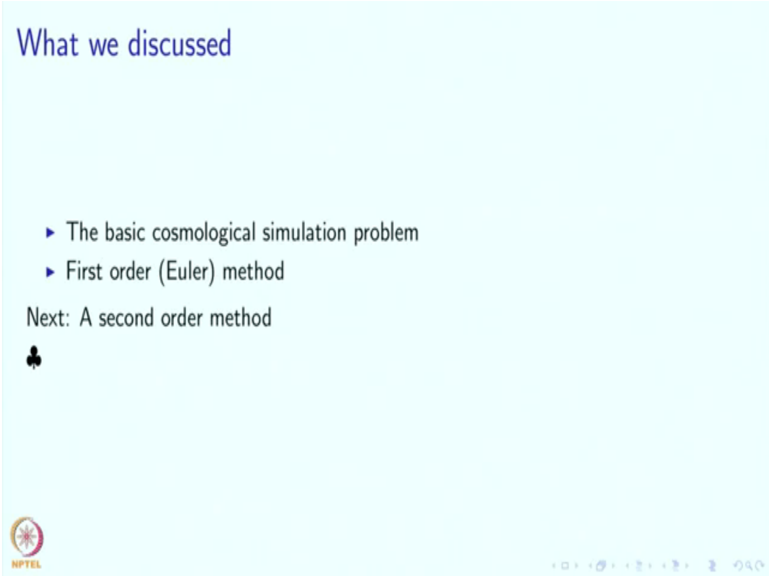
An Introduction to Programming through C++

Professor Abhiram G. Ranade
Department of Computer Science and Engineering
Indian Institute of Technology Bombay

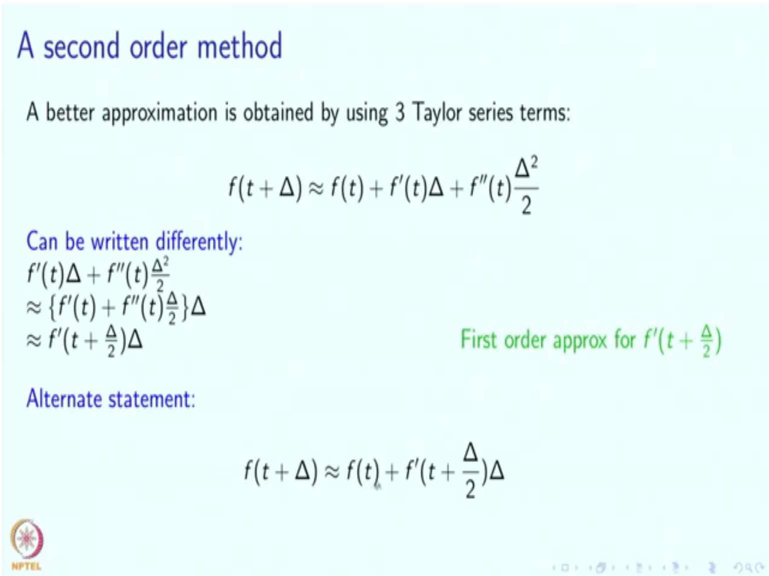
Lecture 27 Part-2 - Cosmological Simulation - Second-order Euler method

(Refer Slide Time: 0:19)



Welcome back. In the previous segment, we discussed the basic cosmological simulation problem and we discussed the first-order Euler method forit. In this segment we will talk about a second-order method which is going to be lot more accurate.

(Refer Slide Time: 0:34)



So the idea is simple. Instead of using two terms of the Taylor series we are going to use three terms. So f(t + delta), f is the variable we want to estimate at time t + delta we are given the value at time t, so f(t) + delta is f(t) + f'(t)*delta and in the previous method we had stopped at this point.

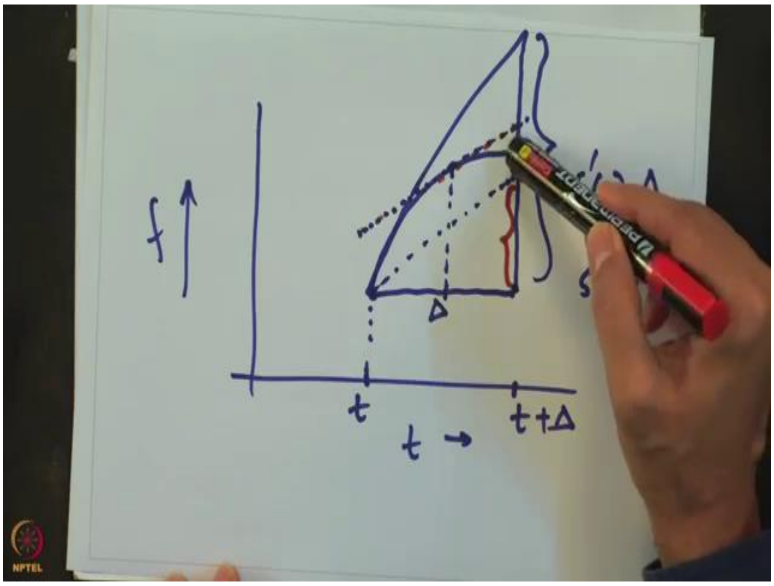
But here we are going add one more term, so + f''(t)*delta^2/2. So this is still approximately equal to because as we know the Taylor series just keeps going and we are not taking all the terms of the series. We are just taking powers, second powers of delta. And therefore, by the way it is called second-order method. Now this can be written in a different manner, so let us look at this part just this part. So I have so this is what I have written down over here and I am going to rewrite it by taking delta common because delta is there in both.

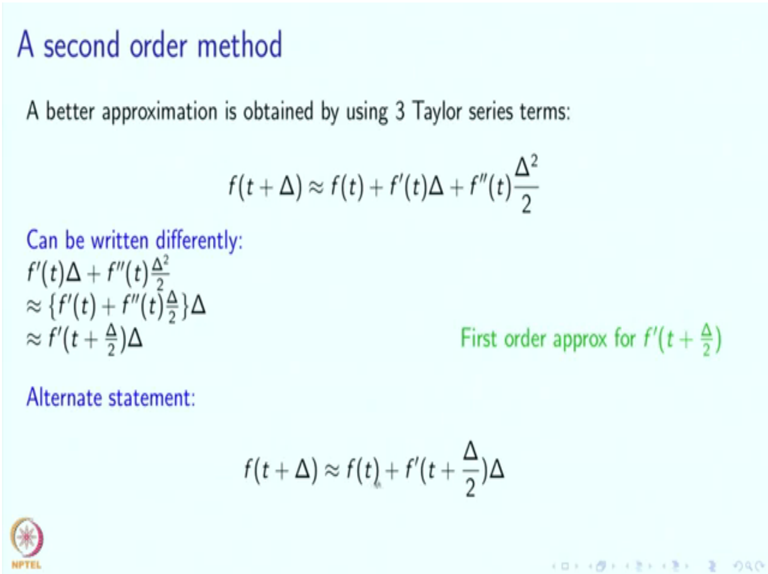
So this I brought out this delta so here I just get f'(t) and here I get f''(t)*delta/2. Now if you look at this you may observe that this is actually a quantity which can be written differently. So in what way? Well I can write it this way. Why is that? So this is just a first order approximation of this. How? Well, if I want to know the value of variable at t + something, if I want to know the value of the variable at t + something, how do I get it? Well, I write it as the value of the variable at t, then plus the derivative of this plus this something in the multiplication.

So indeed in this parenthesis what I have is the first-order approximation of this, we are going in other way of course that we are saying. So we are going to substitute this for this over here. So instead of this we are going to get this term. So here is the alternate statement of the second-order method. So we have same thing f(t) + delta is equal to the first term does not change but this second term looks like this because this taken first-order approximation is exactly what this is.

This alternate statement is nice in many ways. Why? First of all, there are only two terms, so that is good. Second, the second derivative has gone away so it is a simpler thing and that almost looks like the first-order method. The only difference is that in the second term we had, we had t over here f'(t)*delta. So here we have t + delta/2, so that is the only difference but this is still essentially the same expression and therefore in terms of error, this is going to be as good but because it is simpler we are going to use this in our algorithm. So we will use this form but before we get to that, I want to look at what is geometrically just as to just so that you get a betterintuition of what is going on over here.

(Refer Slide Time: 4:35)





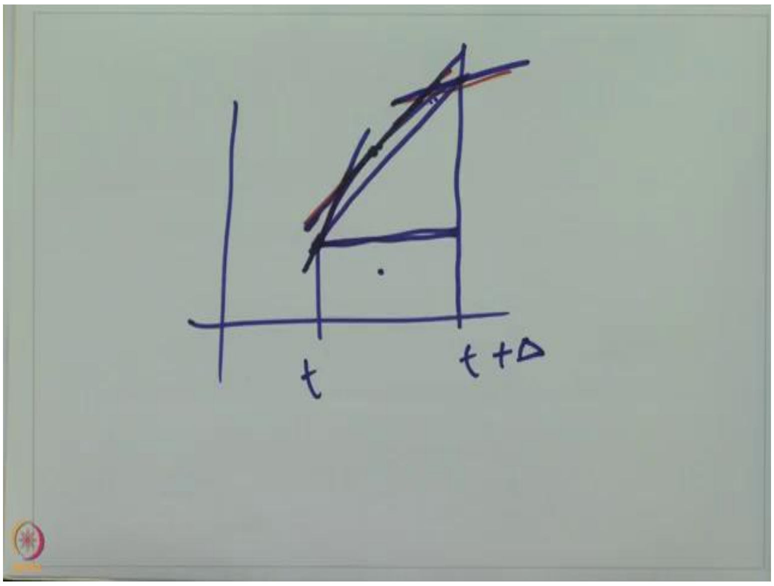
So, again, let me draw a picture, so this is time and say let us say this is t and let us say this is t + delta and on this axis we have f. So let say so I am plotting the curve let say something like this, okay? So this is the plot of f, now what does the first order approximation say? The first-order approximation. The first-order approximation says that the increment that you are going to get is going to be the distance, so this distance is delta so the time difference which is delta times the slope and what is the slope? The slope of the tangent, so the slope of the tangent is going to be something like this.

So this whole thing is going to be f'(t)*delta because this f' is nothing but the slope of the tangent, all right. So what is the second-order term, the second-order method say? The second-order method is saying that look, do not take the slope at time t sorry, do not take the slope at time t but take it at time t + delta/2.

So take it somewhere over here. So take it somewhere over here and now the slope looks something like this, okay. So I want to multiply this slope by this delta and then add that up. So to do that, I am just going to make it appear somewhere over here, so this slope I have just dragged it down, now I am going to multiply it by the delta and what has now happened is that the product is going to get me this.

So this is this slope multiplied by this delta, this whole thing is this slope see that the slope is going really sharp. So now it turns out that taking the slope in the middle is a good idea. In this picture also you can see that what I am getting to over here is much closer to the actual curve than this is. And there is, there is a different way of there is a different reason different way of thinking about it and that is that…

(Refer Slide Time: 7:23)

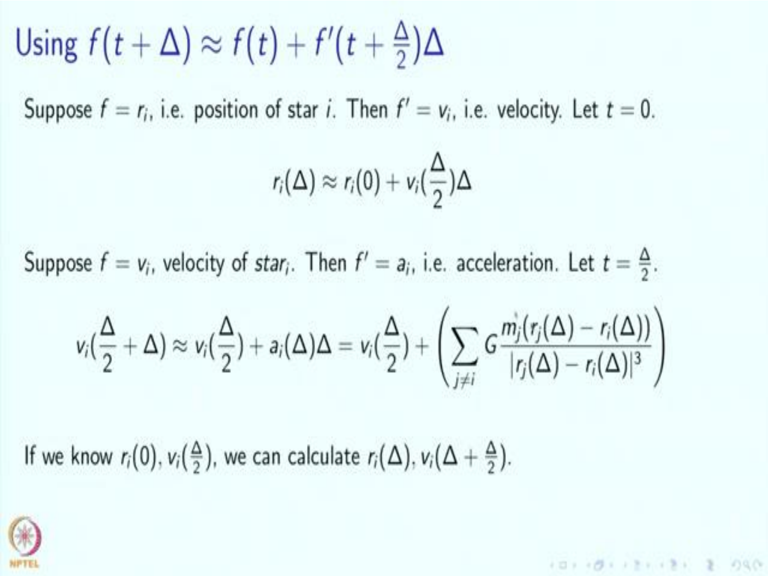


To go from here to here, if I take the so when I multiply this distance by the slope I am sort of assuming that the slope is going to be the same inside this entire interval. So if the slope was really the same in the entire interval then it would be just a straight line. In that case multiplying it would give me the exact answer.

But the slope is not the same, so should I take the slope at the beginning, should I take the slope at the end or should I take slope in the middle? Well, the slope in the middle is likely to be closer to the average slope and therefore we should be taking the slope in the middle. So that is really the intuition behind this formula. It comes out from the Taylor series but the intuition is this.

That I want the slope I want to multiply delta by the slope but I should pick a value which is sort of close to all the values in this interval. So this is t, this is t + delta. I should pick a value of the slope which is close to all the values. So what are the values? Well, the slope starts like this then it goes like this and it finally comes like this. So the slope in the middle which is this, is kind of like this, kind of like this, it is closer to both and therefore we get less error. So we just did this, we got the geometric interpretation and this is just to informally see why this method should be doing better.

(Refer Slide Time: 8:59)



All right, so let us now try to apply this in the context of cosmological simulation. So we will do exactly like how we proceeded before. So let say f is ri, so this f is ri and ri as we know is the position of star i. So what is f'? Well, f' is the derivative of position which is the velocity. Again, we are going to let t be 0. So what does this give us? So this tells us that the position at time delta is the initial position plus the velocity at the middle of the interval times delta. Now this may be another way to see why this new method is better.

Because what we are saying over here is that we are multiplying delta by the velocity in the middle. So we are saying yes, the velocity might be changing and we are hoping that the velocity in the middle is a better representative of the velocities in the entire interval. Intuitively the velocity gets lots of time to change from our selected velocity if we select at the middle but if we select it if we select it at the beginning. If we select it in the middle, then the initial velocity or the final velocity is not too far in time from the middle.

And therefore, the middle is a better representative, so that is as far as the position is concerned. So if we know the position at time 0 and we know the velocity at time delta by 2 then we can calculate the position at time delta. Next let say let see what happens if f is the velocity, so then f’ is the acceleration and again let us take time equal to 0, sorry this time we are going to take time equal to delta by 2, you will see why, Okay.

So what becomes of this? So t is delta/2, so we are going to get delta/2 + delta. This is going to be vi of delta by 2, remember we are substituting vi for f and t and delta/2 for t. And f’ is acceleration and t is delta/2 so this whole thing becomes delta*delta.

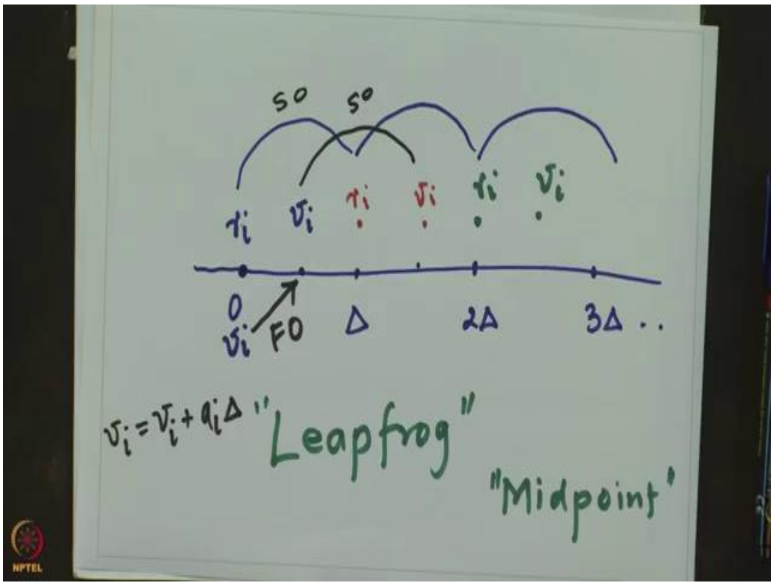
So this tells us how we could calculate the velocity at time delta/2 + delta and for that we need to know the velocity at time delta by 2 and then the acceleration at time delta. Do you know the acceleration at time delta? Well, as it happens we do, so this is the velocity term, this is the for the acceleration what I am going to get. Well, for the acceleration we already know that it depends only on the position oh, I think I need a delta over here, so there is a delta over here which I have which I need to add.

So this acceleration term is this term and this is the acceleration at time delta. Do you know this term? Well, we do if we know ri of delta and by the way when I say we can calculate ri, I mean we can do it for all stars. So we can know it for rj and everything as well, all right. So yeah, there is a delta term over here.

Basically what where, what has this led us to? So this says that if we know ri(0) and vi of delta then we can calculate ri of delta and vi of delta + delta by 2, okay let us just check again. So can we calculate ri of delta? Well, here is ri of delta, well we have ri + 0, yes we said if we know ri(0), do we know vi of delta by 2? Yes, if we, we said if we know vi(delta/2). So yes, if we know these two things, we will be able to calculate ri of delta. Will we be able to calculate vi of delta + delta/2?

Well, vi of delta + delta by 2 we are going to update as vi(delta/2). Do we know this? Well, yes we said if we know vi(delta/2), do we know these things? Well, all ri(delta) we have calculated in this step over here. So therefore, we know these terms as well and therefore we can calculate this. So what does this mean, what have we exactly accomplished? So I am going to draw a timeline over here.

(Refer Slide Time: 13:45)

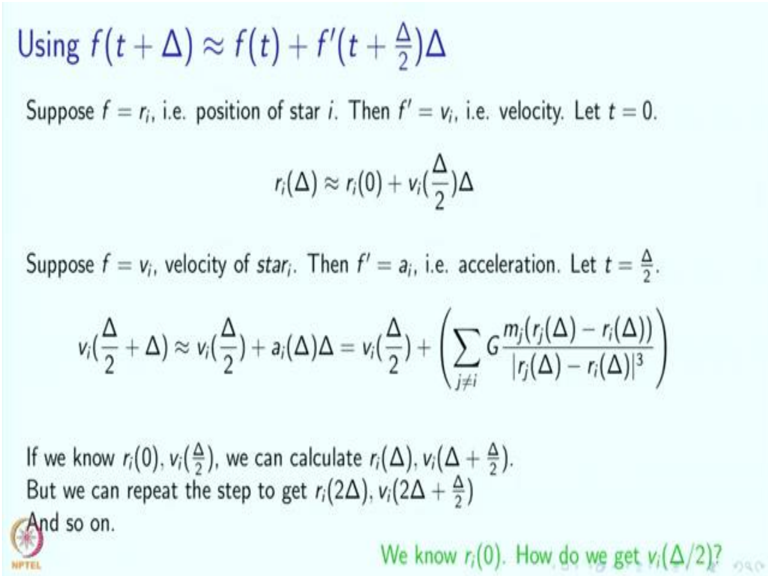


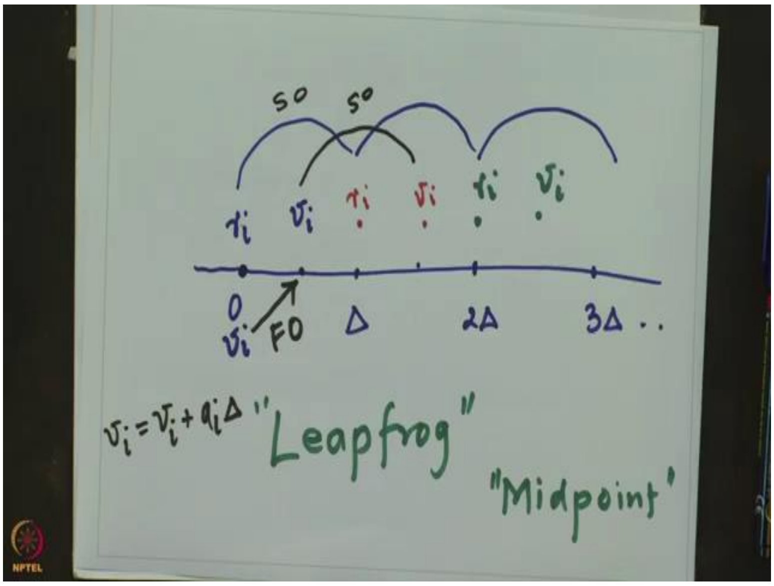
Okay, so this is 0, then this is delta then this is 2 delta, then this is 3 delta and so on. So at this point we know ri and at this middle point we know vi. And what that tells us that knowing these two things knowing this and this, we can calculate this and this but now we can apply the same argument or let me let me write it down properly. We can calculate ri for this point and vi for this point. Now I can apply the same argument to these two terms the same calculation. So I will be able to calculate the ri over here and the vi over here and I can keep going. So it is like the previous thing but ri and vi values are not at the same time, at the same time instant.

Okay, so they are at these staggered time instants. In fact what is going on is that the ri values are calculated at integral multiples of delta, odd multiples of delta/2. So these are the ri calculations and these are the vi calculations. So you can fancifully think of the ri values sort of jumping across the vi values and in fact this method is also called the Leapfrog method or it is also called the Midpoint method.

But anyway so this is how the updates were. We calculate these two first, from these two we can calculate these two, from these two we can calculate these two and so on. So that is exactly our algorithm. What I am going to write down now is just this in a little bit more like code first of all.

(Refer Slide Time: 16:09)

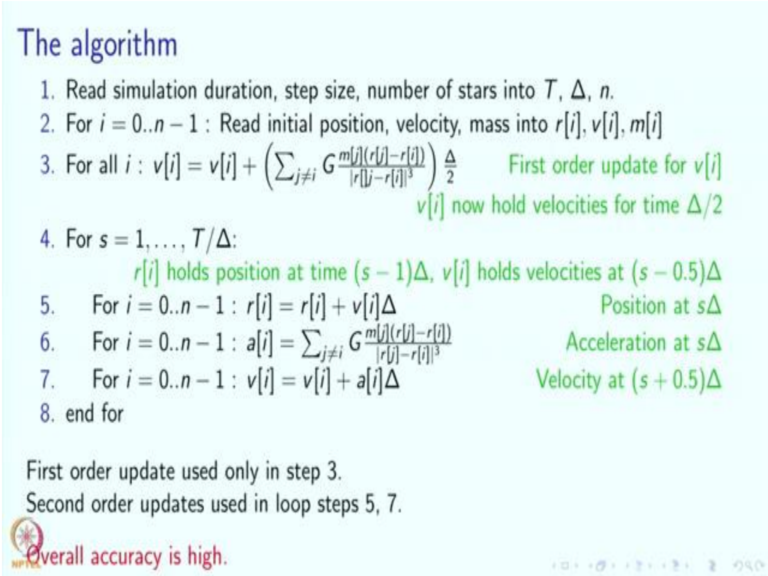


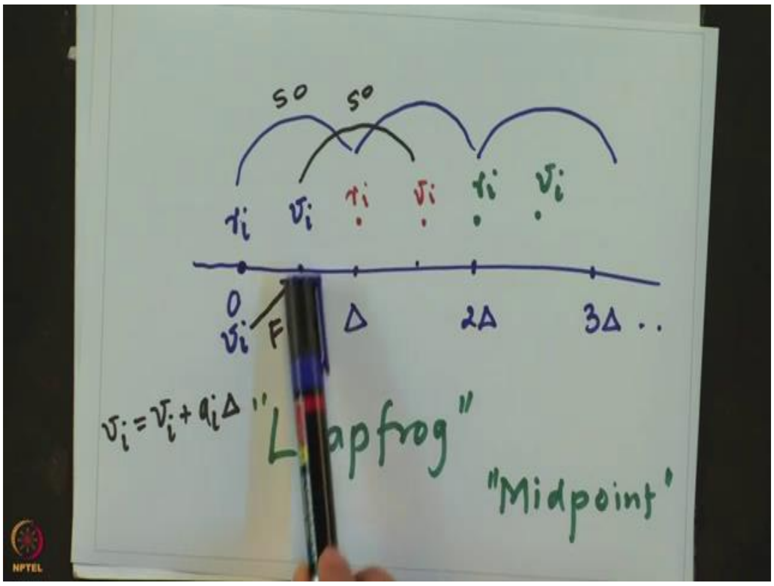


We can repeat the steps to get things for 2 delta and 2 delta + delta/2 and so on. Well, there is a little bit of problem though. So we said if we know these two things, but do we know these two things? Well, we do not, what was given to us were the positions and the velocities at time 0. So we know vi at this time and ri at this time, so our input specification said that we are given the positions and the velocity at time 0, we were not given this. So can we get this? Well, by now you should able to guess this, can we get this? Well, given the velocity over here we can always get this if we know the acceleration because that is we can do this by a first-order Euler expression, we have update. So to go from here to here we will use a first-order update but note that happens only once. So before we get the whole process started, we are going to do this first-order update to get the velocity at this point. So what is the expression needed over here? So we are going to have vi equals the original vi + acceleration times delta, so that is the first-order update. So that is what we are going to use here.

So we were given these two things, from that we got the velocity over here and ri already we have and now we are going to use the method on this slide to get these, these, these and how are we getting these? So this is a second-order update. This is also a second-order update, so all of these are second-order updates. So we do one first-order update at the very beginning and subsequently we only do second-order updates. All right, so that is what the code or the algorithm is going to have.

(Refer Slide Time: 18:11)





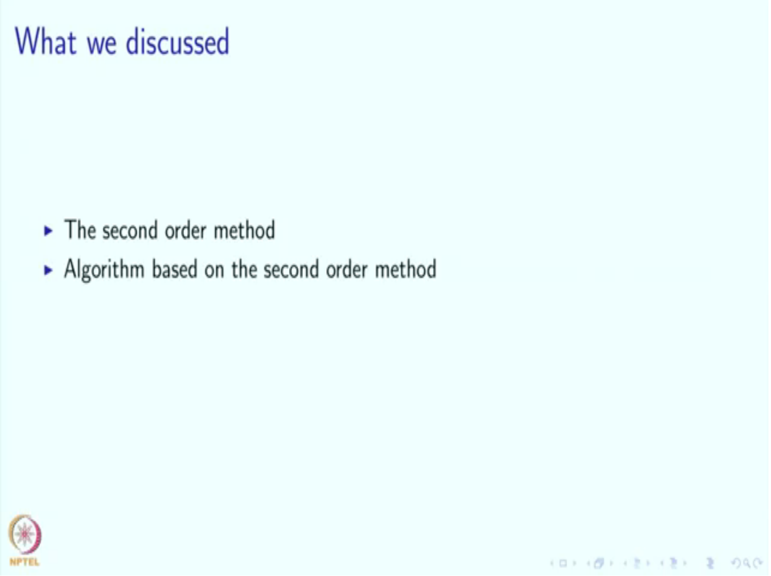
So as before we will read simulation duration, step size, number of stars into T, delta, n. Then we will read the initial position, velocity, mass into those arrays and then this is going to be the first-order update that we talked about, so calculating vi at this position. So at time delta/2 and you can see that that is why we are multiplying by time delta/2 and the values over here are at time 0. So these are values at time 0 and from that we are now getting vi at time delta/2.

And now we are just going to do this t/delta times and at the beginning of the loop ri will hold position at time (s – 1)*delta, vi will hold velocities at time s - half delta. So we are effectively at this point in our execution. So at this point we have advanced to this point, so ri is going to be over here, we have not got to the red values. So at this point ri is holding the positions at 0 and vi is going to be holding the positions at half.

But s is 1 over here, so this is 1 - half, so that is exactly the right thing. So exactly as we said, we will calculate the updated values of the position and that will get us the position at this time. So from this we got to this time. And then we will use the acceleration calculated earlier to go from this point to this point, so that will take us to acceleration sorry it should be no, yeah. So this is really the acceleration at s*delta not earlier. We are going to calculate the acceleration at this point. So we calculated the acceleration at this point and this acceleration we are going to add, so that will bring us to this point.

So as a result our velocity will be available at this point which is s + half, s + half times delta. That is it, that is the entire algorithm. So as we said earlier, the first-order update is used only in this step and over here and over here we are using second-order updates. So as a result the overall accuracy is high. This does turn out to be a good algorithm, all right.

(Refer Slide Time: 20:56)



So what have we discussed in this segment? So we have discussed the second-order method and we have discussed sort of the details of the entire algorithm based on the second-order method. In the next segment I will show you the program but we will take a short break.