Finite Element Method

Prof C. S. Upadhyay

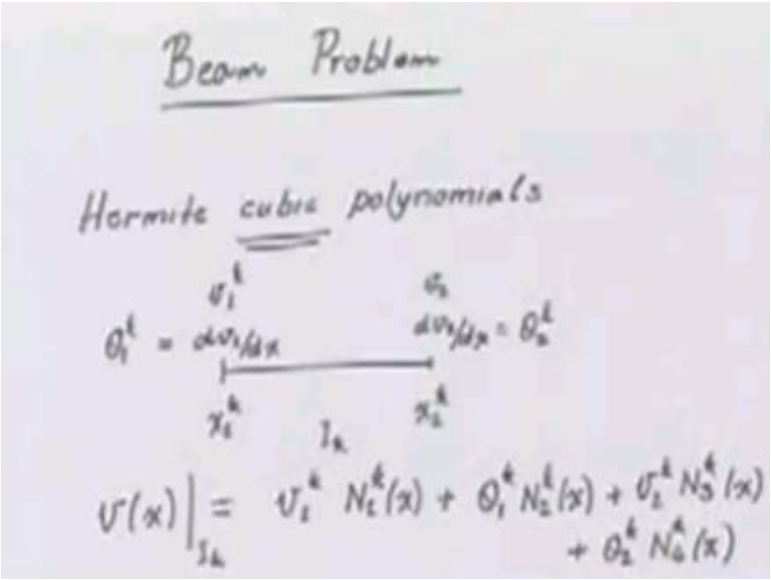
Department of Mechanical Engineering

Indian Institute of Technology, Kanpur

Module – 6 Lecture – 3

In the previous lecture, we had started talking about the beam problem. What was the difference between the beam problem and the bar problem that we had done earlier? The difference is that the beam problem is a fourth order differential equation. For this, we had defined the weak form; we had also talked about the boundary conditions that are possible. We had started talking about the shape functions that need to be constructed at the element level; the minimum order shape functions in order to satisfy the C one continuity that is imposed by the smoothness requirement that comes out of a weak formulation.

(Refer Slide Time: 00:26)

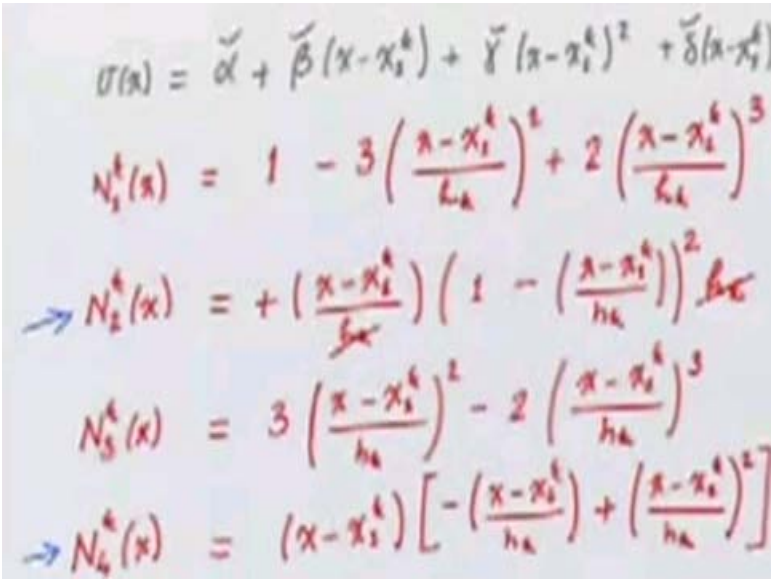


We have said that we are going to construct the so-called Hermite cubic polynomials, which we have said at the element level will lead to an interpolation of the value of…. So this is my x_1 of k element; this is my x_2 of k for the element I_k, so it will lead to a interpolation to the value of the displacement at the element level and dv_1 dx at the element level; similarly, at the two nodes v_2 and derivative here.

So value on the derivative or the slope and the value at the slope at the two ends of the element. So, we need to interpolate between the four values and the minimum order polynomial that will interpolate four values will be a cubic. That is why the starting side of polynomials - the minimum set - is a cubic polynomial. As compared to what we had in the case of the bar problem where the minimum order polynomial was the linear; this was linear; remember that here we start with the piecewise cubic.

We had said that essentially we will write v of x is equal to v_1 of k v_1 over the element I_k v_1 of k into N_1 k plus - this dv_1 I am going to call it as $theta_1$ of k - $theta_1$ k N_2 k x plus v – the displacement at the other end - transverse displacement, v_2 of k N_3k x plus - this I am going to call as $theta_2$ k N_4 k x - the function of x. Now it is a question of constructing these polynomials.

(Refer Slide Time: 03:45)

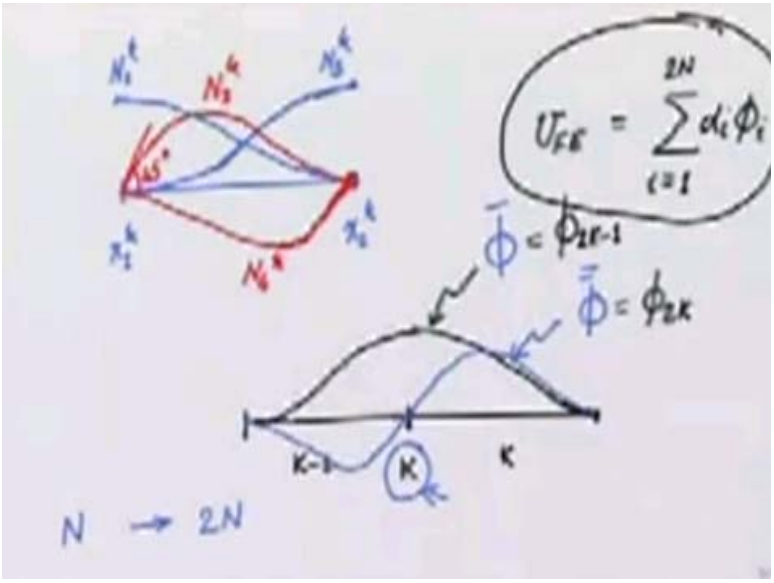


If I go ahead and take v(x), as I told you earlier, is equal to some alpha plus beta into x minus x_1 of k plus gamma into x minus x_1 of k whole square plus delta into x minus x_1 of k whole cube. If I took this expansion which is a cubic, here I go and satisfy those conditions, I should be able to obtain the constants alpha, beta, gamma and delta and from these constant I should be able to rearrange everything in terms of v_1 of k $theta_1$ k v_2 k and $theta_2$ k.

I will get out of that N_1 k of x is equal to 1 minus 3 x minus x_1 k by h_k whole square plus 2 into x minus x_1 k by h_k whole cubed. Similarly, N_2 k as a function of x comes out to be: minus x minus x_1 k over h_k into 1 minus x minus x_1 k over h_k whole squared, this whole square into h_k. I can effectively cancel this h_k out; take this one out. N_3 k of x becomes 3 into x minus x_1 k by h_k whole squared by minus 2 x minus x_1 k by h_k the whole cube. Similarly, my N_4 k as a function of x comes out to be… N_2 k will have this plus sign here (Refer Slide Time: 06:36 min). Again here it will be x minus x_1 of k into minus of x minus x_1 of k by h_k plus x minus x_1 k by h_k the whole squared.

N_1k is 1 minus 3 x minus x_1k by h_k whole squared plus 2 x minus x_1k by h_k the whole cube into k is x minus x_1k into 1 minus x minus x_1k by h_k whole quantity squared . N_3 k is 3 x minus x_1 k by h_k whole squared minus 2 x minus x_1 k by h_k whole cube and N_4k into x minus x_1k into minus x minus x_1k by h_k whole squared plus x minus x_1k by h_k whole squared. These are the four functions that we have at the element level. Look at one thing that these functions N_2 of k and N_4 of k correspond to the slope that is dv dx at the two points. N_2 of k and N_4 of k - in these there is a difference in the dimension by a factor of h_k. That is this into h_k will give you the same dimensional consistency as the N_1 or N_3. Let us now go and look at how these functions look.

(Refer Slide Time: 08:35)



If I am talking of these functions at the element level the first function N_1k should be 1 at this point, at the point x_1k, it should have a 0 derivative at this point and it should have a value 0 and the derivative zero here. Why? Because when I put it back in the expression for the value v at that point becomes is equal to v_1 of k; the derivative becomes $theta_1$ of k and so on. So because it is a linearly independent function, it so happen at when you take a derivative you should only be left with the part coming out of in the $theta_1$ of k. This function will be 1 here, the value is 1, slope is 0 and it comes down to value 0 and

slope 0; this is my N_1 of k. Similarly, let us take the similar one first; N_3 of k will have a value 1 here, slope 0. So N_3 of k will be like a mirror image of N_1 of k this becomes N_3 of k.

Let's make the N_2 of k. N_2 of k what will it have? If I look at N_2 of k, it should have a slope 1 at this point, value 0 and slope and values 0 at this point (Refer Slide Time: 10:17 min). It should have a slope of 1, value 0 and we come down this. This is my N_2 of k. If I look at the slope, this angle is 45 degrees; if you want to say that. Similarly, let me make N_4 of k. N_4 of k corresponds to theta$_2$ k. It will have a value 0 here, slope of 1 at the point x_2 and value 0 and slope 0 at the point x_1. So that function will look like this. This is my N_4 of k. So these are our four shapes functions at the element level.

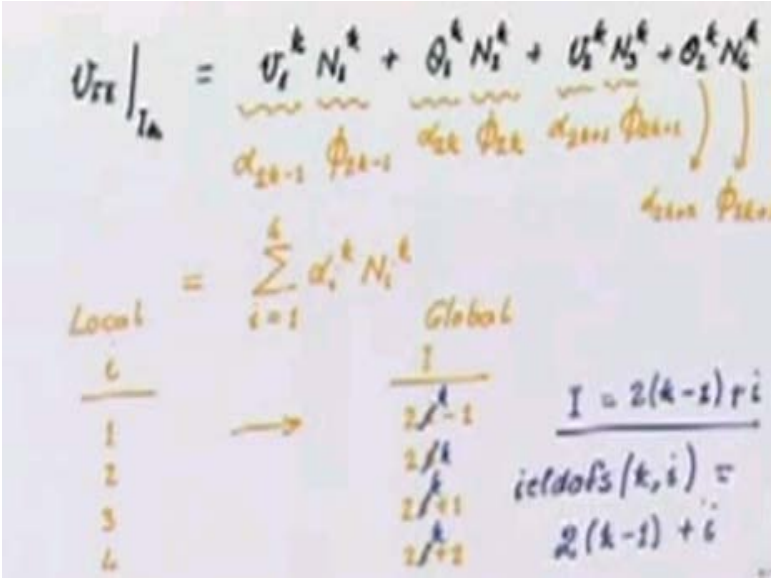
Simple question: what are then the global basis functions? Because remember we are still, as far as the approximation is concerned, we are creating this global basis functions which are C one functions and the pieces of this global basis functions are these shape functions in the elements.

Let me draw two of these representative functions. Let us take for example the function which interpolates the value of the v at this point. Let this be a generic point k. If I look at this function, in this element, this will be element k, this will be element k minus 1. If I look at this, this is going to be my first phi - the global basis function and then this corresponds to the value. The second one, let us look at this same point which corresponds to the slope will have a derivative one here. It is going to do something like this and this is my second function. (Refer Slide Time: 12:43 min)

So corresponding to a generic node k, the first global basis function corresponding to that node is I will call it phi bar, is this kind of a function which vanishes at the ends of the elements k minus 1 and k, and after that its smoothly goes to 0; stays as 0 everywhere else. The second function phi double bar corresponds to this slope being 1 here, which means it interpolates the slope. You see what it does it has the slope 1 here, value 0 and it comes down finally at the ends of the elements k minus 1 and k it becomes 0. This is how

we are going to construct the generic basis functions corresponding to a node k. Our approximation will be in terms of….So how many basis functions we have now each node? We have 2, because each node in the case of the C one approximation has 2 unknowns assigned to it; these unknowns are the value of the function and value of the derivative of a function. If I have N nodes, then there will be 2N such functions. My generic approximation, in this case, I can write that v finite element for an N noded mesh is sum I is equal to 1 to 2N alpha I. Corresponding to the node k which phi I do we have? So corresponding to the node k it will be actually equal to phi, if you check it out it will be 2k minus 1 and this will be equal to phi 2k or we will have the basis functions corresponding to the node k or the phi 2k minus 1, in the phi 2k which goes into the representation of the finite element solution over the whole domain.

(Refer Slide Time: 15:34)



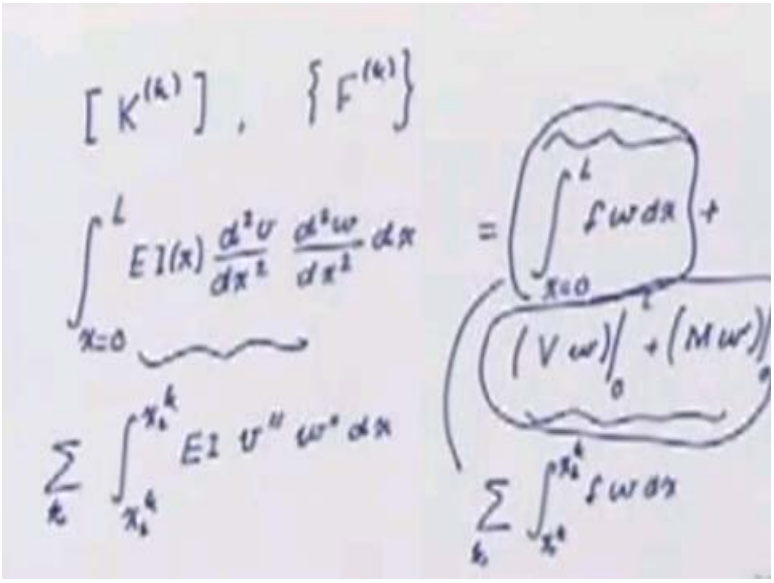
Now, by the same taken by what we have done here, we have done this, what is then my representation of the finite element solution in the element? Just like we did earlier, we will say that v_{FE} in the element k is equal to v_1 of k N_1 of k plus $theta_1$ of k N_2 of k plus v_2 of k N_3 of k plus $theta_2$ of k N_4 of k .

It is quite easy to see that what is this corresponding to? This corresponds to the global alpha 2k minus 1. This is the piece of the global phi 2k minus 1, because the first node of the element k is the node k. This corresponds to global alpha 2k and this corresponds to the global phi 2k; this is alpha 2k plus 1 phi 2k plus 1; similarly, this corresponds to alpha 2k plus 2; in this phi 2k plus 2.

We could also write that this is equal to at the element level by renaming v_1 of k and so on. I is equal to 1 to 4 alpha I of k N_I of k, where we understand what are alpha I in the k mean. If I do this convention of renaming the v_1 of k as $alpha_1$ of k; $beta_1$ of k $alpha_2$ of k; v_2 of k as $alpha_3$ of k and $theta_2$ of k as $alpha_4$ of k. What kind of local to global correspondence do we have? If I write here i 1 2 3 4; this is local; this corresponds to global. Let me write it as big I; big I is equal to 2i minus 1, 2i, 2i plus 1, 2i plus 2; this is the global. Now it is quite easy to write it an algorithmic way that the global I is equal to 2; the global actually should be 2k minus 1, 2k, 2k plus 1, 2k. This one will be 2k minus 1 plus i.

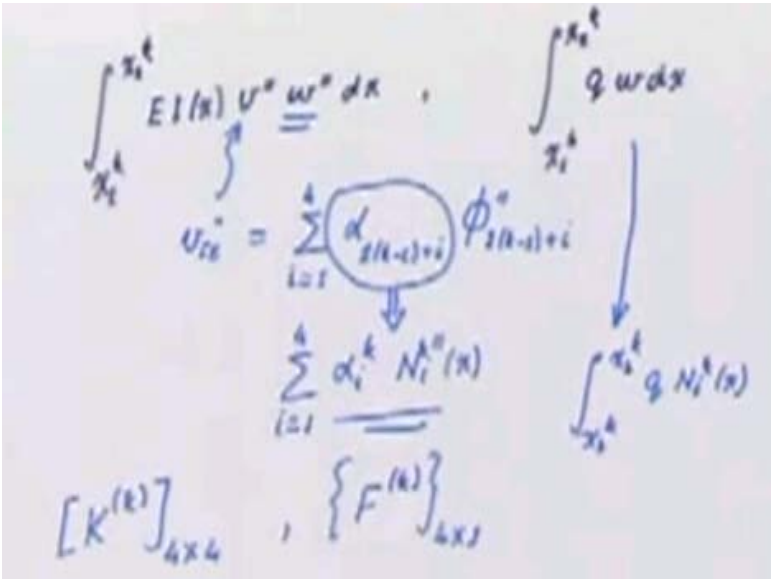
If I write it like this, you check that for i is equal to 1 is become 2k minus 1; for i is equal to 2 it becomes 2k; for i is equal to 3 it becomes 2k plus 1. This I can now load in my ieldofs. So ieldofs for the element k, i for the element k the local ith degree of freedom is equal to 2k minus 1 plus i. This gives me the local to global enumeration. Once we have obtained these things, then I have a complete correlation between the element level approximation and the global approximation.

(Refer Slide Time: 20:12)



If we now see what does the elements stiffness matrix and the element load vector turn out to be? If you see this, the global stiffness matrix will come from this expression. This will give me the global stiffness matrix and this remember that on the right hand side I have x is equal to 0 to L f w dx plus shear force into w evaluated at 0 and L plus bending moment into w prime evaluated at 0 and L. This quantity plus this quantity is going to give me the load vector. Remember that as we have done earlier for the second order problem, these boundary conditions we are going to impose later. This essentially will go into our load vector - this part (Refer Slide Time: 21:33 min). If I look at first stiffness part, where is it going to come from? The stiffness part, this obviously, will come from the partitioning over the element k, sum over x_1 of k to x_2 of $_k$ of EI. I am now writing the second derivative as the double primes; w double prime dx. This is the sum; this integral can be written in terms of this sum; similarly, for the load, I can write sum over the element k integral x_1k to x_2k of f w dx.

(Refer Slide Time: 22:29)

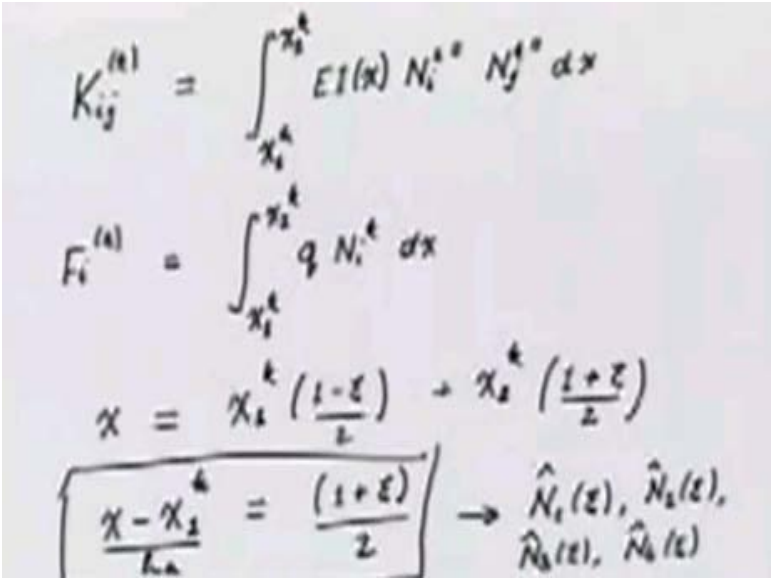


So, at the element level which are the integrals that are active? It is x_1 of k to x_2 of k EI of x v double prime w double prime dx and integral x_1 of k to x_2 of k qw dx; q is my transverse load that I have applied; so I am using f or q accordingly here; I am mixing the two, but it should be q, because according to the convention for beams process load is generally given by a q. If I look at this, the finite element approximation is replaced here instead of v; I know the finite element approximation will be equal to the sum of I going from 1 to 4 alpha 2k minus 1 plus i phi double prime 2k minus 1 plus i. It is a sum of this four and remember that this is equivalent to writing the sum over i going from 1 to 4 $alpha_i$ of the element k into N_i of the element k double prime. Similarly, for the q, I will get it as sum… here there is no sum involved. In the q, I will get the integral as integral x_1 of k, x_2 of k q; it is N_i of k , because here the w also what happens? w is also made of this $N_{i,}$ because these are the only parts of w which is non-zero in this element.

For v, I have this representation; for w, now I will write, I will put w is equal to N_1 of k and 2k and 3k and 4k and I will get the 4 equations to which this element is going to contribute. Which columns will it go to? So if I put w is equal to N_1k, N_2k, N_3k, N_4k it means which rows it is going to? It is going to the row 2k minus 1, row 2k, 2k plus 1 and

2k plus 2; columns will come from this $alpha_i$ part, so columns will be the column 2k minus 1, column 2k, column 2k plus 1 and column 2k plus 2. In terms of this element representation, I am going to have the elements difference matrix K to the power k which is of size 4 by 4 and the element load vector for by 1.

(Refer Slide Time: 25:59)

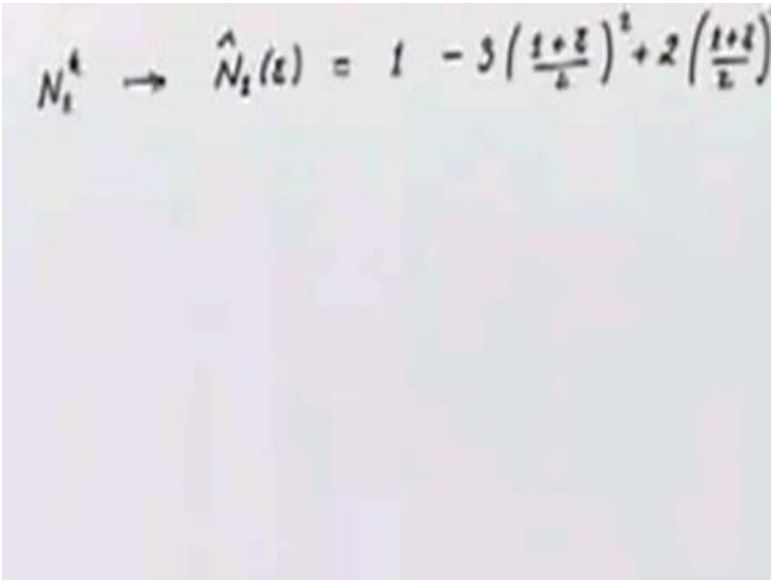


Let us now give the entries of the element stiffness matrix; it is going to be… we have done enough of this, so we can now go back to what we have done to the previous slide and see that it is going to be EI x N_ik double prime N_jk double prime dx and F_ik is equal to integral x_1 of k to x_2 of k q N_i of k dx.

The question is that we have done everything at the master element level. How do I convert everything into the master element level? Again we are going to put in the transformation x is equal to x_1k into 1 minus psi by 2 plus x_2 k into 1 plus psi by two. If I put this transformation then in the definition of N_1k, N_2k, N_3k and N_4k all I have to do is substitute x minus x_1 of k is divided by h of k is equal to one plus psi divided by 2; we check from here that x minus x_1 of k will be equal to what? x_2 minus x_1 of k into psi by 2, and x_2 minus x_1 is $h_{k.}$ x minus x_1 of k divided by h_k is equal to 1 plus psi by 2.

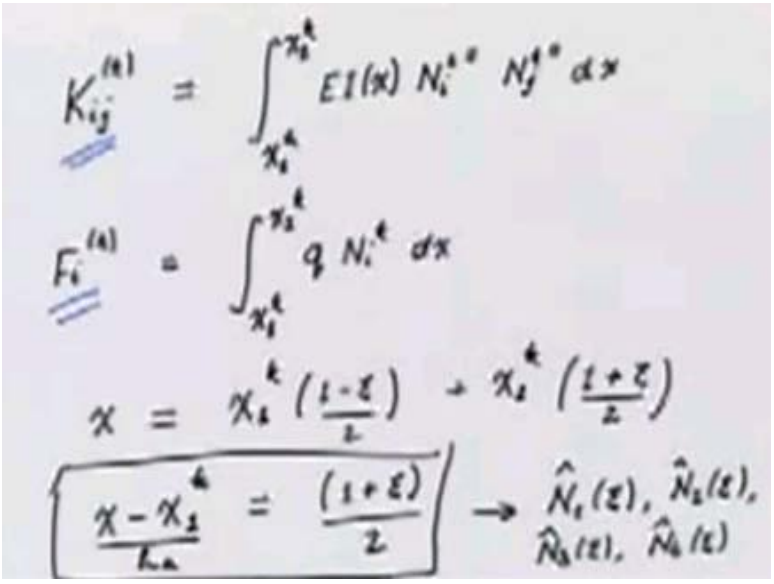
I substitute this thing in the expressions of N_1k, N_2k, N_3k and N_4k to get my so-called N_1 hat as a function of psi; N_2 hat as a function of psi; N_3 hat as a function of psi and N_4 hat as a function of psi.

(Refer Slide Time: 28:45)



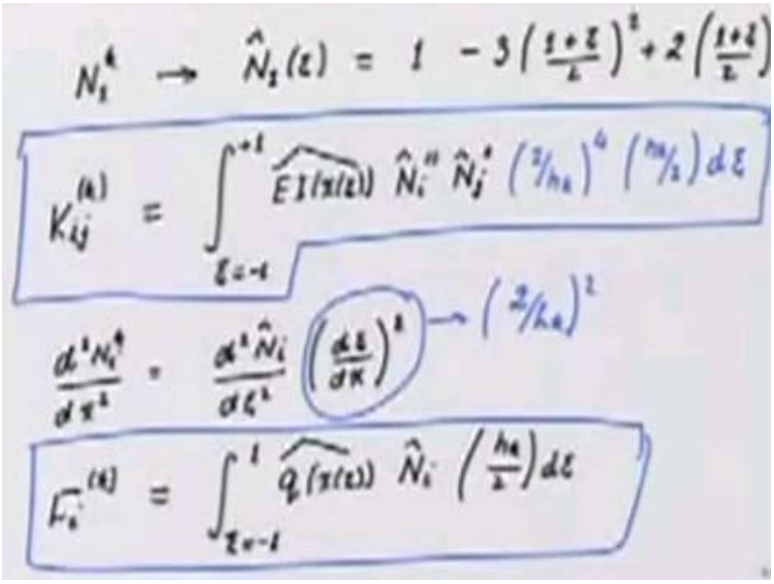
Let me write as an example what will N_1 of k become? N_1 of will mapped to N_1 hat of psi this is equal to 1 minus 1 plus psi by 2 whole square plus 2 1 plus psi by 2 whole cube. With this simple substitution, I am now able to get the definition of these shape functions at the master element level.

(Refer Slide Time: 29:24)



If I have the definition of the shape functions at the master element level, then I can convert these integrals to integrals about the master element level.

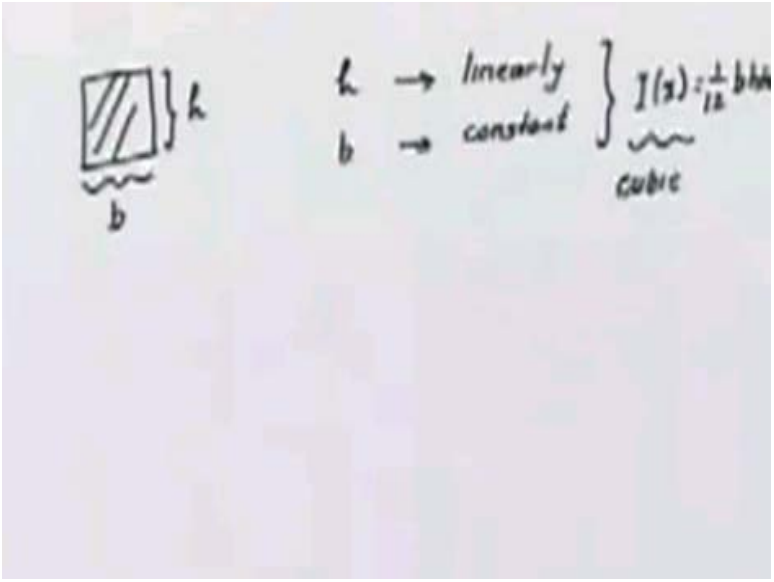
(Refer Slide Time: 29:30)



How will I convert? So what I will get is k_{ij} over element k now by doing the conversion minus 1 to plus 1 EI hat now it has become a function of psi N_i hat double prime N_j hat double prime. N_i hat because d two N_{ik} dx squared is equal to d two $2N_i$ hat d psi squared into d psi by dx whole square. This d psi by dx whole square is nothing but as we have defined the Jacobean earlier 2 by h_k the whole squared. I will have here 2 by h_k whole to the power 4 because for each one of them now the Jacobean. This is what my k_{ij} is going to look like at the element level.

Similarly, F_i for the element k will look like this: minus 1 to plus 1 q becomes now a function of psi. So I make it q hat into N_i hat into h_k by 2d psi. In the element load vector N_3 becomes this. Let use see what is the integration rule we have to use? So we are going to use the [ ] quadrature that we have defined. The question is what is the order of the integral - maximum order?

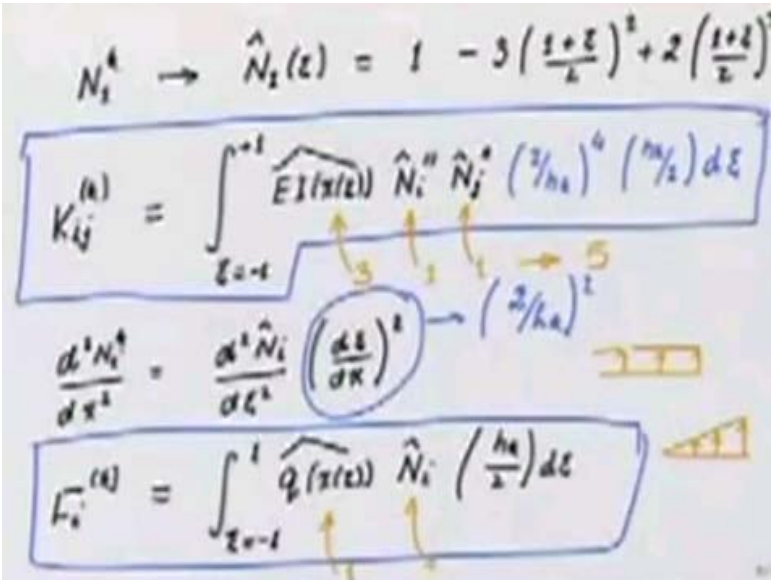
(Refer Slide Time: 32:08)



Let us assume at least for the problems that we are trying to solve, lets say that we have this beam with a rectangular cross section. For this cross section this is the height h, this is the width b. We say that h various linearly and b is constant; that is, the height varies

linearly, that is, I have tapering in the height, while the depth remains the same. In this case i as a function of x is equal to one-twelfth bhh cube. So i varies is a cubic, because h is linear, so i is one-twelfth of bh cube; its becomes a cubic.

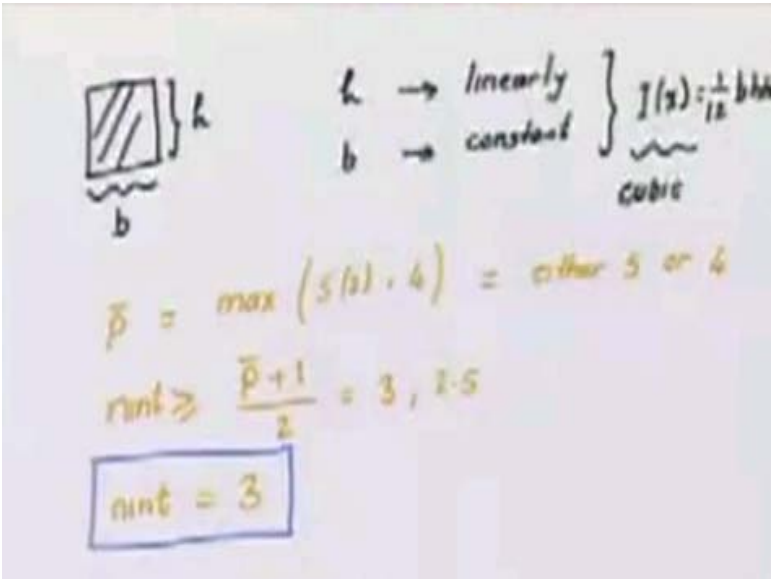
(Refer Slide Time: 33:10)



If i is a cubic, if I go back to my integral here, so EI is a cubic. So if I write it here EI is a cubic, it is of order three. The second derivative of Ni since, Ni was cubic, second derivative linear, second derivative of N_j is linear. The order of the integrant is 5, if I have taken the beam to be a tapered beam; if it is a beam of constant cross section that is h and b are both constant, then this part will become 0. So this integrant is essentially of order 2.

Similarly, if I look at F_i, let us say that I am interested at most in a loading, which is linear; that is, I will have the constant loading and I have the step loading, triangular profile. The loading is linear, if the loading is linear, the q is linear N_i is cubic. The integrand is of order 4. Let me write here within bracket incase my EI is a constant.

(Refer Slide Time: 34:36)



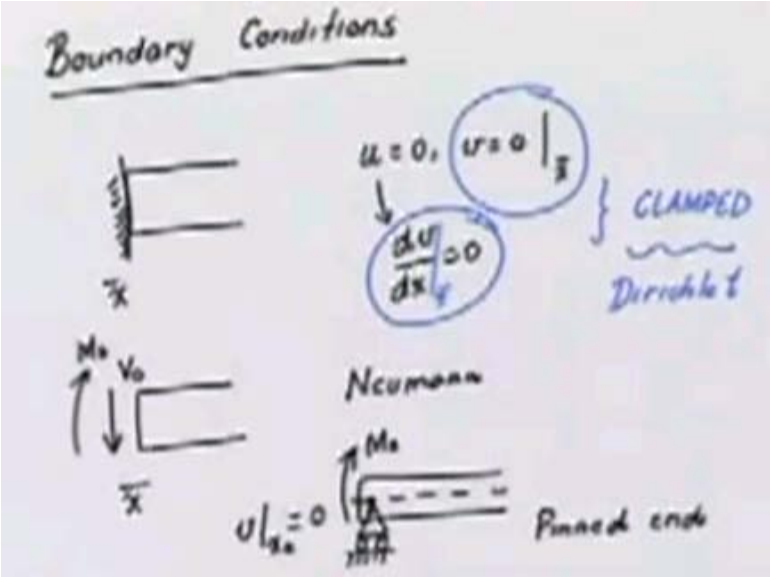
My integrand E bar, if you remember, will be equal to max of 2 or 4. This is equal to either 5 or 4; with the understanding that this 5 when I have a tapered beam and it is 4 when I have a beam with constant cross section.

In this case, if the integrand is of this order, then we know that the integration rule, the number of points, so I have to have nint is greater than equal to p bar plus 1 by 2. In this case, it is equal to if I take 5 then 3 and if I take 4 it will become 2.5. So nint is equal to 3 will do the job. If we take the 3-point integration rule, we will get exact integrals as far as the stiffness part is concerned and as far as the load part is concerned. Remember that these are procedures which are now very similar to what we have done. If I want to incorporate this in my finite element program which I have already written, I do not have to change much; all I say the p processor you see that the mesh will be the same; the degree of freedom numbering how will I do? I will go and simply... If it is the bar problem I will do the degree of freedom numbering according to the bar with the order of approximation we can choose; in the case of the beam we are only going to look at the Hermite cubic we could raise the order of approximation, but let us say it is only Hermite

cubic; so we choose for the beam the Hermite cubic and then we can give a degree of freedom numbering as I have just shown, the ieldof can be created, no problem.

After I have done all those things, then I choose the integration points; I have the integration points and there again at these integration points I can load the values of the shape functions, the derivatives of the shape functions and now in this case the second derivative of the shape functions also we are going to store. Go to the element calculation, do the element calculation, do the assembly and continue from there.

(Refer Slide Time: 37:27)



This is how we are going to do the whole procedure. So adding this part to the one-d code should not be difficult. Now, what about the boundary conditions? So, the boundary conditions, let us take some basic types of boundary conditions, I can have at each of the end, the case which corresponds to a clamped end.

Here I am saying u is equal to 0 and v is equal to 0 at the point x bar. Now u by its representation it means that through the height it is 0, which means that this will give me dv dx is equal to 0, because u_0 was 0 for the problem the way we have done it. So u

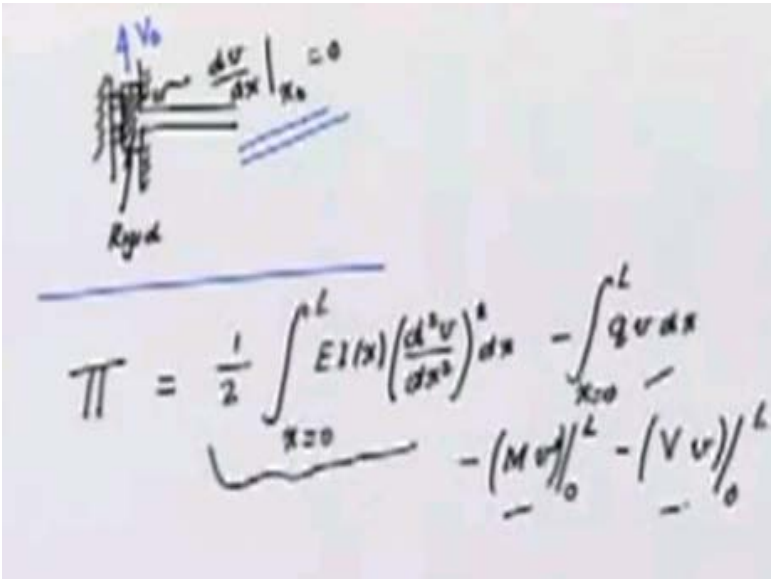
becomes equal to minus y dv dx; now for all y this has to be 0; so dv dx is 0. This condition where I am specifying v is equal to 0 and dv dx equal to 0 at the point x bar - this is called clamped end. We have talked about the clamped end and this is nothing but our Dirichle's boundary condition, that is both u and v are fixed. We can have a mixture of things also, let us see.

We can also have a second type of boundary condition at this point N where I am specifying the bending moment and the shear force. That is the bending moment and the shear forces are given. So, this is our Neumann end. Now the bending moment and the shear force could be 0 which corresponds to a free end. This is a Neumann boundary condition.

I could have another situation where I am fixing the middle line of the bar, have to be at this point. What I am saying here is v at this point is equal to 0. If v is 0, then I am actually allowed to give a moment if I want - a rotation. I can apply moment M_0 or I leave it free. This is called pinned end.

Remember what we had said is the displacement and the shear force occur together in pairs and the bending moment and the slope occur in pairs. If the displacement is specified certainly I cannot specify the shear force, but here I have done nothing about this slope. So, I can apply a bending moment there. I have to apply a bending moment there; the slope could be anything.

(Refer Slide Time: 40:50)



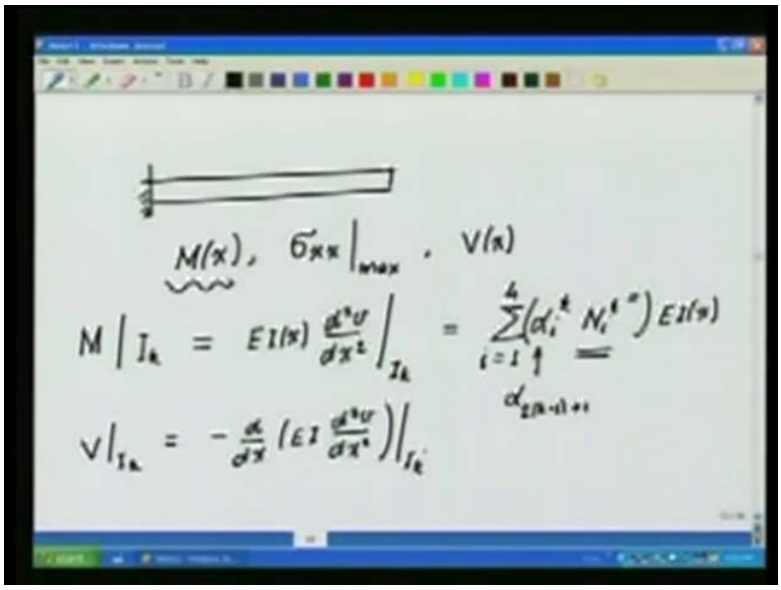
In general, see in most of the examples that you see, this bending moment is said to be 0. Similarly, I could have another special boundary condition, where I have x bar moving in a guided slot and this is our rigid support; the rigid support is moving in a guided slot. So what happens because of this I have a dv dx at x_0 equal to 0, it is not allowed to bend, but I could certainly at the same time, specify a shear force v. So we can have various types of boundary conditions; one can take many more, but these are quite sufficient, at least for most of the problems that we are looking at. These boundary conditions can also be incorporated exactly the way we did it for second order for the bar problem, this can be incorporated for the beam problem where ibc type at the two ends. Now, this ibc type can be given various numbers: 1 mean clamped; 2 means bent; 3 means free; 4 means rigid; rigid free; rigid on a roller. So we can give these flags at each end and according to the flag, we specify what is the data which has to be input. Accordingly, because dv dx, for example here is fixed, so I have to go and constraint the dv dx to be equal to 0 in my final system. So once I have obtained element calculations, done the assembly, then I now appropriately add the boundary conditions; that is add either the boundary forces and moments or constraint the boundary displacements on the slopes to the given values.

Again, the same procedure that we have followed. Once we have this, then now we are in a position to solve the problem. We can solve this problem in terms of how many unknowns, if it has any LEM number of elements, then I have ne LEM plus one node twice into ne LEM plus one is the nndofs, the total number of degrees of freedom that we have; that is all. We solve for that many degrees of freedom, get the solution and we are done.

Another thing that you should be able to do now, we have done the principle of virtual function. We could also derive everything that we have done from the total potential energy, where the strain energy is given by, as we you know from a basic mechanics, EI of x minus integral x is equal to 0 to L q v dx minus M v prime v prime at 0 to L minus v into v as defined.

So if I define the total potential energy like this, which is what we have the strain energy due to the bending, work done by the distributed force, work done by the N moment work, done by the end shear forces, then by taking the first variation of this we should get exactly the weak formulation that we have obtained. Now, the question is that this was just a remark - we could derive it anyway we wish - that how do we post process the solution? What is the information that we need?

(Refer Slide Time: 44:53)

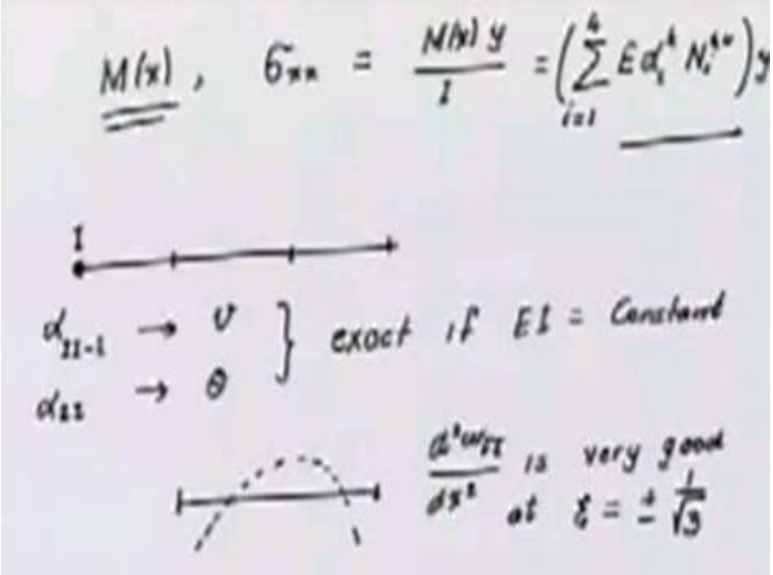


Out of this, if I have this beam, I will be interested in the bending moment distribution, I will be interested in the actual stress $sigma_{xx}$ distribution, I will be interested in $sigma_{xx}$ $_{max}$ at that point, I may be interested in the shear force at that point; these are the quantities that I may be interested in, to get from the finite element solution. How will I get to M of x? So in an element I_k, I could do it globally or I can do it in an element, this M of x is equal to, M in the element I_k is equal to EI x into d two v dx squared in the element I_k, but this if I can write it: i is equal to 1 to 4, $alpha_i$ k N_ik double prime whole thing into E I of x.

So I know what $alpha_i$ k corresponds to from the global to local enumeration. So I find the value; this is alpha 2k minus 1 plus i, I find the value of this degree of freedom multiply it with the second derivative of the shape function that we have obtained the value of the material property – the flexural rigidity at the given point x. This way put it in, I can find the value of the bending moment at any point in the element. So by interpolation we can find. Similarly, for the shear force; shear force, in this case, in this element if you see it will be this (Refer Slide Time: 47:06 min). Let us assume, for the time being that EI is constant, then this essentially means this is nothing but the third

derivate of v, which in the element is a constant, because we have taken a cubic. So VI for uniform flexibility is a constant. So post-processing is not a problem if EI is not a constant, then I can go and put it again in the formula that we have written. So this way we can find bending moment and shear force at every point.

(Refer Slide Time: 47:44)

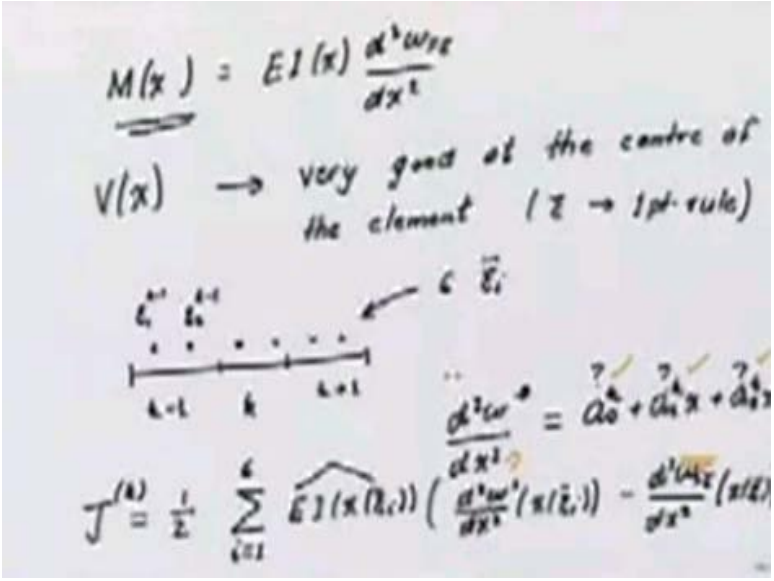


Now given a bending moment at a point x, finding $sigma_{xx}$ is not a problem. It is equal to M_x at this point y by I; so this will be equal to Sigma i is equal to 1 to 4 E $alpha_i$ k N_ik double prime into y. So where is it maximum – at the top or the bottom? So I go to plus minus h by 2 or whatever is my maximum value of the y, because, it may not be that the neutral axis is at the mid point; neutral axis could also be at shifted depending on what is the cross sectional parameter. So I find the value of this at the top and the bottom and see which one is the higher, and that is how I will find $sigma_{x\ max}$, if I need to.

Given a mesh, the values of alpha for a node $alpha_{2i\ minus\ 1}$ and $alpha_{2i}$, for a node I, these values correspond to v at this point; corresponds to theta at this point; values are exact if EI is equal to constant. So for constant flexible rigidity the nodal values of the displacements, that is the alphas that we have obtained, they correspond to the transverse

displacement and the slope at this point and they turn out to be exact, provided that EI is constant. If EI is not constant, but if it is smooth then the nodal values are very good, but are not exact. Similarly, I can talk of now, how the bending moment behaves. If I have this cubic approximation, I will find in an element the bending moment, if I look at the error in the second derivate, it will be something like this. So what I will get is d two w_{FE} dx squared is very good at actually the points corresponding to the two-point integration rule; that is at psi, if I go to the master element plus minus 1 by root 2. So these are the points where the second derivative of the w are going to be very good, similarly, because, the second derivative of the w here is a linear.

(Refer Slide Time: 51:00)

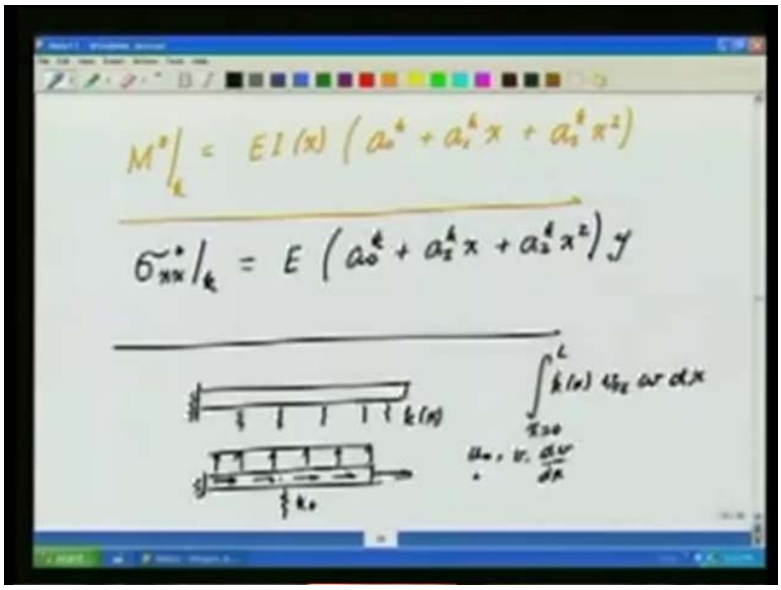


So if I say that, then M of x is equal to EI x d two w_{FE} by dx squared is very good at two points in the element and these two points correspond to in the master element psi equal to plus minus 1 by root 3. Similarly, if I talk of v of x, this will be very good at the centre of the element. Or we can say it corresponds to the point psi corresponding to the one-point integration rule. So, essentially if I really want to pick up the good values of M of x and d of x, I can pick them up at these points.

Now there is another question imagine that I would now like to get a good value of M of x at every point in the element (Refer Slide Time: 52:22 min). So again what would I do? I have essentially, the neighbors k minus 1 and k plus 1; for the neighbors, I know that for each of the elements, there are two points where the second point of w is good. So I will take these points, let us say psi_1 of k minus 1, this is psi_2 of k minus 1 and so on, psi_1 of k, psi_2 of k… I will number them in a sequence; there are 6 such points. So I will call the six points psi_i bar, in this set of three elements. If I look at this del two w dx squared is a linear. I will now say that I want to get an approximation which is quadratic. Then, I will define how I will find these co-efficients? I hope that what I am getting is better than what the finite element solution gave me.

So I am going to define a J which is sum over i equal to 1 to 6; this is for the corresponding to the element k EI at the point x corresponding to the point psi_i bar into d two w star dx squared at the point x corresponding to psi_i bar minus d two w_{FE} dx squared at the point corresponding to psi_i bar whole squared. So I defined this discrete sum over the six points where the six points are the two-point rule in the elements k, k minus 1 and k plus 1. So this I will define for the element k. Now, I want to minimize this. So I want the co-efficients a_0k, a_1k, a_2k such that this is minimized. I get the set of equations knowing what is the w_{FE}; this is known. I can find d two w star dx squared from this fit, I can find these coefficients a_0k, a_1k, a_2k just like we had done in the earlier case.

(Refer Slide Time: 56:00)



Once I have obtained these co- efficient, then I can say that my M star in the element k is equal to EIx into a_0k plus a_1k x plus a_2k x to the power 2. So if I go back again this exercise, this fitting has to be done for each element, where for the element I take two neighbors and from there I am fitting this quadratic polynomial using the values at these six points. Once I have done this, then for each element I can construct the solution and I can construct the M, as I have constructed the M, I can construct the sigma$_{xx}$ star at the element K is equal to E into a_0k plus a_1k x plus a_2k x squared into y. This is how we can construct the recovered value of the stress. This is essentially an extrapolation of what we had done for the second order differential equation. So this is also a patch recovery method, but used for the beam problem. You can see that this will give quite decent values of the recevored bending moment and the actual stress that we obtain.

So, now we have been able to reach a proper procedure for solving for the actual bar problem and a beam problem. Now, in general, I could also look at some extensions of this, maybe the beam is sitting on an elastic support that is on a distributed spring with this I will only change the stiffness matrix part, that is for the stiffness part I am going to add k of x w_{FE} into w dx. I could have my member supported by one spring, then at this

point I have to ensure that I have a node and at this point I simply go and take care of this by adding k_0 to the diagonal entry corresponding to the displacement at this node.

So these are certain things that now can be easily handled. I could also handle the problem I have a distributed load which is transverse and an actual force and an end load. That is you can look at the combined beam-bar problem; that is both impend loads and transverse loads are there; in this case, we will also have the actual displacement; the actual displacement will give me the second order differential equation; the transverse will give me a beam which is a fourth order differential equation. So we can combine the two and solve the problem. In this case what will happen is at every node you will have the u, 0, v and dv dx as the unknowns; so there will be a three unknowns. This is not a big problem. Once we have understood how to do these things, you can handle this. So essentially, in a one-dimensional problem this is what we are going to look at.

Next class we are going to start on the two- dimensional boundary value and we are going to elaborate about how to go about adapting what we have done without major changes to the two-dimensional problem.