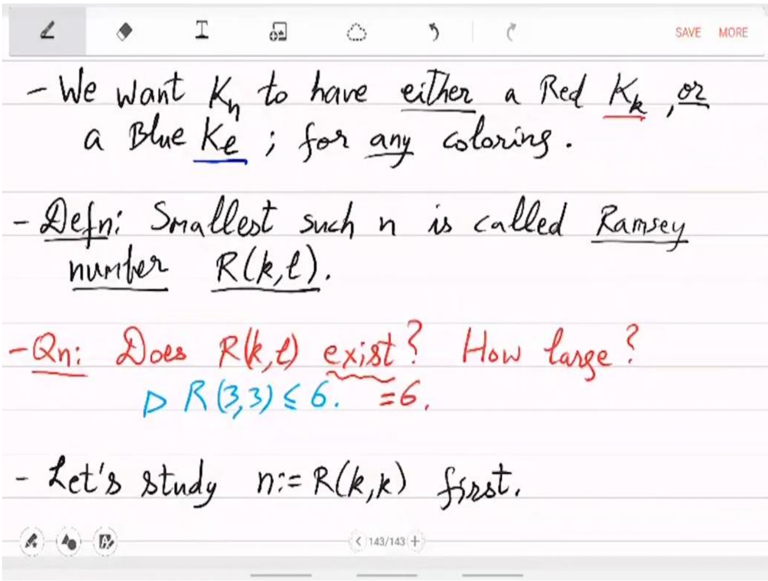
Probability for Computer Science
Prof. Nitin Saxena
Department of Computer Science and Engineering
Indian Institute of Technology - Kanpur

Module - 7
Lecture - 27
Ramsey Numbers, Large Cuts in Graphs

Last time we started probabilistic methods. First example we took is Ramsey number. So, you saw that in the complete graph on 6 vertices, no matter how you colour the edges, red or blue, there is always a monochromatic triangle. So, let us now generalise this in a far reaching way, using probability and other tricks.

(Refer Slide Time: 00:40)

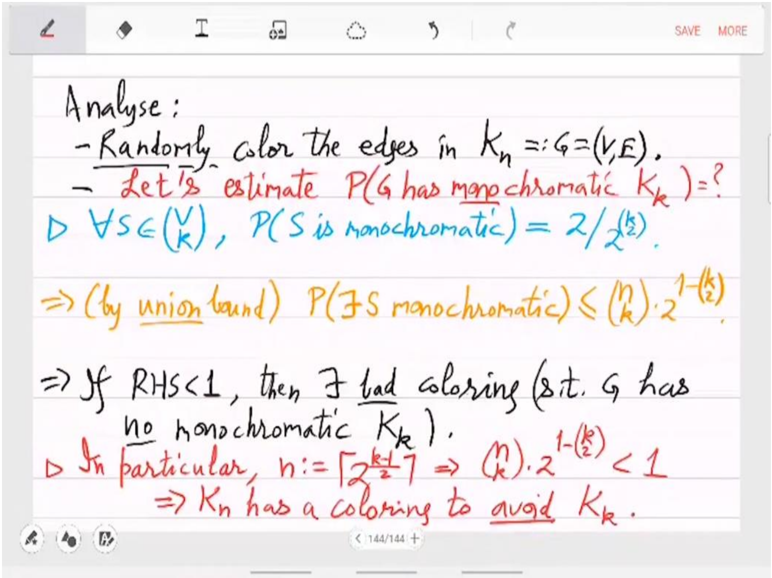


So, we want K n to have either a red, let us say this complete subgraph on k vertices, small k vertices, or a blue K l, for any colouring. This is our goal. So, essentially, this is a question on given k, small k and l; what n to pick? So that, no matter how you colour the edges, red or blue, you will get either this coloured complete graph or this coloured complete graph. So, we define, smallest such n is called Ramsey number R k, l.

So, Ramsey number is the smallest n which is a function of k and l of course, such that, no matter how you colour the edges, there is always a monochromatic complete subgraph of the respective size, either small k or l. So, obviously, first question that arises here is why should such a number exist? So, does k, l exist? Even existence is not clear. And how large? If it exists, how big is it, as a function of small k and l?

And we have just shown that R 3 3 is less than equal to 6. You can actually show that it is exactly 6, but we have at least shown before that 6 vertices are enough to get monochromatic triangle. So, in fact, this is equal to 6. So, keeping that in mind, let us study R k k first, where small k and l, we are taking them equal. So, let us first understand this, what is happening.

(Refer Slide Time: 04:04)



So, we will first actually prove a lower bound using probability, and then we will show an upper bound using averaging argument like before. So, let us randomly colour the edges in K n. Let us call this graph G, which is V, E, vertices V, edges E. So, vertices are 1 to n, and edges are all possible. Obviously, we do not know whether this n exists for R k k, but suppose it exists, then look at the graph G, randomly colour it with red and blue colours, the edges.

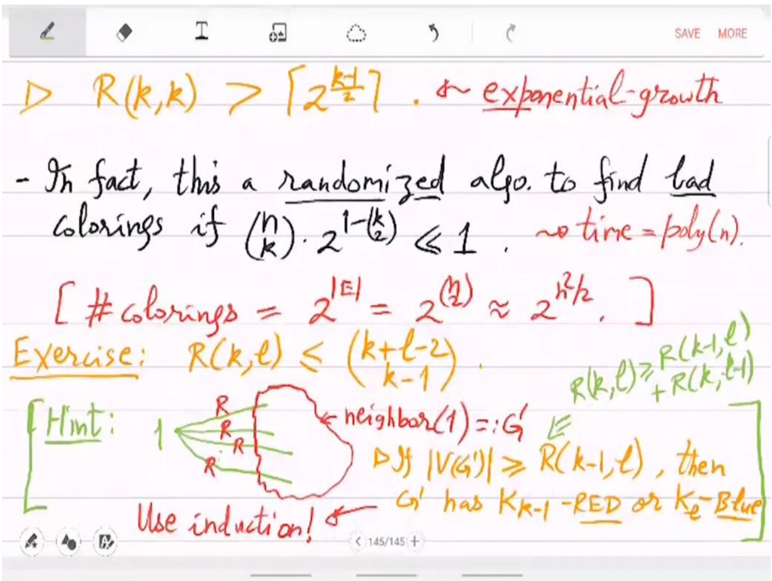
And let us estimate the probability that G has a monochromatic subgraph k, subgraph of size small k. What is the chance of there being a monochromatic subgraph? So, first observation is, any subset of size small k of the vertices, the probability that this S, particular fixed S is monochromatic is how much? So, it can be either red or blue; so, that is 2, and 2 times. What is the chance that all these edges are red? So, how many edges are there?

Number of edges, k choose 2. So, that is, each edge can be red or blue; so, it is 2 raised to k choose 2. So, this is the probability. This is actually, as k grows, this is a very small probability that this particular subset is monochromatic, but you should not be misled by this little probability, because the number of subsets is large. So, overall, what is the chance that some subset is monochromatic? So, let us analyse that.

So, this implies by the union bound that probability that there exists an S which is monochromatic; so, this is just summing over all the subsets; so, which gives you n choose k times 2 raised to 1 - k choose 2. This is the probability that there is some monochromatic subset of size small k, red or blue, complete subgraph. So, this means that, if the RHS expression is less than 1, then there exists a bad colouring such that G has no monochromatic complete subgraph of size small k.

So, this probability is less than 1, then actually there is a positive chance. So, less than 1 means, the opposite probability is positive. So, there is a positive chance that for all the subsets, the colouring makes each of them non-monochromatic. So, in particular, if you take n to be 2 raised to k - 1 by 2, you take the ceiling of this. So, if n is this much, then you can see that n choose k 2 raised to 1 - k chose 2 is less than 1, which means that; so, K n has a colouring to avoid a complete subgraph of size small k that is monochromatic. So, which means that, if this n exists, it has to be bigger than this exponential bound, it has to be quite large.

(Refer Slide Time: 10:06)



So, which means that R k k has to be bigger than 2 raised to k - 1 by 2. So, this still does not prove that R k k exists, but if it exists, it is quite large; this is exponentially large. So, in other words, what we have shown is that Ramsey numbers are very large. You need actually very large graphs, so that any 2 colouring gives you a complete monochromatic subgraph that is complete.

In fact, what this argument shows, this is a randomised algorithm to find bad colourings. If this bound n choose k 2 raised to 1 minus k choose 2 is smaller than 1. If it is sufficiently smaller than 1, then, what it is saying is that, when you pick a random colouring, it will be bad. So, this actually is a probabilistic algorithm, fast algorithm, which is an interesting thing; I mean, you are actually looking for the bad colouring in a huge space; the number of possible colourings is like 2 raised to n square.

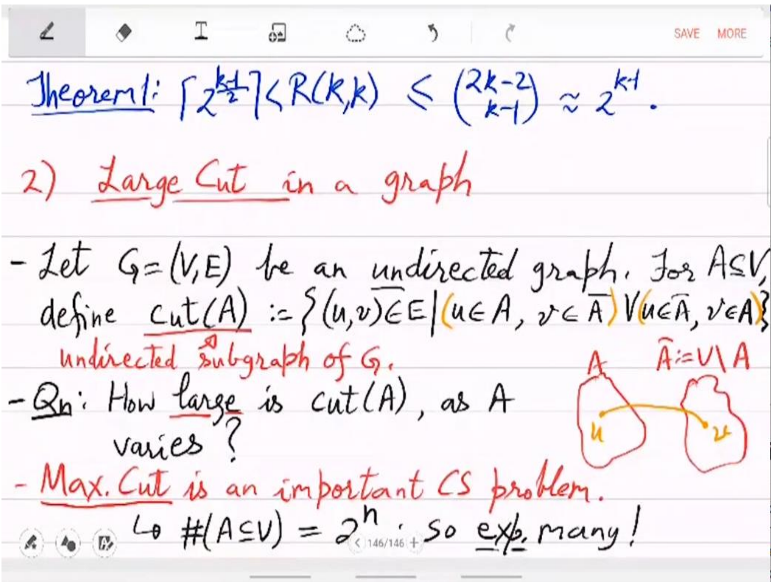
So, number of colourings is around 2 raised to number of edges, which is 2 raised to n choose 2, which is like 2 raised to n square by 2. So, this is a very big space, but the time that this algorithm is taking is only polynomial in n. So, this is a very fast algorithm compared to the space that it is looking at. So, I leave you with this exercise that R k, l indeed exists and it is actually smaller than k + l choose k, kind of this, k + l - 2 choose k - 1.

So, it is this binomial number. So, hint is, you basically attempt in a similar way as you did in case 6, you used pigeonhole principle or averaging principle to look at the neighbours of first vertex. So, you look at vertex 1; and these are the neighbours. These are all red edges. To see the neighbours of 1 via red edges, they are more than blue edges; so, you get this neighbourhood of 1. And let us call this graph G prime.

Now, what happens here is, if you show that the graph G prime is large, either you want k - 1 monochromatic; I mean, k - 1 complete subgraph of colour red, having k - 1 vertices, then you can use these red edges to get a complete subgraph of size k. So, if the vertices are R k - 1 l in this, then G prime has either k - 1 red colour or k l blue. So, if you find k l blue, then you are done. If you find k - 1 red, complete subgraph monochromatic, then you can use the red edges.

So, basically, you use induction. So, use induction here; that is the full hint. So, using this idea, you can actually look at the neighbours of 1 and then accordingly show that R k, l = k + l - 2 choose k - 1 actually works. And this is in fact implied by R k l greater than equal to R k - 1 l + R k l - 1. This is the hint for the induction. So, if you take a function R k l which satisfies this property, then you can either look at the red neighbours of 1 or the blue neighbours of 1; one of them will satisfy, will help you go 1 step down in the induction argument. So, what have we learnt then?

(Refer Slide Time: 17:02)



So, let us now join all these conclusions, combine these conclusions to deduce theorem 1, which says that R k, k, it has to be at least 2 raised to k - 1 by 2 ceiling, which we showed by probabilistic argument. And it does not exceed 2k - 2 k - 1 by using this R k, l formula or upper bound. So, this is around 2 raised to k - 1. So, you have a pretty good idea of what this R k, k Ramsey number is.

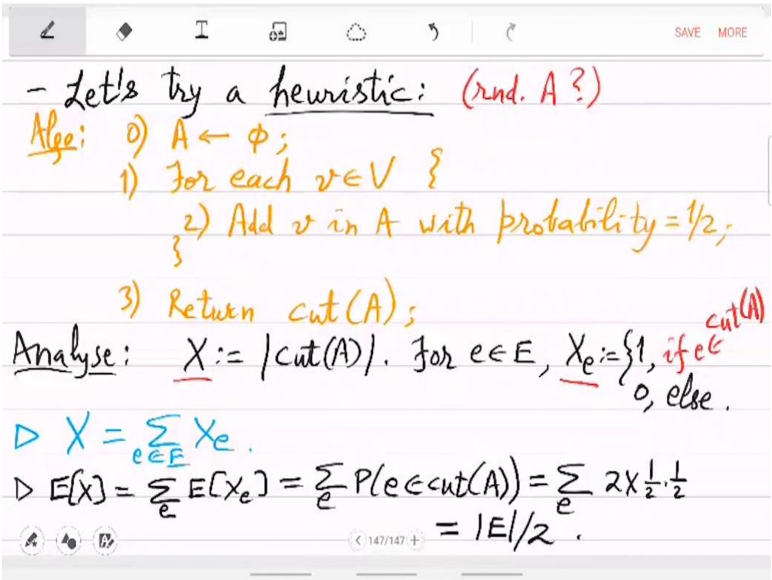
It is at least square root 2 raised to k - 1, and it is at most 2 raised to k - 1. So, that is a beautiful way to get these 2 bounds, and a good understanding of R k, k. So, that is the first example. Let us move to the second example of probabilistic methods, which will be cuts in a graph. In the first example, we looked at colouring in a graph; now we look at some other property called cut in a graph. This is again to do with the edges. So, what is this?

Let G be an undirected graph. So, for a subset A of V, define cut introduced by A, subset A of vertices to be essentially the edges which go out of A to A compliment. So, the edges that go from A to A compliment; let me draw this for clarity. This is A; this is A complement. So, V - A. And this is u here, and this is v. So, this is called a crossing edge. So, you want to collect the crossing edges to define the cut.

So, I will also need the other direction. So, either u in A, v in A bar; or u in A bar, v in A; those are the cases. So, now, cut of A is also an undirected subgraph of G. So, u will be given in the input G, A is not really known; you want to find in A such that the cut is maximised; it is the max cut problem. That is the question. So, how large is cut, as A varies. So, you want the largest, the max cut. So, max cut is an important computer science problem.

So, you want to maximise the cut over subset A of vertices. Now, remember that number of A is just too much, this is 2 raised to n. So, number of subsets A of V is 2 raised to n. So, this is exponentially many. So, how do you search in this space for the best A possible, with the largest cut. In fact, a priori it is not even clear a theoretical lower bound on the cut. It seems to depend too much on the input graph. So, now we will prove an interesting lower bound and also give an probabilistic algorithm. So, let us try a heuristic first.

(Refer Slide Time: 23:01)



So, instead of trying to analyse it very properly, let us just jump in and say that we will pick these edges of the cut, or we will pick the subset A in a random way. We will look at a vertex u and decide whether to put it in A or A bar. So, first step is, you initialise set A to be empty; it has nothing. For each vertex, what you do is, add v in A with probability half. This is a trivial step; you can achieve it by just tossing a fair coin.

So, if the coin gives you head, you decide to put vertex v in A. If it is tails, then you do not put it in A; you have basically put it in A compliment in a way. And just do this for all the vertices and finally return the cut. So, that is the heuristic. This is the simplest probabilistic algorithm you can think of for this problem; just picking A randomly. That is really the idea here. Now, is it any good?

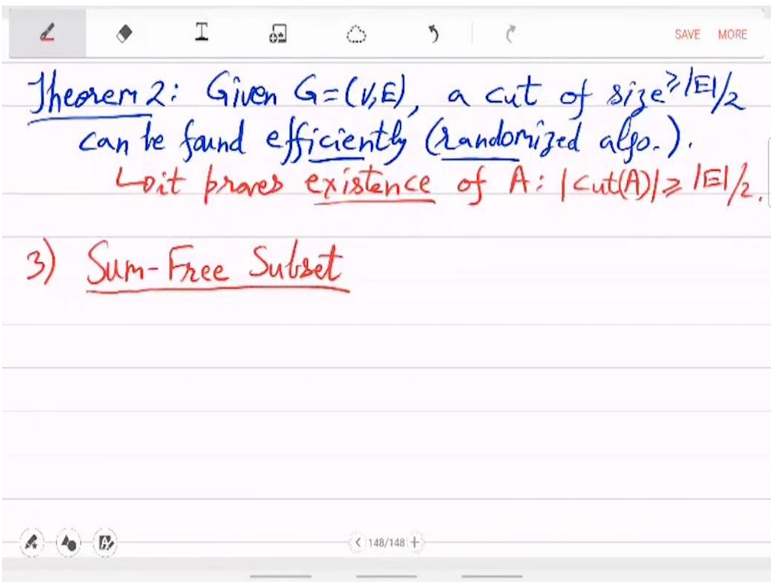
Is this heuristic algorithm giving us new knowledge about the cut and how big a cut? So, let us analyse defining the main random variable here we are interested in, which is the cut size. And for edges in the graph; so, given graph is v, e, right? Edges are e. So, for every edge, define a random variable which is 1 if e is in the cut, and it is 0 otherwise. So, it simply indicates whether this edge was chosen.

Well not chosen; so, the n points of the edge e. One of them was chosen in e, and the other was not. So, that indication is given by the random variable X e. And then, the sum of this, these X e's is the size of the cut. That is another random variable which is sum. So, let us remember that. So, X is equal to sum of X e over all the edges. And the other thing is, if you look at its expectation, this is the main thing. What is this cut expected to be?

So, we look at the expectation over these coin tosses that we did. So, expectation of X e is basically probability that; since it is an indicator variable 1 0, it is just probability whether e was put in the cut. Now, e has obviously 2 vertices, 2 different vertices; and independently, one was put in A and the other was put in A compliment. So, there are 2 possibilities. This event has basically partitions into 2 events, and so, say e = u, v; u goes in A, v does not; so, that is half times half probability.

And it is the same for every e. This is happening by linearity of expectation, it does not require any assumption. So, you get in the end, number of edges by 2, which is amazing. So, you know that this heuristic algorithm which is just being random A, actually gives you a cut half the size, half the number of edges. So, this is not at all clear when we define cut that such a thing exists. So, that is our theorem 2.

(Refer Slide Time: 28:42)



Given G, a cut of size half the number of edges at least can be found efficiently. That is a randomised algorithm, and it is very fast. So, although the number of subsets is 2 raised to n, your algorithm is not doing that. It is actually polynomial in n. And it is very interesting. So, this is an example of probabilistic method. It is actually telling you existence of A. So, thus, it proves the existence of A such that the cut is at least half the number of edges.

And it is proving this apparently without using any graph theory. This is a very simple heuristic. So, this brings us to the end of second application. And the third application we will do is about sum-free subsets.