Computational Fluid Dynamics

Prof. Sreenivas Jayanti

Department of Chemical Engineering

Indian Institute of Technology, Madras

Module No. # 07

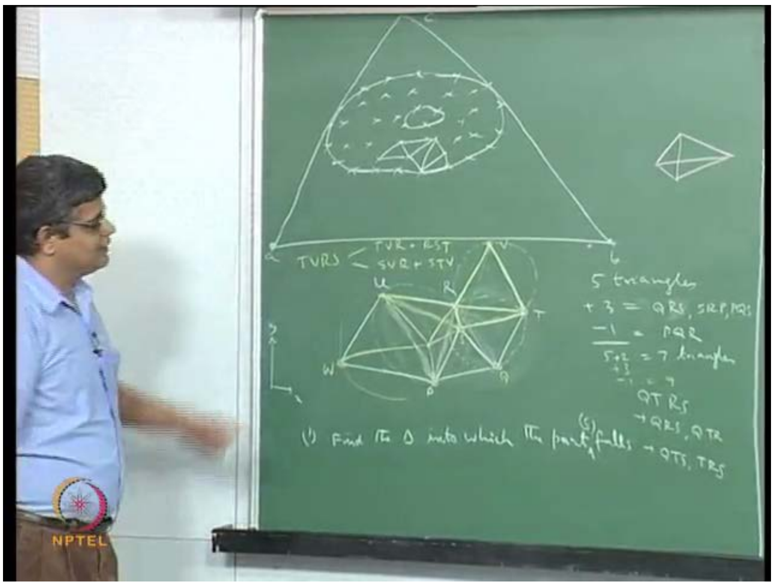
Dealing with complexity of geometry of the flow domain

Lecture No. # 46

Delaunay triangulation method for unstructured grid generation

Having done the nodalization, we have looked at the advancing front method by which we want to do the triangulation. So, that is adding, joining the nodes in order to form triangles what will ultimately form the basis for our control volumes. We have seen how we can do this using the advancing front method. There is another method which has some advantages and that is the Delaunay triangulation method in which were trying to create triangles, which have better aspect ratio, that is, which are more like equilateral triangle, this is the point which we have already touched upon last time. What we would do first is to look at algorithm by which we can do the triangulation according to the Delaunay triangulation method, and then see its advantages, and its specific disadvantages. What we are looking at is a case where already the domain nodalization is done.

(Refer Slide Time: 01:21)



Therefore, we have for a given domain flow domain like this, which is in between these boundaries, we have points which are on the boundary and which are also on the interior. So, we do the boundary nasalization, and then we draw the horizontal lines here, and then we put many points inside with some gap. Let us say that we have these many points and the idea is to join them systematically in such a way that we get triangular tiles, which will fill up this entire available volume.

In the Delaunay triangulation method, in the advancing front method, we start with these boundary edges (Refer Slide Time: 02:22) and then we look at, we start with this and then see we see which of these points lies to the left and which of them is the nearest; so that, we can start the triangulation like that. In the advancing front method in the Delaunay triangulation we do not specifically consider the boundaries like this we only have the set of nodes which have to be joined and that includes the boundary and the interior nodes and we start with our super triangle with vertices such that it is a bit difficult to do it in this (Refer Slide Time: 03:11), but its vertices which are such that when you make a big triangle all the nodes that you want to join are inside this.

So, you have for example, a b c sufficiently spread out. So, that you havea super triangle which completely envelops this now the idea is that you start with the triangle you already have a triangle here and you pick one of the nodes that has to be joined with the other nodes and let us say that there is a node and we have in the process of triangulation we have done some triangulation like this. So, that this is not permitted; let us say that in the process of triangulation we are already through with some things then we are looking at what next

So, at this point these are the nodes with and some of them belong to the super triangle also, we pick one more node which has to be triangulated and that node in this x y domain has certain x and y coordinates and based on that maybe that nodes fall here. So, this is a new node which has to be a triangulated.

So, the first thing that we do is that we have the existing nodes and the existing triangles and we pick one more node which has to go into this and which has to be joined with other nodes to form a triangle; obviously, when you have a node here (Refer Slide Time: 05:11) we do not want to join it with nodes which outside which fall outside this triangle because then that would mean that you will have overlapping triangles and all.

So, the best way to triangulate this is to join with the 3 nodes in the triangle in which this one falls here. So, in that sense we would first of all identify for a given x i y i y j of this particular node into which of the existing triangles the point falls. So, that is the first thing: find the triangle into which noise the point falls and how can we do that? We can do by the left hand rule. That is, if the point is inside a triangle let us say, if it were inside the triangle then, that point should lie to the left of all the edges of that particular triangle or a polygon. For example, if we have if we are going in the clockwise direct in the anticlockwise direction along this edge from this point to this point (Refer Slide Time: 06:30) this is lying to the left and as we now go along this line here we can see that it is lying to the right. So; that means, that this point is not in this triangle again if you go to this triangle here you are following a clockwise direction this point is obviously, lying to the right. So, this is not in this; if you come here again you are following the counterclockwise direction as you go along this node this is toward the left as you go along this node this is toward the left; but as you go along this node this is falling to the right

So, it does not fall into this or this and one can see that when you go along this point here this is not falling to the left of this. So, it does not fall into this, but if you come to this triangle here, you follow along this is to the left of this and then to maintain the clockwise the anticlockwise direction you come here and then this is to the left of this (Refer Slide Time: 07:25) and then you come back here and this is again this is to the left of this. So, by finding out that triangle which has edges such that to each edge the point the new node that we want to join up with falls to the left of that particular edge we can locate the triangle in which that new node falls. So, we go to that triangle and that triangle has obviously, 3 nodes and then we create 3 new triangles by joining these like this because, the idea is to, all these are nodes which have to be joined together and these are to be joined together in the form of triangles. So, that the complete area is covered.

So, we are saying that this bigger triangle obviously, has an interior node which is what this node which has come in and; that means, that you cannot have a big triangle like this because this point this node will eventually have to be joined with other nodes and if it is joined with other nodes then it will lead to overlapping triangles.

The best way to do is to accept this as a new node and accept that this old triangle that you had cannot be sustained because there are interior nodes to that and. So, make this into 3 triangles by joining like this. So, let us say that we are looking at P, Q, R and this is the S (Refer Slide Time: 09:12).

So, in the process originally you had 1, 2, 3, 4, 5, at the beginning of the insertion of this node s let us call this as point s you had 5 triangles and now you have created 1, 2, 3 plus 3 more triangles created and which are this Q R S and we also know that the original triangle p q r cannot be used to triangulate because P Q R will have an interior node. Ultimately, when we triangulate everything here like this where one can see logic and then one can fit by the eye, we can see that in each of these triangles there are no interior nodes. The fact that there are no interior nodes inside a finally, made triangle made a triangle here will mean that once we know that these P S R triangle and Q S R triangle all these things are possible triangles then P Q R which contains this as an interior node s, as a interior note is no longer tenable.

So, we subtract the original triangle into which point s has fallen. So, that is we add 3 new triangles which are Q S R let us maintain some resemblance of counter clockwise direction and then S R P and then P Q S and we subtract out the P Q R triangle.

Therefore, at the end of this insertion we have 5 plus 2 that is 7 triangles and these 7 triangles are such that now, if this were the domain all the way from 1, 2, 3 like this then this domain is triangulated into so many triangles and there is nothing overlapping this. But, the triangulation here does not stop here because, one can immediately see that in the process of creating this we have made these triangles which are not very good triangles from the point of view of aspect ratio; one side is too small the other sides are very large.

Now, is there a possible cure? So, there is a possible cure. So, there is a possible cure and the possibility that we are looking at is that we cannot shift this node here because there are many nodes and at this stage we do not know if there is going to be some other point which is coming here for example, what will make the whole aspect ratio. So, at this stage we do not know what other nodes will be coming into this domain and we have to make adjustments with the current state of knowledge we have to come up with the best possible way of triangulating this current domain that we are considering.

So, our domain consists as of now of as going from here (Refer Slide Time: 12:42) up to this; these things with these identified nodes. Now, if you want to do this then what we e say is I do not have freedom to move this or these things these positions are fixed. For this fixed position is this the best triangulation? Or, is it possible to have a better triangulation with better aspect ratio?

So, for that I take the freedom to consider a different triangulation and I am not happy with this particular triangulation here. So, I consider points q r s q and let us say this is t. So, I have a rectangle a quadrilateral q t r s and this can be triangulated into like what it is q r s and q t r (Refer Slide Time: 13:36).

It can also be triangulated into this way. So, that is I can have q t s and t r s. So, the with the existing domain and with the existing nodes, if I consider just these 4 points here, I have two possible ways of triangulating this quadrilateral; either into Q R S and Q T S, Q T R or T R S. So, that is this one and then T S Q or Q T S like this. which of these two is better? One can visually see that a triangulation like this is better than triangulation like of this. So, the original triangulation that we had for this set of 4 points is not good and we would like to change in into this. Are we allowed to change? Yes, we have, yes, we are allowed to because we have, these are the domain points that we have these are the nodes and we have to come up with the best triangulation. Now, how do we objectively say that S T R and Q T S triangulation is better than Q R S and Q T R triangulation?

We would like to see whether we can mathematically show that this triangulation is better. So, we would like to put a circumscribing a circle through points R T and Q here and that circle may be something like this (Refer Slide Time: 15:45). So, a circle which is unique for the given 3 points Q T R and the idea is that if the point s which is forming the other node of this quadrilateral. If this falls within this, within this circumscribing circle then, it is necessary to swap from this triangulation to this triangulation here. So, considering this circumscribing circle going through R T and Q we can say that this triangulation is better than this triangulation. So, we remove this and then we keep this triangulation here.

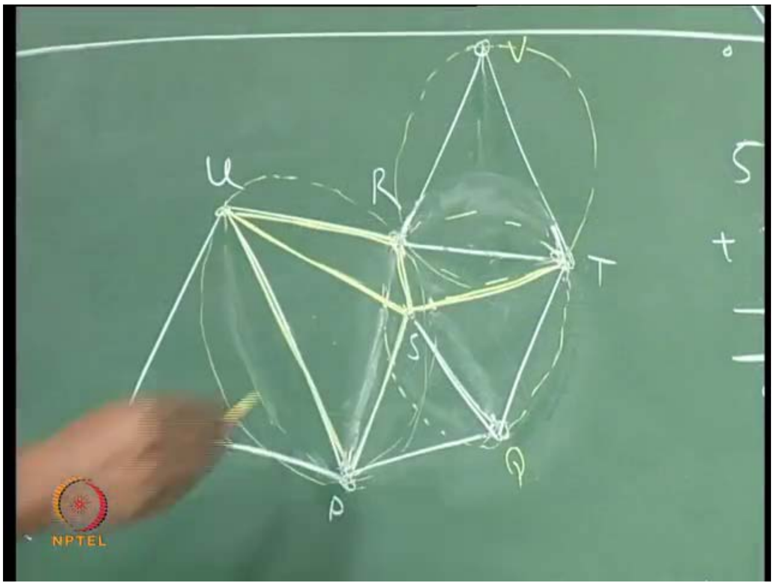
So, as a result of this criterion, here we have decided that when we consider this particular quadrilateral, this triangulation is slightly better than the other triangulation. So, having done this we can do the same thing for any other quadrilateral.

For example, now we have we can look at T S P U, this particular thing this can again be done into either this way this triangle and this triangle or this triangle and this triangle which of these is better we can try to use the same argument of again circumscribing quadrilateral which again circumscribing circle. I am making a guess here that it will be something like this and that this particular node here falls inside this therefore, a triangulation like this is better than the existing triangulation.

So, we remove this and then we remove this (Refer Slide Time: 18:03). So, we have a triangulation which goes like this and like this. Now, we have created a triangulation like this at the end of this step; but now we can see whether when we consider these 4 points is this the best or not.

So, we have R T S U let us say v here then we can go through the same argument we can draw a circumscribing circle through this these three points and we can see that point S lies outside this circumscribing circle. So, that means, that the existing pattern is better than having it like this. So, you when you consider T V R S can be made into either T V R or and plus R S T and or into S V R plus S T V.

(Refer Slide Time: 18:35)



So, in order to verify which of these is possible which of these is better then we can do this circumscribing through these things and see whether the other node falls outside this and if it is outside then we do not have to make this change if it is inside this then we have to make the change. So, from this point of view we retain this triangulation and we can do there we have already done for this triangle and we can do the same thing for this quadrilateral. So, that is what we have actually done here (Refer Slide Time: 20:00).

In that sense, with an existing position of 5 triangles we introduce a new node S here and then, we create new triangles by first of all finding out to which triangle it belongs and then, once we find that out we create 3 new triangles by joining this node here to the 3vertices of the triangle and then since we have made smaller triangles we remove the original triangle P Q R from that. So, we have 7 triangles and we see at this stage whether or not for the given existing nodes and for the existing domain as we have seen here whether or not this is the best way of triangulation without changing the location x I y j of these points here.

So, and that process will result and at the end of every step in the best triangulation that is possible for the given geometry and we have forgotten we have a quadrilateral which is now created like this because, of this is affected and we have to consider this triangle this rectangle also P S U let us say and W here (Refer Slide Time: 21:15).

So, for this quadrilateral is this triangulation better or is this triangulation better? That we can look, do by making a circumscribing triangle circle through this and see whether this point lies outside that and one can readily see that it will be lying outside this see that this point lies outside that and one can readily see that it will be lying outside this. So, that means, that this particular triangulation is better (Refer Slide Time: 21:50). So, in that way at the end of the introduction of a new node we have to consider for we have to consider whether the triangulation is the best for all the quadrilaterals that is that are possible.

In this process when we started out we had this triangle and we created 3triangles out of this and we can if you were to consider this then we had we have 4 possible quadrilaterals. That we have to check; 1, 2, 3, 4, this is another one and if we had some other point here then we would also have to consider this.

So, we considered this quadrilateral and then we came up with this as the best thing. Now, we have we now consider these 2 adjacent quadrilaterals are corresponding to the new diagonal here and then we see which of these is better and then we do it like this (Refer Slide Time: 23:00). So, in that sense if there is a change and if there is a change in the in the quadrilaterals in the triangles that they form we want to see that in every possible quadrilateral that exists in its current way of joining ok. For example, we can consider this quadrilateral is this the best way or is this is this better we can consider this quadrilateral and then we can do we can see whether it is better or this is better we cannot consider a quadrilateral which is like this because that is not actually joined in the form of 2 triangles.

So, the quadrilateral obtained by 2 adjacent triangles must be triangulated optimally; must be triangulated in the better way - better of the two possibilities where we have better aspect ratio. Is this better or for the same points is this better? So, for every quadrilateral we have to look at the 2 possible ways of doing this and once we exhaust all the possibilities for existing adjacent triangles then we come to the conclusion that for these set of nodes P Q R S T U V W, this is the best triangulation possible.

So, now we have 7 triangles 1, 2, 3, 4, 5, 6, 7 which are optimally triangulated and then we introduce one more node and that one more node and that one more node may have x i y j and it may fall into something here. Now, we do the same process; we identify which process it falls into by going into this counterclockwise and we see whether there is any triangle which to which this particular points always lies on the left side and because we are starting with a circumscribing triangle which contains all possible points there is at any point of time the new node must fall in to one of these existing triangles.

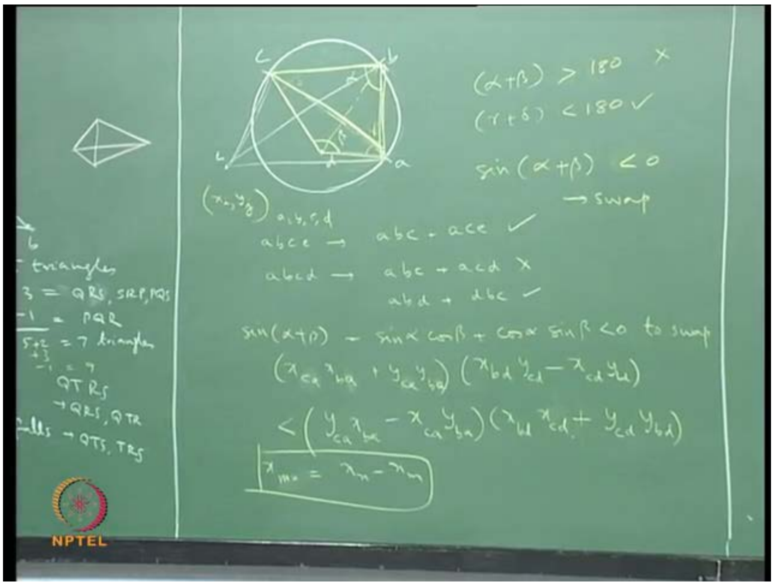
So, we find that out and then we make the triangles like this is provisional subject to the verification that with this existing thing that the adjacent triangles forming a quadrilateral are optimally triangulated. So, we again go through the process of for example, examining this is the, are the 3 new triangles (Refer Slide Time: 26:06). So, we take each face of this and for example, this face which belongs to the triangle in which the new node has happened we see whether for this one this triangulation is better or this triangulation is better so; obviously, for this is better.

So, we eliminate this and then we retain this and this is again another new quadrilateral and we see whether this is better or this is better; this looks like it is better; this is the other edge is this better or is this better it looks like the other may be better (Refer Slide Time: 27:00). So, we have this and then we can see if these are already old triangles we have already verified this. So, we do not have to worry about these things, but any new quadrilaterals that are found for example, now we have quadrilateral like this point this point this, point and this point.

So, is this better or is this better this looks like this is better. So, we can live with this and this triangle we have just now this quadrilateral we have just now just now verified that this is better than this s one. So, we have now introduced one more point. So, we have plus 3 minus 1. So, that is there are 9 triangles are there and we have nodes.

So, in this way we introduce each node and we then we form 3 new triangles and we then eliminate the older triangle into which this falls because that would mean overlapping this and create in a sense 2 new triangles and then we go to the quadrilateral argument that is for the given set of possible quadrilaterals obtained by merging 2 adjacent triangles. We want to see for each such quadrilateral do we have the best possible triangulation and that triangulation depends on the aspect ratio and the verification of this is can be done by doing this thing. But, we can also have a simple formula for checking the verification - the triangulation. For example, if you have if we have 4 points let me first draw a circle.

(Refer Slide Time: 28:52)



So, a b c and we can consider two cases for example, d which is here and we are looking at this being the triangulation and another point e here and this being the proposed triangulation. So, let me just put the different colors. So, that we can understand this better.

We have drawn a circle and this is the circle which is circumscribing the points a b and c and we have a node d here this is obviously, falling inside this circle. So, that means, that the existing triangle over which we have made this circumscribing thing is not good and we should swap it into this. Now, we take, if we take points of the quadrilateral a b c e, then we are proposing that it should be made into a b c e should be made into a b c that is this triangle plus a c e that is triangle here (Refer Slide Time: 31:05).

So, and if this is what is proposed then over this a b c we draw the circumscribing circle and we see that point e lies outside this c circumscribing circle. So, this is better, but if we consider quadrilateral a b c d and we propose to divide this into a b c plus a c d then this is the existing proposal and we draw the circumscribing circle going around it and see that the point d is lying inside this. Therefore, we say that this is not good and that we should make it into a b d plus d b c and to prove this we have to draw a circumscribing circle which is going through these a b d and see that it is going beyond that and one can see this drawing a circle like this.

So, this is what we are trying to do here and to put it in a more quantitative way if we consider this quadrilateral here with a proposed triangulation like this which we know is obviously, wrong here and if we call this as alpha angle and this as beta angle then we would like alpha plus beta to be less than phi less than 180 degrees.

See, right now, for these 2 opposite angles to the common line here to the common diagonal of this quadrilateral that we are proposing. So, this is greater than we 1 80 degrees and this is not good. So, we would like that this be the common this be the diagonal and this be the triangulation; so that this say gamma plus delta here. So, we can see that if you say that this is gamma and this is delta you have gamma plus delta in the existing formulation this is greater than 180 and. So, this is preferable and this is not preferable (Refer Slide Time: 34:11).

So, if we have an alpha and beta which is greater than 180 then, we should do the swapping of the diagonal from a c into b d. So, this same thing can be put in the form that sine of alpha plus beta should be if it is greater than if it is less than 0 then this is then you have to swap and we can come up with a criterion that writing sine of alpha plus beta as sine alpha cosine beta plus cosine alpha sine beta should be less than 0 to swap and these angles can be expressed in terms of the coordinate points of the 4 corners.

So, it can be written as x a c times x b c plus y a c times y b c let me just check I think it is better to put it like this x c a times x b a x b d y c d minus x c d y b d should be less than y c a x b a minus x c a y b a times x b d minus x x c d y c d y c d times y b d, ok.

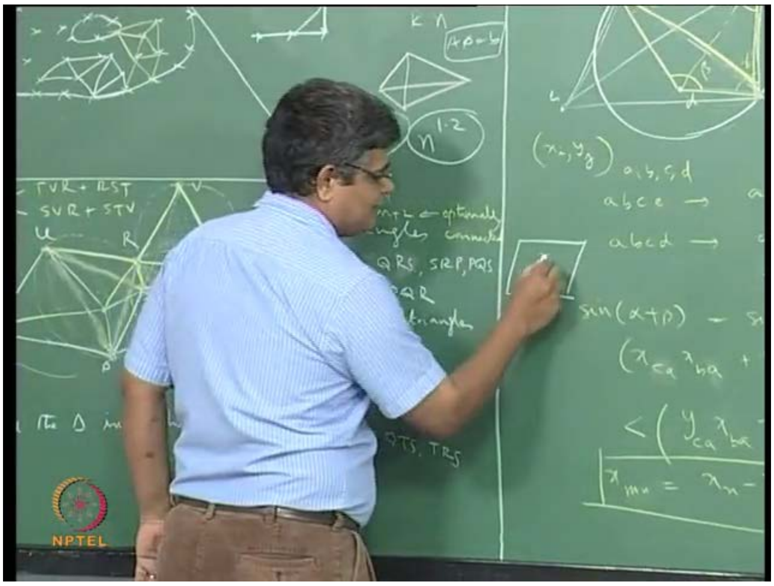
The point essentially that we can do it in probably different ways, but this condition for the swapping that alpha plus beta if it is greater than 180 then it is not desirable and that we should go for swapping can be expressed in this way and this can be converted intoan algebraic criterion like this where x c a is defined as x m n is defined as x n minus x m so, the condition of swapping need not be done by drawing the circumscribing circle and all that it can be programmed very easily.

Once we know the x I y j of the 4 quadrilaterals that are considered and one can use a formula like this where x m n x c a is x a minus x c x b a is x b minus x a and for example, y c d is y d minus y c like that. So, what we need to know in order to verify this condition is the x I y j of all the nodes a b c d if we have this then we have an algebraic condition for verifying this. So, we have to the mathematics involved in this is rather straightforward because, we are only using algebraic conditions, but we have to do a smart programming in order to keep track of the number of existing diagonals triangles and then the existing triangulation and then adding and then deleting there is a lot of bookkeeping to be done and thena lot of searching to be done, but all of the searching and all of the thing involves simple algebraic expressions involving only the nodal points only the coordinates of the nodal points.

So, that is the important thing here that if we know the x I y j of all the nodes then we can come up with a clever program to a smart program to implement this and then we go through like this until all the nodes that are to be joined are considered.

So, if you have 100 nodes we put all this 100 nodes in some array here and then we consider this super triangle and then each node into this we take out of that array here and put into this and then we have a separate array of triangles that are created at each step and then we have the triangle information to be stored in some ordered way and then as and when these triangles are modified the triangles that are under consideration have to be also to be modified.

(Refer Slide Time: 41:21)



So, at each step before each step we have m number of triangles at the end of this step at the end of insertion of one of these nodes we have m plus two triangles which are already optimally connected. So, this calls for some facility with programming, but it is possible to do that and then if we do that then we can make this triangulation work.

At the end, we go on with this insertion of more and more of this points and then creating more of triangles and going through this verification process that until all the 100 nodes that we want to nasalize are taken out. So, that is that array which contains the existing number of nodes yet to be connected becomes an empty. So, at that point we will have all the nodes are joined with the other things and then we may also have these remaining the three nodes of the super triangle some of them may get eliminated if they do not get eliminated then we have to take out these nodes and then we have to take out from the set of triangles those triangles which have this these super nodes corresponding to the super triangle (Refer Slide Time: 42:43). So, at the end of that for a given domain, we will have come up with a triangulation which is optimal when wherever you consider 2 adjacent quadrilaterals having common faces.

So, if you were to make a quadrilateral by joining these two then the corresponding triangulation that we have is already optimal in the sense it has chosen the correct way of ordering this and that is the advantage of this particular algorithm for triangulation and this incorporates this idea of this and this works well for convex surfaces because in the convex surfaces we can guarantee that the triangulation this particular the optimal triangulation is actually done, but if you have a the concave surface then this particular algorithm may run into difficulties.

We may get a (Refer Slide Time: 44:00) concave surface for example, a corner in such a case we may have boundary nodes like this and the algorithm which also has some super nodes. All those things may join may form in the process of generating these nodes for example, at this stage the information for the boundary coming in this and it may form a triangle like this and this; obviously, is outside the domain and this triangle should not have been formed, but at some interior stage in the process of triangulation we do not if we do not have the information that this triangulation is prohibited then it is not possible to avoid this kind of thing.

So, this is possible for a concave surface; but if you have a convex surface like this then this criterion will always make sure that this kind of possibility does not happen that it would chose a triangulation like this rather than a triangulation which is falling outside because that is not optimal. So, that that condition that the optimality e verification here will makes sure that a triangulation involving odes which may make this fall outside like this will nodalize in the process of triangulation. So, and there is no way of avoiding this if we follow this algorithm only at the end we have to see whether we at the boundary points what kind of triangles are generated and if there are any triangulation triangles which fall outside the domain then we have to remove them.

So, that is one of the disadvantages of the Delaunay triangulation and also the advantage of Delaunay triangulation that it is very good at doing comp convex domains, but in concave domains like in corners and those kind of places. We have to be careful that that the triangles which are which fall outside the computational domain are not made and if they are made you have to take them out.

So, this is how we can do triangulation either using the Delaunay triangulation method or the advancing front method. In the case of advancing front method, this situation does not arise because we are always defining a boundary and we are always looking to make sure that for example, something like this will be going in this direction.

So, we are always conscious of the advancing front and which makes this kind of solution unobtainable it would not arise in the case of advancing front method now before we complete the discussion on the grid generation we should see some idea of how difficult it is.

We can immediately see that when we talk about an instructed mesh it is very it has to be a smart program it has to be a clever program and in terms of mathematics it is not such a demanding thing in terms of computational effort it is not. So, demanding in pure number crunching kind of thing, but there is a lot of logic that is required for this and the generation of the grid itself is not expected to take too much of time, but one can see that there are a larger number of possibilities.

As the number of nodes increases the computation time required for verification of all these things increases and it is generally thought that the unstructured mesh like this involves a computational time which increases with the number of nodes to the power 1 point 1 or 1 point 2; let us say that 1 point 1type of increase of the grid generation time with the number of nodes.

So, that means, that as the number of nodes increases then it does not increase linearly because there are more possibilities, but as we have seen in this process you do not have. So, many new rectangles, new quadrilaterals that you have to verify; so, it is not n square or n cubed, but its only n to the power of 1 or so.

In the case of the structured jess mesh generation, for example, using the elliptic generation method that we considered we have to solve we have partial differential and we have seen a number of methods and typically there it is of the order of n square if you if you have an iterative method for the solution of this and it may be n square or more because you have to do it a number of times.

It may be some k times n square where k may be some ten or twenty depending on the number of iterations that you have to make to get the matrix which are required as a part of the solution which is produced as a part of the solution process although this is the type of computational method that is required this pales in comparison with the computational time required for the overall solution of the equations. Because, in those kind of equations we are solving a similar type of a phi is equal to beta type of equations, but we are solving that many times because we have an iterative process and then we have to do that and the computational type which is required for the grid generation also pales in comparison with the time it is required to convert the for example, an engineering drawing into the specification of the boundary and all those kind of things. There is usually a lot of time that is spent in identifying all those things precisely what is the computational domain and how the computational domain is, can be specified in mathematical form such that you can generate the overall shape of the domain. So, that you have this is a part of the triangle this is a part ofa cube this is part of a sphere.

So, all those things will have to be added together. So, you have to make use of several building blocks to come up with the definition of the surface topology of the computational domain. At that point, if you especially if you have a lot of internals then the generation of the surface topology of the internals itself will take long time and that is usually much more time consuming than the raw computational time that is required for the grid generation after you have defined the surface topology of the computational domain.

So, from a practical point of view, grid generation is very important. It is the essential it is computational process there are methods that are available for the generation for any domain of arbitrary complexity, but it is not a trivial task whether it is using the structure generation method, body fitted method, or in the unstructured method, there is quite a bit of programming that needs to be done and we have to take care of so many things. So, that we do not get into problems with overlapping meshes extra volumes and holes in the computational domain, all those kind of things are avoided. Then, the quality of the grid in terms of the aspect ratio of the cells is again further point orthogonality at the boundaries at another point which is a the desirable feature.

So, there is a lot of aspects of grid generation one has to study quite a bit more before one can become very good at grid generation. So, what we have tried to do is touch upon the important elements concepts involved in this and for fairly simple geometries whatever we have discussed here is possibly sufficient for us to venture into the programming.

So, using what we discussed we can easily generate the grid for example, a domain like this (Refer Slide Time: 53:24), with some internal packing or if you have a corrugated flow domain like this with one fluid going in this direction, another fluid going in this direction its again possible to generate a grid like this, for using this kind of approaches. But, grid generation is not a trivial task and it has to be done professionally and even if we have a very good grid generator the generation of a grid starting with an engineering drawing is not a trivial task.

So, one has to allow for sufficient time and comprehension of the details of the geometry of the flow domain to and understand how it can be converted into an expression which is usable by a grid generator. So, that kind of time is also not to be underestimated.

So, having considered all these things, one can say that grid generation is now a routine task for a trained specialist. It is not to be done for one has to train oneself into various aspects of the generation strategy and then the correction strategy, mesh refinement and mesh amelioration all those things are to be done before cone can come up with an optimal geometry. So, in the last lecture of this course we will try to see, try to address some of the computational aspects of solution on an unstructured or a body fitted mesh.