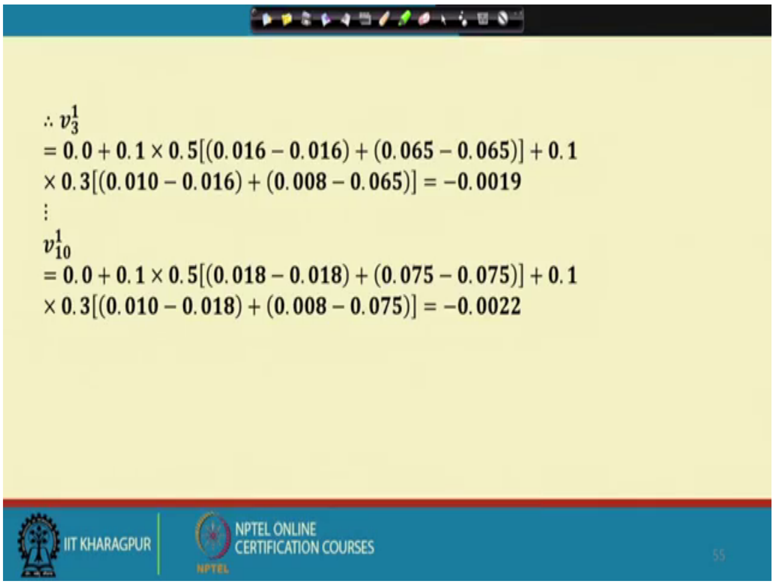
Next I consented on the second particle that is v 2 with respect to 1 and once again the initial velocity is assumed to be equal to 0 plus C 1 is 0.1 or 1 t is 0.5 now the personal best corresponding to the first design variable with respect to the second particle that is nothing, but 0.0 0 8 minus the value of this particular x i t corresponding to the second particle in terms of b 0.0 0 8 plus 0.0 8 5 is nothing, but the personal best of the second particle corresponding to the design variable h and 0.0.5 is nothing, but the value of h corresponding to the second particle plus 0.0 1 that is C 2 multiplied by r 2 with respect to t is nothing, but 0.3 multiplied by the globally best 0.0 1 0.

Now, the globally best corresponding to the first particle the second particle and the past design variable that is 0.0 1 0 minus. So, 0.0 0 8 is nothing, but the first design variable corresponding to the second particle then comes here 0.0 0 8 is nothing, but the globally best solution corresponding to h, then 0.0 8 5 is nothing, but the value of h corresponding to the second particle and if we simplify we will be getting the modified velocity, that is v 2 with respect to 1 that is nothing, but the velocity the updated velocity of the second particle in the next iteration and that is nothing, but 0.0 0 2 2 and here I can assign negative value, because this indicates actually the vector direction.

So, just to indicate the direction so, I am using the negative sign here.

Now, here the next we try to find out.

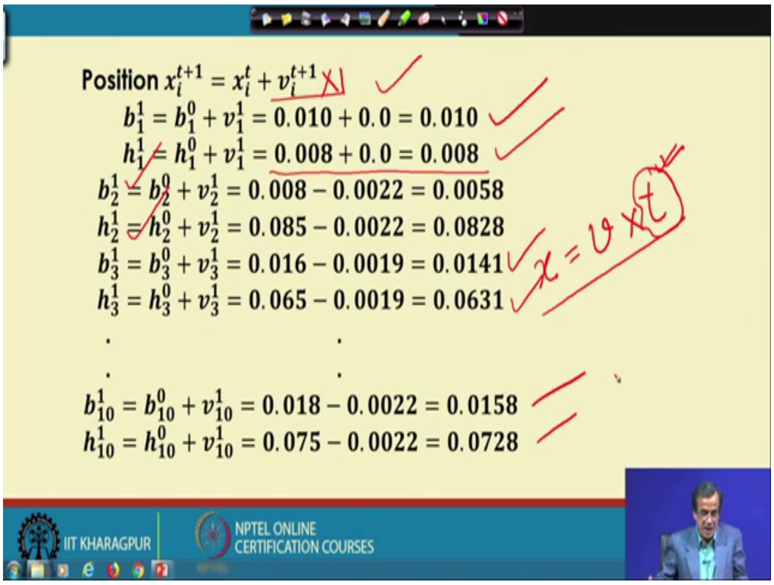
(Refer Slide Time: 25:42)



The velocity the updated velocity for the third particle in the next iteration and exactly in the same way I can find out what should be the v 3 1 and the same procedure I will have to continue, then ultimately I will be getting this v 10 with respect to 1 and that is nothing, but the updated velocity of the 10th particle in the swarm corresponding to the next iteration. And using the same formula so, I can find out what should be the value the updated value for this particular for the tenth particle in the next iteration.

So, using this particular principle so, I can find out the updated value for the velocity of each of these particular the 10 particles and 1 shape got the updated values for this particular the velocity. Now we are in a position to find out what should be the updated position of these particular the particles. Now let us see how to find out the updated position.

(Refer Slide Time: 26:56)



Now, to find out the position so, this is actually the formula which I will have to use that is x i t plus 1 that is the position of I th particle in the next iteration that is t plus 1 th iteration is nothing, but x i with respect to t plus v I, with respect to t plus 1. So, this is nothing, but the how to update the position of the i th particle in the next iteration.

Now, here I just want to mention 1 thing to find out the position once I know the velocity. So, I can find out what should be the position using the expression like the position x is nothing, but the velocity v multiplied by time x equals to v t. Now here t indicates the time and here it is nothing, but the number of iteration. So, truly speaking this is multiplied by 1, because I am trying to find out what will happen to the position in the next iteration. So, t indicates here is nothing, but the iteration number and truly speaking x i t plus 1 is nothing, but x i t plus v i with respect to t plus 1 multiplied by 1.

So, this is actually the formula by using which we will have to find out the updated position. Now v 1 with respect to 1 that is nothing, but the position of the first particle in the next iteration in terms of the first design variable that is v is nothing, but v 1 0 plus v 1 with respect to 1 and that is nothing, but if I just substitute the numerical values like be 1 0 0 point 0 1 0 plus v 1 with respect to 1 is nothing, but your 0.0 this I have already calculated.

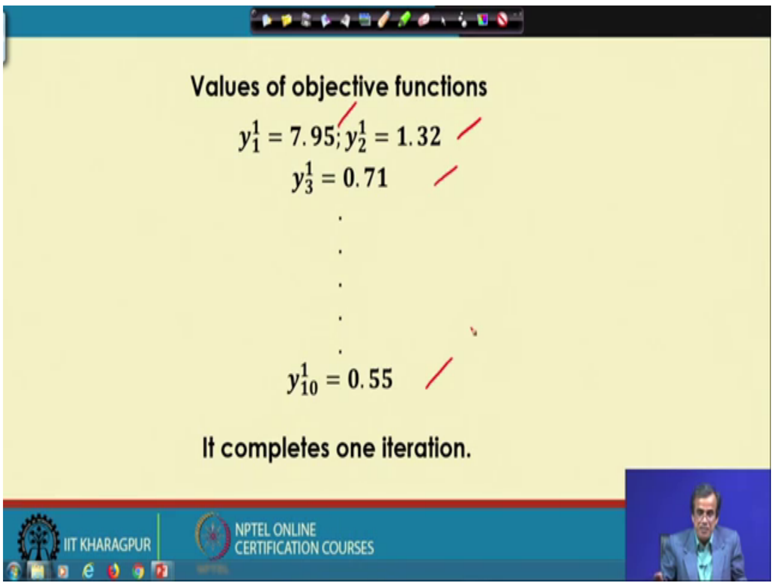
So, I will be getting this particular expression that is v 1 with respect to 1. Similarly I can find out h 1 with respect to 1 that is nothing, but the position of the first particle in the next iteration in terms of the second design variable that is h 1 with respect to 1 and that is nothing, but h 1 with respect to 0 plus v 1 with respect to 1 and if I substitute all the numerical values I will be getting 0.0 0 8.

Now, following the same principle I can find out what is v 2 with respect to 1 what is h 2 with respect to 1; that means, in terms of the b and h these 2 design variables of the second particle in the next iteration. Similarly for the third particle I can find out the updated values of the design variables; that means, the position of the design variables that is b 3 with respect to 1, h 3 with respect to 1 and this process I can continue and for the last particle that is the trained particle I can find out what should be the values of the variables like b 10 with respect to 1 is nothing, but this then comes h 10 with respect to 1 is nothing, but this.

So, now I am in a position to find out the updated the positions of these particular the 10 particles and using this particular information of these particles of positions actually this completes actually 1 iteration of this particular the PSO. So, let me repeat little bit. So, we will up to assign the initial values of the position and the velocities in a at the beginning at random and up to that.

So, I will have to update the velocity of each of the particles in the swarm then using that information, I will have to update the position of the particles and these are the updated position of the particles in the at the beginning of the next iteration.

(Refer Slide Time: 31:28)



And once we have got this particular thing actually. Now, we are in a position to find out what should be the values of the objective function. So, if we just substitute the values of the variables. So, I can find out y 1 with respect to 1 is nothing, but 7 point is nothing, but 7.9 5 y 2 with respect to 1 is 1.3 2 y 3 with respect to 1 is 0.7 1.

Similarly y 10 with respect to 1 is 0.5 5; that means, at the beginning of the next iteration I will be able to find out what should be the values of the objective function.

Now, I will have to repeat the procedure which I have already discussed. Now this is the scenario at the beginning of the next iteration. Now what you will have to do is I will to assign the personal best for each of these particular particles in terms of the 2 design variables v and h and at the same time.

So, I will have to find out the values of the objective function for each of these particular the particles and then I will have to find out corresponding to which particle I am getting the maximum value of the objective function and accordingly, I will have to find out the globally best solution. And the procedure which are discussed that I will have to repeat and this process will go on and go on through a large number of iteration this PSO there is a particle swarm optimization algorithm is going to indicate what should be the globally optimal solution.

So, this is in short the working principle of the PSO and using this I can find out what should be the optimal cross section of this particular the single point cutting tool.

Now, as this algorithm ensures both the local search as well as the global search and at the same time it starts with some the initial momentum of the solution there is a possibility that, this particular algorithm will be able to hit the globally maximum solution in a end up within a number of iteration. And it is going to indicate what should be the globally optimal solution for this particular the single point cutting tool. So, that it can ensure the lightweight after ensuring the condition of no mechanical breakage.

Thank you.