Finite Element Method

Prof. C. S. Upadhyay

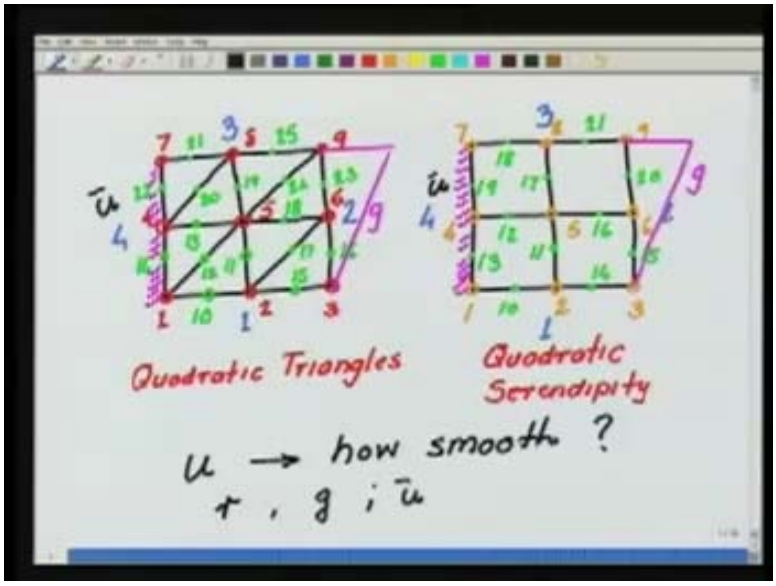
Department of Mechanical Engineering

Indian Institute of Technology, Kanpur

Module – 8 Lecture – 3

In the previous lecture we elaborated on how to use various types of elements and the families of shape functions defined over these elements in a two-dimensional finite element computation.

(Refer Slide Time: 00:37)

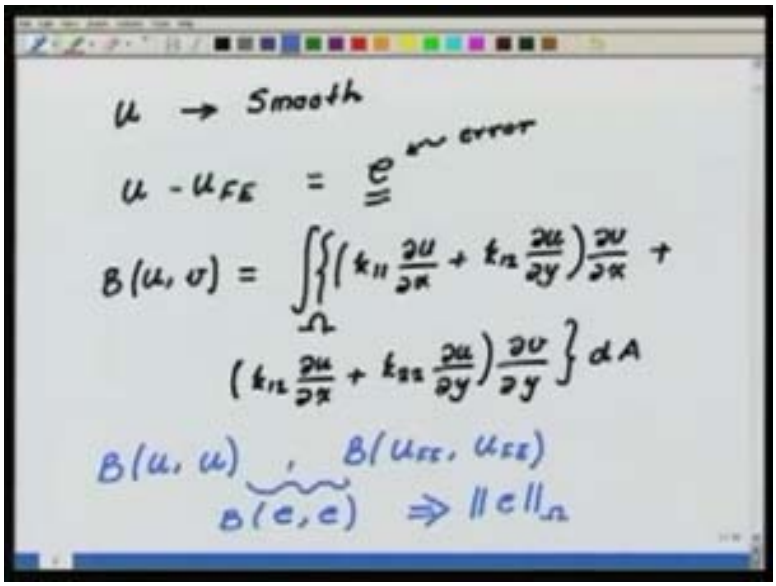


We had talked about a rectangular domain as an example, where we had chosen either to make a mesh of triangles or a mesh of quadrilaterals. In this case it would be rectangles. The convention we had followed was: this is the edge 1 boundary, boundary edge 2, boundary edge 3 and boundary edge 4. Let us say that I would like to do a quadratic approximation in either of the cases. So here I will use quadratic triangles and here I will use quadratic serendipity. If I do quadratic triangles, I will do the numbering in the particular way, 1, 2, 3, 4, 5, 6, 7, 8, and 9. I will call the mid side ones 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 and 25. As discussed last time, the corner nodes of the elements of the domain, the vertex nodes will be given the degrees of freedom first. That is my convention and it has its advantages. After that we will name the side degrees of freedom element by element.

First, I go to the first element number, the degrees of freedom corresponding to that element, then the second one and so on. Here I have twenty-five degrees of freedom. If I use the serendipity, I will have these nine corner ones and will start numbering the edges. In this element, I will have edges 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20 and 21. So, in the case of serendipity, we have a total of 21 unknowns and in the case of triangles, we have 25.

Let us talk about the boundary conditions. I could have edge 4 as u specified and may be edge 2 will have some variation of g, which is the flux. So these are the boundary conditions that I may have for the problem. Given these boundary conditions, given the domain data, do I know what the smoothness of the solution is? So the exact solution to the problem is u. The question is how smooth is it? By how smooth we mean how many x and y derivatives of this function u derivatives are going to be bounded. That is, they have finite values. It turns out that the smoothness of the solution not only depends on the body source term r and the boundary flux term g or the boundary displacement u bar, but it also depends on the domain and the way the boundary conditions switch.

(Refer Slide Time: 06:15)



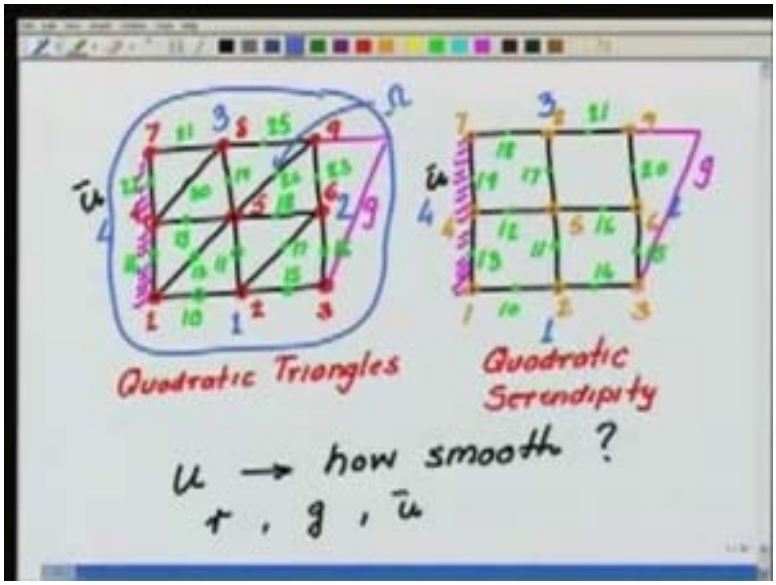
Let us say in this case the solution is smooth, that is, u is smooth. We will be interested in knowing how well is the finite element solution performing as compared to the exact solution. So we will be talking about u minus u finite element, which we are going to call error. This is error

in the current finite element solution. I have put a mesh, taken one approximation, solved for the finite element solution and now what is the error? This error is represented as e.

How are we going to measure the size of the error? The size of the error will be measured in terms of square root of some factor into the strain energy of the system. We are more familiar with strain energy kind of terms. Here we define B (u, v) from our variation formulation of the weak form, which is the integral over the domain omega of (k_{11} del u divided by del x plus k_{12} del u divided by del y) into del v divided by del x plus (k_{12} del u divided by del x plus k_{22} del u divided by del y) into del v divided by del y into dA. We are defining the bilinear form. We should remember what we had done earlier in terms of this quantity, which is nothing but the right hand side of a weak formulation.

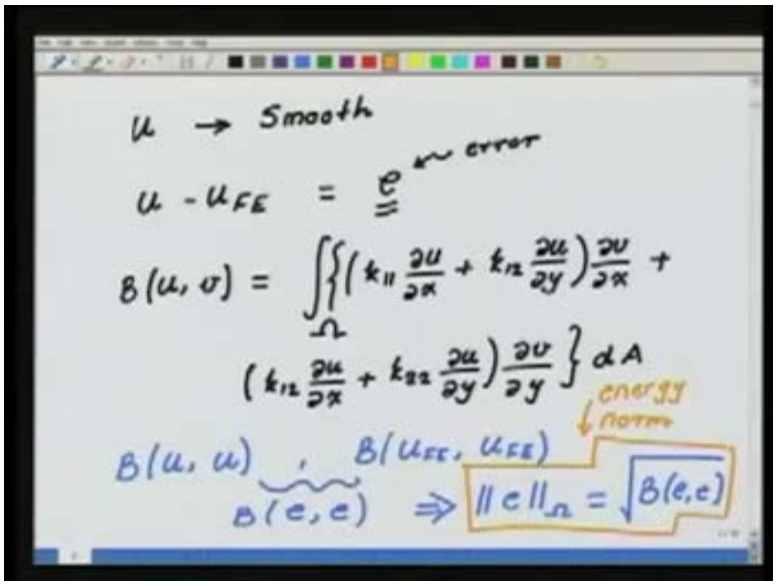
I can define what B (u, u) is. Similarly, I can define what B (u_{FE}, u_{FE}) is. We see that the integral over the omega will be understood in terms of sums of integrals over each of the elements. Similarly, by element sums, I can define B (e, e) and we had said that we would define the size of the error in terms of the energy norm of the error. Let us give it by e within two lines defined over the whole domain.

(Refer Slide Time: 09:14)



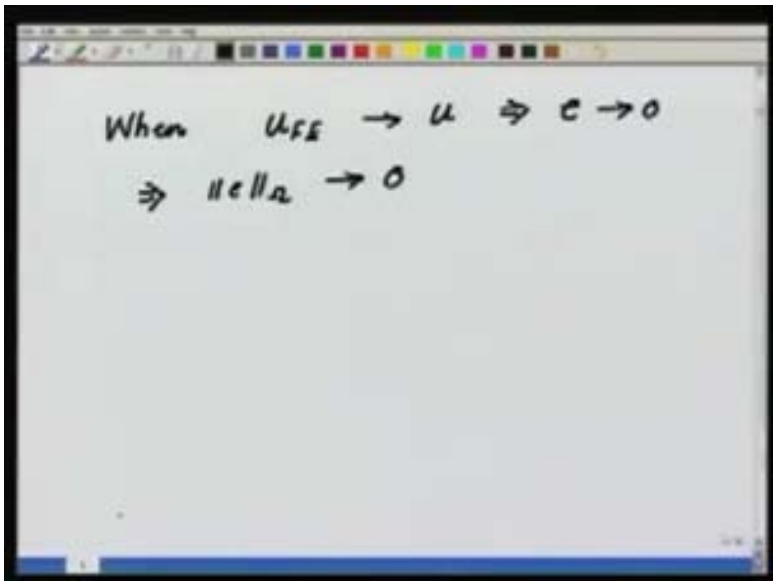
This is the domain omega. It is defined over this whole domain.

(Refer Slide Time: 09:24)



This thing is equal to nothing but square root of B (e, e). We are going to define this quantity, which is energy norm of the error. Why are we calling it energy norm? Because it is in some way related to what we know as the strain energy. Though this is not a problem of mechanics, but we define it like this. Once, we have this energy norm defined, the question is, with respect to this quantity, what do I expect?

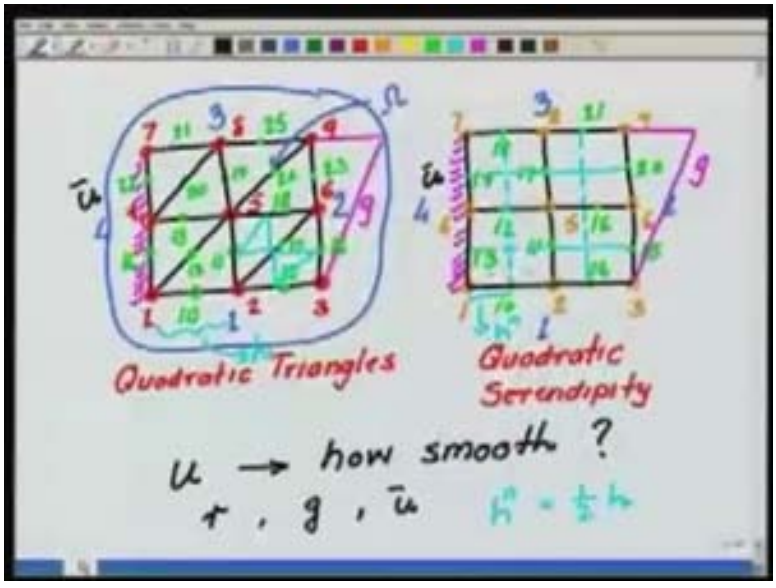
(Refer Slide Time: 10:15)



When u finite element tends to u, it implies error tends to 0, which implies the energy norm of the error should tend to 0. This is what we mean by conversions. I refine the approximation in some way and when I refine the approximation, the error should go down to 0. We measure it with respect to the energy norm.

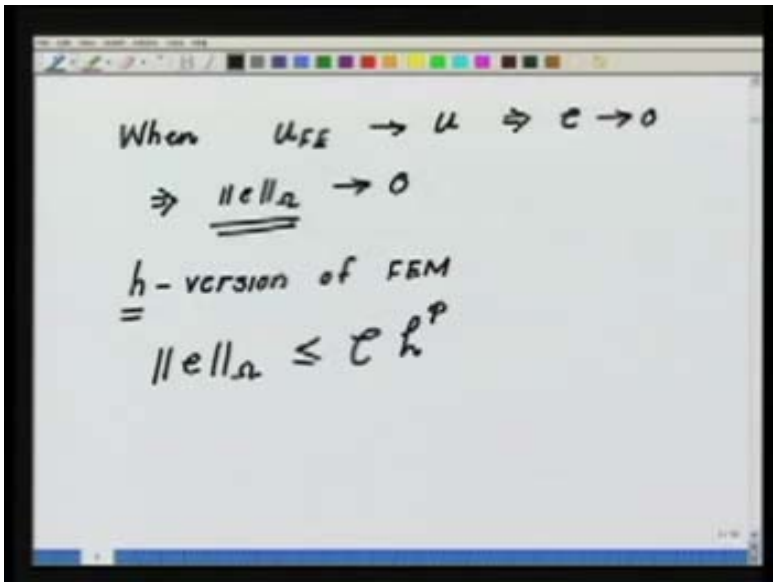
The question is how do we refine the approximation?

(Refer Slide Time: 10:55)



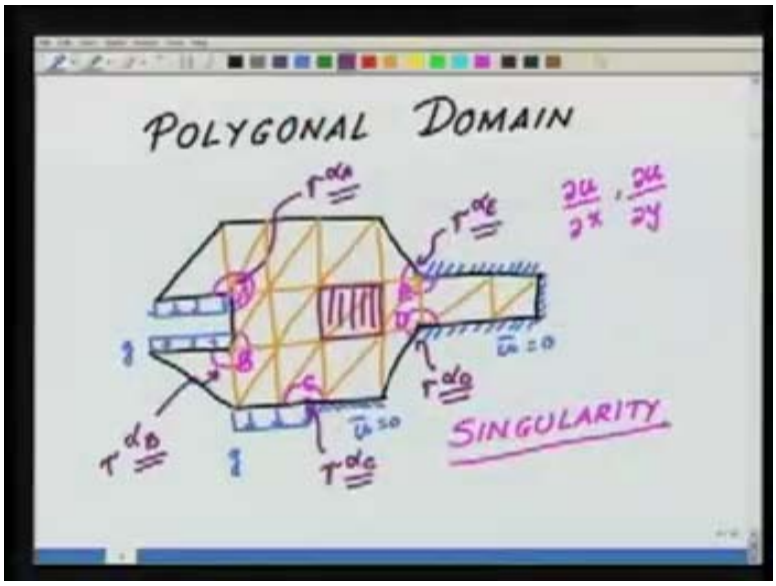
In refining the approximation, we could do two things. We could either decide to refine the mesh, that is, break the existing elements into uniform smaller pieces. This will be the new mesh, which will have four into 4, 16 elements. So out of a mesh of 4 elements here, we have 16 elements. If I have to do it for the triangles, I will break the existing triangle into four equal triangles. I can keep doing it and the mesh is getting refined. Like in the one-D case, we have to define a mesh size. We will take this horizontal length h. So for the smaller mesh, the mesh size is less than the bigger mesh, in fact, it is half of that of the bigger mesh. So h new is equal to half of h old. This is what we mean by enhancing our approximation using mesh refinement.

(Refer Slide Time: 12:10)



As discussed in the one-D case, we are going to be specifically talking about the h version of the finite element method, where improvement in the solution is achieved by keeping the P, that is, order of approximation fixed and only refining the mesh. If the mesh is getting refined, we will like to have the measure of this energy norm of the error in terms of a function of h, the mesh size, and we expect that as the size h goes to 0, this should also go to 0. When I use smooth, it turns out that this quantity goes as some constant into h to the power of P, where P is the order of approximation and h is the mesh size. This is similar to what we had obtained in the one-D case.

(Refer Slide Time: 13:28)

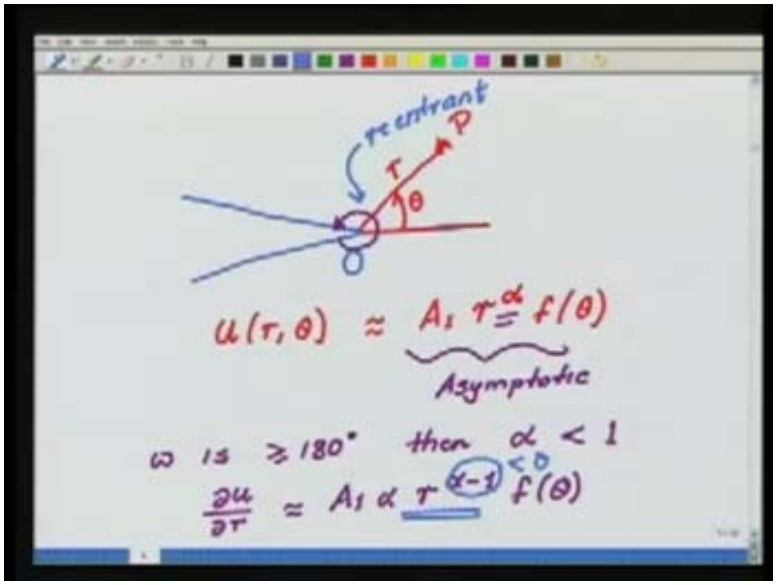


What happens when the solution is unsmooth? We have been so far only talking about polygonal domains. Let us take a polygonal domain, which is like this. I am fixing it here and here. Here I am giving a flux condition and here also I am giving a flux condition. I may also have a flux condition here in this part, while here I am fixing the solution. u bar is equal to 0. Here also u bar is equal to 0 and here g is specified and here also g is specified. How do I solve the problem over this domain? I could make a mesh of triangles or quadrilaterals. Let us make a mesh of triangles. It doesn't matter whether we have a mesh of triangles or quadrilaterals.

The mesh size is the maximum of the length of the edge of a one of the triangles, because here the sizes of each of the triangles may be different, because I have an irregular domain. In this case, I have these interesting points. If I look at it as a problem of elasticity and fix displacements here and have this kind of a corner, I expect stress concentrations to happen. Similarly, at these points ABC (C, because here the boundary condition is changing and suddenly switching), I have a flux condition and suddenly I am fixing the displacement. At these points, I would expect stress concentration to happen. Similarly, for the single variable problem also, I would expect the flux or the derivatives to become high in the vicinity of these points. In fact, they are not only going to become high but they are going to become infinite. That is, they are going to have singularities in the derivatives. That is, here, in the vicinity of these corners, I am going to have singular behavior of the solution and then the singular behavior will very quickly die away. It is localized

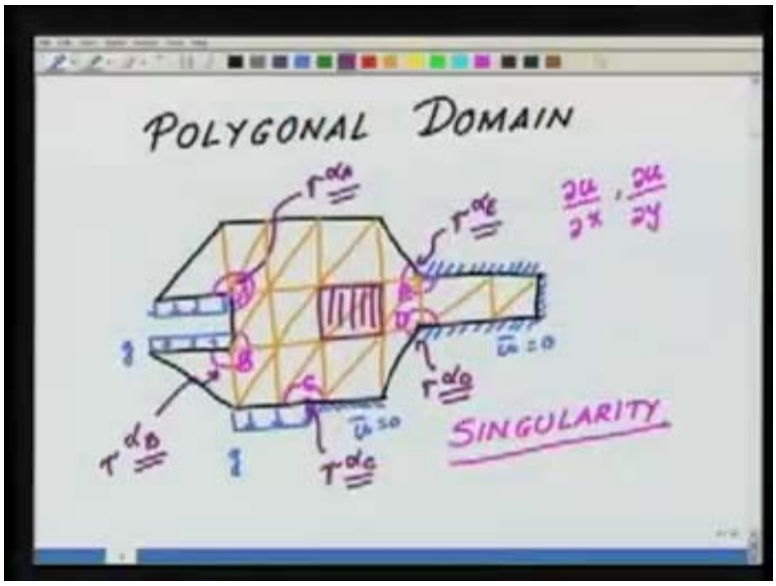
in some vicinity of these corners. Because of this, the question is can we characterize the singularity? Yes, in the vicinity of these corners, I can characterize the singularity. Let us take a generic corner.

(Refer Slide Time: 18:07)



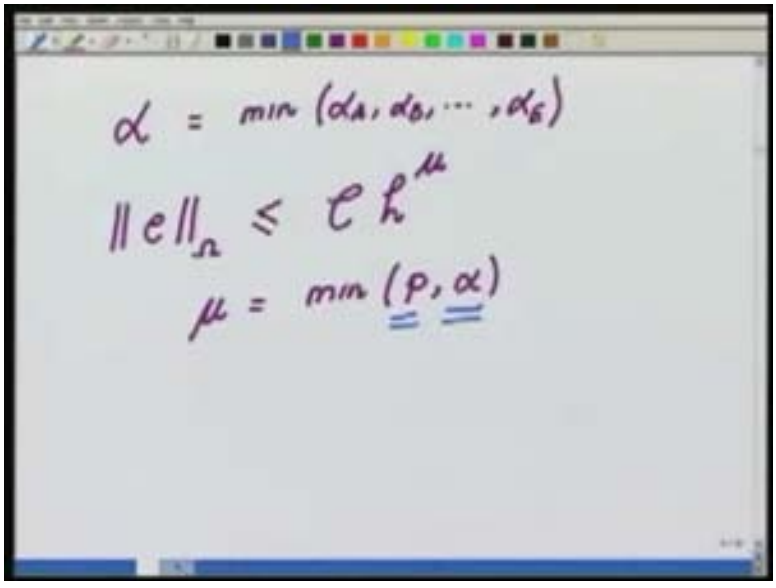
This is the generic corner O. I fix an axis with respect to this. For a given point P in the vicinity, I will measure this angle theta and the length r. That is, I am using a local polar coordinate system in two-D. The solution in the vicinity of the point O will be: u is a function of r and theta in the vicinity of point O which is equal to a constant into r to the power of alpha into a function of theta. This is how the solution in the vicinity of this corner is going to look like. This is obtained from an asymptotic expansion. In fact, this expansion is not only one term, there is an infinite series in terms of many of these alpha i. We are taking the first term that corresponds to the smallest alpha because that is the one that dominates the whole behavior. When this angle omega is greater than or equal to 180 degrees, then alpha is less than 1. If alpha is less than 1, del u divided by del r will be A_1 alpha into r to the power of alpha minus 1 into function of theta. del u divided by del r is now r to the power of alpha minus 1, where alpha minus 1 will now be less than 0. As r comes down to 0, this quantity shoots up, that is, this becomes singular at r equal to 0. This is the nature of the solution in the vicinity of these corners for which angle is greater than 180 degrees. These corners are called re-entrant corners.

(Refer Slide Time: 20:58)



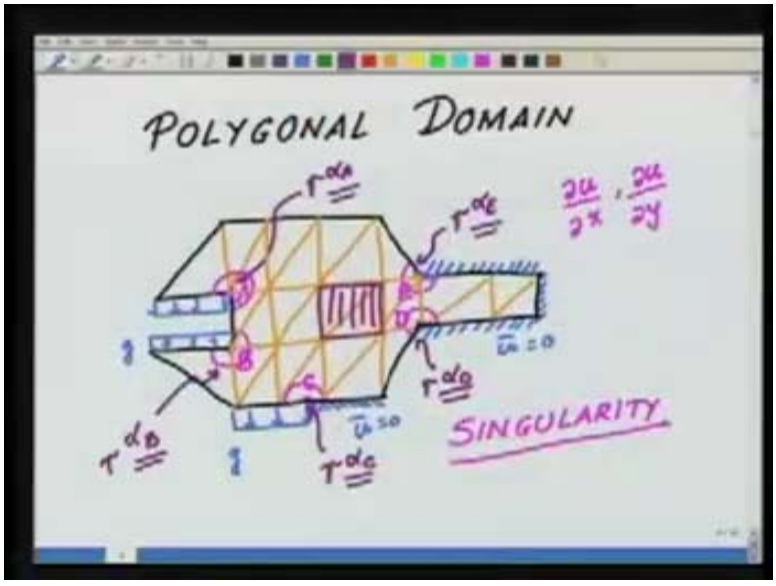
Let us go back to the previous figure. Corresponding to each of these corners we put a local polar coordinate in the vicinity of the corner and do expansion of the solution in the vicinity of the corner. I will have for each of the corners a corresponding exponent of that r to the power of alpha term. Here also we will have r to the power of alpha c. I will have all these exponents.

(Refer Slide Time: 21:53)



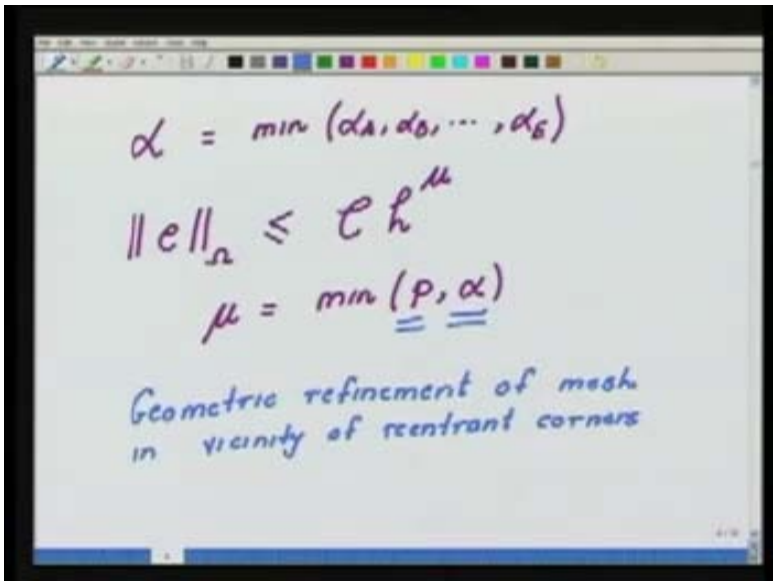
Alpha is equal to the minimum of all of these: alpha A, alpha B, up to alpha E. If I do a mesh refinement for the given domain in terms of alpha, it will turn out that the error is of the type c h to the power of mu, where mu is equal to minimum of P and alpha. The rate of convergence is now driven by the smoothness of the solution, that is, by the exponent of r to the power of alpha term in the vicinity of the re-entrant corners. The domain may have multiple re-entrant corners because we face this kind of domains all the time. These are quite common in engineering practice to have this kind of domains. So we are going to have this and then it does not matter what P we use. The rate of convergence is always going to be minimum of P and alpha. If I use higher alpha P, I will still get the rate of convergence alpha, if alpha is less than P, which is now a sub optimal rate, because raising the P has not helped us. Then the question is how do I do a better job?

(Refer Slide Time: 23:42)



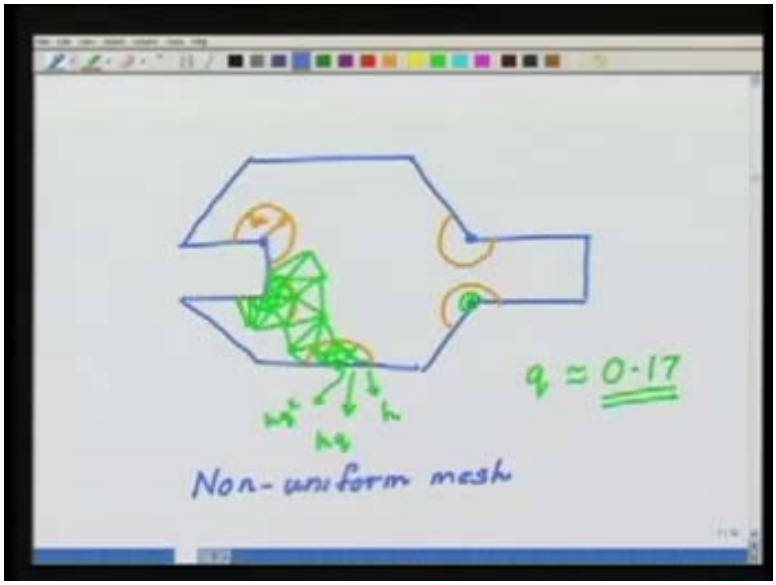
To do a better job, let us go back to our problem. It is quite obvious that here we are not able to capture the singular behavior because of these large element sizes. We should do a proper refinement of the mesh in the vicinity of this corner, so that I can capture the singular behavior of the solution. It will not be captured by just raising the P, but by a refinement of the mesh.

(Refer Slide Time: 24:12)



We are going to do a geometric refinement of the mesh in the vicinity of corners.

(Refer Slide Time: 24:50)

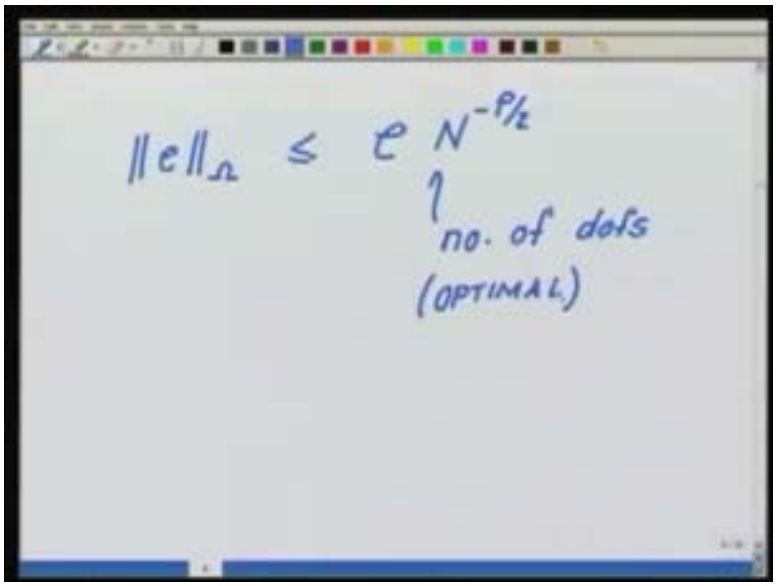


Let us go back to the domain that we have made. I will make this domain again. The problem areas are these points. We are going to take these rings in the vicinity of these points. They need not be rings. They could be approximated by polygons. Let us say this is of size initial h. Now we make smaller rings inside this ring. That is, I am going to make a smaller ring such that if this

is at distance h from the singular point of interest, then this is at a distance hq and this at a distance hq square and so on. I can have n rings, where the nth ring will be at a distance hq to the power of n, where q is generally taken to be 0.17 or 0.15.

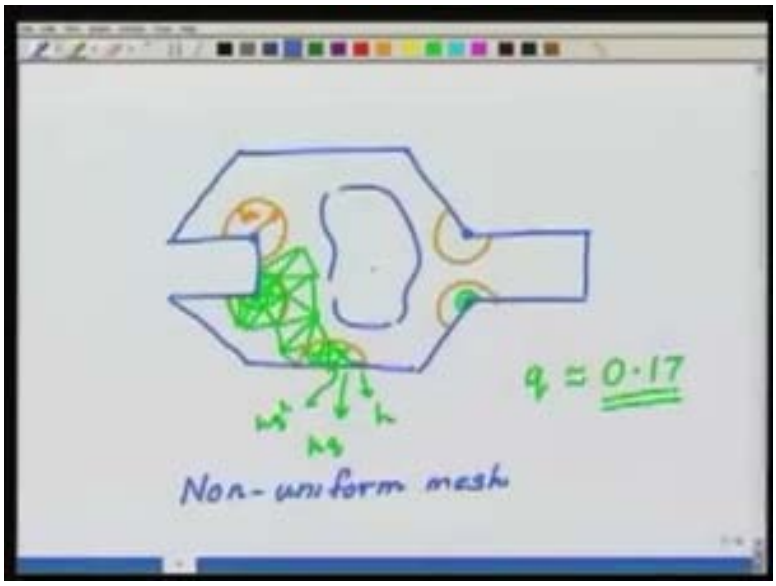
If I do this, then I can make these rings in the vicinity of these corner points, then in this rings, I can now make the elements. I can make the elements in the rings and as I said the rings need not be circles. They could be approximated by polygons. I make these triangular elements and then extend the mesh outside and so on. I will be getting the mesh like this. In the vicinity of the corners, I make these geometrically refined meshes. Once I do that I can put large elements outside these corners. I make a non-uniform mesh.

(Refer Slide Time: 28:03)



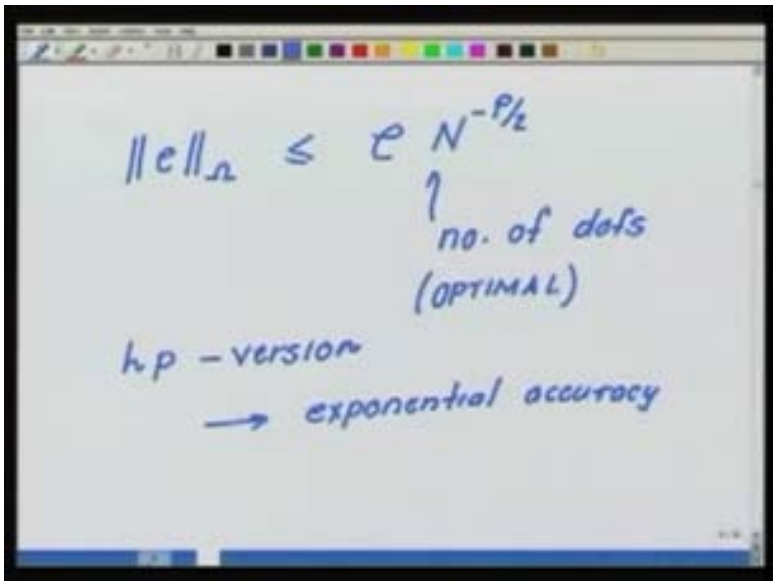
If I take sufficient number of rings, then the energy norm of the error will now go as c N to the power of minus P by 2, where N is the number of degrees of freedom. This is now optimal for the h version of the finite element method. I can use a combination of h and P. That is, I start raising the P.

(Refer Slide Time: 28:36)



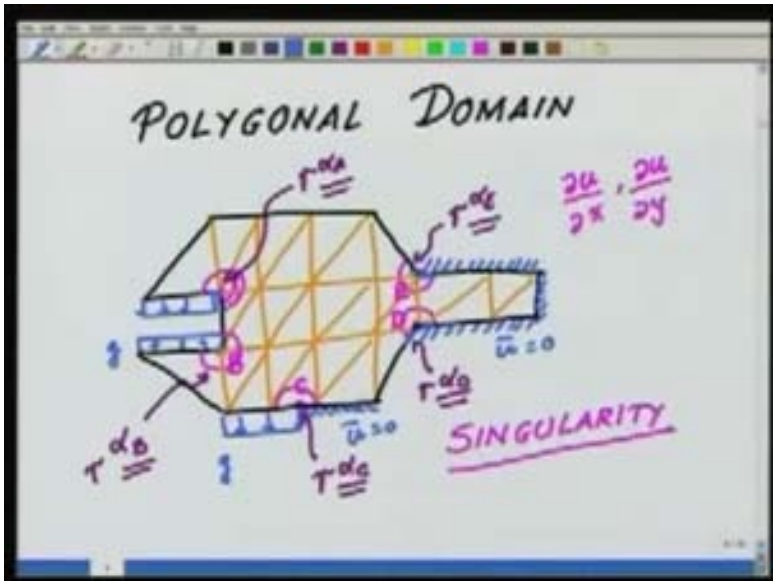
In the vicinity of the corner the solution is singular. If I take the refined meshes and a low P and I keep raising the P such that the solution away from the corner is smooth. Solution is only unsmooth in the vicinity of the corner. Using large elements with high P in that region will give very good accuracy and that is exactly what we should do. Use large elements here with high P, small elements with low P, and we can get exponential accuracy.

(Refer Slide Time: 29:07)



When we do hp version of the finite element solution then I can get exponential accuracy.

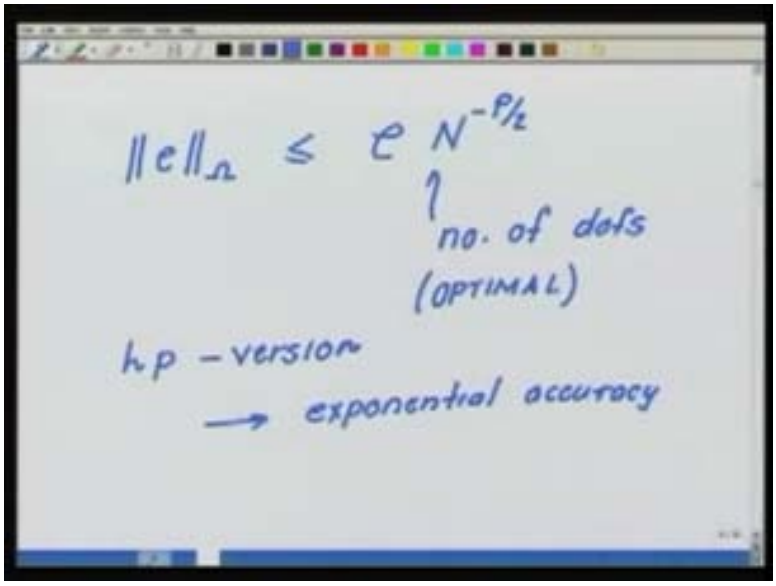
(Refer Slide Time: 29:36)



In real life structures or components that we have, we will never have these corners. We will have a fillet here. The corners will be rounded off. But when we make a mesh to represent this domain, the domain is discretized. It is represented in terms of a mesh that we put and the approximation that goes with the mapping that we use. This corner represents the fillet when we

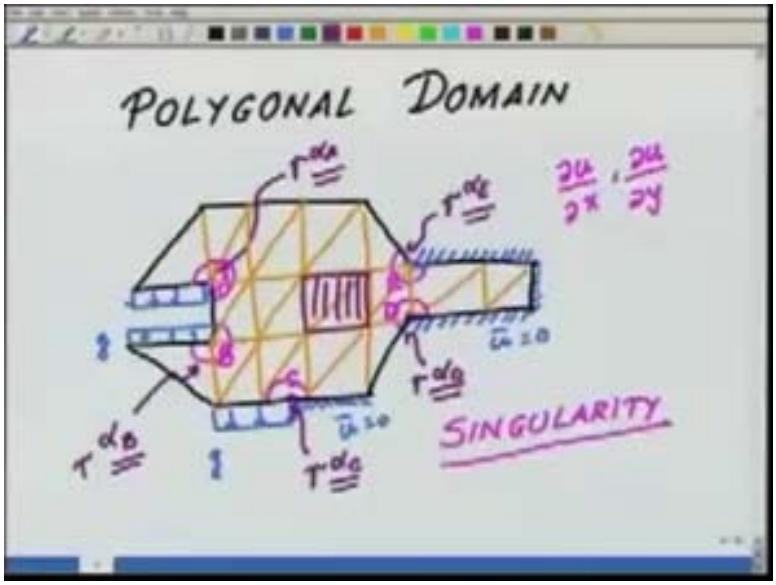
put a mesh, that is, when we use linear mapping. Even though the actual problem does not have a singularity per say, though it has a stress concentration, the problem that the finite element formulation sees is a problem with a singularity, and that is what it tries to solve. For that we are still going to get the same problem as we have outlined.

(Refer Slide Time: 30:44)



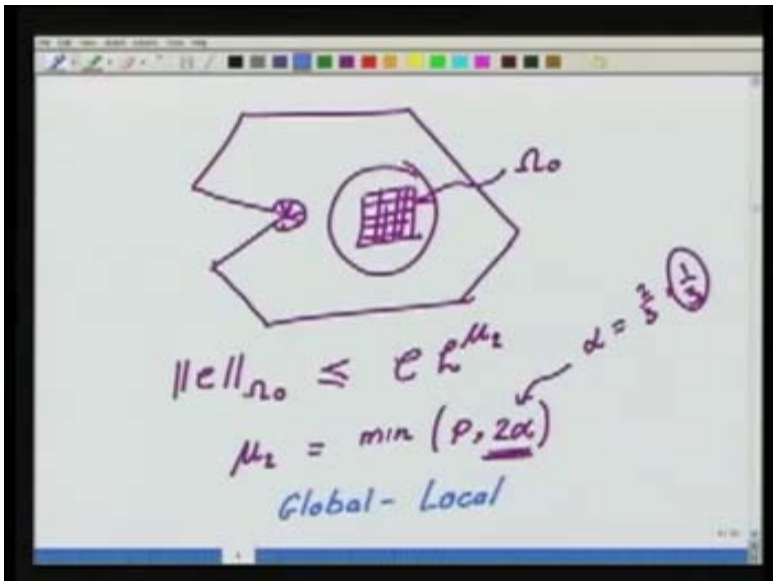
This was about various ways of controlling the accuracy of the approximation.

(Refer Slide Time: 31:05)



Another issue that can crop up is when one is interested in a region that is far away from the boundary from any of these singular corners because that is what may be the region of interest. For example, I may have a cutout here. I may be only interested in the vicinity of the cutout. The question is when do we have to do the refinement and all other things to get an accurate solution? In principle, from the point of view of physics, it looks like the effect of the singularities is local and this effect dies off very fast. Away from the singularity, the solution is smooth and everything should be fine.

(Refer Slide Time: 32:00)

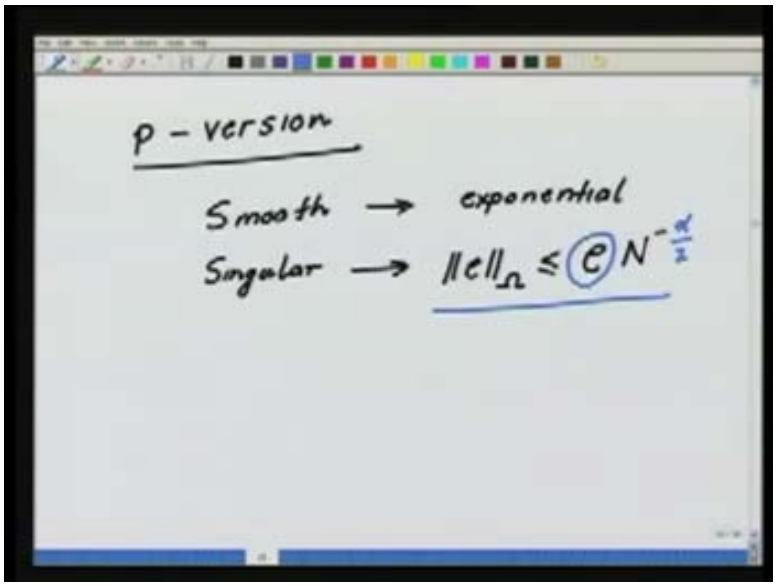


I will draw a domain with 1 re-entrant corner. I am interested in this region. Let us look at how to measure the accuracy of the solution here. I will talk again of energy norm of the error, now in this sub domain or the region of interest omega naught. This error goes as c h to the power of mu_2, where mu_2 is equal to minimum of P and 2alpha.

If P is less than 2alpha, then as far as accuracy is concerned, it goes at, what I got for the global optimal rate. But what if 2alpha is less than P? I have used high end of P and in most of the engineering problems, this is always true even for P equal to 1, because for the domain, I have shown, alpha could be two-third, one-third and so on. For the one-third, 2alpha is always less than 1. So in that case, the effect of the singularity dominates the accuracy of the solution here. This is called pollution effect. That is, numerically, the penalty that I pay for whatever mistake in

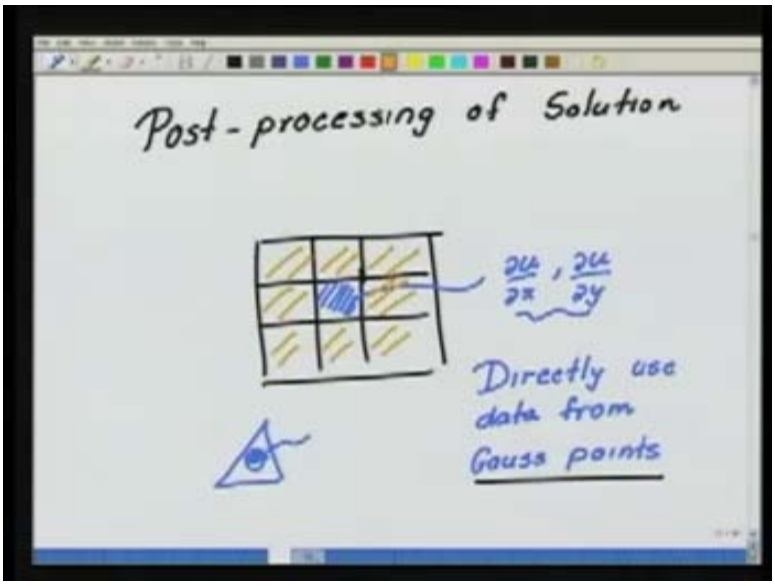
representing the solution in these unsmooth regions, that penalty I have to pay everywhere also. That is why this 2alpha comes into play. Even if I am interested in this region, I will have to do the mesh design near the corners in such a way that this influence of the singular term becomes small here. So the refinements have to be done and this is one of the drawbacks of what is known traditionally as global local methods, where this was traditionally ignored and mesh at singular points that were far away from the region of interest were not refined. But we have to do it because it directly affects the numerical solution. We can use mesh refinement in order to get better approximation, better quality solutions.

(Refer Slide Time: 35:02)



In the P version, when the solution is smooth, that is, I do not have these singular points etc., I get exponential rate of convergence. When it is singular, what do we mean by the P version? It means I fix the mesh and raise the order of approximation. Then the error is lesser than or equal to c N to the power of minus alpha by two. If I have equivalence with the h version, then error is h to the power of alpha. Here again, in the P version, the error does not do better than what we have for the h version. Only thing is that the constant may be small. This is how we do the mesh design and control the accuracy of the solution.

(Refer Slide Time: 36:57)

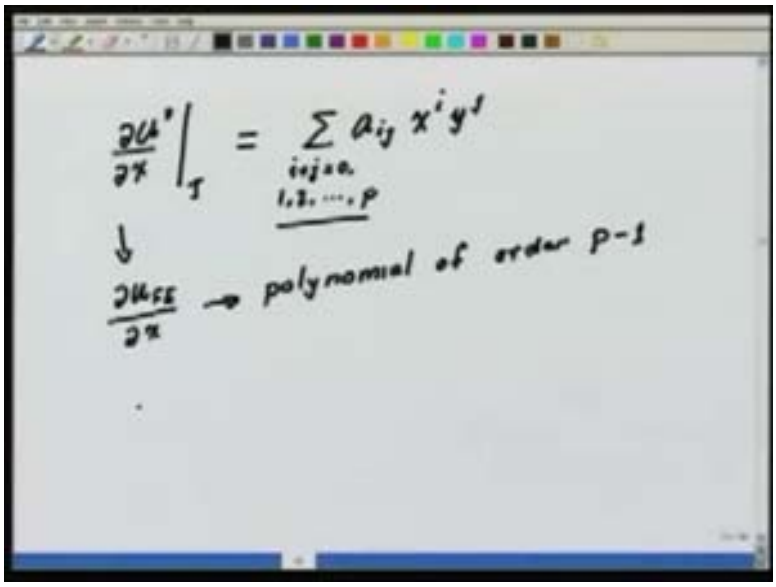


I have now solved the problem. I would like to now obtain the flux information or the derivative information out of the solution of the finite element problem. In that case, we will have to talk about post-processing of the solution. Given the mesh, given the domain, let us say I am interested in the values of del u divided by del x, del u divided by del y in this element. How do I use the finite element data to get a better value with respect to the exact solution? Because, the finite element solution as such will have jumps in these quantities at the interface boundaries, while the exact one is nice and smooth. Therefore, obtain better information about this data at this point.

Directly use data from gauss points which is especially true for the quadrilaterals, while for the triangles, if I have the region in the vicinity of the centroid, it can be used to obtain the better values of this derivative information; which is not always true but in many cases, this is quite good. This is just the use of the direct points from the mesh or from the element that we have in order to extract data at the so-called good points. This is not going to be true for the elasticity problem, but for this problem these points are good enough. Now I want to recover good values for the whole element. In the one-dimensional case, we had talked about the super convergent recovery techniques. That is, by fitting the finite element data over a neighborhood of the element of interest, we can extract seemingly better representation of the derivatives. We will take a neighborhood of this element, one element neighborhood which incidentally in this case

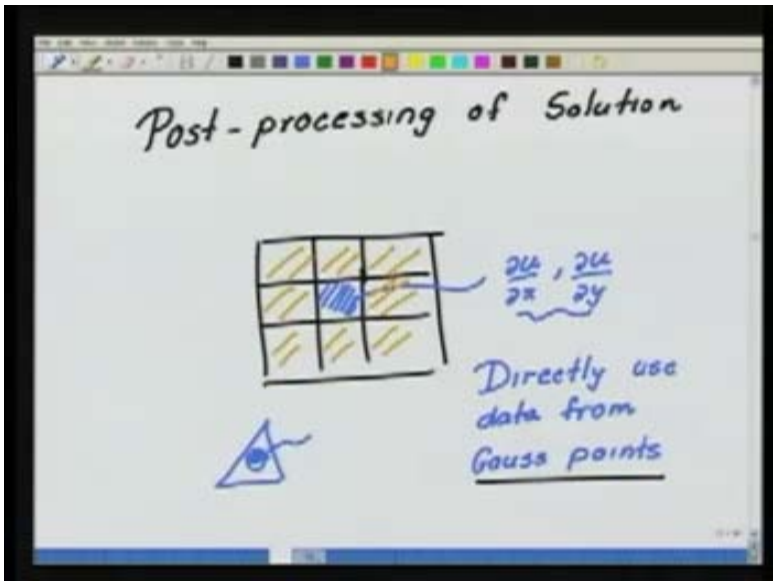
turns out to be the whole domain, the whole mesh. We take all the elements that are connected to this element. That is, they share either a vertex or an edge with this element.

(Refer Slide Time: 40:28)



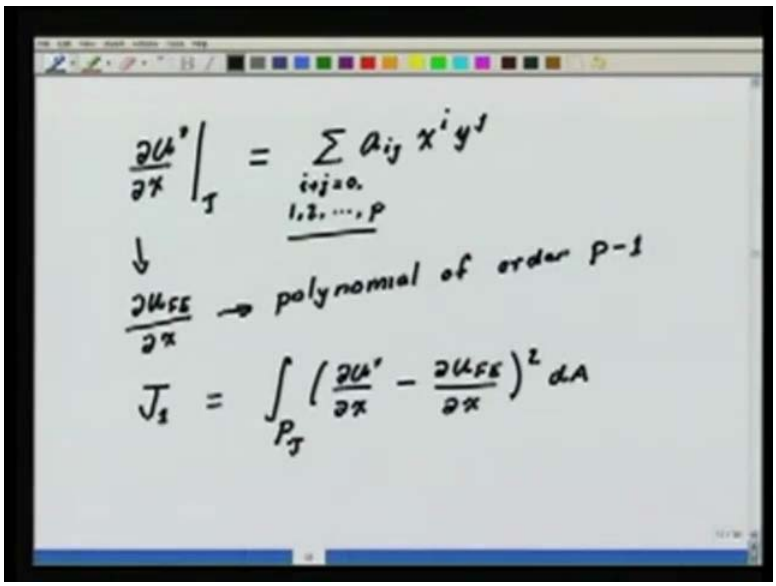
I can have a very simple fitting over these elements: del u star divided by del x with respect to the element of interest tau is equal to sum of a_{ij} x to the power of i, y to the power of j, such that i plus j is equal to 0, 1, 2 up to P. This represents the polynomial of Pth order - a polynomial, P+1, where P is the order of approximation. P+1 or P will do fine because the finite element solution del u_{FE} divided by del x will be a polynomial of order P-1. For this, we take one order higher or even two orders higher and will take it as a polynomial of order of P. This is what we would like to obtain from the finite element data.

(Refer Slide Time: 41:54)



This patch of elements will be called P_{tau}. This is the element tau. All the elements, including the element tau, are called the patch P_{tau}.

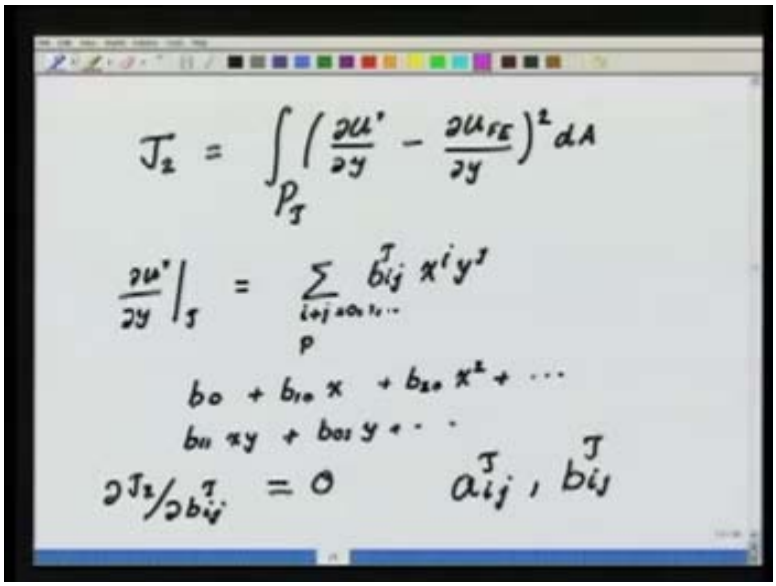
(Refer Slide Time: 42:20)



Over this patch, we will define J_1 is equal to integral over the patch of del u star divided by del x, minus del u_{FE} divided by del x whole squared into dA. I will find the coefficients a_{ij}s such that

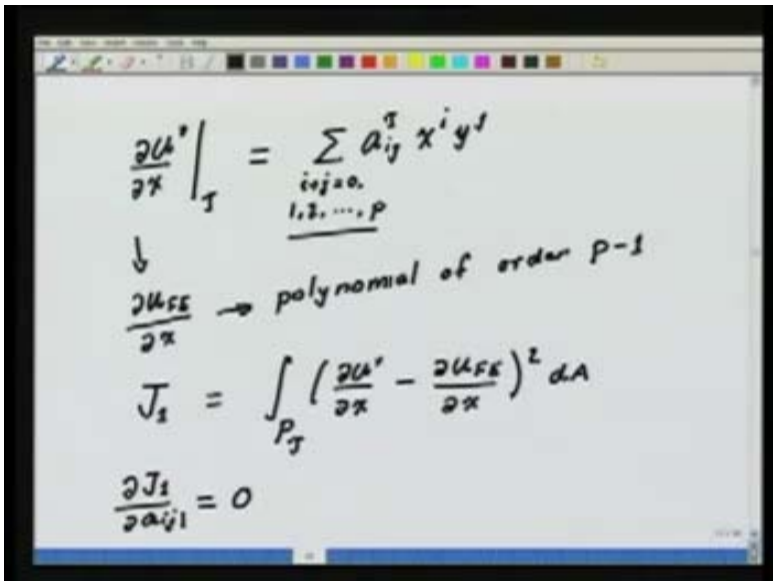
del J_1 divided by del a_{ij} is equal to 0. That is, I find a minimizer of this functional that I have defined over the patch and these are the a_{ij}s corresponding to the element tau.

(Refer Slide Time: 43:09)



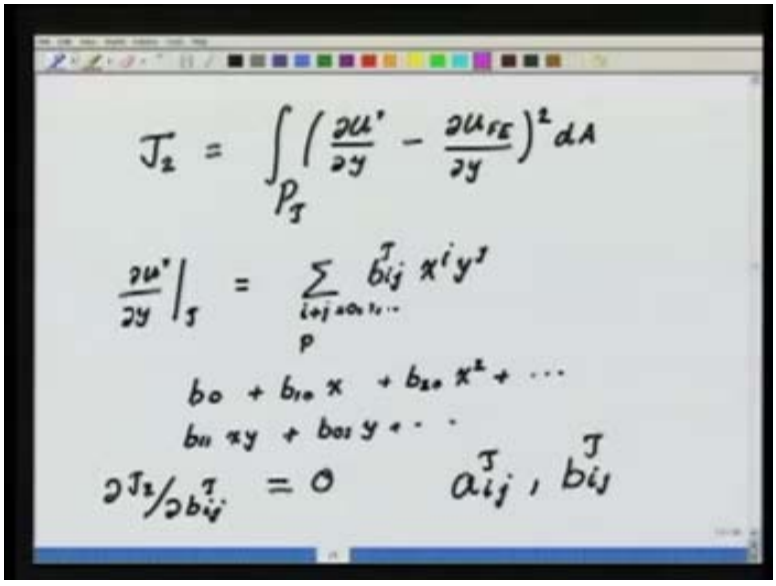
Similarly, I am going to define another functional J_2 is equal to integral over the patch P_{tau} of del u star divided by del y, minus del u_{FE} divided by del y whole squared into dA and find the a_{ij}s in this case, where del u star divided by del y over the element tau is equal to sigma i plus j is equal to 0, 1 and so on, up to P, of b_{ij} x to the power of i, y to the power of j. It will have the expression b_0 plus b_{10}x plus b_{20}x squared plus so on plus b_{11}xy plus b_{01}y plus so on. It is in terms of all the monomials of order up to P. We want the b_{ij}s to be such that del J_2 divided by del b_{ij} for the element tau is equal to 0. This will give the equations in terms of these coefficients b_{ij}.

(Refer Slide Time: 44:49)



Similarly, the previous one will give me the equations in terms of the coefficients a_{ij}. Solve these two at the element level and this is going to be a very small problem. The size of it is going to be simply (P+1) into (P+2) by 2 for each element.

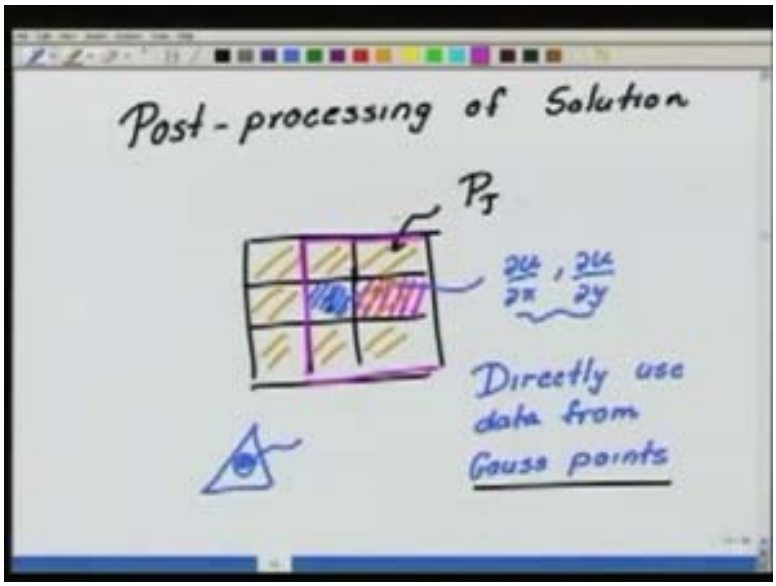
(Refer Slide Time: 45:10)



These small problems can be easily solved and I will obtain the set of coefficients a_{ij} for the element tau, b_{ij} for the element tau. Using these set of coefficients, now I will go back and

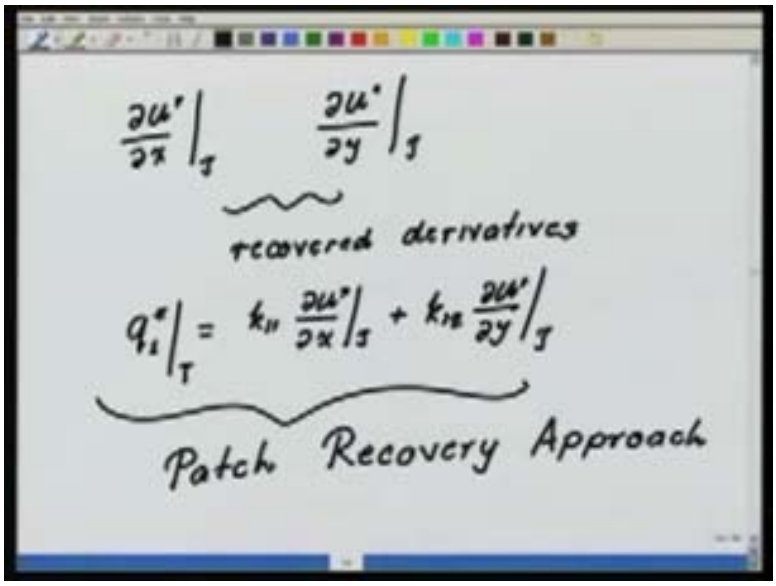
construct the del u star divided by del x in the element. I am doing this projection over the whole patch, but I use the data only in this element.

(Refer Slide Time: 45:38)



Now if I have to go and do it for the next element, let us say I am interested in this element, then the patch for this element will contain all these elements. I will do the same job for the next element by taking this patch and constructing these coefficients a_{ij} b_{ij} and out of that I get del u star by del y and del u star by del x for that element.

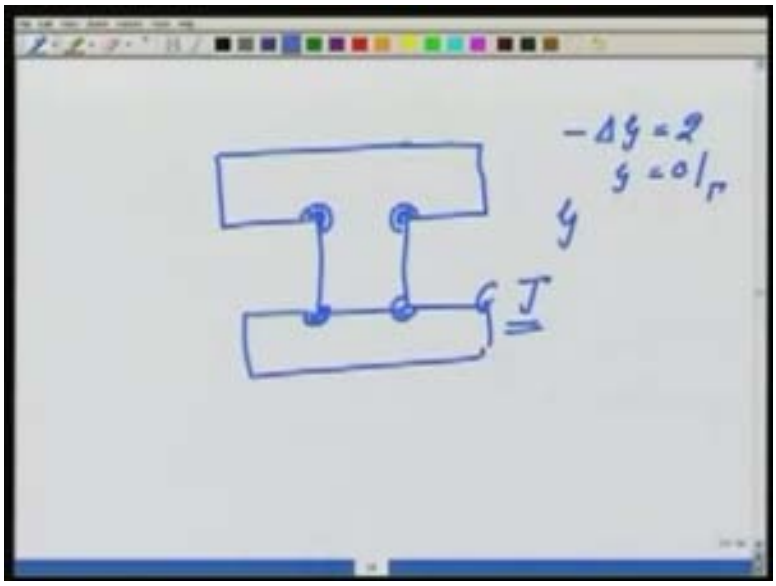
(Refer Slide Time: 46:05)



I can use this information del u star divided by del x in the element tau and del u star divided by del y in the element tau as recovered derivatives and in terms of these derivatives, I can define the flux quantities q_1 star in the element tau as k_{11} del u star divided by del x in the element tau plus k_{12} del u star divided by del y over tau. This is a simple recovery procedure with which we can expect to get better flux quantities. In general, for measures that are not too distorted, elements that do not have very small or very large angles, this does a pretty decent job of a recovered stress field. This is one of many methods that are available to do this kind of a post-processing. It is a patch recovery approach. We can define various other definitions of J_1 J_2. We can do recovery of the flux terms directly. That is, we can write q_1 star, q_2 star in terms of polynomials and get the recovered quantity of interest.

With this, we are now pretty much in a position to solve a finite element problem for this model problem that we have taken which is the generalized parson equation, which is applicable to the steady state heat conduction problem. It is also applicable to the torsion problem where we talk of the prandtl stress function problem, which could also be written in terms of a parson or a laplace equation, depending on the boundary conditions we use.

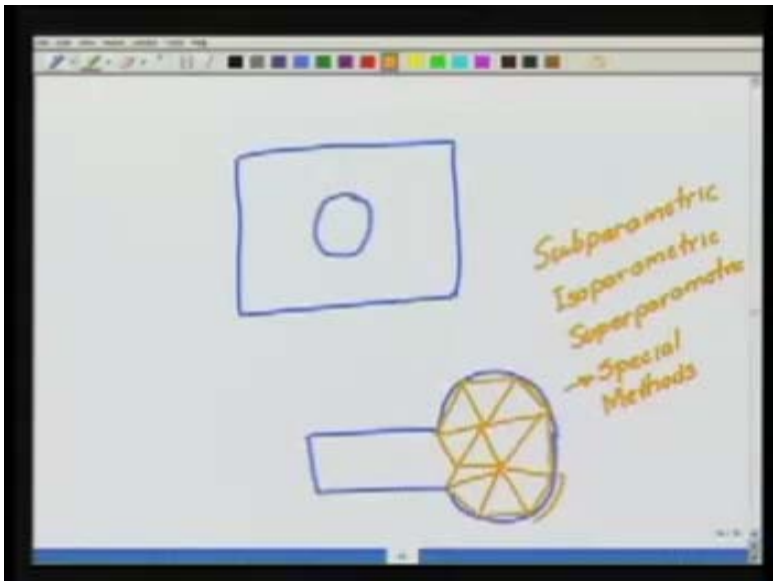
(Refer Slide Time: 48:39)



So all those problems can now be solved using what we have discussed. For example, if we talk of the torsion problem, then we may be interested in this kind of a cross section. For such cases, when we talk of the shear flow and so on, we know that there is going to be a stress concentration. How do we compute the stress function zeta and the j, which go into the modulus of the torsional rigidity GJ? We can use our finite element method. Solve this problem, which is a Poisson problem: minus delta zeta is equal to 2 with zeta equal to 0 on the boundary, on gamma. Solve this problem and we are done and from where we can get J.

Here again, there is a singularity in the vicinity of these corners for one of the very popular domains that we have and this can be handled by doing mesh refinement in the vicinity of this. Another section we are interested in is the ‘i’ section. Here again, we will have these corners and so on. Thus, we have a tool with which we can solve all these problems for polygonal domains.

(Refer Slide Time: 50:15)



Next what we talk about is the case when the domain is not polygonal. For example, I may have a domain, which has a cutout, say, a circular or an elliptic cutout; or I can have a domain, which has a curved profile. In this case, as far as mapping the geometry is concerned, if we do what we have been doing till now, that is, we take the linear map and then make a mesh, I will only be approximating these curves by straight lines. Let us say I make a mesh of triangles. I will be only approximating these curves by straight lines. This is a bad idea because it depends on how coarse or how good the mesh is. If the mesh is coarse, I will be making a lot of error in representing the geometry of the domain. We would like to have a better representation of the geometry of the domain. So in that case, the linear mapping will not do and we will have to use a higher order mapping that is mapping in terms of higher order polynomials. We will be talking of sub parametric, isoparametric and superparametric maps. I will also indicate some special methods with which we can have exact representation of the contour of the domain, if I know the parametric representation of the contour of the domain.

So this is what we are going to discuss in the next class and from there we are going to proceed to some other special problems like the elasticity problem, which is what is of great interest to many of us. For the elasticity problem we will be talking of two variables, u and v and the two planar displacements because we are talking of planar elasticity u and v.