Finite Element Method

Prof. C. S. Upadhyay

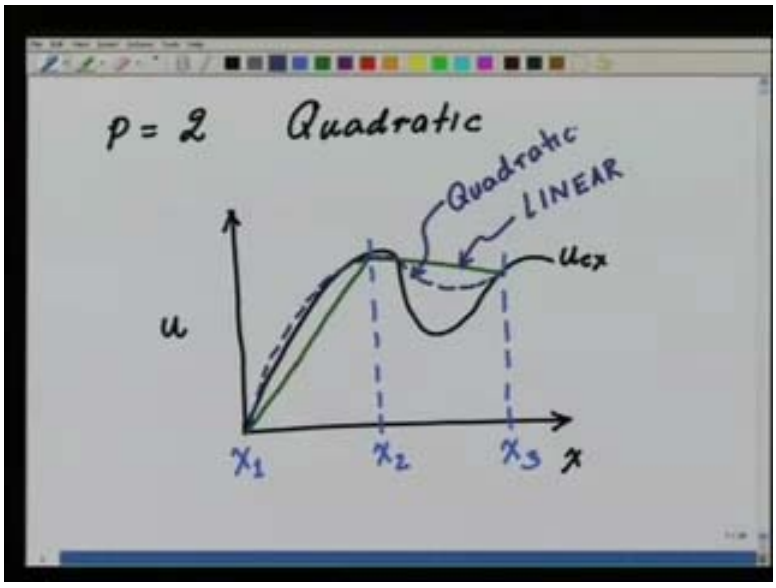
Department of Mechanical Engineering

Indian Institute of Technology, Kanpur

Module – 3 Lecture - 2

In the previous lecture, we had seen how to do a quadratic or P equal to 2 approximations. We had defined the global basis functions which was piecewise quadratic in nature with the local support and then we went ahead and in an element talked about the quadratic shape function in the element. Piecing together the shape functions, patching them up, we got the global basis function as we have seen through the drawing in the previous lecture.

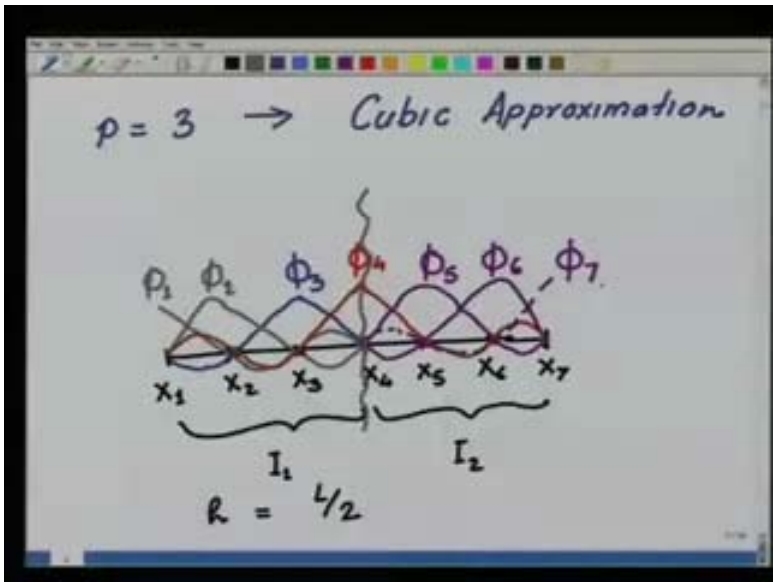
(Refer Slide Time: 00:26)



Obviously, we should ask the question, why do we need to do a quadratic approximation, when linear approximation seems to be doing a good job? Let us see this thing through a simple example; this is my domain. Next, for exact solution to the problem do something like this (Refer Slide Time: 01:25). So for this exact solution, we are going to make, for this domain let us make a two element mesh. This is my point X_1 X_2 X_3. If I did a linear approximation using the differential equation that we had defined then my solution should be as we have seen, should give us the exact solution at the nodes; we should get a figure like this. The question is for the

same mesh that is fixing X_1 X_2 and X_3 can I improve the solution? What should I do to improve the solution? Fixing the mesh. One simple thing I can do is I can take a quadratic approximation. If I take a quadratic approximation, what most possibly will happen is this. The nodal value will be exact again and then from here the quadratic will be something like this. If this was linear, possibly this is going to be my quadratic (Refer Slide Time: 03:20) and this is obvious that the solution to the approximate problem will be closer to the exact solution, if I took for the fixed mesh, a higher order approximation piecewise. If I went and get a higher than P equals to 2 approximations so that next thing that we can talk of is the cubic approximation.

(Refer Slide Time: 03:48)



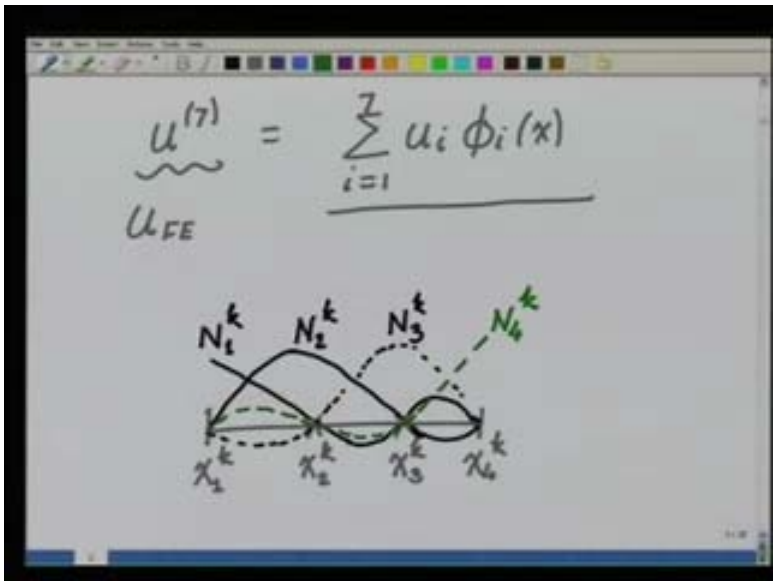
Let us do the cubic approximation or we said that the order of approximation is 3 and this is defined by the value of P. When I do the cubic approximation, let me take it through a simple two element mesh. When it is the cubic in an element that is c square is a cubic, we know that we can define the cubic using four independent functions. We need to have, because we have to have linear independence and completeness for the basis functions. We need to have in an element four such functions. Let us not bother about that now; let us draw. How will I draw? This will be point X_1; this will be X_2, X_3, X_4, X_5, X_6 and X_7. This is my element I_1. This is my element I_2 and extremities of the element have given by X_1, X_4 and X_4 and X_7. For each of the element have introduced two interior nodes, for the element 1, we have interior nodes X_2 and X_3 for the element 2, we have 2 integer nodes, X_5 and X_6 such that they are equally spaced in the

element. If we have the uniform mesh, that is the mesh of the same size everywhere, so the h will be equal to L by 2 in this case. What is the size of the element? That is the distance between X_1 and X_4 is h and between X_4 and X_7 is there. So the distance between X_1 and X_2 will be equally spaced, will be h by 3. This distance is going to be h by 3.

For this, let us through a picture define cubic functions, such that at the point 1, X_1, it has the value 1 and it has X_2 X_3 X_4 and all other points, it has the value 0. Let me just draw this function. If I want to draw this function, I can think of a function like this. This will be a phi_1. Similarly, I can draw the function which has the value 1 at point X_2 and 0 at all other points. What is the cubic functions which has value 1 at X_2, 0 at X_1 X_2 X_4. So the cubic function is, if I draw, it will look like this. This is phi_2. The function which is 1 at the point X_3, 0 at X_4, X_3, X_1 that function will be phi_3 and then finally the function which is 1 at the point X_4 and 0 at X_1 X_2 X_3.

First is I am defining the function in the first element, and then I will continue to the second element. So this function at least, it should look like, this is going to be part of phi_4. This function is also non zero in this element. So it should do the same job of vanishing at point X_5 X_6 X_7 in the second element. This is how we are going to define phi_4. If I have to define phi_5, phi_5 have to vanish at X_4 X_6 X_7. So it is going to be like this. This is phi_5. Similarly, phi_6 will be like this and phi_7 like this. So if I had more elements in the mesh, I could have drawn the basis functions. These are the global basis functions and as we have been doing till now, let us now look at restriction to an element. So what will do from the basis functions?

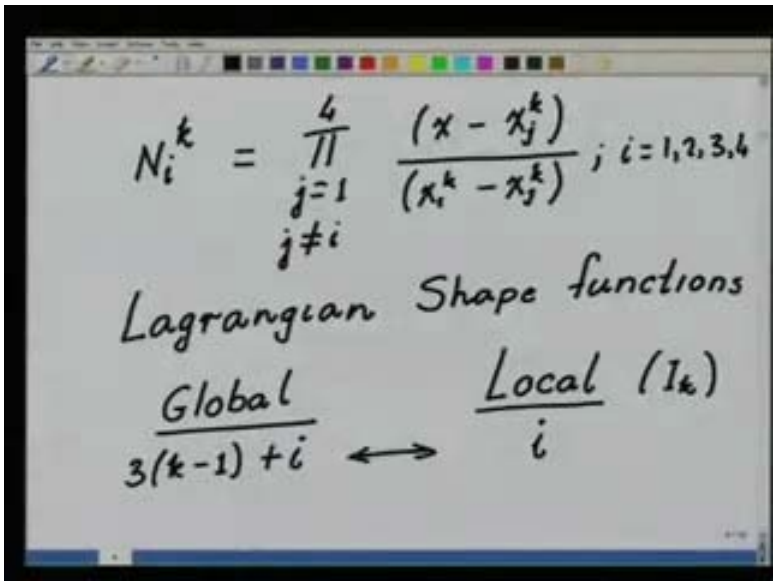
(Refer Slide Time: 09:38)



We know that u seven in this case, is equal to sigma i is equal to 1 to 7, u_i phi_i of x. I will not carry this upper subscript throughout my presentation from now onwards. I will switch on to u_{FE} where u_{FE} will stand for the particular series that we are talking about. So this is the representation of the finite element solution.

How are you going to do this thing at the element level? In the element level, we have to define now, the element level shape function. This is X_1. This is element k. So this is X_1 of element k. X_2 of element k, X_3 of element k and X_4 of element k. At the element level, what is going to be the N_1? This is N_1 of element k; this is N_2 of element k. Similarly, N_3 of element k will be, it should be 1 at the point X_3. Similarly, I will have the N_4 of element k.

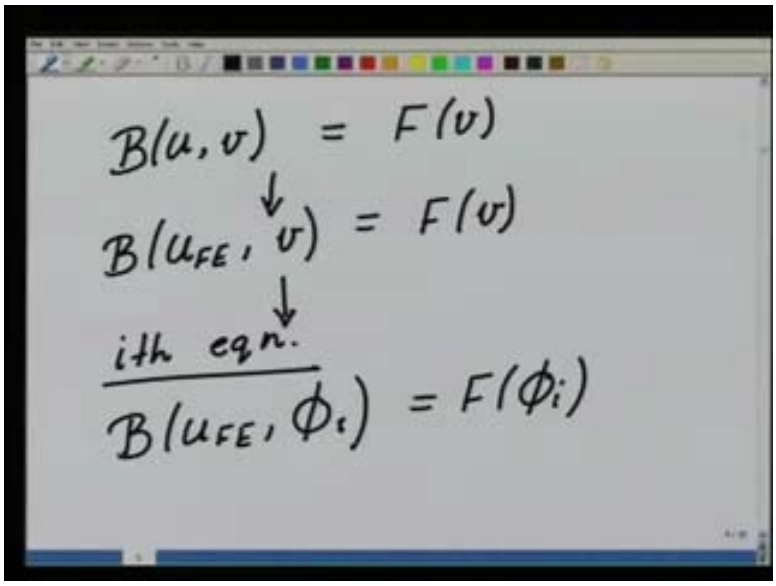
(Refer Slide Time: 12:02)



By now, it should be obvious what the functional representation of these N_i's is. Given these 4 points in the element, this N_i^k is equal to product of j going from 1 to 4, j is not equal to i, $(x\text{-}x_jk)$ by $(x_ik\text{-}x_jk)$. For example, if I am talking of N_1, so in N_1, in this product, we are going to ignore j is equal to 1. We have $(x\text{-}x_2k)$ into $(x\text{-}x_3k)$ into $(x\text{-}x_4k)$ divided by $(x_1k\text{-}x_2k)$ into $(x_1k\text{-}x_3k)$ into $(x_1k\text{-}x_4k)$. This is how we are going to define the generic i^{th} shape function of an element which is of degree 3, so given this, now all the shape functions can be defined to the product rule. This is the Lagrangian definition as we have pointed out earlier. These are called Lagrangian shape function. When I give the definition of the Lagrangian shape function, then we have to remember that we have to go from the element level definition to the global level definition. So we have to have the local to global numbering, how will it be done? Here, I will have i is equal to 1, 2, 3, 4.

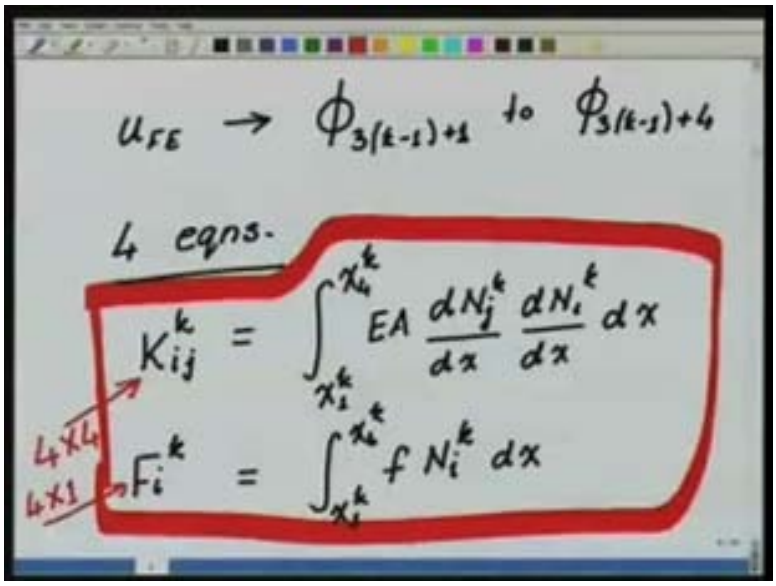
If we check with the mesh that I had drawn the recipe is very simple that the element level i for the element k, so we are talking of element I_k is equal to globally which corresponds to 3(k-1) plus i, the way we have done the numbering of these points of the node that we have placed. It gives up one to one correspondence like this, so the local i corresponds to global 3(k-1) plus i, we should check that.

(Refer Slide Time: 15:10)



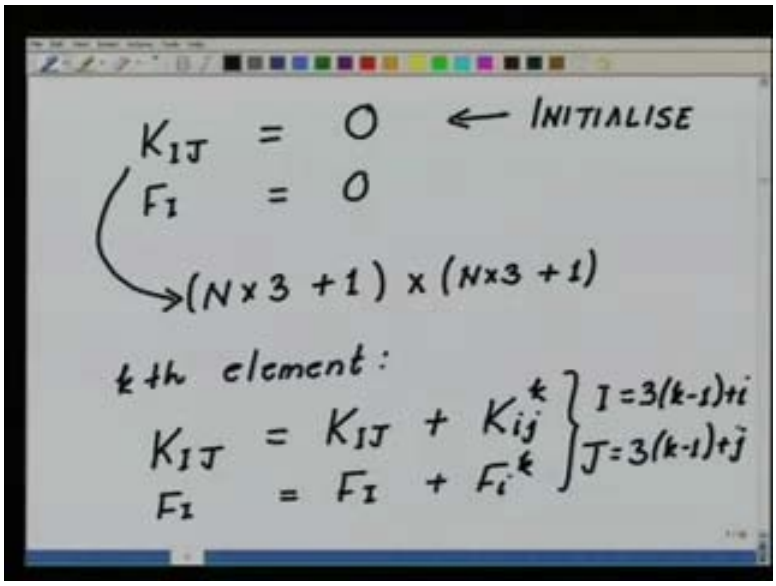
Next is what is the element stiffness and load vector contribution to the global system? For that, we are going to go back to our representation of the problem. Here, I am going to substitute the finite element solution and the ith equation which we remember comes from F phi_i. So this was what we had globally and as we have done earlier, let me go back and point out a few more things in terms of terminology. This N_1 of the element and N_4 of the element are, as we have done in the quadratic case, are called the Edge or the Corner Shape function. This N_2 and N_3, if we see, only defined in the interior of the particular element. They are 0 everywhere else while N_1 and N_4 are also non-zero in the neighboring element, that is why we call edge shape function or vertex shape function and N_2 and N_3 are called internal bubble functions and if we see that this N_2 k and N_3 k is actually the total definition of the global basis function. Let us come back here.

(Refer Slide Time: 17:08)



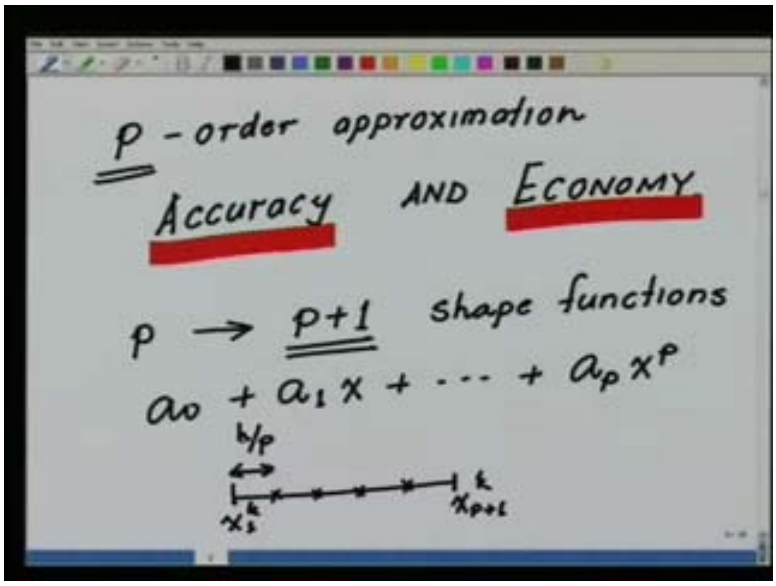
In this expression, we would like to get, this is the expression element level contribution. We know that at the element level this will get contribution from phi$_3$ into $_{k-1 \text{ plus } 1}$ to phi$_3$ into $_{k-1 \text{ plus } 4}$. When we make the element equation, we will have how many element equations? Because P is equal to 3, we have 4 shape functions in the element; we will have four equations at the element level. What are these equations? These equations will be K_{ij} of the kth element equal to integral over, I will go from $x_1$of k to x_4 of k EA dN_j of k divided by dx into dN_i of k divided by dx into dx, I am not putting the distributed spring part. This is only for the bar. If we have the spring, we will have to add the additional contribution from the spring. Similarly, the F_i for the element k is $x_1$of k to x_4 of k f N_ik dx. I am not adding the boundary part because that will be added through the boundary condition. This is what I want to get at the boundary level. These are going to be my element level equations and we see that this one (Refer Slide Time: 19:30) is 4 by 4 system matrix and this one is the vector of size 4. We have done sufficient amount of assembly to know how to assemble these things.

(Refer Slide Time: 20:04)



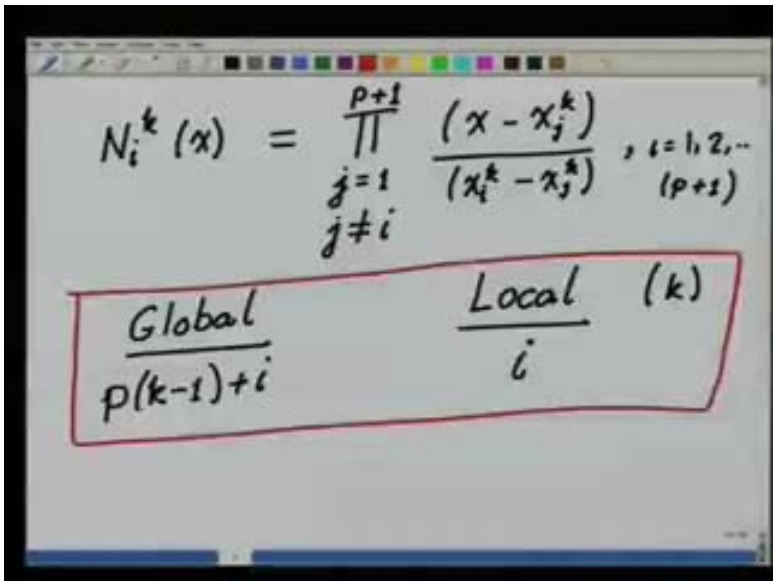
To do the assembly process for the element, we would start of by defining the global stiffness matrix, K_{IJ} such that it is given the value 0; this is initialized and similarly, F_I is equal to 0. Can you tell me what the size of this matrix is? This global stiffness matrix will have a size; it will be equal to number of elements into 3 plus 1, cross number of elements into 3 plus 1. So the matrix of size 3N plus 1, similarly F_I is a vector of dimension 3N plus 1. When we got the kth element, we will take K_{IJ} global is equal to K_{IJ} global plus K_{ij} of the kth element. Let me first do the load vector part also; F_I is equal to F_I plus F_i of the element k. Here, I is equal to 3(k-1) plus i and J is equal to 3(k-1) plus j. We know which global entry, which global row and column should might element level row and column entry go to or assemble entry and similarly, for the load vector. So this is the assembly process.

(Refer Slide Time: 22:20)



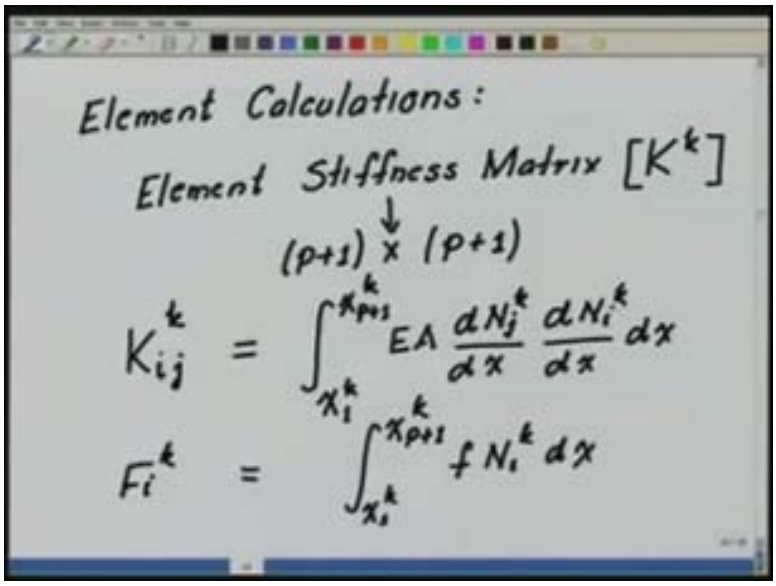
The question is can I generalize it? Can I generalize to any P order approximation? Nobody tells us that the approximation should be of order 1, 2, 3 or 4. What govern whether I use an approximation of order 1, 2, 3 or even 100 is? First of all accuracy of the solution that we decide and also the economy of the computation, so accuracy and economy, these are some of the primary consideration that we have to have. Whatever key order whatever mesh size gives the best solution is what we want and as quickly as possible. Can I generalize what I have done earlier for the P equal to 3 case, to a generic element of order P. So generic element of order P mean it should have P plus 1 independent shape functions. This comes from the completeness in the sense that, we would like to represent a polynomial which is degree P exactly, using these functions polynomial of degree P will have P plus 1 term. It will have P plus 1 independent term because monomial 1x and so on x of P are independent monomial. So when I want to represent it using the shape functions, we need at least P plus 1 shape functions and that's what we are doing. So we will have this P plus 1 shape functions to be defined. We would like to define them at the element level. So to do that at the element level, the same name, we go from x_1 of k, now we will go up to $x_{p \text{ plus } 1}$of k. This is the whole element. It is of type h. This element of size h is broken into P equal part, by putting these interior points. Interior points which are located at distance h by P from the previous point. This way I can construct the intermediate points. We have to define this functions which are 1 at one of the points and 0 at all other points.

(Refer Slide Time: 25:43)



So can we generalize the definition of the ith shape function that we have done earlier for the cubic and the quadratic to the piece one, answer is here. We will take the product of j going from 1 to P plus 1 such that j is not equal to i product of ($x-x_j$ of k) by (x_i of k -x_j of k). This is for i equal to 1, 2 up to P plus 1. This definition satisfies all the properties that is at the node x_j of k, this function has a value 1 and all other nodes, it disappears and it is polynomial of order P or degree P. This is how we can construct generic peak order shape functions. We have to do the global to local or the local to global enumeration. That again is very easy. We have the global there and local here. It should be quite obvious that this is going to be P(k-1) plus i global is equal to the local i, this for the element k. This is all that we have as far as to the global to local enumeration is concerned. Provided we put the point in the way I had shown for P equal to 3 case.

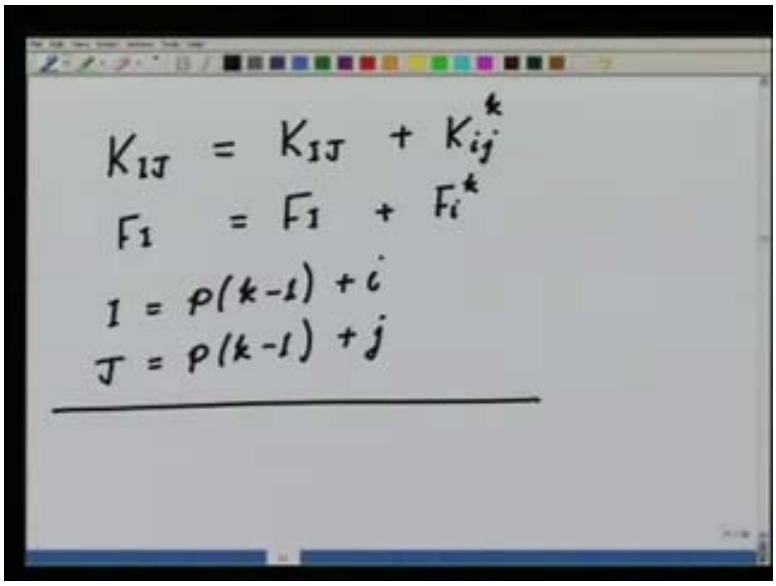
(Refer Slide Time: 27:48)



After this, once we have done this, the next job is to do the element calculation. So at the element level, element stiffness matrix will be of size P plus 1 by P plus 1 and the element load vector is of size P plus 1.Then I can talk of the ijth entry of the element stiffness matrix. This will be equal to integral from x_1 of k to x_{p+1} of k. This one should be very careful about the limit of the integration EA dN_j of k divided by dx dN_i of k divided by dx into dx. Similarly, F_i of k is equal to integral x_1 of k to x_{p+1} of k, f dN_i of k into dx.

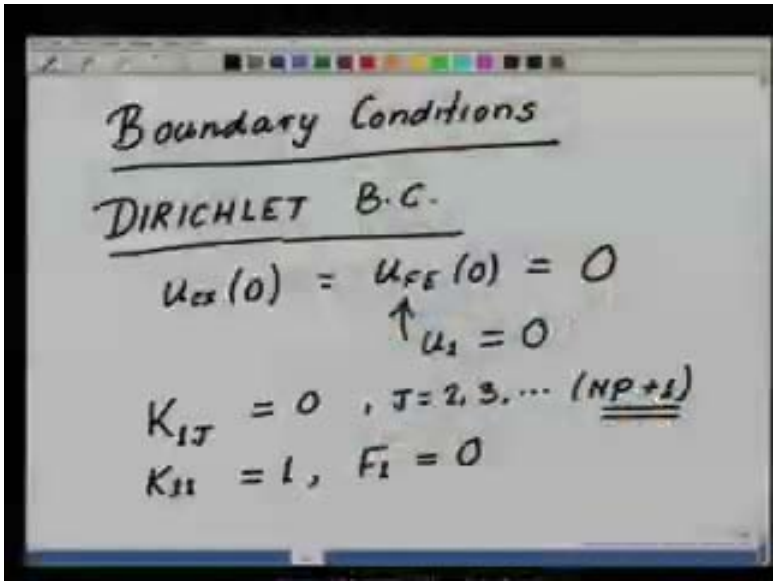
We have actually given the algorithm or a process or procedure by which we can now construct the element stiffness matrix and load vector for an element of any order P. We are going beyond P is equal to 1, P is equal to 2, P is equal to 3, it could have any P and after that obviously, the assembly procedure and so on will follow.

(Refer Slide Time: 30:09)



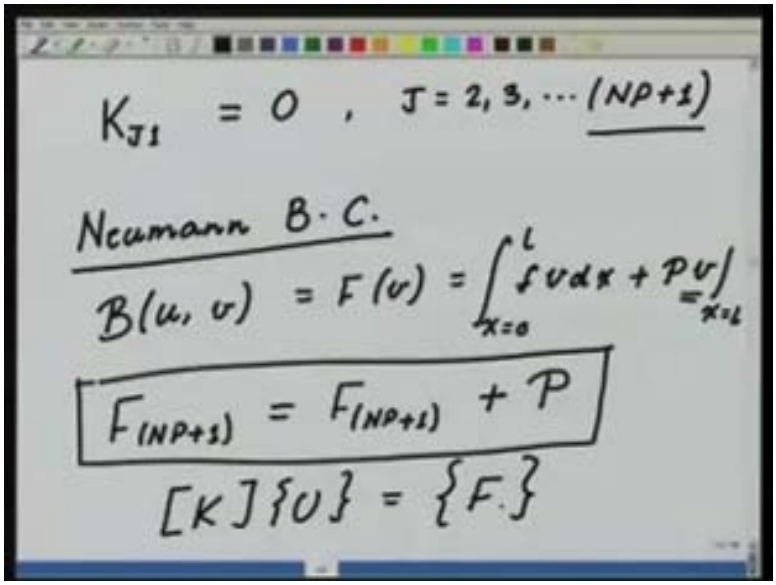
Assembly from the element will be true, where we know that I is equal to P(k-1) plus i and by j is equal to P(k-1) plus j. This is how the whole assembly will be done. After we have done assembly, then we will have to apply the boundary conditions. How do we apply the boundary conditions? The approximation of the order P, remember that we have talking of P define everywhere, everywhere we have plain P in all element.

(Refer Slide Time: 31:16)



So boundary conditions, remember how did we apply the boundary conditions for the linear approximation? That is P is equal to 1, we started of while starting the Dirichlet boundary condition. For our problem we know that u is that at x is equal to 0 is equal to u finite element, because they have to be the same, is equal to 0, so its equal to value 0, which means this needs that the coefficient u_1 has to be equal to 0. How did we go about enforcing it? The first we did was, we set that in the global stiffness matrix K for the first row corresponds to u_1. We are going to set all the entries equal to 0, for j is equal to 2, 3 up to NP plus 1, we should note that here we have written that total number of unknown in the problem as, N is the number of element, P is the order of approximation plus 1. One should check that this is what we get, total number of unknown for the degrees of freedom, as they are called for the problem or NP plus 1. So this we are going to do then we said that K_{11} is set equal to 1. Similarly, F_1 is equal to 0. After that zero or equal to the value of the applied displacement at the end x is equal to 0.

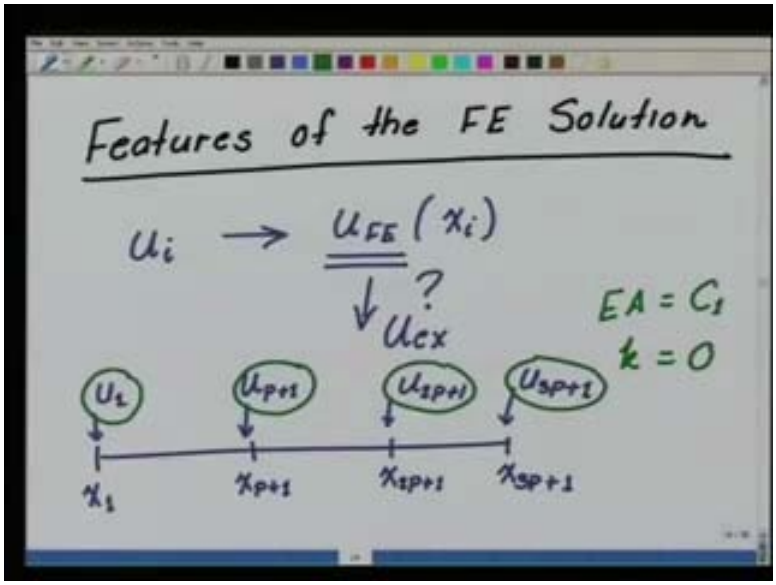
(Refer Slide Time: 33:49)



For our problem, it is zero then we had gone and then the next part that is if I had a non zero value of u_1 then I would have to correct the load vector as we have done in the case of the linear approximation, but here it is not 2, $u_1$is equal to 0. I am going to now set K_{J1} is equal to 0 for J is equal to 2, 3 up to (NP plus 1). This takes care of the Dirichlet boundary condition. The diagonal entry of the first row is set to 1; the load vector entry of the first row is set to 0. All other column entry in the first row is set to 0 and all the first column entry below the first row are also set to

zero. That is why, we are K_{J1} is equal to K_{1J} is equal to 0. This is all as far as the Dirichlet boundary condition is concerned then we have the Neumann boundary condition. In the Neumann case, if we remember the weak form, we had B(u, v) is equal to F(v) which is actually equal to integral x is equal 0 to L, fv dx plus P into v evaluated at L. Which of our global basis functions are going to be non zero at the point x equal to L? The answer is simple which is going to be the NP plus 1 global basis function. It has the value 1 at the point x is equal to L and all other basis functions are going to disappear at the point x is equal to L. In the case of the Neumann boundary conditions, we simply go to the last entry. Last load vector entry that we have which is $F_{(NP+1)}$, this is set equal to $F_{(NP+1)}$ plus to this I am going to add P. So Neumann boundary conditions are applied. We have the system of equation which is of size NP+1 by NP+1, we can go use any standard solver which can solve global problem, K U is equal to F. Solve it, find the values of this coefficient u_i.

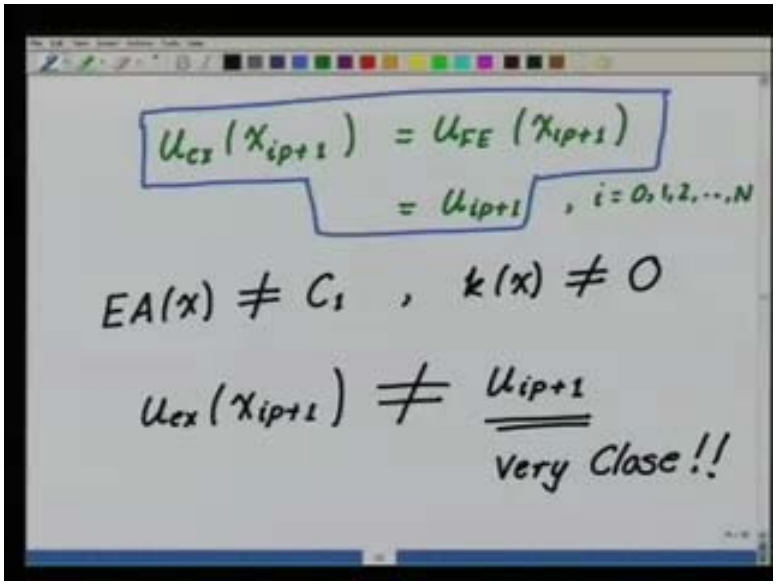
(Refer Slide Time: 36:50)



Once we have done it, let us look at some features of these solutions of the finite element solutions. What are the features? One should note that all these u_i's, the coefficient u_i in the series representation of u_{FE}, in this case, corresponds to the finite element solutions evaluated at the point x_i. These are the properties of the interpolation functions that we are used or the basis functions that we have used at the Lagrangian functions. One can always construct other families of the shape functions and basis functions for which this is not going to be true. In this case only,

the particular case, the coefficients turn out to be the finite element solutions at the nodes or at these points. We can directly look at the value of the coefficients. I will say that the solution at the points, the point x_i is equal to this coefficient. So that is the nice things to have. The question is which is equal to the finite element solution at the point, how good is it as compared to the exact solutions? As compared to the exact solution, let me make the mesh, simple one and not going to add the intermediate points, x_{p+1}, $x_{2p+1,}$ x_{3p+1}. It terms out, this will have correspondingly to u_1, u_{p+1} u_{2p+1} u_{3p+1}. It is very interesting property of the solution which we will see, when we do the programming. One can easily program, what I have such a long and see that these things are indeed to whatever I am going to say. These u_1, these values for the case EA equal to constant and k is equal to 0, in this case, that is the problem where there are no action spring attached to the member and the member has uniform cross section and the same material everywhere. In that case, there is very curious thing that happens. It turns out that these values are actually equal to the exact solution at these points.

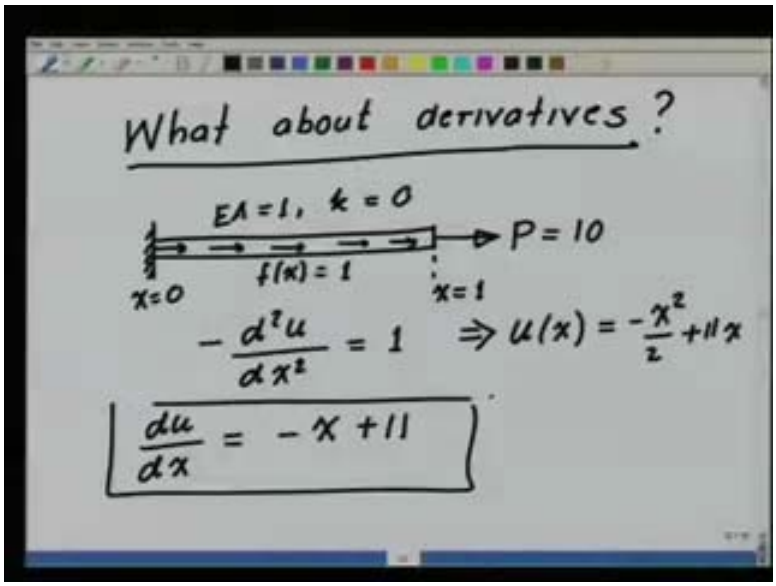
(Refer Slide Time: 40:16)



So I would say, at the point ip plus 1 exact solution is equal to u finite element at this point ip plus 1 which is equal to u_{ip+1} for i equal to 0, 1, 2, … N. This is the very curious property which only the solution to the problem described about like this. The question is remember that other points at all the interior points, the point lie in the interior of the element, this is not true. Let us look at the other cases. In case EA of x is not equal to constant and k(x) and or k(x) is not equal

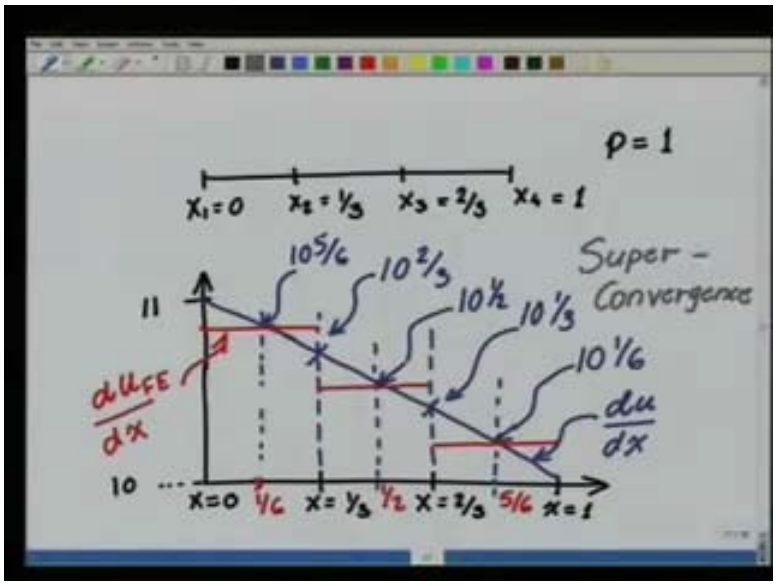
to zero that is either of this is true or both are true. In that case, this is not going to be true. That is u is that at the point ip plus 1 is not equal to u_{ip+1}. Remember that in the case of the bar with a variable cross section or the variable material or a bar with all of these and or the action string, this is not going to be true. However is everything loss? No, it is not going to be true, but this will be very close to the exact solution. This is the feature of the finite element solution that is again we need to the one dimensional problem that we are talking about. We will not able to generalize it to the 2 and 3 dimension but anyway nevertheless, this is the very nice property of the solution. Then the question is what about derivatives.

(Refer Slide Time: 43:06)



Finally, what we are interested in are the derivatives? The answer is the following that let us do the simple problem. Let us see the case where the bar is subjected to the end load P is equal to 10 are uniformly distributed load f(x) is equal to 1, material is EA is equal to 1, k is equal to 0. We take this bar and in this case, for this bar, if we know that the differential equation will be end up getting u(x) is going to be –(x square by 2) plus 11x. We can check that u at 0 is equal to 0. Here we are taking bar of length 1. So the length is equal to 1. In that case, u(x) is –(x square by 2) plus 11x such that vanishes at 0 and has derivative of value 10 at the other end and x equal to 1. So du divided by dx for this problem is equal to –x plus 11.

(Refer Slide Time: 45:25)



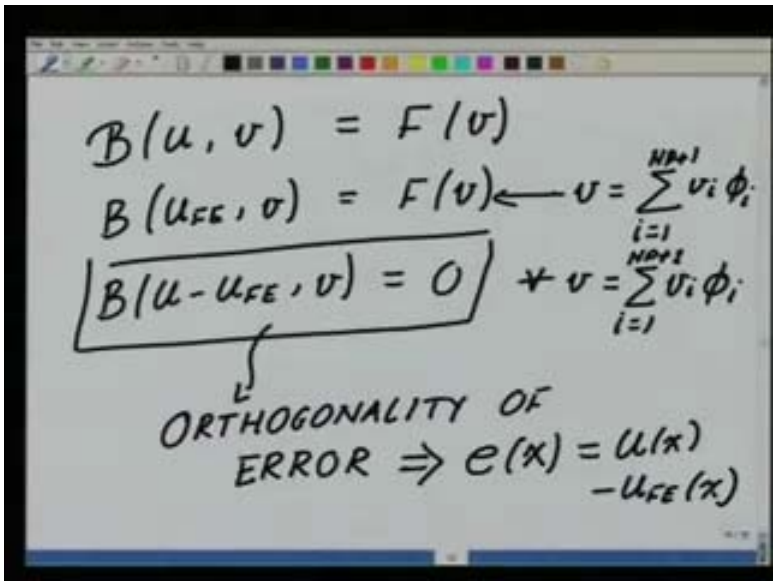
Let us try to solve the problem using the 3 element mesh, 3 element mesh with linear approximation. This is x_1 is equal to 0, x_2 is equal to 1 by 3, x_3 is equal to 2 by 3 and x_4 is equal to 1 and the P for this approximation is 1, that the approximation is piece wise linear. So the derivative of the finite element solution in each element is going to be a constant. If I plot here on this scale, the derivative of the solution then we will call this number and starting here from 10 and I am going up to 11. The exact solution is going to be a straight line which goes some value of the 11 at the point x is equal to 0 through a value of 10 at the point x is equal to 1. Let me extend this further. Let me now take the mid point. So this function, the derivative of exact solution which is du divided by dx at this point, we have the value of 11-x. So it is 10 and 2 by 3rd, at this point we have the value of 10 and 1 by 3rd, here the value of 10 and 5 by 6th. Here it will have the value of 10 and 1 by 2 and here it will have the value of 10 and 1 by 6th and what is the du_{FE} is that if I plot the derivative of the finite element solution is going to be like this. So this is du_{FE} divided by dx.

What can be P out of this? This is the picture of the two things but we can extract some more information also out of it. The derivative of the finite element solution for this particular approximation is exactly equal to the derivative of the exact solution. This is the point 1 by 6th. This is the point ½. This is the point 5 by 6th. So exactly equal to the derivative of the exact solution at the mid point of the element. This was the element 1, this is element 2 and this is

element 3. This is the very important property that we have for the derivative of the finite element solution that is for the linear approximation, for this kind of loading that we have taken it is exactly equal to the derivative of the exact solution at the mid point of each element. This property is given a name. It is called super convergence. Even though we show it with very specific example, for linear approximation, we can show is that then we have any other loading in all those cases, the mid points turns out to be a point of very good derivative information. That is the derivative of the finite element solution at the mid point of the element is very close to the exact solution, at in fact closer to the exact solution then the derivative evaluated at any other point in the element. That is why it is called the point of super conversion. Remember that in this case, the mid point now for P equal to 2, 3, 4, 5, 6 for all these keys, we will have such points existing. This will be called point of super convergence and j will be specific to the interior of the element. These become the existence of such point for super convergence becomes very important then we want to force process the derivative information.

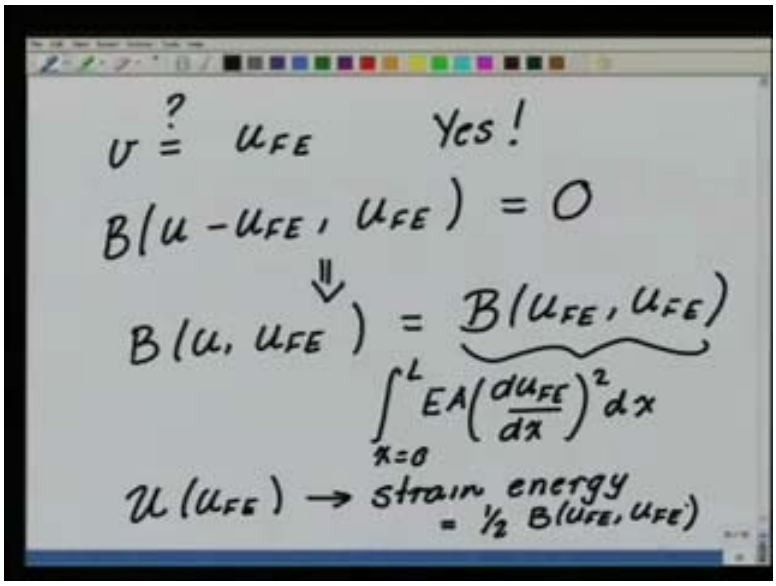
The question is why do we want to force process? Because, if we look at this here the finite element solution is doing the terrible job at the internal element into (51:30). Here, I know that derivative should be the same that is stress, because the material is the same should be same from the both direction, but finite element solution is doing something completely different is see the jump. The question becomes how we can use available information to construct at derivative information. This does the much better job than what we have obtained directly. That part goes into something called post-processing that is, we take what we have computed and out of it, we get supposedly better information as far as the quantities of interest or concerned and what will be ideal case here is? We would like to have the post processor in such a way that follows this statement. Question is in this problem, can we do it? Answer is yes. We will see later on. Not in this lecture. Let us now look at one more property of the finite element solution in this lecture. This comes from what we have defined as the bilinear form. So this is our weak formulation in terms of the bilinear and the linear form for the inverse solution.

(Refer Slide Time: 53:25)



We know that the finite element solution also satisfies this, but for a particular type of v, such that this is true for v, for all v(s) which are given as summation of v_i phi_i. For such v's, this is also going to be true. For the exact solution, it is true for all admissible v(s). So if it is true for all admissible v(s) for the exact solution, it is going to be true for these v(s) also. If I take this set of v(s) and put it here and take this difference then I will get B(u - u_{FE}, v) is equal to 0 for all v equal to summation of v_i phi_i. This property is called Orthogonality of Error. Till now we have not introduced this term? This error is given by the function e(x) which is equal to u(x) -u_{FE}(x), the terms that for all the v's u(x) -u_{FE}(x), if we take bilinear form of x which is equal to 0. How does this property really help us? We see that out of this we are end of getting of very curious thing.

(Refer Slide Time: 55:07)



For our problem, can we choose v equal to u_{FE} as the special case for our problem, the answer is yes, because the u_{FE} is also 0 at the point where we have given the Dirichlet boundary condition and everything else is satisfied. If u_{FE} was not 0, at the point where the Dirichlet boundary conditions are applied then this is not going to be true but for our problem this is going to be true. If this is true, then I can also say from what we have done earlier which means that the B(u, u_{FE}) is equal to B(u_{FE}, u_{FE}). What is this quantity? If I go to the definition of the bilinear form, this is equal to EA du_{FE} divided by dx whole square dx. This also remind us of physical quantity, at the physical quantity remain with us is the strain energy of the finite element solutions. So this strain energy as we can see is equal to 1 by 2 of B (u_{FE}, u_{FE}).

We will continue from here in the next lecture and using this definition of the strain energy of the finite element solution and similarly, the strain energy of the exact solution. Strain energy of the exact solution will be 1 by 2 of the u, u. We will talk about how does the finite element solution approximate the exact solution with respect to the strain energy. Thank you.