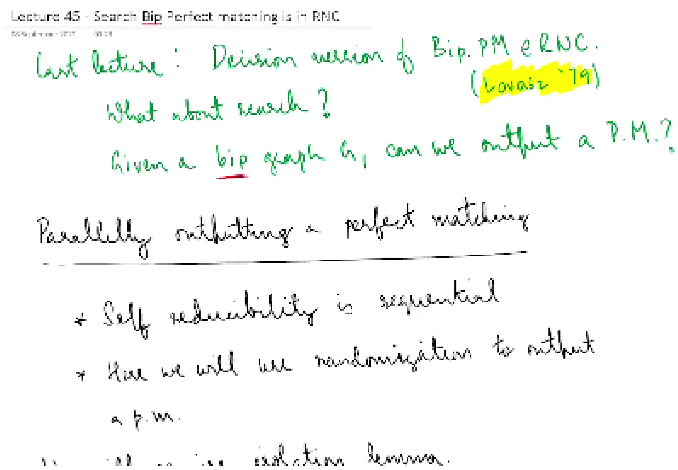
Computational Complexity
Prof. Subrahamanyam Kalyanasundaram
Department of Computer Science and Engineering
Indian Institute of Technology, Hyderabad

Lecture - 45
Search Bipartite Perfect Matching is in RNC Part 1

(Refer Slide Time: 00:15)



Hello and welcome to lecture 45 of the course computational complexity. In the last lecture we saw given a bipartite graph G how to decide if the graph G has a perfect matching. And we did that in randomized NC which is a randomized equivalent of parallel computing and NC is also we saw the circuit characterization of NC. In this lecture we will see how to output the perfect matching.

So, there is a decision version which says whether yes or no whether this graph is a perfect matching or no. And now we are going to see how in the same class in the same randomized NC in the parallel computation class we can output the perfect matching. One thing that I want to say here is that there are other algorithms in the serial world to decide and even output a perfect matching for both bipartite as well as general graphs.

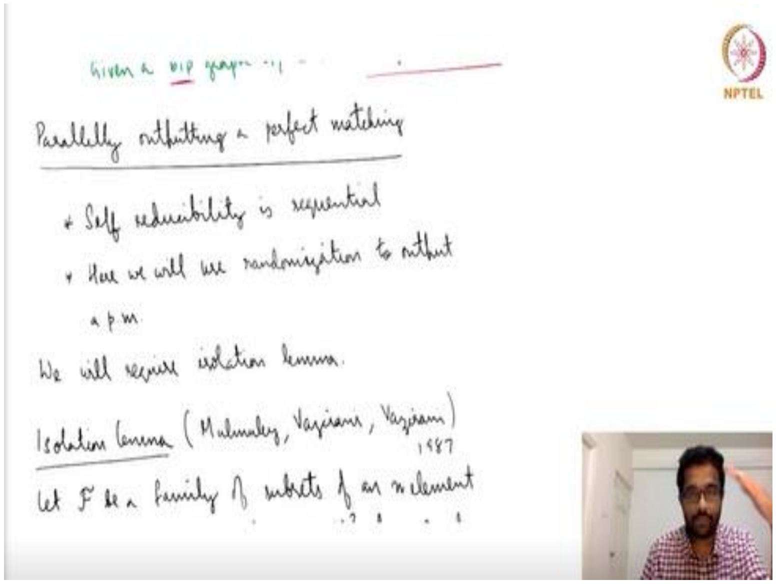
But the speciality of this algorithm is that, this is doing the same thing in a parallel setting. And, another thing that I did not mention last time is that the algorithm that I described in the previous lecture was by Lovasz in 1979. So, in today's lecture we will see how to output the perfect matching. So, given a bipartite graph G can be output a perfect matching, so it is a bipartite graph still and we want to output a perfect matching.

So, one technique that we have seen, once in the proof of Karp Lipton theorem, is the self-reducibility. So, you could do the following for instance. So, this is what we did in the case of Karp Lipton theorem as well. We could take the graph and check whether there is a perfect matching. If it does not have a perfect matching there is no need to proceed further. If it has a perfect matching, then you take it you pick one edge from the graph.

Remove it and check whether the remaining graph is a perfect matching. If the remaining graph has a perfect matching, then fine you can continue. If the remaining graph does not have a perfect matching that means the removed edge was essential for the graph to have a perfect matching. So, that means that edge head was there in all the perfect matching, so you retain that edge and then compute the perfect matching for the rest of the graph.

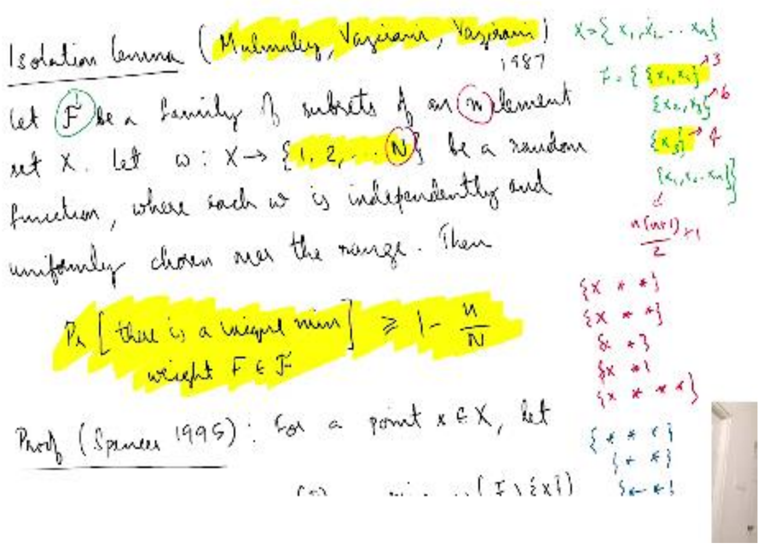
And then you can do this recursively. The problem with this approach is that this is a very sequential approach. It is inherently sequential. So, this is by the self-reducibility and this is a very sequential approach. And this if the graph is of size if it has N vertices then it will require N by 2 iterations of this thing. So, this is certainly not something that can be simulated by a login or polylog and depth circuit.

(Refer Slide Time: 03:17)



The depth of this will be at least order of N, if not more. So, here we will see a completely different approach. One which is very clever also uses some elements of the characterization that we saw last week. So, and randomization will again be a key ingredient.

(Refer Slide Time: 03:41)



And this approach that we are seeing today was by Mulmuley, Vazirani, Vazirani from 1987. So, for that they first proved what is now known as isolation lemma. And this is a fact that is in general about choosing random values and weights for a set and this also has found applications in other outside the computation of perfect matching. So, this is something that has been studied that can be studied even independently. And so, I will shade it separately.

And we will see the proof of this first before going into bipartite graphs and perfect matching. Suppose there is an n element set x and let F is in the script F. So, be a family of subsets of an n element x, so let us say this set contains x 1, x 2, x n, and maybe F contains some subsets let us such as x 1, x 2 maybe subset maybe x 2, x 3 may be a subset, maybe just one alone is a subset maybe all of it x 1, x 2 up to x n is a subset. So, this is F, it is a collection of subsets of x.

So, this is x. And, now we want to find, so let us see what and then what we want to say now you let us say there are weights assigned to each of these elements. Let us say x 1 is assigned away w 1, x 2 is assigned away w 2. So, let us say x 1 is assigned weight 1. This is just for easy simplicity let us say x 1 is assigned weight 1 x 2 is assigned weight 2. And, so on up to x n is assigned weight 1 n.

In this case the set x 1 x 2 has weight 1 + 2 which is 3 x 2 x 3 has 2 + 3 which is 5 x 1 alone has 1 x 1 to x n will have n into n + 1 by 2. Just for the sake of interesting example let me say this is x 3 and, in that case, this set has weight 3. Now you can see that what is this set in the family script F, with the smallest weight. In fact, there is no one set with the smallest weight both x 1 x 2 and x 3 both have weight 3. And weight 3 is the smallest weight in this family.

So, the goal is to assign weights in such a way that there is one unique smallest weight set in F. So, the current assignment has not created one unique smallest weight set, so the setting is that if the sets are already determined now is there a way to assign weights such that there is a unique smallest weight set. So, this particular assignment of weights does not work for this family. So, now suppose we had given x 3 the weight 4 then everything else being the same.

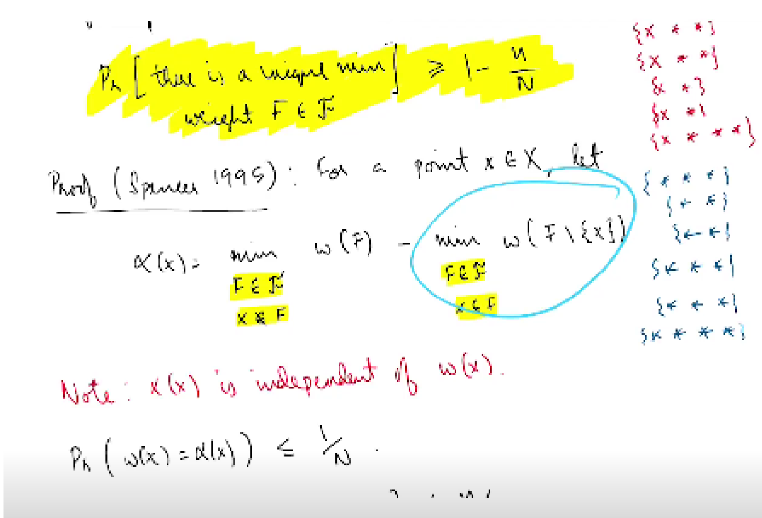
Then this would have been 6, this would have been 4 and this would have been plus 1 here. So, in that case now in we had a unique minimum wait set which is the first set. So, the goal is to given a fixed family of subsets F can we assign weights in such a way that there is a unique minimum weight set. You should not have two sets with the same weight and them both being the minimum.

So, you could have two sets be the same weight as long as they are not the minimum. The minimum one should be only one set; it should be unique. What this is saying is that? There is a reason reasonably chosen random assignment random weight assignment which will ensure that with good probability there is a unique minimum weight set. So, that is the isolation lemma. So, suppose you just pick a random assignment from the x to 1 2 up to n.

So, notice that the number of elements is small n and x is chosen. The weights are chosen from 1 to capital N. So, these two are different, capital N is different from small n. Each element of the x is randomly assigned a value from the set 1 to capital N. Random independent and uniform. Meaning an element of x is chosen given the weight 1 with probability 1 divided by capital N, 2 with probability 1 divided by capital N and so on. And they are all assigned independently.

Then, the probability is such that there is a unique minimum weight set in the family capital F is at least 1 - n divided by capital N. That is the statement of the lemma. If you assign it in this fashion then the probability of there being a unique minimum weight set is at least 1- small n divided by capital N this is the main statement of the isolation lemma. And the proof is very interesting and once again this proof has nothing to do with graphs or bipartite matchings or anything. It is just random assignment and computational probability. So, let us see why this is true?

(Refer Slide Time: 09:50)



And this proof is by spencer this is not the proof in the original Mulmuley, Vazirani, Vazirani paper. So, we defined something called alpha X for each x, for each small x. Where small x is an element of capital X. And the weight function is called W. So, we define something called alpha x. What is alpha X? You look at all the sets F in the family script F. You look at all the sets F in the family script F such that X is not there in F.

So, you look at all the sets that do not have x as an element where x is the argument of alpha. Now and look at the minimum weight such set. You look at the minimum weight such set. And then you look at all the sets F that have x in them and then you look at the minimum weight of that set but after having removed x. So, maybe let us say these are some sets that do not have x. I am just giving a very pictorial representation. So, you compute the weight of all of this.

These are all elements. But, none of them is x. So, this looks like an x but these are all asterisks. And then you look at the sets that have x x star, star. So, maybe x star star. This is all be x star, something. And then you for the red things correspond to the second term and the blue things correspond to the first term and then the second term we exclude x and look at the remaining sum.

In the first you just consider all the sets that have x that do not have x in them for the first term. So, this is what you do, maybe I will just move this around. And you take the difference. So, notice this the first term does not really, the first term that is the one that I have circled here. This does not depend on the weight of x because, all the sets that are considered do not have x. So, this first term does not depend on the weight of x.

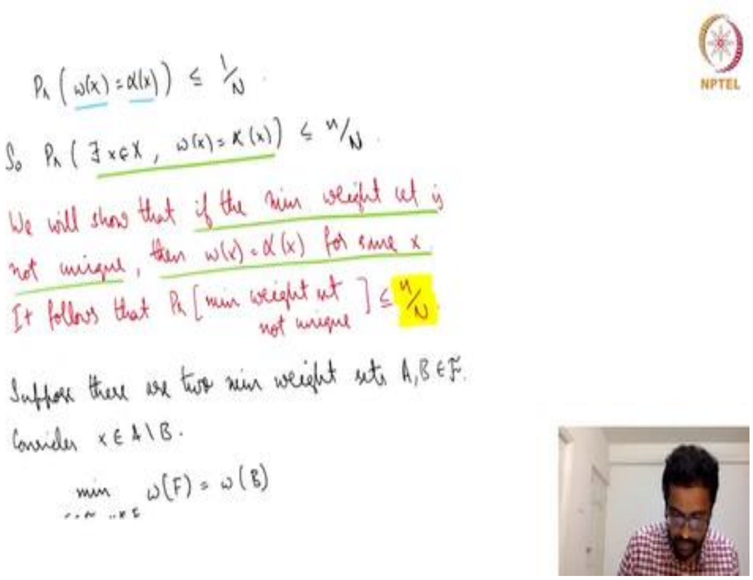
And what you can notice is that even the second term this also does not depend on the weight of x because, we are choosing all the sets with x, but we compute the weight after having removed x. So, even the second set, second term, we do not is independent of the weight of x. It is dependent on x because x determines which way the sets are partitioned into the first and second. But it is independent of the weight of x.

So, alpha is the difference between the first and second terms and both of these terms are independent of the weight of x. And notice that, so what is the probability that the weight of the element x is equal to alpha X. So, notice that alpha once the weights of the remaining elements are fixed x is one element so if all the other elements of capital X are fixed. Then, alpha X is fixed what is the probability that the weight of small x is equal to the fixed value alpha X?

So, the weight of x could be any one of the values from 1 to capital N. Now, what is the probability that you pick the value that is equal to alpha X. So, whatever alpha x is? If it is in the range 1 to N there is one possibility out of n capital N possibilities. So, the probability of alpha W X the weight of x being equal to alpha X is at most one by N. If it is out of this range then no problem.

So, for instance if alpha X is negative, then there is nothing to then the probability will be 0, but anyway I am just writing upper bound 1 by N.

(Refer Slide Time: 15:00)



Now, so for a specific x the probability that alpha X is equal to the weight of x is at most 1 by N. What is the probability that? There is some element of capital X such that for some element x of capital X such that for that x, that small x the weight of x = alpha X. That is what we write in the second line what is the probability that there exists a small x in the whole set such that for that small x the weight is equal to alpha.

So, there are n possibilities, that small n possibilities, so for the first element this could be same, second element this could be the same and so on. So, there are n elements smaller elements, so the probabilities at most the probabilities at most of the union. The union of the probabilities at most sum of the individual probabilities it could be smaller, because they could overlap. But this is a union bound it is 1 n divided by capital N.

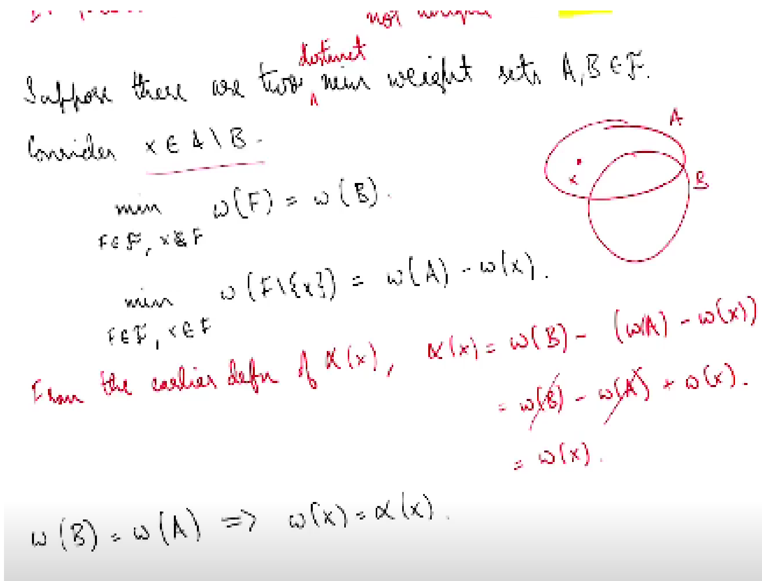
Small n divided by capital N. And this is the same small n divided by capital N, that also features in the probability in the statement. Now all that remains is that to be shown is that, whenever so the claim is that whenever there is a unique or the probability of a unique minimum weight set F in script F is at least 1 - n by N. Now, we are saying that the probability that the minimum weight set is not unique or rather, the minimum weight set is not unique.

If the minimum wage set is not unique then w x = alpha X for some x there has to be some small x for which w x = alpha. So, which means the probability of the minimum weight set not being unique, is so whenever the minimum heat set is not unique then for some x w x = alpha X. And the probability of that happening is at most n divided by capital N. So, the probability of minimum weight set not being unique is upper bounded by n divided by capital N.

Because, whenever the event that minimum wage set is not unique happens the another event happens that is that for some x alpha X = w x. And the probability of this even second event happening is at most n by N. So, the probability of first event happening is also at most n by N. So, the complement if you take the complement the probability the minimum wage set is unique is at least 1 - n by N. So, that is immediate that is how you get the statement.

All that remains to be shown is the statement I have underlined here in green, whenever the minimum wage set is not unique, then the there is some element small x for which the weight is equal to alpha X. So, now all that we will show is that statement.

(Refer Slide Time: 18:20)



Suppose, there are two minimum weight sets. Two distinct minimum weight sets A and B in script F. Because they are distinct either so there is an x that is in A but not in B. Of course, A could be a subset of b in which case you just take an x that is in B but not in A and you just change the role of A and B. It cannot be the case that there is no x that is in A but not in B and there is no x that is in B but not A if that will happen means the sets are the same.

So, consider x that is in A but not in B. Now for this x we will show that alpha X = weight x let us see why. Now consider the minimum weight set F in script F that does not contain x. We know that the minimum weight sets are A and B, not A and B are two minimum weight sets. There could be more. Now out of which we know that x is not there in B. So, x is in A but not in B. So, x is in we do not know if it is a content set so suppose A is like this B is like this.

It cannot be a strict subset because a strict subset means B will have a smaller weight. So, x is here. So, the first term you are picking the smallest F for which the weight is the smallest and x is not in that set. So, B will feature in that summation or not summation B will feature in that minimum. So, one of the sets for which this will be minimized this B so, w B. And now you look at all the, now you look at the second term.

So, we are now looking at the second term of this alpha X computation. Consider all the weight of the sets F that have x in them but after having removed x from them. So, there could be multiple sets that contain x and B minimum. But we know that A is one of the minimum weight sets that have x in them. We know B is not one of them, but we know that A is a minimum weight set that have x in them. There could be other minimum weight sets that have x in them.

And the weight of x is same for everything, whichever A or C or whatever. So, this quantity the weight of F - x this is equal to weight of A - weight of x. And what is alpha? Alpha is the first term first quantity minus the second quantity. So, it is double alpha is nothing but w B minus so maybe I just write here from the earlier definition maybe I will just make some space. Sorry it is taking some time, from the earlier definition of alpha X, alpha X is nothing but w B.

It is the first quantity minus the second point minus w A - w x. And which is nothing but w B - w A + w x. And by assumption w A and w B are two distinct minima sets. So, they are both the minimum so which means these two are equal these two will cancel w B and w A. so which means alpha X = w x, which is exactly what we wanted to show. So, we said that if the minimum weight set is not unique if there are two distinct minimum weight sets.

Then there is an element x for which alpha and w are the same and that is it. So, this is the isolation lemma. So, again just this I will restate the statement suppose x is a set containing n elements and then we are assigning weights to these n elements. And F be a family of subsets script F be a family of subsets of F x. Now if you choose weights if you assign weights randomly to the elements of x.

So, the weights are same, meaning the weights are just assigned to the elements of x. And then you compute the weights of the sets in script F, as the sum of the weights of these elements, in that case if your weights are assigned randomly. Then, the probability that there is a unique minimum weight set F in script F is at least 1 - n by N. Where small n is the number of elements in x and capital is x, capital N is the range from which you choose weights.

So, the if you want to improve the probability of getting a unique minimum weight set you just increase the range from capital N you can double capital N or something. So, this is the isolation lemma. So, perhaps I will continue with assuming the isolation lemma in the next lecture. Thank you.