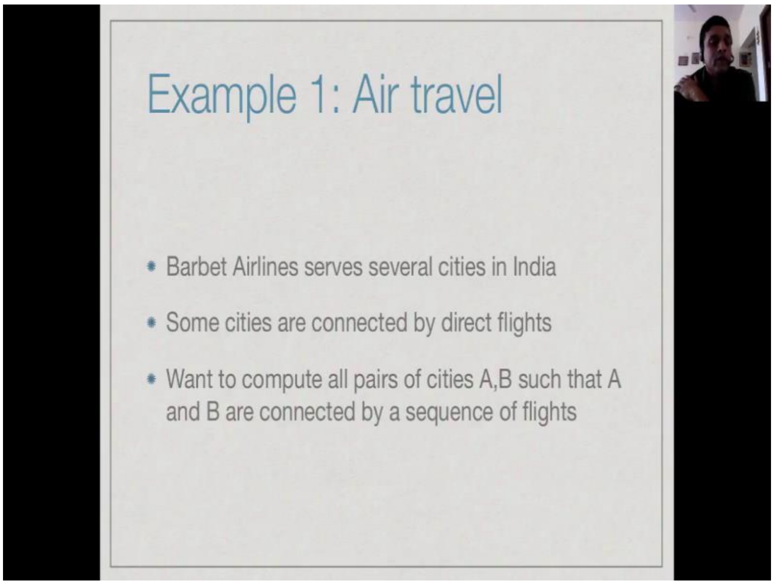
Design and Analysis of Algorithms
Prof. Madhavan Mukund
Chennai Mathematical Institute

Week - 01
Module - 02
Lecture - 02

So, before we start the formal part of the course, I would like to discuss a few examples to motivate the kind of problems we will be looking at.

(Refer Slide Time: 00:10)



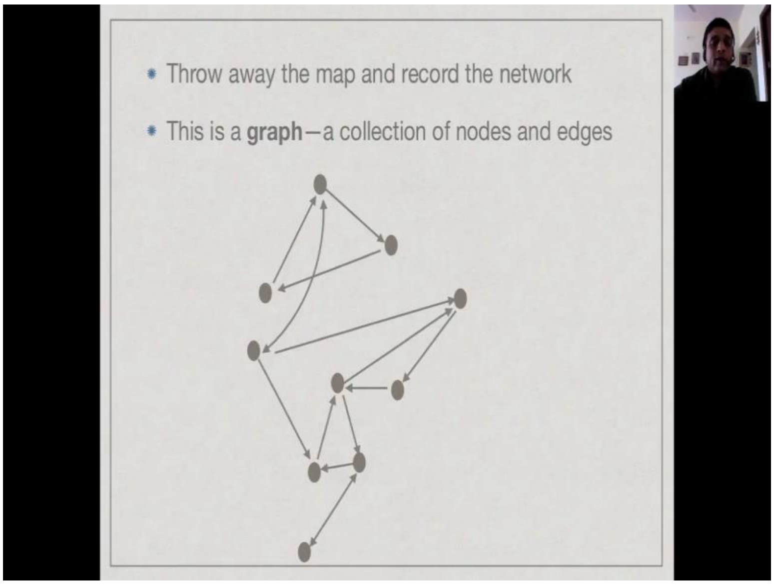
So, we start with a problem of air travel. So, we have an airline; Barbet airlines, which serves several cities in the country. And of course, although it serves several cities, it does not really connect all these cities directly. Only some of the cities are connected by direct flights. And for other pair of cities you have to take a hopping flight. You have to go via an intermediate city. So, our first goal maybe to compute every pair of cities, which are actually connected by this network served by this airline. So, how do we find out all pairs of cities A, B? Such that A and B are connected by a sequence of flights.

(Refer Slide Time: 00:50)

So, first we need to look at the network. So, this is a typical way that we might find the network. For example, if we open the in-flight magazine of an airline, you find a route map. And it is written like this. You have a physical map of the country and you have the cities which are served; marked out. There are about ten cities; Delhi, Varanasi, Ahmedabad, down to Trivandrum in the south. And you have some arrows indicating the flights. Now, some of these flights are in one direction. You can go from Delhi to Varanasi, but you cannot come back directly to Delhi. You must go to Ahmedabad and then come back. So, this is quite common.

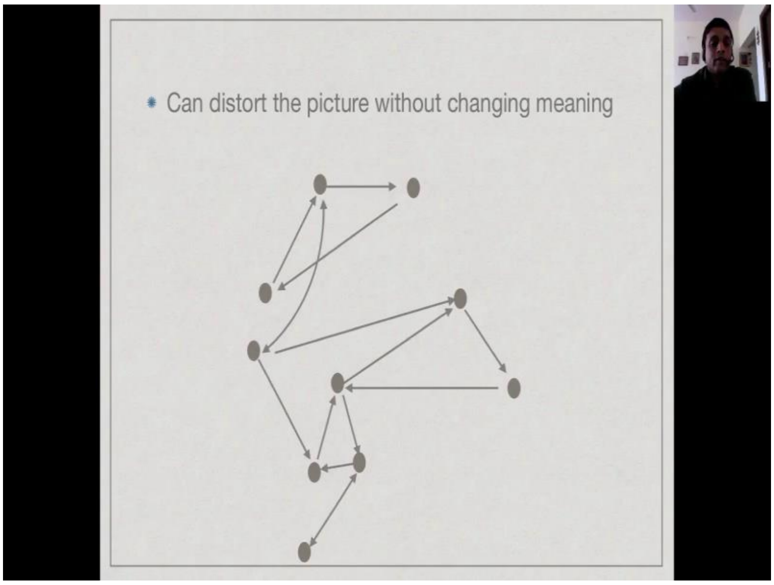
If you look at airline's schedules in airlines, you will find that there were these kind of triangular routes. Where you go around a triangle and you cannot go back directly without hopping in between. Some pairs of important cities; like in this case Mumbai and Delhi might be connected by flights in both directions or even Mumbai and Calcutta. And so now we have these ten cities. We want to know is it really possible go from say Varanasi to Trivandrum or is it not possible, is it possible to go from Hyderabad to Delhi or is not possible. So, our first step is to model this problem in such a way that we retain the essential details, which are relevant to solving the problem and get to do for the unnecessary details.

(Refer Slide Time: 02:03)



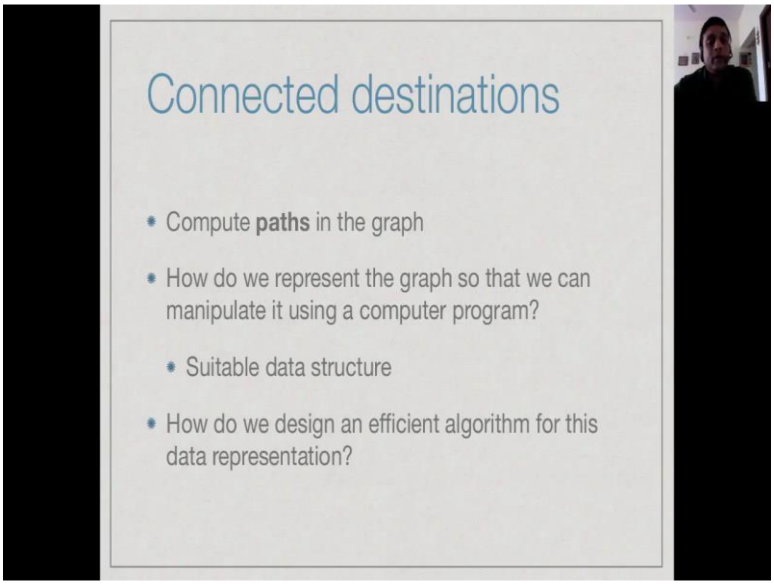
So, in this case what we really would like to know is the structure of this network. Right. So, the map itself is not relevant. We just need to know how many cities are there, which are on the network and how are they connected by the flights. So, the picture below which has these gray circles and arrows represents this network. And these cities are the gray circles and the flights are the arrows and the arrow heads indicates the direction. So, if there is an arrow head in one direction, it is a one directional flight; if there is an arrow head in both ends, its means it is a bidirectional flight. The actual names of the cities I am not so relevant. So, we can call them 1, 2, 3, 4, 5 or a, b, c, d, e or whatever and solve the problem. So, this kind of a picture is called graph. We will study the graphs more formally when we come to this module in our course. But a graph is just a picture of this kind. So it has some nodes, this dots and edges.

(Refer Slide Time: 03:00)



So, now one nice thing about moving to this abstract level is that the actual picture can be distorted without changing its meaning. So, we can move. For instance, if we look at this city here, right, we can move it to the right and it does not make any difference in terms of solving the problem. Or we could simplify the picture by moving, for instance, this edge on side, so that we get no crossing edges. And this is again the same picture. Though it looks quite different from this picture that we have started with, this is again the same network. Now, in some situations it is useful to realize that the graph that we have looks like this; that there are no crossing edges. Technically, such a graph is called a planar graph. It can be drawn on a flat piece of paper without any edges crossing. For planar graphs, we might have better algorithms and for arbitrary graphs. Now what do you want to do in such a graph?

(Refer Slide Time: 04:00)



So, in this case we want to compute what we call a path. That is the sequence of edges going from one city to another city; where of course the direction must be correct. So, you cannot go backwards, across and edge which is flying from A to B. You cannot take this flight B to A, unless there is another flight.

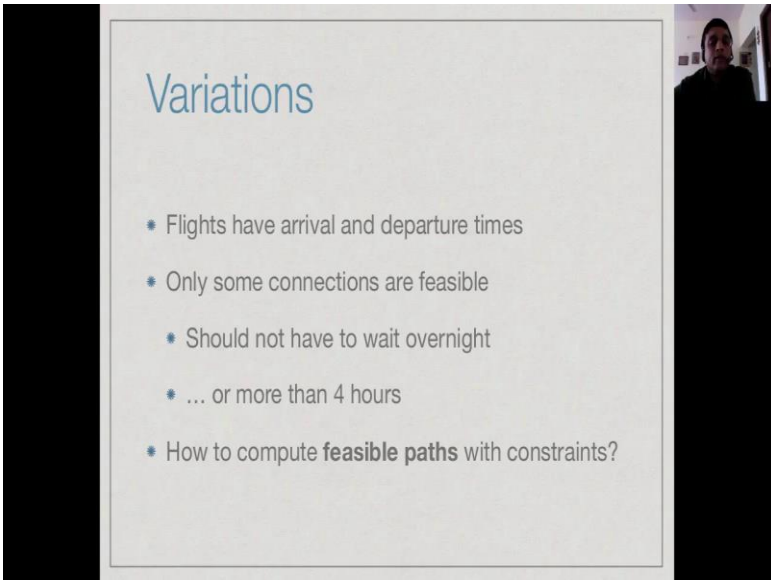
So, our first question is how do we take this picture and put it into form that we can manipulate using a program or an algorithm. So, we need a suitable data structure in order to represent this graph. Now given the way we represent the graph, we need to manipulate it to answer the question that we handle. In this case, connectivity. How do we go from A to B or can we go from A to B or which all cities B can I reach from A? So, how do we design such an algorithm, given the way we have represented these cities in the graph? Does it depends on the representation or there multiple representations, some of which gives us more or less efficient algorithm? These are all the questions that we need to answer before we can decide on whether we have got the best solution at act.

Now, in terms of efficiency we have to look at what are the things that determine the complexity of a problem. It is fairly obvious in this particular case that if we have more cities, the problem is more complicated. So the number of cities, which we can call N, is certainly one parameter which determines how complicated the algorithm is going to be, or not how complicated the algorithm is going to be, or rather how long it is going to take to run. The other question which determines how complex the network is this how many direct flights there are... Obviously there are fewer flights, there are fewer places, which can be connected and we have to explore fewer possibilities.

So from this, it follows that computing the paths depends on both N and F. So, we will not have an algorithm which will always takes a twenty steps. It will have to depend some number of steps depending on N and F. Now what is this dependency, how does it grow? If N doubles, does our algorithm take two times more times? Does it takes four times more times? If N term grows to factor of ten, does it takes ten times more or hundred times more time?

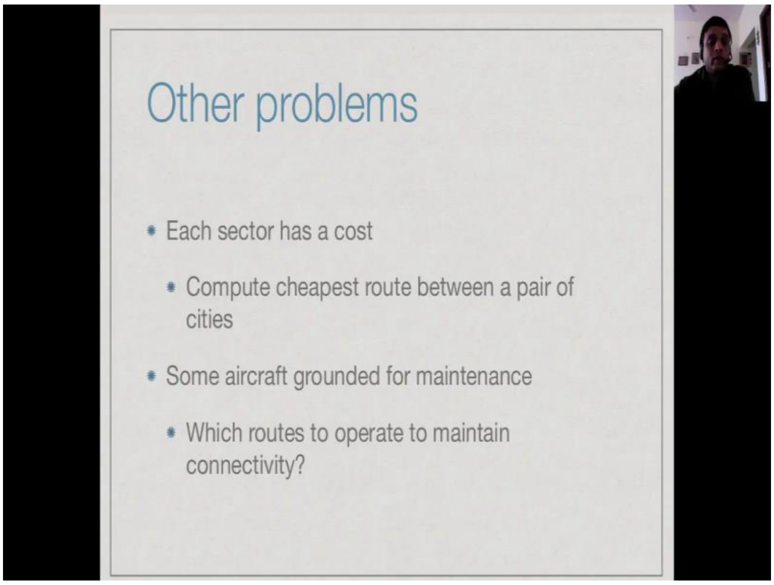
The other question related to this is given this dependency on N and F what realistic size of networks can we handle? If the airline grows to twenty flights, we will still have be able to give our answer in the reasonable time. Remember that this kind of an answer is typically required when somebody is making an online booking or something. And you will not allowed to reply in a few seconds, right, it is not enough to come back after an hour and say “yes, there is a flight from Trivandrum to Calcutta”. So, what is the limit of our efficiency? Can we scale this algorithm to cover airlines, multiple airlines? So, we have a website which actually sells. Across all airlines I can take you from place A to place B. That depends on how larger value of N and F we can handle.

(Refer Slide Time: 06:55)



And then of course the problem that we have looked at is a very simple problem; can I get from A to B. But very often it is not good enough to get from A to B. You want to get from A to B within some reasonable time frame. For instance, it is not usually acceptable to break journey overnight on aircraft. At the same time, you also do not want to spend more than the certain amount of time meeting in between flights. So, there are only some connections. Although there may be; erratically there may be the connections, only some of them may actually feasible. So, now our problems becomes little more constraint. So, we do not just want to look at the connected paths from A to B. But connected paths A to B, which meet some additional constraints in terms of timing and other things. So, can we solve this problem with the same approach that we solve a simpler problem or we need to take a radically different approach or do we need more information in order to decide or solve the problem.

(Refer Slide Time: 07:53)



Suppose, as you would you expect each sector on this thing has a cost. As a passenger, the cost would be the price of ticket. So, if you are trying to compute the best way to go from A to B, your motivation might be to choose the cheapest route in terms of the ticket cost. Of course cost is not only money, cost could be time as well. You might also want the quickest route from A to B, the one which involves the least waiting. So, it depends on what your priority is. Or you urgently required from where, in which case you do not mind paying more. Where, I am going for a vacation with my family; in which case, you have a relax time schedule, but you want to make sure you get a value from money.

from the airlines point of view there may be other questions. Periodically aircraft have to be brought down for a day for maintenance. Now, you do not want to have so many aircrafts that you keep all the routes flying and wastefully keep planes unused. At the same time if you keep two few planes, then when you bring an aircraft down for maintenance you have to sacrifice some routes. Now, which routes should we sacrifice? So that, you ensure that the connectivity of the network remains the same. If you could go earlier from Trivandrum to Calcutta, during a maintenance shutdown you should still be able to go from Trivandrum to Calcutta; maybe by different.

So, this is the problem to be addressed by the airlines stop; where as the cheapest route might be a problem to be addressed by the customers. So, there are very many different points of questions you can ask about this basic air network that we have described using a graph. And we will see answers to some of these problems in this course.