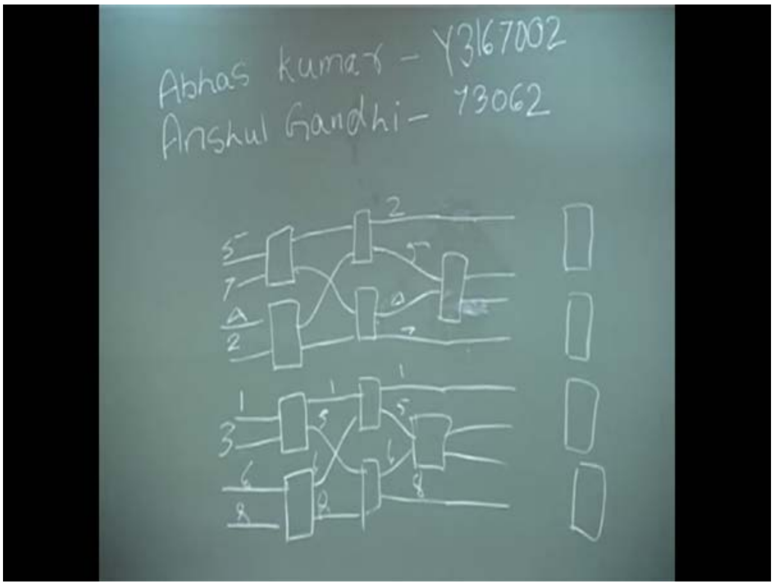
**Parallel Algorithms**

**Prof. Phalguni Gupta**
**Department of Computer Science and Engineering**
**Indian Institute of Technology, Kanpur**

**Lecture - 15**

It is it is group all is whatever group consisting of one or two or none. There are the different types of problems that you are thinking to handle, and we will be selecting one or two college groups, so that this will be you can understand in the later days. And also while presenting the problem you have to tell also when do you aspect to finish your training. So, that accordingly you can have your presentation right, that is the idea but do not tell that I am ready with the problem and solutions I am ready to present today, that is not allowed. So, first you are...

(Refer Slide Time: 01:19)




So, since you have stand lot of (( )).

Student: Can I start sir?

Of course,

Student: All right. So, the problem network thinking to work upon with the odd-even merging problem. This was discussed in the class little bit, the problem as that given a set of numbers we have to sort them and them using odd-even merging in this. The device that has been used here is comparators and what they do is the, they take two numbers as inputs and they will give you the result of they will give you the result of the comparisons between these two numbers.

Student: What we plan to do in this is to trying reducing firstly the number of comparators used throughout and in secondly whatever comparator we are saving at each slate. We would like to use them to do some more power calculations and thirdly the algorithm as it stands already, is not very flexible. So, we might not be able to use all the same comparator. So, if possible we try and make a few modifications of the algorithms. So, that, we can use whatever comparators may saving at in each stage. Now, that idea would be, we will see what are the maximum amount of patterns that we are using at any given stage and then in the later stages. We will see how many comparators we will save and then, those save comparators we will try do some future comparison if it can be possible.

(Refer Slide Time: 03:20)

Like for example, say we have something like this, where we are comparing a, b and c, d and say we have some extra comparators saved here already.

Student: So, we can see that, at the next stage at whatever next stage comes we will be comparing with the output of this. So, if in case we have extra comparators, one possible idea would be to right. Now, we could compare, let us say we could compare b and c, if we have extra comparators, because it they not doing any work. So, we can use them to do something. So, it may be possible that at later stage we have to compare b and c at sometime. So, because the result is already known here, we could reduce that stage. So, this will be roughly idea.

Student: Yes sir.

These are comparatives. Now, you are thinking about the network model, right?

Student: Sir.

The model is given to you.

Student: Yes sir.

Now, how to say the data from then the data b into these comparators? How do you thinking that one?

Student: Due to one of this, you see you are saying it is already going into one comparators and how to. So, may be at the beginning we could make copies of these data, say that if this is the initial stage we could make copies of this d be forehand

So, you assume that applying b has been operating in different places.

Student: Yes we would want to use b modern once. So, we would have to make copies of that.

So, you keep it in mind this you understood that.

Student: We will have to make more than one copy.

Your aim is first one is that optimum use of network.

Student: In network model.

Student: Of operators of a network model.

And second thing is that you want to try to make the processor busy.

Student: Obvious yes sir.

And another one is that.

Student: We want to modify the algorithm, if you give the algorithm as it is, if it is not very flexible. So, we will change it, for that we can make use of a comparator.

Can I make it little? I want to suggest one thing that one that if you gone through the minimum comparison based network of.

Student: We will be we will show that little bit we have seen at chapter 5 points.

So, yes there is a one thing you can think that you have 5 points and you can have optimum network (( )) even 5 elements.

Student: Finally.

That optimum network is given

You see the diagram.

Student: So, 5 it exists.

5 it is there, 7 is also there something most competitive there, 5 it is there suppose you have the 5. So, after 5 you get sorted and 5 is sorted.

Student: We can make use of that model.

So, what should be your network in the equipment?

Student: It should be something.

So, see you are trying to use then you get output is 5.

Student: And then we use that.

Then you can use this is another one thing another thing is that, this is a tau is 5s order, sevens order.

You can think about those things, 5s order means that 5 elements. So, I have a 5s order, because this nothing but, we are assuming that given 3 elements I can have.

Sleep level sorter whose takes constant amount of that right. So, please go through those materials, before you think about merging odd-even, merge network and other thing. So, that you know little modification. We 5s order, sevens order that can be taken into account, also combination of 5s order or sevens order, whatever it is with odd-even merge, that also can be right and any other problems you are thinking.

Student: No sir.

So, that you will be working on that. So, anybody one else to give any input to them? Now, what is the forgetting?

Sorting network, your problem is the sorting network right? Sorting network and respected date of the little ten is one (( )) eleventh is.

Student: Eleventh and thirteenth.

Thirteenth, which one you want?

Student: Eleventh one.

Eleventh one and write up.

How many problems you will be complete?

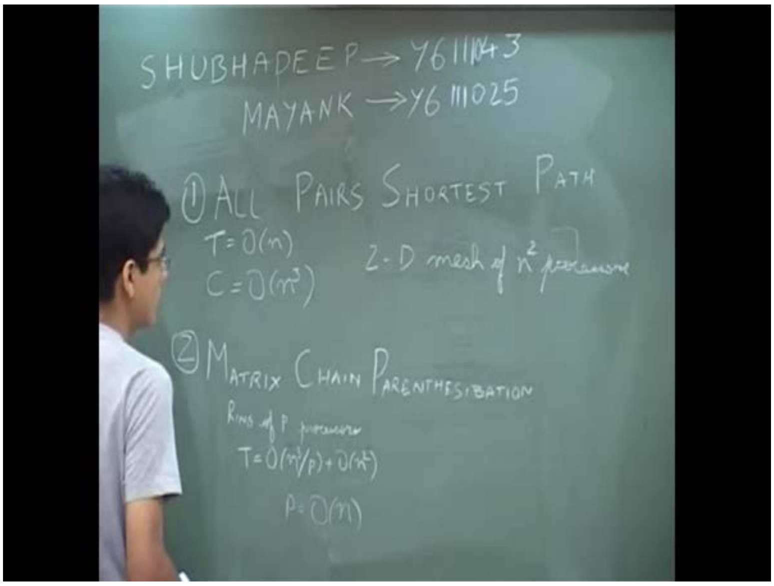
Student: 4 or 5.

Why 4 or 5 why? Would not 5 or 4 either it is tough for write them all the 5 problems.

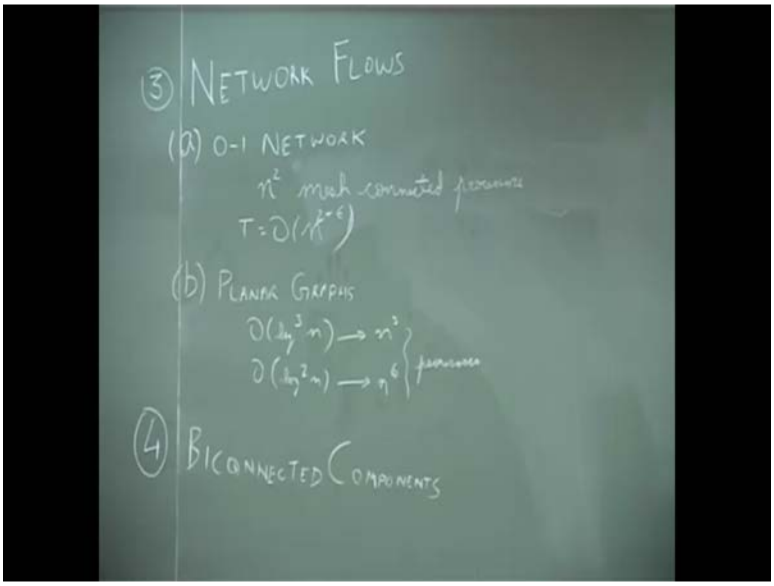
Student: 4 sir.

Write down your name first.

(Refer Slide Time: 09:42)




(Refer Slide Time: 13:43)




Student: Let us looking at some of the dynamic programming problems.

Say you will, why they two roll numbers are different?

Student: Sir.

Student: Let us look at some of the dynamic programming problems of sequential algorithm, that we have studied earlier like [third] versus algorithm for shortest path and matrix parenthesization problem. For Floyd versus algorithm would little bit inputs. So, parallel algorithms, what we have got is, we have an implement this on a mesh 2-D mesh network with n square processors. So, each processor will compute the d i j k term of the [third] versus algorithm in order n time. So, the ultimate the entire algorithm will be solved in order n time and the cost will be order n cube. So, it is d cost optimum and for the matrix parenthesization problem. We have this given a series of matrix and we want to find such a order of parenthesization, that will minimise the number of multiplications for that we have this rink of key processors and for that we allot time requirement will be order of n cube upon p plus order n square plus order n square. So, six p is order n.

First one is done.

First one you want to discuss first one

Student: all pairs shortest path.

No, I have several questions on that.

See all pairs shortest path getting several algorithm parallel algorithm it is simple.

Student: The Floyd versus algorithm version directly we are just looking at that.

No, I am talking about have you seen any parallel algorithm of all pair shortest path

Student: Just a logical description of it.

Have you seen any parallel algorithm?

Answer is yes or no?

Student: Yes.

Which are those one?

Student: Only the Floyd version algorithm.

There does not at least more than only one algorithm exist on parallel.

Student: No, obviously there exist more but, I have seen only one.

Now, is there any algorithm for 2-D mesh?

Student: The one we are talking about is implementing on 2-D mesh.

Now, do you feel, then you do not take in to better you able to break that one?

Student: One thing is that we can just try to implement this or we can try to have some different we can implement the same problem on a different architecture.

The second one is will happen in parenthesization problem.

Say matrix parenthesization is lot of work has done again.

Have you seen any algorithm?

Student: One of them.

Why would not try to understanding? Whenever I am asking you are telling one of them. Have you read the paper?

Student: We are looking at one of the same paper. In that paper actually we are talking of a number of dynamic programming problems like normal common subsequence.

Student: First one is flow network, we are looking at two cases one talks on planar networks and the talks on the next one planar graph but, this planar graphs paper no algorithm for planar graphs. So, there is only some maximum flow network algorithm existing for parallel planar for only some class of graph, not for all possible graphs for, if we try to do for this, maximum flow networks then it can possible to more kind of all possible algorithm paranthesization.

Student: The fourth one sir, is the one way discussed with you earlier bi connected components, where in we will take the various components of the train get the articulation vertices with respect to these components for each articulation vertex, find the next to be added and then add. So, the input tree is not bi connected with the output is a bi connected tree graph. Sorry, bi connected graph.

See now, I have these four problems you are planning right and what are these? Are available that p is order n?

Student: Yes sir these are available.

Student: Of n square processors

And network flows, what happens to this bi connected thing?

It is informs of tree right?

Originally yes it was a tree.

And output also become a tree.

Student: Will be a graph, yes will be a graph not the tree but, a graph because, it will be bi connected that will not be a tree sir.

Student: There are two these things parts from any tree.

For in that case are you going find out the (( ))?

Student: Yes.

So, if one parallel algorithm (( )).

Student: Parallel, did not find any parallel as reaction.

Then, why did you try that one only? It is a bi connected component.

Student: Serial there are two different once less this one.

Can you find out?

Student: Parallel algorithm if not them.

Then another thing what I wanted to do find out in the bi connected component if you observe what happen say at least an articulation point and then after the new earth connected another one that is the simple method.

Now, we define we like to define this each characteristic.

Each characteristics, the each characteristics means that I have a very strong connection, width connections moderately good connection right say I have connected but, that connection is bit, I have a connection cannot be able to.

Student: Strong, weak as in capacity not a laboratory.

Sound wait vector if you put in.

You feel, and then the problem can be defined into better say.

Student: Saying we need a minimum x weight between them.

I have that band width connections and another things.

Yes sir.

Right now, the some corrections are mid corrections, some corrections very strong connections.

Student: Yes sir.

And my fault is that, if some fail then, I have correction but, that correction is v

So, I cannot take over loop

Student: On this one.

On this something like this one can we find out those type of can we introduce that type of issues

Something like network mesh

Can we introduce some issues of bi connected algorithm? You have to define little problem little carefully.

Student: Sir

Right you start working on bi connected component.

Student: Yes sir

And in the mean kind we try to re define the problem of bi connected components and we say these two problems. I am sure this is not be able to do anything on that better and network provided you can some write the dynamic record.

Student: Yes sir.

That evens parallel algorithm small units of problem. Now, the whole changes in between them.

Student: Yes sir.

Right, then can you write a parallel algorithm, say if you then keep the dynamic or some connection.

Student: Brakes or is lost.

So, in that case you are this algorithm will be. So, dynamic algorithm you can write, dynamic parallel algorithm. So, you can think about that or you can think about bi connected component and why do you consider the bi connected component problems. We will try to design the problem where the connections also have some weight vector.

Student: Sir.

Right, because uniform correction, now a days it is not recognized.

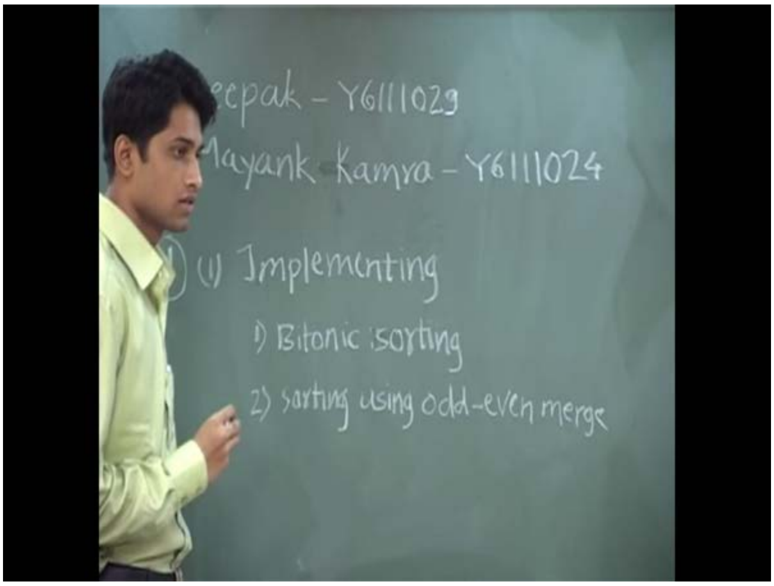
Not of that much.

(( )). So, if it is a case one can, you can be. So, one of these two units, one is a dynamic network problem.

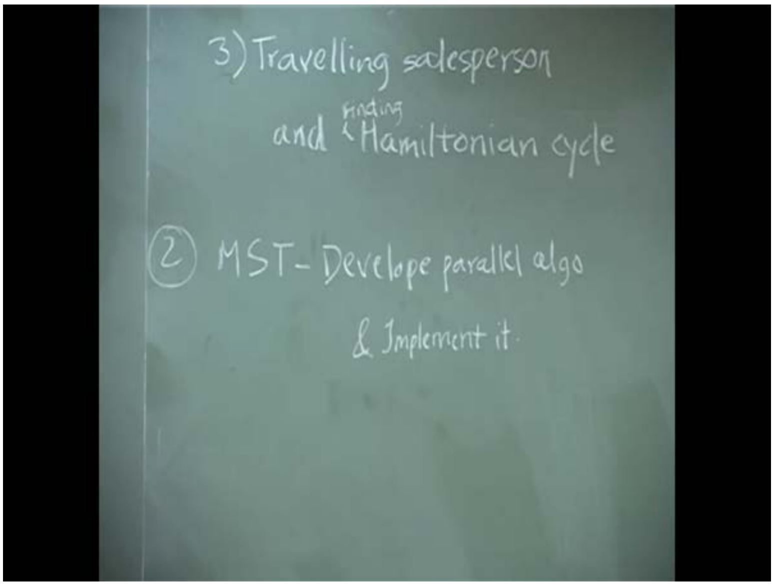
Student: Sir.

You can think, this is dynamic one, dynamic parallel, you understood what do you mean by dynamic parallel algorithms right? That is one or bi connected component you need to some weight vector right.

(Refer Slide Time: 22:44)




(Refer Slide Time: 24:51)




Student: So, we are thinking about actually two problems, one implementation of the algorithms covered in class and starting with the some algorithm. We well know to sorting next we were a use sorting algorithms and different networks and different models and then it is timing formats we could also.

So, your problem is on just on implementation. So, which one [matrix] should be using.

Student: Sir, we are trying to implement on the different architectures and comparative the performance.

Can you see you using s r p machine? Which machines where are you planning to establish?

Student: Sir, we are planning to implement on m p i.

Where you need n number of machine right?

Where are you going to get them?

Student: C c machine

C c , s m p s machine right? Another one is just you have the transformation also, you can think for it but, that you have to design your model base on that m p i.

Student: Sir and second one is the problem, that means what happens may be look at the m s t algorithms? There are travelling sales person.

On what model you are planning that for m s t?

Student: We use actually we start parallel algorithm.

Student: We start with different

See the problem with m s t, that you have to find out minimum positive right, that you want network model. I think that minimum forty, you can also think about in network model in which we generate m s t the model itself in your life network model for per day similarly. So, you have to some both on there or either of these two.

Student: Probably we will try to implement some other algorithms.

So, here you can have several right some of them.

Student: That can.

Then you will have library type thing.

[FL].

So, I must get advice of this all the course along with, I will have the some 5 processor, I will write put in some array coma p processor and it will in both we are agree the same, right?

No, say you do not that work you have either of this say what you want. This one, if I get library form very good this is required.

(Refer Slide Time: 28:43)


Simulated anteing outline which is (( ))

Student: Simulated anteing.

Student: It is a standard bottom to some optimization technique in which we have one configuration and we will be apply some energy function algorithm and got a new configuration. Compare the every function for new configuration, comparatively main basics idea of that is and we are correlating symmetry and it is problem with the t s p problem. In this we are having a complete weighted graph and we are a function is to finding cycle of minimum. So, this problem is available with the linear connected array and it is time complexity provides them and it is also available.

Student: So, we are trying to do this algorithm and [arrive] this problem and other points and second are to parallel B plus Tree, in which like a in peoples we everything is on minimums and we are trying to decision, this notes on parallel processor in which, like you want to exist or awake anything on both statement. Take a and third problem is to solve T S P, that is take T S P problem in which we are having call some sorting to solve this problem and actually take which is very less.

Student: Sir, actually we have problem to take something o of n to the power n factorial and we found for algorithm which is taking o of n square only and cost is, if we are using n cost is a n plus n cube which is much less than n to the power.

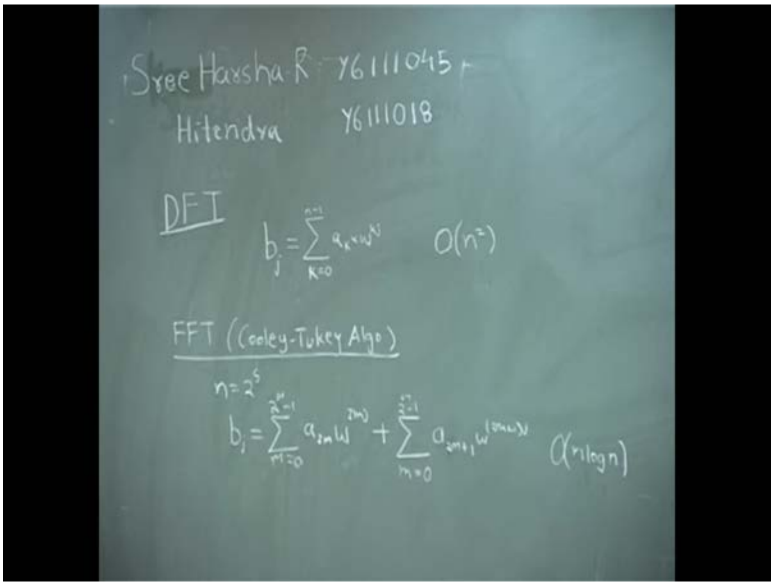
Are you sure?

Student: Sir, means we are.

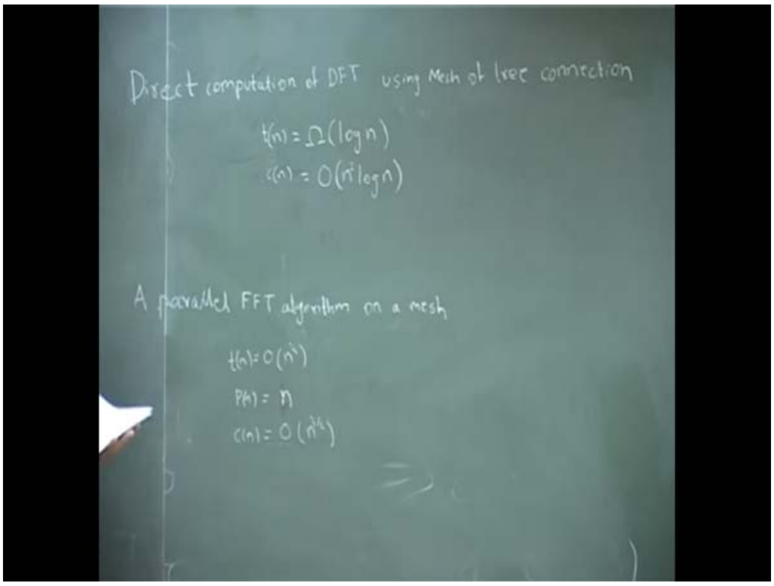
Student: And. So, think implementing that parallel red black tree.

Yes, say I need to volt or need to a update myself on parallel algorithm structure, right? There are people working on, if we have on parallel algorithm along with the some idea on parallel p plus 3 and parallel algorithm. One of them some idea you do not to solve it, right? First one is that, whatever parallel structure that kind of full right of along with, then there exist the other data structure and you take one of the false item one, the possible rate can solve or provide the right (( )).

(Refer Slide Time: 33:40)



(Refer Slide Time: 37:30)




Sree Harsha R.

Student: Sir we have minimum parallel.

Student: Right now, already we have butterfly mesh 3 and mesh. So, but, actually the right now, this butterfly thing is but, we want to see we want to construct mesh p and mesh actually. The thing is mastery, what happens it becomes very in efficient, because of the cost remember the time is very optimum example, 3 log n the number of [means] that are there between in this notes there are so many means that become completely. So, we want to look at algometric parallel let us see on mesh. It is a finite number of extra notes like these are along with the new set, we have on the internet we want to add some on the regular mesh, we want to add some links and see how the actual algorithm reacts with and how did cost like?

So, first what you want do is it you are not direct you want define mesh around forty 5 the mesh connected are required right and. So, that it can take a log of right and then you try to implement on to it.

Student: One of the machine right and we want to see how the machines on the same architecture? How they compare? So, that like be an otherwise say and you get even and odd obviously.

So, only thing is that, if it is a network connection then.

Student: It also remember thing that 3 d profession like butterfly. So, if we can hangover.

If it is a butterfly, then you should work better way on [hider clue] you can think about, I bulking also and also you can think on mesh should be the idea for us and particular, because it should gives those implementing at fifty Organization of better result in some cases.

So, may be what they want to tell you here at the end of the implementation I must able to use it as a right way right say we are going to get that. So, I want to say that at least our think should be equivalent to their right whatever they have.

Then only and now you should becoming how to provide the input. So that, it is the big problem that I know you take minimum of time.

Student: Sir, actually once we start implementation be coming out.

Yes. Now, tell me your suitable trick whether you trying.

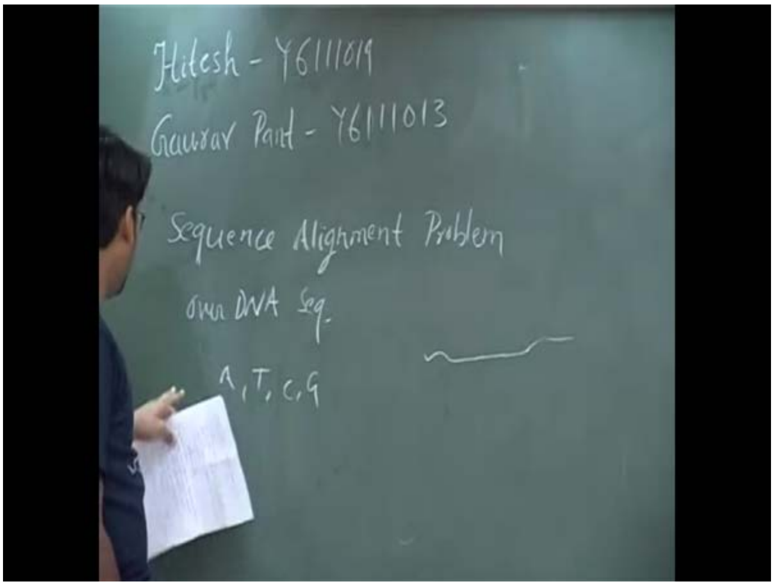
Student: Sir, D F T using mesh of tree connection it takes order of log n time but, its cost is too much. So, F F T on a mesh, it cost less than this. So, we are trying to few connections only not the whole.

Say you can instead of say, why not mesh is having four connections, right? Could not generate, you can use think about the (( )) also, instead of constant number of parallel algorithm.

But, by adding the possible algorithm you must be able to achieve right. Say, here the time is you observed the time is order n order of n, even go your possibilities later.

See once I for few day machines once I take basic cost is not important right, time is important. How much time I have occupied by machine right? That cost is important before I compute the machine see somewhere. What is we have got that? What a single machine there exist a single machine, which is generic importance, right? That all problems cannot be solved efficiently on a concluded parallel machine provided you assume that it is fully corrected machine.

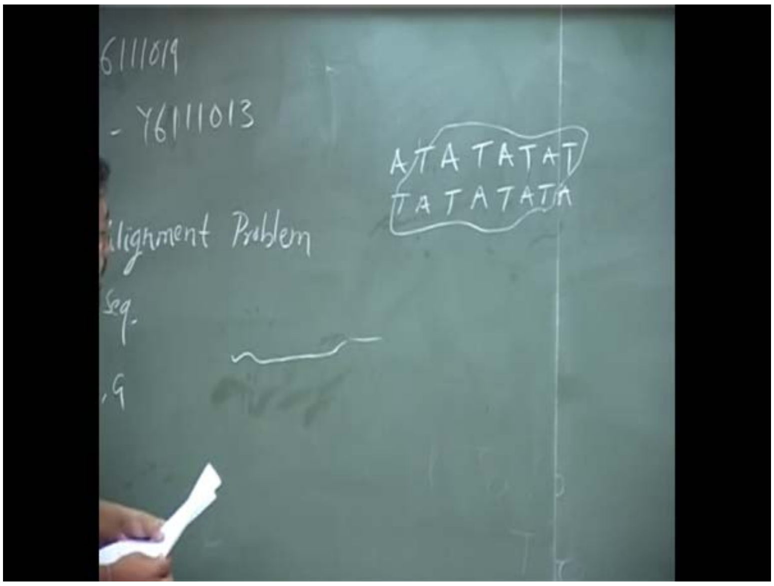
(Refer Slide Time: 41:09)




Student: Basically, we are trying to solve this sequence Alignment problem over the DNA sequence; basically it is a problem of [bulk] matrix domain and here as we know DNA consists of liquidized A, T, C and G and basically the motivation for this. Problem is that we want to infer the functionality of some new genome every time, we found some new genome and we do not know the function it what we try to infer the functionality of that particular new genome from the existing DNA and at genome and. So, what we try to match the particular sequence suppose, I have a sequence of this A, G and C. This nuclides and I got some new genome. What I will do, I will try to find out, I will try to match this particular sequence to some existing particular DNA sequence or genome sequence and if I got some match, there are some different types of match local different types of alignment local and global over all. We will talk about the entire thing but, what we try to find out that particular match and if I got some particular match and to from that particular that existing genome functionality. We try to infer the functionality of this new genome.

Student: So, here basically first of all, if I want to compare two sequences. So, the very basic concept of having distance but, here is some problem in having distance.

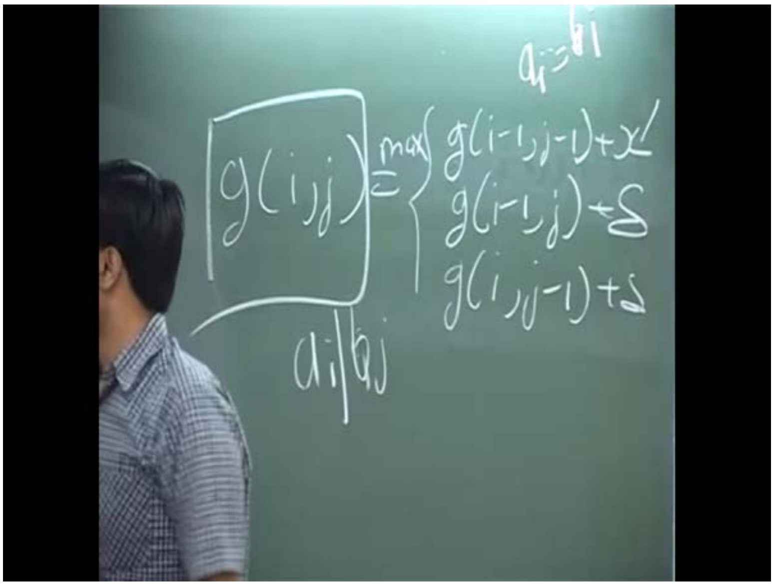
(Refer Slide Time: 43:48)




Student: Just consider the sequence and i have another sequence here, if i consider the having distance, then having distance will not work. So, and you can just see if i just shift this the whole listing by one position right towards right, then it will got a match but, this portion will got match to this particular portion. So, here the rather than having distance we take the concept of added distance sphere. In added distance sphere, what happens basically? We insert some we will make some insertion some delusion and some replacements.

Student: So, for every match we reward some points and for every insertion or deletion, combined and we tell that indent. So, for every insertion deletion, we pay some penalty. So, there are mainly some methods for alignment and there are some algorithms for these things. So, basically what we are trying to do, we try to basically this genome sequence consist of millions of these nucleotides and if we have quadratic time algorithm, like order of n square, then you can just consider if [values] very large and. So, even quadratic time algorithm will take lots amounts of time. So, there is a lot of the scope for parallelism for this particular problem.

(Refer Slide Time: 45:32)

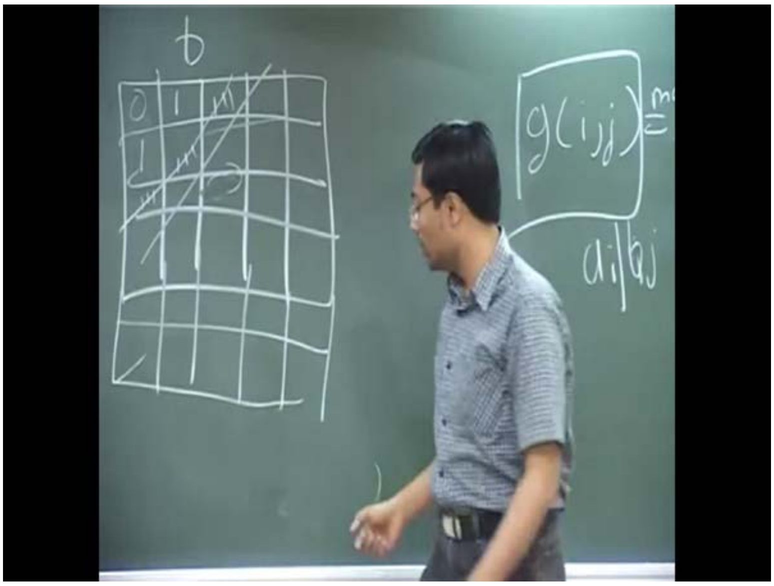


Student: Sir, basically we have, till now we have dynamic programming solutions for these problems. Where in the basic recurrences that, this cases when a i string a i matches a i-th matching with b i where we are matching this, a and b where b is, if a match occurs, we get some odd points and if I get some insertion division, then we deduct this can plus with data will be negative. So, we deduct some kind of.

Student: So, at any position g i j this denotes the score, optimal score of matching up till a i to a j to b j. At that means, if I am at this position that means, I have matched this string. So, this gives me the optimal score. So, finally, what at a m, b m that the final position. We get the optimal maximum score of matching two string a and b.

Student: Now, the problem here is that, this algorithm solution is order n square and. So, what we got n plus, it takes n square space and since n is very large. So, this is very costly. Now, if we try to parallelize it what we can see over here is, at the first step what we have is the value of this location. Now, in the second step we have we can parallely calculate this and this. In the third step, because any for calculating any value we need 3-dimension g I, g i minus one j minus one i minus one.

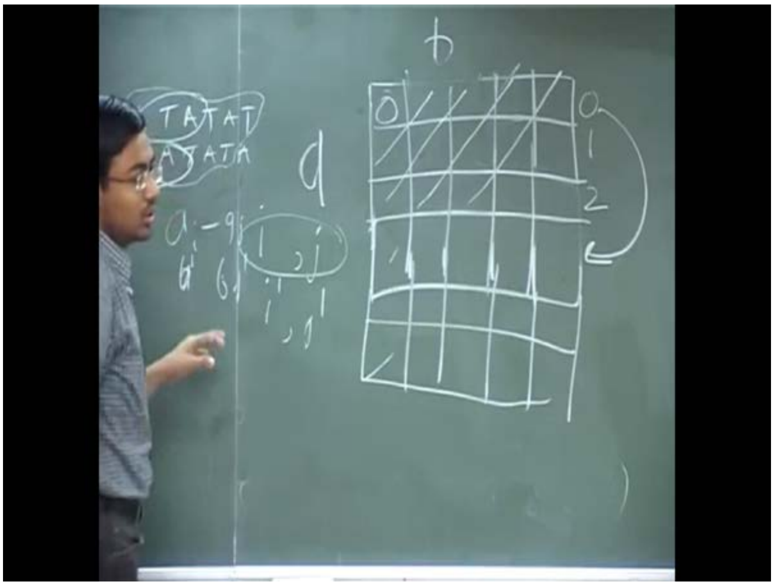
(Refer Slide Time: 47:41)




Student: So, all these 3 can be calculated parallel.

Student: In the next step we can calculate this parallely this is called wave front parallelism. So, but, the problem here is that n is. So, large that we cannot just go on increasing the number of processors. One time we require order of n processor where we wasting the processors because only at the big time is in the rectangle q c diagonal and it is actually not possible because n is. So, larger we cannot have so many processors. So, what we do is that, instead of that.

(Refer Slide Time: 48:12)



Student: We signed this first two to the 0 processer, first processor, second processor and this is calculate the first instant, there are first step, then at the second step. These two are calculated at the third step. These 3 are calculated and this is calculated like this and when it reaches the end in the 0-th processor is replaced this location. So, this is the basic problem and what we are trying to look into it. As how, which is the architecture, which can give best results for these. So, this sequential laminar problem and what optimization or what implement can be do in the existing algorithm to make it more efficient.

Student: Although because, basically the problem is that, this is not easy to visualize what sort of architecture will be useful. Here only one thing we can see every processor require 3 connections, like it requires the data from 3 neighbors. So, we can think of such an architecture in which a particular processor have 3 connections and then what can be the maximum means group, what we can have 0 for example, I am having 3 processor, how many in with relation to n, what will the optimal number processors.

No but, you can always assume suppose, you have then you can always staying sequential where that maximum p.

P diagonal.

Student: P diagonals right.

So, that is your first level of and then you have to design all the network on this p and this is the best one, then it will be duplicate right rest part is duplication.

Student: And then this is the global alignment problem, we have a look a bit harder problem, which is local alignment but, what happens is that, it may be that in the two strings, the whole string is not matching to the second string but, it may be that the small portions are matching very well, which is called local alignment. That means we have to find two positions i j and i dash j dash, where in the first string corresponding a i to a j matches well to a i dash sorry b j b i dash to b j dash. This is called the problem local alignment. This is a bit harder problem because, here we can simply go from right to the end this source to the.

Now, tell me one thing what are the different from others.

Student: I can match.

Sir, l c s is just a special case of this problem where, we have 0 for this entire 0, for this and 1 for this.

So, you cannot use algorithm.

Student: Sir actually that is, that we are just matching one.

Student: Sir, we are not sir here the concept of a d y distance comes up in k m p.

Most of the you are also assume that one may be instead of every t a t a t a t somewhere you got a t a c a t a t. So, some weight factors because of the mutation.

That is, can you see that part also in what way it is? Give for from here it is algorithm.

Student: Sir, actually those string matching algorithms look for correct exact matches not so, because did not for everything here. We are just finding the optimal score, what optimal score we get between two strings.

So, can you re-design k m p s algorithm, for not exact matches you are ready to suppose I am ready to sacrifices 20 percent of twice right.

Student: Sir, we will try to see.

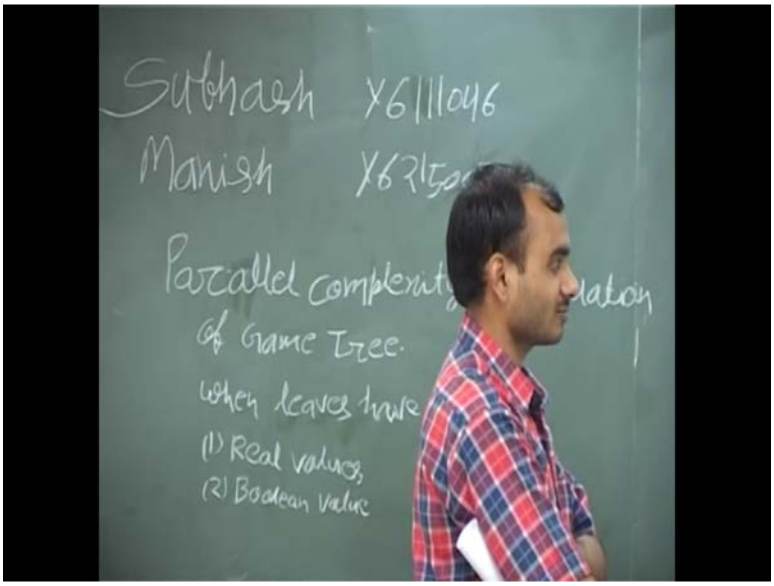
Or the other way here, we got matching g i j. Now, instead of that g i j i exact matching if i say hundred then know if it is a lack of some number I will assume that.

Some of you know then only you will be getting the problem will be little complex, you know? You have to make some that. So, that otherwise you will be taking that p diagonal number of these are some modifications, and then you will be able to do. That is you get some problem at the same time you want to put little.

Student: So, we can look into modification supreme.

This is that point.

(Refer Slide Time: 53:38)




(Refer Slide Time: 54:15)


Student: Now, we taken as the land and algorithm we have for any just for example, in any.

No, first you tell the problem.

Student: The problem is to play a game in which first player is trying to maximize this programs, maximize this game, means if he wants to maximize the chances of [win], I think this is a famous. Here is a two things although both has same thing by Min Max is that thing is that we can represented by met, also met first coming this tree where in the first player node. This is all settling works means, where we can maximize the game and all the square nodes, where this is the move for a second player and where he wants to at least where he wants to minimize the game I think.

Student: Another one.

Student: No, he wants to means, totally is a game where he wants to minimize the total game means, second player means totally second players game is reduce first game.

Student: See that, if I get maximum one, that you get the minimum one.

So, what the aim is to earn more?

Student: No, because game is single game when he that the gain of game is a single. Means this is not different for means I want to maximize that gain in game. So, if second player wants that my gain is reduce, then he needs for example, in any game in case for, if I wants win the second player was that my gain is reduce means, if he means or even my gain is less. So, he wants to minimize the game, he wants to maximum first possibility.

Tell me what is real I have in the game you are even find out the now you tell.

So, now what is in main thing, what that is?

Student: Here it is that

When you will be [gainey] what he gets rupees pocket right?

Student: No, means I want to means if we are playing any game that I want solve this, that the gain of means his gain is already reduces that.

But why that the square box?

Square box is means, the gain of means, this is total game but, he cannot give that, this game is for player one, not this game is means player game is single game, where one is one wants to maximize the game and second one wants to minimize the game.

Automatically right, suppose you want one is the automatically (( ))

So, these are the questions he wants.

He wants I think it is defining game to respective single game yeah yes

But you are telling. Now, it is a formal game because one player.

Student: It is one player game but, this is total game means, both are happening but, the definition of means, gain of game is single game here. We very two things means, the in sequential algorithm, we find that for example, he is coming means, we needs some nodes by alpha, beta pruning where if it things that mention, first player goes two choices, a first player have two choices then he knows that, this choices anyhow better means all those nodes have less game and he knows that means all of further nodes are less game. Then we remove this node, this part. We are not chooses this part and he only chooses first player chooses only this span and then again the same thing for second player, if he wants that or the game because second player wants minimize that game, if he knows that all these are minimum Kinetic.

Student: So, there are many part we need the many parts have not taken at all so

So, you will be looking some

So, what in sequential algorithm? We remove these things in parallel algorithm, we cannot do like this, we have to take.

Say in sequential algorithm, we are inspecting each and every leaf is beats and but, in parallel algorithm we are trying, if we have the p processor then divide this within the p groups and check.

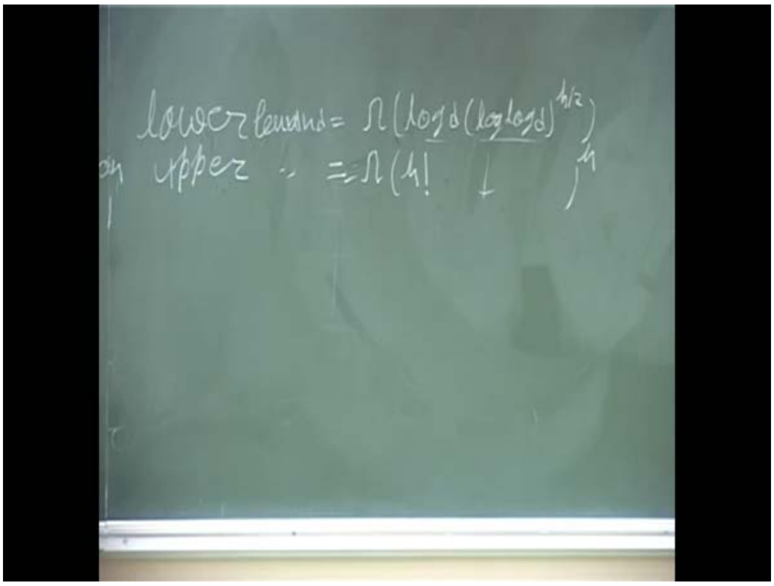
Which is?

Which is the best result and remove all the. So, we only have to check one group and move for p groups if you take you may find that is a you know although p x number of this in form one sub tree this and other sub tree.

What is p minus x? if processor x some other how to tell that, you have to see that part there whether the highest sub tree. You are going consider or not collapsing. How you are going to and if question collapsing right and there, then you will be collapsing and then finally, it will go up to that, it could be right. So that collapsing technique for parallel processor.

Student: Actually we are also calculating that, how much we can enter this situation means because, there is some Engineering algorithm exist.

(Refer Slide Time: 1:02:43)




Student: Yes sir, exist but, h is the degree and here it comes, that real h and two to the powder h, the power is also the square.

Student: Whole to the power h here it is also means the power is also a square of that in end.

What model within?

Student: Sir not for a particular model.

Where is the way for, If you consider butterfly possibility, can you think about the butterfly model?

Student: Sir, actually we have to gives, we have to see which model.

No, of course. So, what 8 to 1?

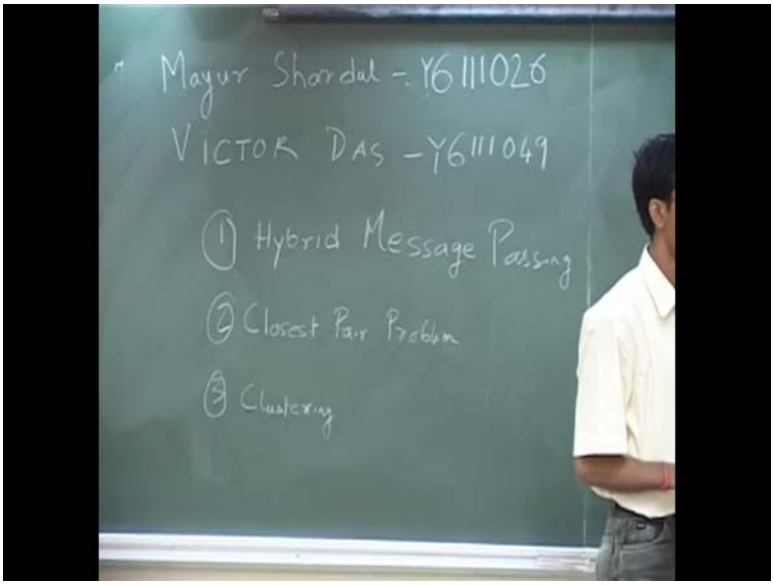
Student: Eleventh.

All of you 11 cannot go up particular one.

Student: Sir?

(( )).

(Refer Slide Time: 1:04:35)




Student: We have network of multiple methods and that is could not running over the network. So, for message passing, if the message is passes through a pocket, it will first of all there will. So, for those matrices, sum from local network into the game. So, we plan to implement library over shared memory as well as m k and it will dictate in the remote process in on local machine or remote machine and according you, the underline open m p I does not have any but, there is an implement, that have a similar feature but, we implement this feature and evaluate the game and comes by reduction.

We just will change the part of this, if it is in the same not them where just to you.

Student: Sir shared memory there will be it should be a management but, a basic simple implementation assuming the call messages can be.

Now, what you want to do? You have and all the types of machine, then you have to pass the message to all the nodes. So, what will the adjustment?

Student: So, say we have many, multi core system. So, multi core system, if it is within the same system different code. So, we can use the shared memory. So, in that case we will not, we will just change the send and we will check that if it is within the same system then we just use the share memory or else if it is in different one, we will just use this empty one norm al m. Then we have to game on the closest pair problem means, in this we have one gone through. It is just they are taking every point and they were comparing with every other that is sequential is order n square and they were in 2-D mesh they were doing it in order n.

Student: But, we mean, we thought of parallelizing some other means sequential algorithm, means where the sequential is of order n log n.

Student: Means, divide n into two parts and recursion.

Student: And then and where we can divide the rate of points into various groups and then recursion they will merge.

Why do not you try this one?

Student: No sir which we have one, we have got hireling to clustering.

Student: Yes sir.

So that game is

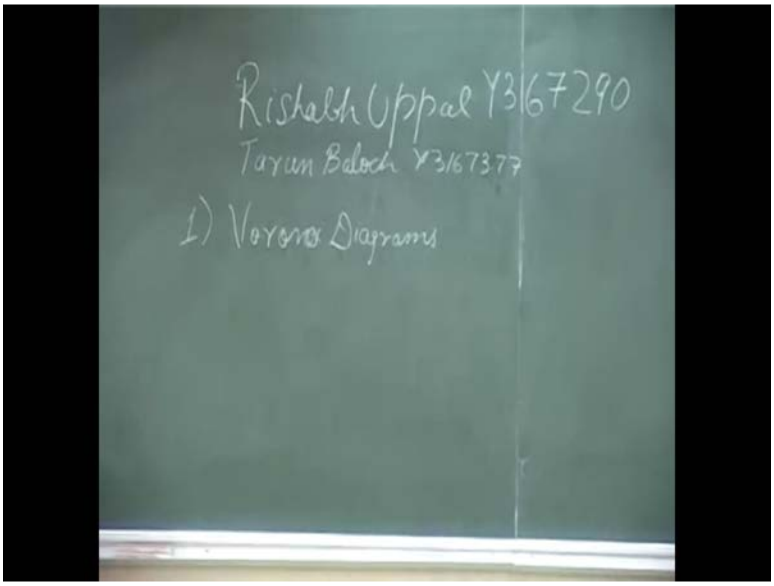
Student: So that is also exists.

Now, in that case you use one and implement clustering.

Student: Eighteenth or thirteenth?

Eighteenth say eighteenth I am keeping as the resulting.

(Refer Slide Time: 1:11:00)




Student: Sir, actually I started like we should start. So, actually we have thought of doing survey on the various kinds of parallel algorithms on the corollary diagram. Basically various kinds of works have been done already in the area like, in that we have various models like in the hypercube model. We have an implementation which has log q of complexity and they some c r e brand model which has a time complexity of which is cost optimized as well with the sequential algorithm. So, we were thought of doing some survey or if the time commits, we can implement that on the c c purpose.

Parallel complexional geometry book.

So, most of basic problems is where do we computational geometry convex hull and diagram(( ))

Student: We can basically generalize this thing on actually sir we found a lot of application of these corollary diagram.

Now, can you split from that point of view, the application point of view?

Student: Actually sir, it has many applications in the as well as like in the windows application, like if our curser is somewhere and from the corollary diagram, we can find like which I can is closest one. So, it points to that as well as like in a motion movie we have to find some good still image, if you have to capture that but, like as we like capture them image from movie it is not like it is.

Student: So, using the corollary diagrams we can capture like a from which image suits the best, which is the like, which has the move still image to make closest to the like to get a perfect image of the whole sequence using like that.

So, are you going to mention in your all those things something you know. So, that it becomes compete one, otherwise only one another you see the thermal computation geometry book copy.

Student: Sir we can try to implement it is.

Student: We will try to implement, then I think it would much better to maintain either sir we will decide about, which models we want to implement is it fine sir.

That's while know as thickness is there. Only you know after reading you're coming directly from computational geometry.

Student: No sir we do not (( )).

Student: Also available.

No that is but, you introduce some application also (( )).

Student: Yes sir.