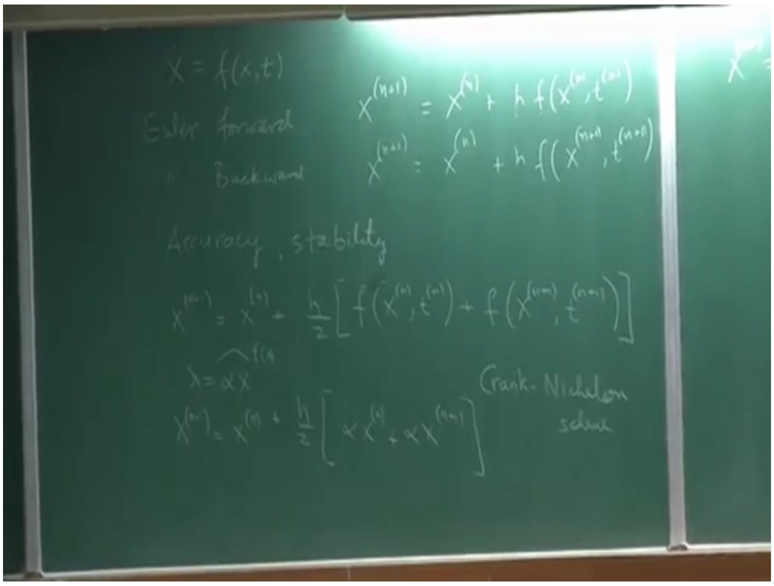
**Computational Science and Engineering using Python**
**Prof. Mahendra K. Verma**
**Department of Physics**
**Indian Institute of Technology, Kanpur**

# Lecture - 16
# ODE Solvers Continued

Let us do some more schemes which are better schemes than Euler, then I will show some demos. So, I (Refer Time: 00:25) ODEs today and then we will start PDEs.

(Refer Slide Time: 00:34)



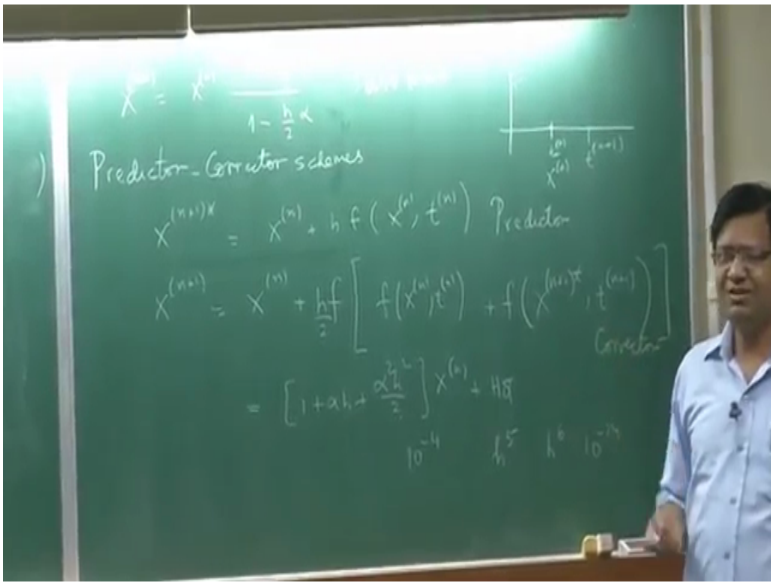

So, just to recap, we wanted to solve x dot is f x t numerically. So, I showed 2 algorithms Euler forward and Euler backward. I am going to change a location a bit for future. So, this was x, I wrote subscript n plus 1. So, I will write like this n plus 1 equal to x n plus h f x n p. So, that is forward Euler and x n plus 1 equal to x n. So, this is time something like n th step this is the number which step are you in that is what is telling you plus h f.

Now, here I have to compute at n plus 1. So, this is the better notation, it will become clear towards the end of the class. So, the 2 important concerns one is accuracy. So, it should be accurate method; not inaccurate. So, then your answer may be very much off and second is (Refer Time: 02:11). So, this stability is somewhat different than your usual notion it is defined for. So, if my solution is going to a constant value or going to a some value asymptotic value my numerical solution should also go to the asymptotic value, but not that value asymptotic; should not should not deviate it; it do not go to that value.

In fact, is quiet clear if your method is not accurate your solution from numerical scheme will could be very different than the asymptotic value, but it should not blow up. So, that is the condition I hope it is clear to everyone and we need to with certain that these 2 conditions are satisfied by the algorithm well I mean sometime we will still use scheme which are unstable, but you have to be careful sometimes is very difficult to prove stability for all parameters. So, now, today I want to discuss one more scheme which is stable, but which is better than Euler. So, that scheme is trapezoid rule based on trapezoid rule. So, I can solve this one as x n plus 1 is x n plus.

So, trapezoid rule is combination of the 2. So, I just combine the 2 f x n t n plus f x n plus 1. So, I put a factor half. So, this system is more well is who will be more accurate because I am taking 2 of them x n; x n plus 1 also it happens to be more stable. So, how do I prove the stability for the problem? I was doing in the last class suppose my equation is x dot is equal to alpha x then how does it reflect. So, x n plus 1 is x n this f x please remember this f x. So, h by 2 this will be alpha x n plus alpha x n plus 1, right, I just I am substituting.

(Refer Slide Time: 05:08)




So, this implies that x n plus 1 is x n. So, I will this is this. So, 1 plus h by 2 alpha and 1 minus is (Refer Time: 05:24). Now for alpha negative; the problem you remember; it was for alpha negative for alpha negative what is this; quantity is always less than 1, correct, alpha negative. So, (Refer Time: 05:45) is stable; what about alpha is i pi to only complex, if is purely complex then again mod is 1 purely complex. So, is unstable he set the border line, but is better than being unstable. So, this is another implicit scheme, it is this scheme, we use it quite often this called Crank Nicholson scheme. So, I will show you. In fact, we need this for future.

So, some schemes we have to remember like Crank Nicholson will be a important scheme this scheme we will use it in when we solve PDEs. Now let us do slightly better schemes; so 1, 2, 3. So, this is third scheme, I have told you. So, will have lots of them, I will not cover lots of them, but I will just show you similar 4-5. So, another class of scheme called predictor corrector schemes; so predictor corrector schemes. So, let us do predictor one; how does it work; predictor corrector? So, look, I do not know this guy know; that is the; that is one of the key problems here I do not know this; how do I compute it for arbitrary effects this was easy.

But for arbitrary effects will be a problem. So, what we do is I first predict not the (Refer Time: 07:45) with exact value; what I predict. So, I predict x n plus 1, I call it star. So, is a predicted value by Euler scheme? So, just use Euler scheme to ; so is going to be x n. So, h f x n; t n. So, I made a prediction; what is a x star; so a question.

Student: sir (Refer Time: 08:22).

Ok.

Student: (Refer Time: 08:24) f x n plus 1 t n plus one.

You do not know because. So, please remember. So, we are going from time t n to t n plus 1. So, I am; I know the value x n and t n and my and given function f; f is a derivative; numerical derivative based on f I want to go to the future.

Student: (Refer Time: 08:58) x n plus 1 (Refer Time: 08:59).

This one, yes.

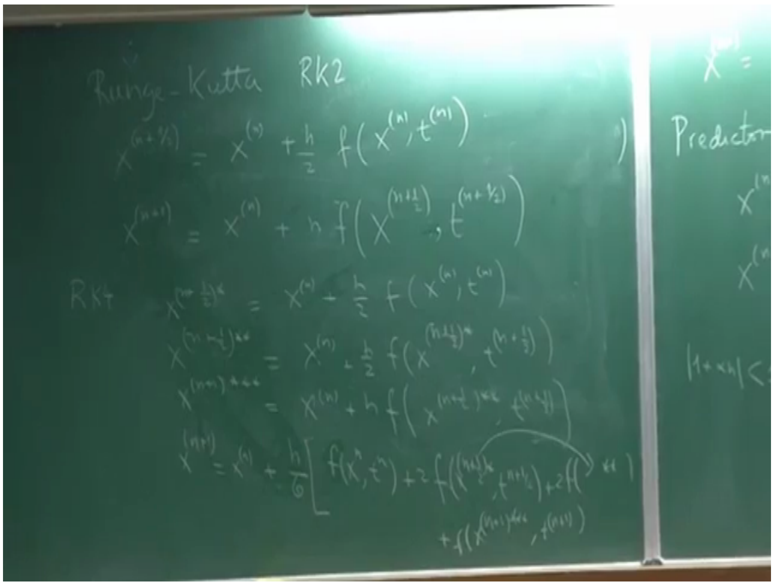
Student: (Refer Time: 09:03) x n (Refer Time: 09:04) n plus 1.

No, no; that is that is called lubricate scheme. So, I am writing with solution in terms of x n plus 1. So, it is true that I do not know it, but for some times is easy to if we equation is linear than I can solve it, but equation is non-linear than I can do iterative, but sometimes is simply not possible like for PDEs partial differential coefficient. So, one scheme is to do this. So, just wait. So, this is a predictor scheme this called predictor then I say that well my solution corrector x n plus 1 is one such scheme is x n now look at this I do not know this I think probably we have use. So, I am going to use this value which is my predicted value.

So, what I will say h half f of x n t n which I know this I can compute, but f x n plus 1 star t n plus 1. So, this called corrector scheme. So, I combine this one with predictor corrector and I will really solve it, I will really solve it explicitly. So, please remember these are the words which is quite often used called explicit implicit scheme. So, these are explicit scheme given x n I can find x n plus 1, right away, this called implicit scheme because I well I cannot solve x n plus 1 directly because x n plus 1 depends on x n plus 1 itself. So, for complex problems I may have to do iteratively or use something like predictor corrector, but these schemes have lot of value.

Because this schemes or this scheme is stable and we want stability. So, we can do Taylor expansion and we can show. So, this part is there in my notes. So, I will I think I will skip it. So, this one is one plus alpha h plus. So, this accurate up to second order you can do just Taylor expansion and figure it out is accurate taken out which is better than Euler scheme which is only accurate up to first order. So, in a project you should do at least second order not the first order.

(Refer Slide Time: 12:21)




So, another important scheme; so, I will erase this. So, this called Runge Kutta scheme. So, it has second order third order forth order. So, I will only mention second and forth also fifth order; sixth order.

But I do want to make a remark here you have suppose to take a small h right. So, let us say h is 10 power minus 4. So, is good up to 4 significant digits. Now if I use fifth order scheme which is accurate for fifth order. So, fifth order is scheme will be accurate to h 5 let us say. So, what is the accuracy of we got in this scheme 10 power minus 20. Now, should I go for this kind of schemes or sixth order why not say sixth order the problem with that will be this number will be 10 power minus 24. So, my last 6-7 digit will be comparable to the random numbers to the round of errors, I can only represent with double precision only up to fifteen digits or sixteen digits anything beyond sixteen digits is not reliable.

So, we should not also go very very high accurate schemes. So, that is something we should keep in mind that we should go to like h for six if our h is term of minus four, but indeed if h was 10 power minus this scheme I can use 10 power minus 2 as h. So, I can take a bigger time steps. So, these are some engineering issues you know we have to be careful when you apply your method is an art I mean you have to you have to be careful. So, how does Runge Kutta second order work? So, the idea is to go to the x mid. So, I go to the middle point and find the slope at the midpoint and use the slope to time at once.

So, x plus half equal to x n plus h; so, it should be h by 2, right, I am taking only half step. So, h by 2 f x n t now I say x n plus 1 is x n or you can call this is a midpoint x n plus h f x n plus half. So, this my answer is again accurate up to easily show this is accurate to second order h square is error of the order of h cube. So, I will not prove the stability of this; this is again more work. So, there will be a region of stability for Euler remember it was stable for 1 minus alpha h mod less than one or 1 plus alpha h for explicit. So, we can also derive what is the region of stability.

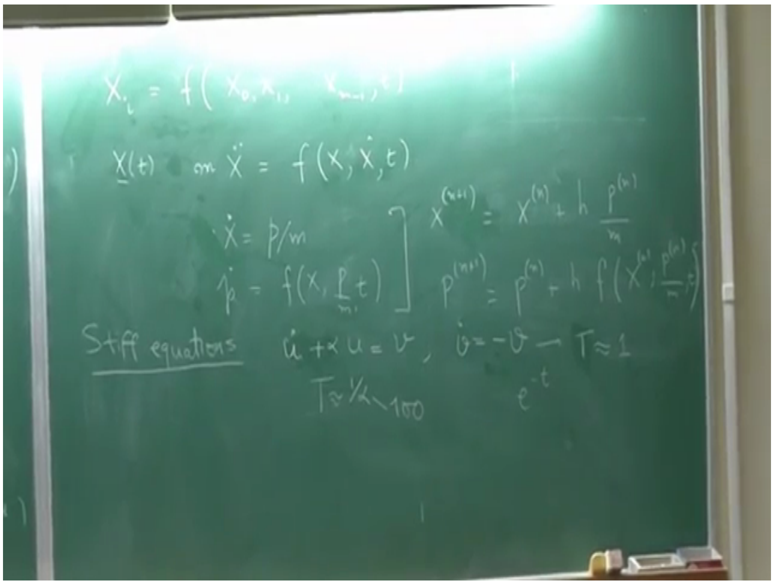
So, what alpha and h combination it is stable, but no this not the course I mean we can do a full course on ODE solvers numeric, but I will not get into that now one famous scheme is. So, this called r k 2 short hand r k 2 we also have r k 4 Runge Kutta 4. Now I recommend r k 4. Now write down the steps its bit long, but let me just write down the steps. So, it is combination of predictor corrector. So, you can also call it predictor corrector. So, this is like I am predicting half value and using that I am correcting. So, x n plus 1 star half star I just write it down h by 2 x n t n second step x n plus half double star.

So, this is another predictor x n plus h by two. So, I go to this using the slope using this values half values now then x n plus 1 3 stars is x n plus h f . So, this is third one now we can combine all of them the last step. So, x n plus 1 is x n using Simpson rule. So, h by 6. So, I am going to write go x n, please, this is not a power. So, bracket I am kind of tired, now 2 f x n plus half star now I have 2 more terms let me just write it here 2 same thing with the double star this values.

So, is remember Simpson has 4 midpoint this is factor 4 for the midpoint. So, you combine 2 for this one and 2 for this one now this guy's of course, a clever people the design all the schemes and this accurate up to h power 4 error is an h power 5. So, it accurate by h power 4 the last step is n plus 1. So, f x n plus 1 triple star this one this is Runge Kutta forth order. So, again you can implement it in your for your home work or project now all these are for solving one equation. Now what if I have more than one equation is a clear. So, far I mean you have tons of schemes I showed how many of them 5 of them; no 6 Euler forward Euler backward that was trapezoid.

Which is Crank Nicholson, then when did one predictor corrector that is 4, 5 and 6. So, 6 scheme that is good enough for your course there some more I may do maybe one or 2 when I do the PDEs quite let us start with this steps. So, I have said these equations.

(Refer Slide Time: 20:34)




So, one thing could be let us call them x; x is variables x (Refer Time: 20:41) x 1, x 2, x 3, like that x i. So, x i dot will be function of all the x is. So, x 0 x 1; I call it up to x m minus 1. So, there x m variable m variables n time; so, these I have these many variables that is a notation. Now you can see why I have changed it. So, i is the number of variables and the this is for your time step now. So, how do I solve this one? So, I have m equations.

So, you write down time step in scheme for all of them. So, given vector x. So, I call it a vector. So, given vector x a time t I can compute vector except t plus delta t for t plus h you can apply the same scheme you can choose one of them and you can apply it. So, let us do an example. So, in mechanics we have equation in 1 d for a particle of mass m; m is solved is force which is function of position velocity or time or and time I say and time. Now this second order equation first you first covert it to first order. So, how to I do first order? So, m x dot is momentum. So, this is first order then p dot is f x p by m t. So, I have 2 equations 2 first order equation. So, I can solve it. So, if I have a Euler scheme.

So, I am going to take to the left at to the right if I do Euler scheme, then this translates to x t plus delta t; x t plus h. So, I will say x n plus 1 flowing the same notation is x n plus h p n by n straight forward. So, given the momentum and position at time t, I can find out the value of these variables at time t plus delta t. So, p n plus 1 will be p n plus h f t. So, this how I solved it. So, it could be huge numbers if particle is move in 3 dimensions; how many equations would I have; 6. So, we (Refer Time: 23:46) 3 second order ODEs, but 6 first order ODE and what do they correspond to this access what are they called.

So, why do I need 6? In fact, it is a very important physical implication. So, this is the phase space. So, phase space is six dimensional and given initial condition which involves 3 position and 3 velocity or 3 momentum I can find the future uniquely. So, phase space trajectory you can draw. In fact, I really solving phase space trajectories is that. So, this is for set of equations which you will need it. So, when I do PDEs, I am going to convert to PDE into set of ODEs. So, I am. So, that is the idea convert PDEs instead of ODEs and solve ODEs by one of that I described, if I want to do Runge Kutta second order; what should I do I first go to the midpoint know from midpoint for both x and p, Then use a midpoint values to go to the t plus delta t.

So, I will not write it, I mean, but the; you should first think of the idea and then you just code it and do not code in the computer right away just write in the piece of paper now the last algorithm part is called stiff equations. So, this is also important. So, when I have set of ODEs, then they have different time scales or they could have different time scale. So, let us take an example. So, I will just show you by an example and we find it all the time. So, the equation I am going to use is x dot plus alpha, actually u dot; I am using u and v u dot at this v and v dot is minus v now all that non dimensionlized we did is not useless for many reasons and one reason is to solve this equations.

So, what is the time scale for this 2 equation what is the time scale for this; what is the time scale what come on what is the time scale sorry v dot is minus v. So, at what time the function will change come on this is disappointing it is one is already non dimensionlized if you think the time scale for this problem is one or lets call it t right is a solution is power minus t. So, in t equal to one it has dropped by 1 by e; please remember how to do non think of these scales; what about this. So, one time scale will come from here. So, that is that time scale another time scale comes from where.

Student: alpha.

Alpha; so, alpha is dimension of.

Student: (Refer Time: 27:26).

One by time; so, time scale here is one by alpha. Now here time scale is 1, here is one by alpha if alpha is too large, then these 2 equation are different time scales. Now you can think of 2 oscillator one has very high frequency another one is order one frequency. Now I would solve it in a computer then what should be my h or delta t what should be my delta t. Now this is a physics aspects; that means, you guys what should be my delta t. So, let us say alpha is 100; this 100 and this is of course, time scale is one. So, what should what should be my h.

Student: (Refer Time: 28:15).

Less than (Refer Time: 28:17) million.

Student: (Refer Time: 28:18).

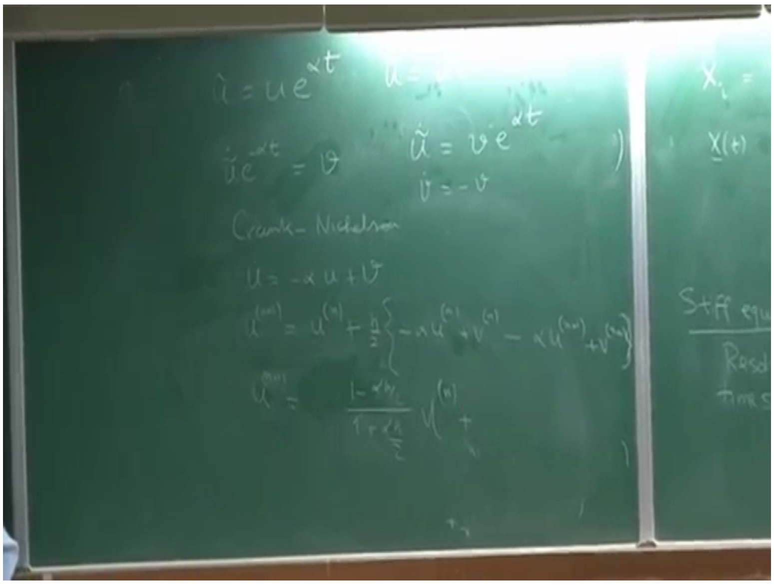
(Refer Time: 28:19) there is no second year, I am not (Refer Time: 28:24) 1000; 1 billion, 1 million, 1 by 1000. So, what should be the number?

Student: (Refer Time: 28:29) less than.

Less than 1 by 100; so, it should be resolve the time scales. So, these oscillating you know I am not oscillating; this is degrading for this guy trying to push it forcing. So, it should be able to resolve the lowest time scale. So, resolve the lowest times scale, if alpha instead of 15, there are suppose I am stuck there will be too much of problem. So, then we need to come out with some more clever scheme if 10 by fifteen means my computer will just sync it is not going to do anything that is 2 tiny time scale 10 power minus fifteen will be too tiny.

So, these are the situation which occur in nature like oscillations it all this big fast lasers, they have very high frequency femto second laser or what fed watt laser. So, they can correspond to this kind of situation. So, how do I handle this kind of problems? So, this called these are called stiff equations is a 2 time scales are far apart. So, we need to worry about stability and accuracy of the systems. So, one ways to; so, one thing is that you I hope you understand that you need to resolve both the time scale for accurate simulation, but there ways to overcome this. So, one clever way I will tell you is that I can make a change of variable for this equation. So, if I make a change of variable.

(Refer Slide Time: 30:32)



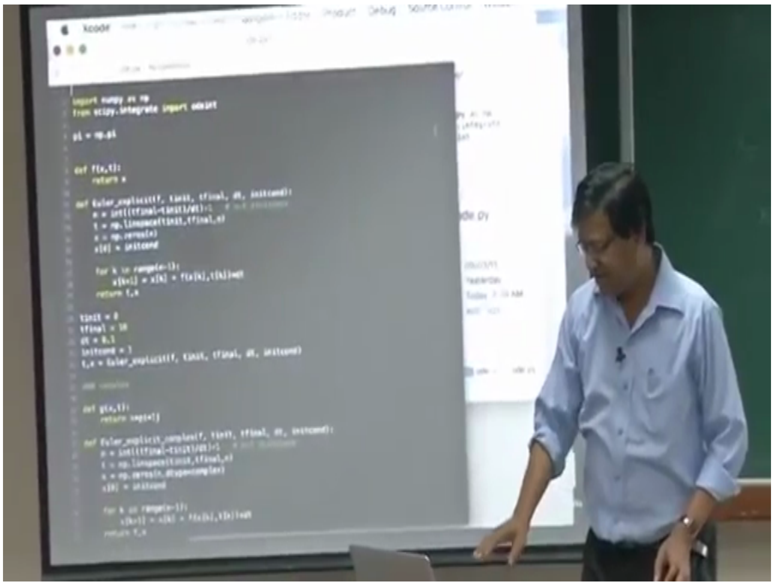
So, u tilde is u e to power alpha t. So, u tilde is a new variable let us take a derivative u dot is u dot alpha t plus alpha. So, now, I got u dot plus alpha u. So, I can now u idea. So, the left hand side can be written as this right u dot plus alpha u is that clear equal to v. So, my u dot is v alpha t. So, I got rid of; I got rid of my problem case alpha; alpha u term. Now I have this is order one no problem. So, you can solve this v dot is minus v. So, this exponential this called exponential trick and this has solved my problem.

So, this one standard trick to able to use higher delta h other thing is we use Crank Nicholson scheme, the one which I just said Crank Nicholson scheme. Let me use bigger h because this is stable for bigger h, it will not be accurate, but you do not need to take a tiny time steps now; so if I use Crank Nicholson scheme; what will be the time stepping for this. So, I use Crank Nicholson. So, it is going to be u dot is minus alpha u plus v. So, u n plus 1 is u n minus; so, a plus h by 2 at n and n plus 1. So, minus alpha n plus v n minus alpha; now rearrange it.

So, take n plus 1 to the left I get u n plus 1 is u n plus where this going to be exalt here one minus alpha h by 2 where d one is alpha h by 2. So, this u n part and v n part we have to also divide now this is the stability this will give you stability. So, Crank

Nicholson is a nice scheme for solving systems where alpha is. So, gives stability of 2 stiff equations that is. So, you can remember this one and will do it when we do PDEs. So, I solved I mean I finished ODE solvers. So, let me just do some demos is that clear or is that like not clear.

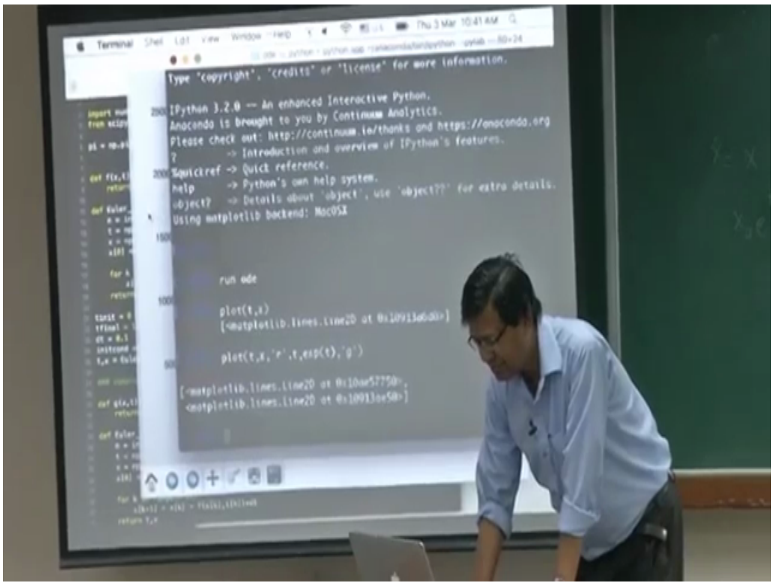
(Refer Slide Time: 34:51)




So, this part requires proof, but you can see this is a nice trick and you can use it also. So, this is there are 3 solvers for Euler explicit Euler implicit. So, is straight forward I mean this writing is easy right; I mean; so, Euler explicit is clear. So, given f any function f I have to give initial condition and final condition d t sorry t in it t final and d t. So, this is initial time final time d t and initial condition. So, give me this I will tell you the time and x the full time series. So, so the loop; so, this is going to be in a loop. So, I just say how many n i require t n x x 0 is a sign with a initial condition and I do a loop.

So, s k is f x k t k times d t d t is h. So, this is Euler explicit everybody agrees with that. So, I can get written t x and that is a answer. So, let us do an example. So, this my f. So, I put my f here t n it is 0 t finalize 10 and d t is point one initial condition I say my value of function is x one unit. So, my equation remember is x dot equal to x equal to x. So, my solution is x 0 e to power t it is growing my d t is point one you have to come closer. So, let me done this one; actually this does not come very nicely in let me just open a new window.

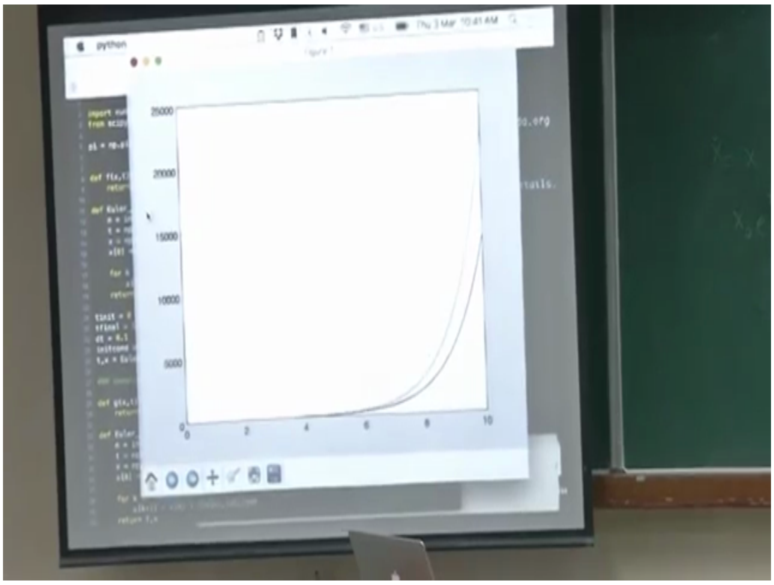
(Refer Slide Time: 37:22)




Run ODE. So, my answer is an t comma x my answer is here t comma x. So, I can plot t comma x plot t comma x.

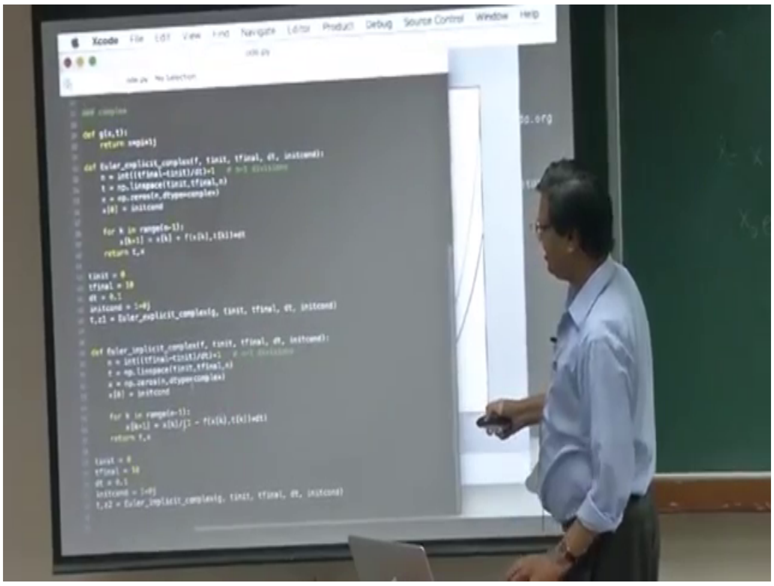
So, these are plot; now how good it is exponential is growing indeed. So, how good is it? So, I can put the; to put the exact value this is straight forward know exact values. So, let us put that t comma. So, I will put this is in red t comma is a exponential e x p t comma g 3.

(Refer Slide Time: 38:26)



Now, these; a correct value and these; my value coming from the numerical port I can see the error is huge. So, this around close to fifteen thousand fourteen thousand as a exact value is close to 22000 at t equal to 10; my h of course, too tiny not too tiny point one too big and this is the first order scheme an error will keep growing. So, it is a bad scheme; let us do the implicit scheme.

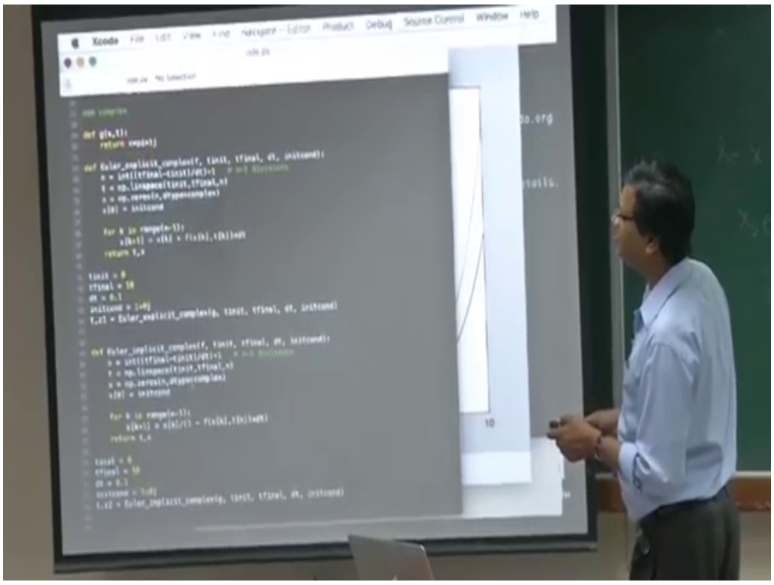
(Refer Slide Time: 39:00)



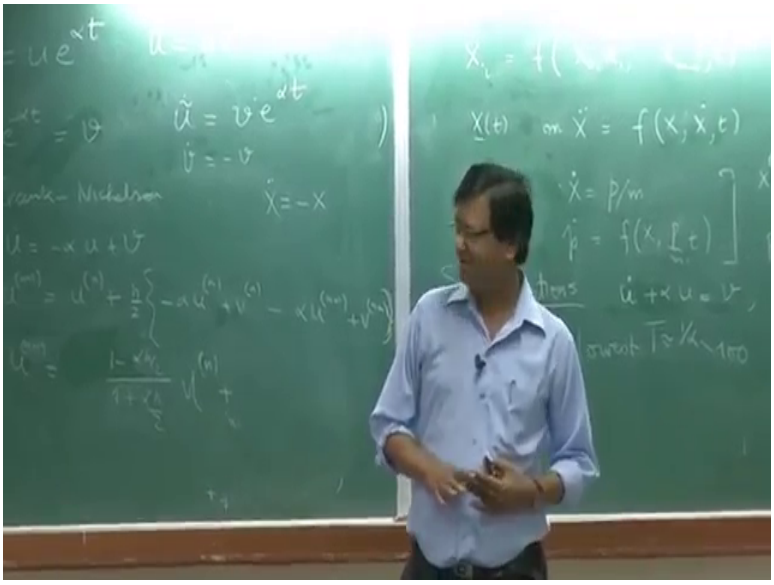
So, you play around with this various options you change h. So, implicit; so, this is implicit scheme is here now. So, I said x k divided by one minus. So, 1 minus alpha 1 divided by 1 minus alpha h; so, this I have derived in the class this of course, works when I can take it to the left hand side for linear equation, I could take it this to the left hand side. So, this is now you have to take a look at your notes. So, this; what I have done in my notes 1 x k divided by one minus alpha h. So, this alpha h. So, this is stable and you can also do.

So, let us do this one implicit. Now I want to; so, this for complex z my initial condition is one and. So, remember I had this is for where is my. So, this 1 g.

(Refer Slide Time: 40:12)



(Refer Slide Time: 40:15)



So, z dot is i pi z. Let us put a minus because in the class, I had put a minus sign, all right. So, let us run it and my answer is in z 2; my answer is Euler implicit z 2.

Student: Sir.

Yeah.

Student: (Refer Time: 40:45) Euler explicit (Refer Time: 40:46).

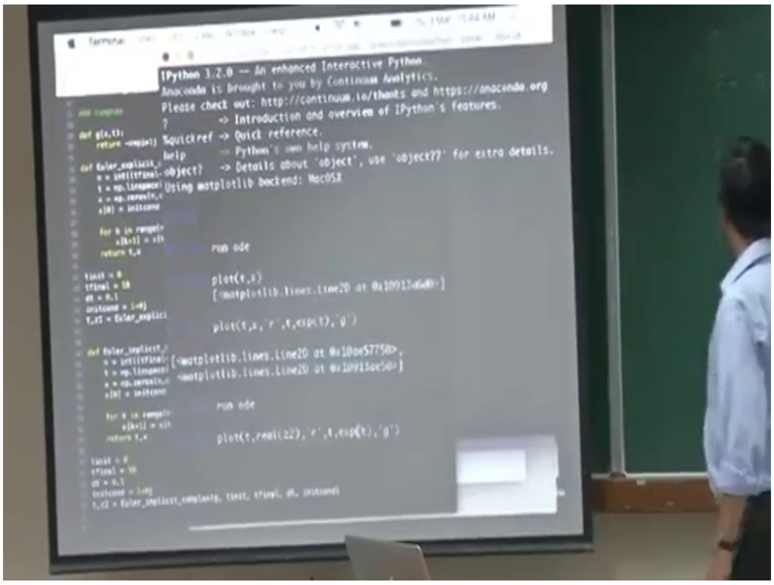
Student: (Refer Time: 40:48).

This f you mean;

Student: (Refer Time: 40:56) function (Refer Time: 40:57).

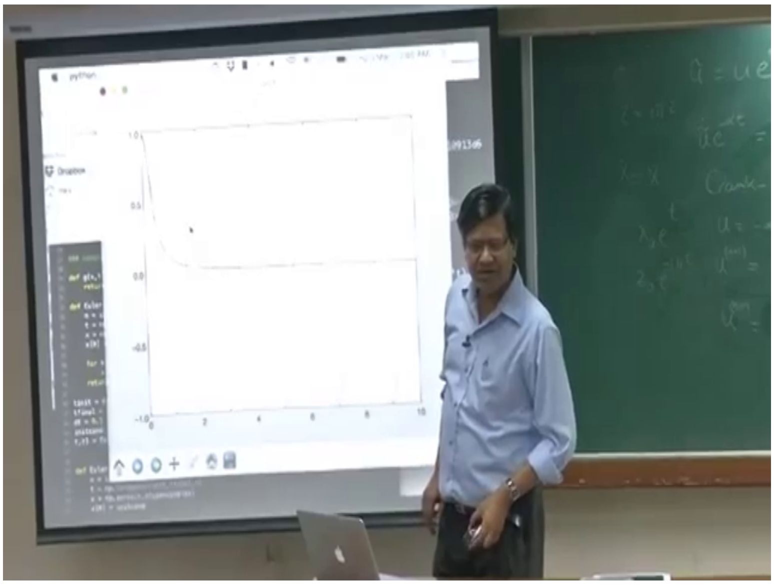
The function itself; so the function; this function I call from here and this function is defined here. So, you send the function itself how it does inside is not our problem you just send the function. So, I am going to look at z 2; z 2 is a complex; it has real part and imaginary part and so, solution is straight forward know is z 0 e to the power minus i pi t and z 0 is one is oscillating.

(Refer Slide Time: 41:41)




So, let us run again. Now I have to real of z it was p know t real of z 2 and what is real of this one; cosine pi t cosine pi t cos; it should work where is the figure. So, let us fill this first.

(Refer Slide Time: 42:17)



So, which is correct value and which is my numerical value correct value is this oscillation, right, cos t my numerical value is going to 0 it is not unstable. In fact, it is stable, but it is going to 0. So, these are the issues you may have computer we have very good computer, we do not have good scheme is not going to solve your problem. So, I had used my x small that you work it out I mean my time is running out very soon. So, you have small h and higher order scheme. So, Runge Kutta fourth order is going to do better; now Runge Kutta fourth order you please implement it.

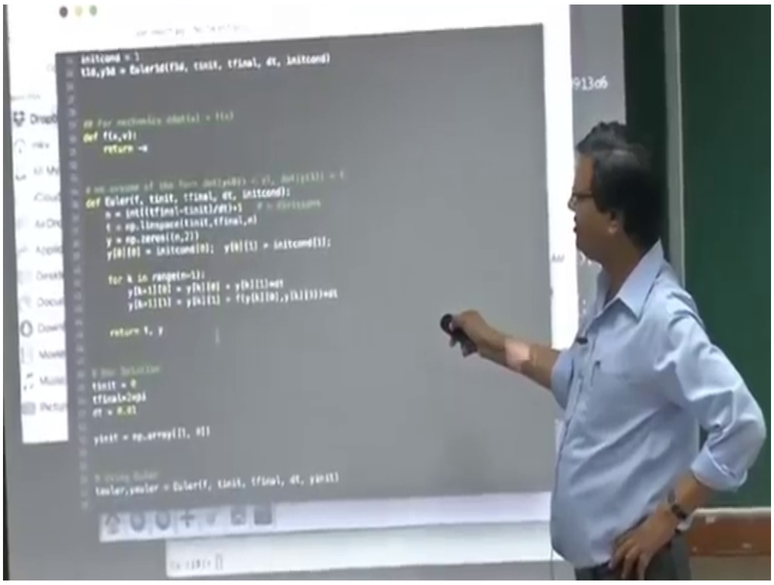
Student: (Refer Time: 43:16) x time p (Refer Time: 43:17).

That is right, it does not matter. So, it is going; if it is t is not, there it will just put x.

Student: (Refer Time: 43:28).

So, whatever x is there we will put the first argument the second argument the computer only understand what is the first argument and the second argument. So, let us solve one mechanics problem. So, I will close this one. So, I will leave these codes in the drop box. So, we want to solve oscillator problem. So, oscillate problem is here. So, x double dot is minus x oscillator with frequency one. So, I convert it 2 two first order equations. So, I have written here dot of y 0 is 1.

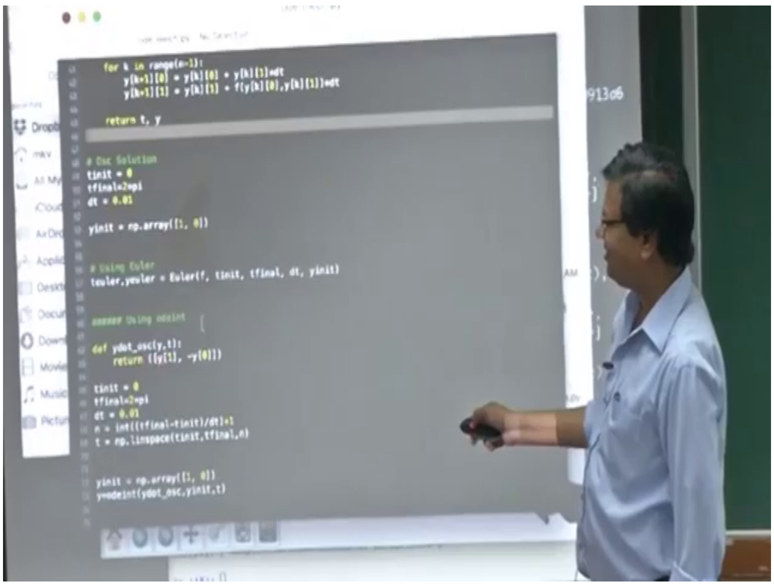
(Refer Slide Time: 44:27)




So, this like position and this will have momentum and dot of p is force. So, I can do Euler. So, this is what is coming here. So, first is these are time stepping up momentum this is a force here and this time taking a position now is going to give you y 0 y 1.

So, this something we have to keep in mind y 0 is position and y one is momentum it will gives an array of. So, you have to create an array the first one is first argument second argument. So, for my mechanics problem I say that first one is velocity position and second is velocity. So, this going to do that work and I can solve it; so t Euler and t u y Euler. So, I saw; I push in here f with initial condition final condition d t.

(Refer Slide Time: 45:33)



So, let us see quickly how it works oh sorry, I made one mistake. So, if I had 2 of them then I have to give you I have to give you both. So, x dot is momentum. So, now, this is x dot is momentum and p dot is minus x.

So, this is x, y is 0 and this y 1. So, what is y dot of 0? y 1 and what is y dot of; y 1 dot? Minus y 0. So, this has to be provided (Refer Time: 46:26), I am mixing up, I am slightly mixing up I also do not remember all the code. So, for my code I need only f this is what is coded by me? So, my code I just need f and if you run it you will get t Euler and y Euler and all my results are in y Euler. So, you can run it and you will find it will be oscillating with errors now python also gives its own ODE solver. So, python ODE solver is called ODE int, this one ODE int now what does python ODE int want; it wants what is y dot. So, you can it can also solve set of ODEs.

So, you have to give derivative for each variable. So, this; what I was writing y dots and that for ODE int I need this, this, this y dots. So, ODE int is a very useful function you do not need to code yourself and this reasonably accurate. In fact, you can also specify accuracy. So, y int and t y int will be it has 2 arguments y int is first is position and second is velocity. So, initial position is 1 initial velocity is 0. So, this is I supply all this here and I will get the result in y result will be y and of course, t you have to create beforehand. So, this time lean space. So, you give it t y initial and your y dots you do not need to give d t; d t it will compute itself; so t array.

So, it will not tell you what function is using which scheme is using where you specify the errors if you want and they will give you reasonably accurate answer, but. So, this what you can use it for your project no problem, but you should also if I tell you that use Euler scheme you should use Euler scheme is also good idea to know what these schemes do because a large project you are not going to use python ODE int; these only for a course tiny 5 minute project, if I am running on a super computer, I cannot use any of this.

So, that time I have to be careful which scheme I should use. So, I will stop.