PROFESSOR: All right, so let's go ahead and proceed with vehicle scheduling. What we try to do in vehicle scheduling, and that's up to this point, we have frequently determined for different routes, identify to deliver the service that this specified on the timetable.

And then vehicle scheduling, which is the main part, is basically how we are using the existing fleet to deliver the service that is specified on the timetable, how we are basically defining the duties for vehicles from the time that they pull out from depot till the time that they pull back in to depot again. What is the sequences of the revenue trips or revenue movements, and where that involve deadhead arcs, layover arcs, and et cetera. We will discuss this in this lecture and we will get to that.

So the task of developing the timetable is basically the task of translating frequencies into a timetable. We may all be familiar with what a bus service or train service timetable is. It basically includes the list of departures and arrivals to different stops by different routes. There are alternative ways of determining the timetable. The simplest way is to just use equal head ways. And the frequency that you allocated to the route the time period, during the time period, just distribute them evenly over the time period.

If you look at the figure on the bottom left here, we will have an example of even head ways here. However, if the demand is not uniformly distributed during the period, for which the example could be peak of peak, that is usually happening-- depending on the network, it may vary-- but usually happens between 7:00 and 8:00, 7:30 and 8:30. Then we want to work out the head ways, and work out the timetable in a way to result in balanced loads and equal loads.

So in order to do this, what you need to do is to generate the cumulative passenger flow arrivals. And then if you distribute your headways on the time axis, you will get even headways or equal headway service. If you distribute your head weighs on the cumulative passenger flow evenly, then you would get balanced loads. As you can see, the headways which is here,

one, two, three, four, five balancing loads for time table development.

The other alternative way of developing timetables would be trying to design a weight clock-face service. Why would we want to design a clock-face time table? Does anyone have any suggestions? Yep.

AUDIENCE: Sometimes it makes scheduling easier if we have some sort of cyclical pattern. It's easier to assign drivers that way.

PROFESSOR: Right, yes. So that is actually one reason-- it may eventually, in the next step, which is crew scheduling, makes the service more efficient and cost efficient. Is there any other-- yep.

AUDIENCE: It's easier for the passengers to remember the schedule.

PROFESSOR: Exactly. If the waiting time estimations that we are doing, with respect to frequencies and with respect to headways, if passengers are able to coordinate with the service, the expected waiting time would be less than the situation where they cannot coordinate with the service. That's only the case when headway is 10, 15, 20, 30 or minutes or 60 minutes. And then we would have service every-- on minute one, for example, that that would be 7-1, 7-11, 7-21, and so on and so forth. So it's easier for passengers to remember the service that is designed clock face.

So again, for the two approaches to headway distribution, equal heavy distribution is basically, as it can be imagined, each headway would be calculated as peak period duration divided by number of vehicles or frequency that we determined for the route at the time period. And in terms of balance load solution, we basically need to first realize what the loads are, the balance loads are, by dividing the passenger flows to n and then work out, on the cumulative passenger arrivals, to realize what the timetable arrivals and departures are. Is there any questions about this?

AUDIENCE: Yeah. Looking at the graph on the left, let's say PE4 minus PE [INAUDIBLE] is greater than the capacity of a bus, then we have a problem. Because if the number of passengers there is [INAUDIBLE] problem with the clock faced--

PROFESSOR: Mhm. Yeah, that's-- however, what we basically did in previous step, in frequency determination, that's one of the checks that we do, the maximum load. And we basically make sure that maximum load is within the allowed standards.

AUDIENCE: But basically, so what we're saying is if we do clock faced, then we look where we have the maximum flow of passengers and take that headway and replicate that headway all across the rest of the time period, is what you're saying if I understand correctly. I'm saying I have to design a headway to serve the maximum, and then the rest of the time period, which has lower flows, will have, you know, excess capacity.

PROFESSOR: Right. Yeah. So yeah, that makes sense. It really depends if we design based on the time period and we distribute the frequencies evenly and make sure that we are checking the requirements that are related to maximum load factors, then we won't have any problems.

However, the other issue that could actually be the counterpart of the issue that you suggested is that when we're designing the timetable based on balance load solution, we may actually end up having to use more vehicles required for that solution. And the reason for that-- does anyone have any suggestions for the reason for [INAUDIBLE] if the number-- why the number of vehicles that we may want to require for balance [INAUDIBLE] could be [INAUDIBLE]?

AUDIENCE: Because if you have a peak right at a certain time, the vehicle would be able to make one trip, and then you'll be sending out a bus just to make a single trip and go back, which adds a lot of deadhead time on either end.

PROFESSOR: Right. Yeah. So that could be one reason. The other reason would be if we have a peak of peak in our passenger arrivals, then we would have really short time intervals or short head ways during peak of the peak, and rather longer headways during the off peak. And then what happens is that if these longer headways exceed the maximum headway that is allowed, then we would have to basically throw in more vehicles to make sure that we do not exceed the maximum headway allowed, which is one of the considerations related to maximum waiting time for passengers and level of service. So that's one of the issues.

So as a result, in general, it is better to assume that design of timetable is going to be based on equal headways, and when we do the frequency computations, be consistent with that.

AUDIENCE: So when we [INAUDIBLE] and they are, like, strictly bound to follow the movement they are set on [INAUDIBLE]?

PROFESSOR: I'm not sure I understand your comment. Can you repeat that please?

AUDIENCE: If there's [INAUDIBLE]?

PROFESSOR: Right.

AUDIENCE: [INAUDIBLE] then the number of vehicles is going to go up. But since the number of vehicles required during off-peak [INAUDIBLE] then it necessarily [INAUDIBLE] you need more vehicles during service [INAUDIBLE].

AUDIENCE: That's interesting. I have a question-- why do you think that a balance load will require more vehicles? I tend to believe that it's somewhat more efficient, capacity wise, than the alternative. Why do you think that it has more vehicles?

AUDIENCE: I thought it was because the frequency is higher.

AUDIENCE: Only at certain points. It's higher only--

AUDIENCE: You have enough vehicles to cover [INAUDIBLE].

AUDIENCE: But what you were saying is that you can't cut service so short that you don't need those load factors. So in the end, you end up using more vehicles.

PROFESSOR: So the reason that we would basically require more vehicles, in balanced load, is that for some of these headways, we need to stretch the headway in a way that it basically meets the balance load on the other headways. And then, once we do that, we may actually go over the maximum allowed headway, that is basically trying to take care of maximum waiting time and level of service in the end.

In that case, then we need to throw in more vehicles during the same period of the route. And if we need to do this for all the routes at the same time, then we would eventually observe more deficit, and more vehicles to be used. Is this the point that you tried to--

AUDIENCE: Yes, but [INAUDIBLE].

PROFESSOR: On reverse?

AUDIENCE: [INAUDIBLE].

PROFESSOR: Well, these are two different factors and both are important in terms of level of service. Headway consideration is related to maximum waiting time. You don't want to have like reading the long headways that would force people to wait longer. And at the same time, you

don't want buses that are overcrowded, and there's discomfort resulting from the crowding effect.

So these are two different sources of this utility, basically, in the service, that we try to balance somehow.

The other issue that you brought up first, in terms of the fact that the service may be directional, and demand may be directional during peak, and there could be possibility of using vehicles in other directions, or using deadheading, and situations like that, we are actually going to cover that. And that's true for both of the two scenarios.

AUDIENCE: It seems like there should be some kind of hybrid way to say, this is the amount of vehicles I want to use-- and there's a way to not have to go over schedule based on equal headways, and still [INAUDIBLE] demand.

PROFESSOR: Exactly, yeah. And so there is all sorts of tricks in these steps, where you tweak some of the solutions from one level and go back to the previous levels, and adjust those. There is all sorts of engineering judgments, and the planning judgments that are involved, and can actually take place.

We will see a couple of those in this lecture, too. OK, so let's go.

So now, once we know what the timetable is, the sequence of arrivals and departures for different trips, for all routes, then it's a matter of determining what's the fleet size that is required for delivering the service that is timetabled.

There is this famous theorem in the literature that deals with an estimation of number of fleet that is required. It is basically an upper bound to the number of vehicles that are required to deliver the service. The reason that it's an upper bound is that it basically assumes that we are going to stick to the original timetable, and we are not allowed to do any trip shifting, or we are not doing any trip shifting in this step to accommodate some of [INAUDIBLE] that are one or two minutes, or a few minutes basically conflicting.

And then the other assumption that is made here to compute the upper bound of fleet size is that we are not allowing deadheading trips to balance the deficit among the terminals. So let's start with the definitions. The function l of k, t, and s-- k is the index of terminal, t is the index of time, and s is the index of the schedule that is already developed.

So l is basically the cumulative number of departures from terminal by time t following the schedule. And a would be the cumulative number of arrivals at terminal k by time t following that schedule.

Then we define a function, which is the difference between the two, it's basically the deficit function which is the cumulative departures minus cumulative arrivals. We use this deficit function to estimate what's the required to deliver the service. So the fleet size theorem says that, if we basically compute the maximum number of deficit over all time intervals for each terminal, and we add those up, we will get the total number that is required-- number of vehicles that is required to deliver the service. So it's basically this equation.

The summation over all terminals, maximum over t of the deficit function. So in this figure, we basically on top, we see different trips that are scheduled, and each trip is basically identified with a number and a pair of terminals. That's, for example, trip one, it starts from terminal m, and ends at terminal u.

And then, as you can see, these trips all start at terminal m, some of them go to u, some of them go to v, and there is also trips between u and v, and basically some of them come back to m again.

So if we basically work out the deficit function for terminal m, what we see is that, for example, at time around 5:00 AM, we will have one trip scheduled here, trip number six. And at 6:00, there is another one. Our deficit is basically growing, until at noon, where the first trip comes back to terminal m.

And then, at that point, it continues at a deficit level three, where there is one departure and one arrival scheduled at 17, at 5:00 PM. And then it continues until the last trip arrivals, at a little be before 10:00 PM. So in this case, if we look at the maximum deficit over time for this terminal, it would be four.

And that's for terminal m. When we look at all terminals in the network, and some of the maximum deficits, the summation will give us the number of fleet that is required.

However, there are a couple tricks that we can basically improve the efficiency. One of the tricks would be shifting the departures or arrivals a little, bit so we accommodate more trips. If they're required, for example, in this case, like the deficit part that actually gets to level four, is as the result of trip nine being scheduled around like 11:15, or so.

If we can manage to shift trip nine a little bit forward, up to noon, then we may actually get rid of the fourth unit of deficit here, and make the total deficit, or maximum deficit, here equal to three.

There's also the issue of reliability that you may want to consider for these trip arrivals and departures. For example, if you look at this point here, where trip two arrives, and it's basically based on this method, it is going to be assigned to trip eight. However, what if trip two is late by a few minutes?

What should be doing in that case? Are we going to delay trip eight to accommodate that? One way of looking at this is that instead of working out the deficit functions with trip times, and then schedule trips, we can work it out with half cycles that already includes layovers, and 95 percentiles at the terminus, to make sure that we have a leeway at the terminals. That will definitely require and result in higher deficits.

But again, if we are allowed to shift the trips a little bit back and forth, then that's something that would be recommended-- to work with half cycles, instead of working with the scheduled trips.

Is there any questions about fleet size required computation here?

AUDIENCE: [INAUDIBLE] to reduce fleet size deficits? Do they actually employ the techniques of shifting?

PROFESSOR: Yeah, of course. That's a part in timetable-- it's going back to one of the steps that was already decided and optimized in the past. But there are all sorts of these revisiting that is going on in transit agencies. Because we do not optimize all the components in an integrated manner, all at the same time, and we basically do a sequential optimization, there is always this requirement to go back to some of those decisions that you thought were optimal at the upper level, and adjust those to account for gains in the lower level.

AUDIENCE: I assume-- suppose that this graph represents an entire agency, and we'll reduce that agency's fleet size from four to three, saving money on another bus. That's a lot of money in savings, by not having that fourth bus. Do they look--

PROFESSOR: When they design timetables, they try to have some sort of coordination with arrivals and departures to the terminals. There is an example at the end of this lecture that we're going to talk about, how you can coordinate back and forth trips, so it will result in lower deficit in the

first place.

But yeah, sometimes it may actually be at the level of the fleet size determination, that you do this these shiftings again to improve the efficiency. Any other questions?

It's a bit faster on these parts. So when it comes to matter of vehicle scheduling problems, it is a very interesting problem to optimize. The objective of this problem is to define vehicle blocks, or sequence of duties for vehicles that includes revenue trips and non-revenue trips. In other words, deadheads to rebalance, or to balance the deficit at terminals. Or layovers that would actually take place for reliability considerations.

So the objective is basically to define these vehicle blocks, given that we are going to cover all the trips that are already scheduled, in a way to minimize the fleet size as a measure of the capital cost, and minimize the non-revenue time and mileage. As a measure of a non-revenue time, could actually account for crew cost, and mileage could account for both crew cost and fuel and basic operational costs.

One important observation is that these are proxies for us. Again, in the [INAUDIBLE] step of the work when we are doing the scheduling for crew, we will realize some of the costs actually depends on how these vehicle blocks are defined. And probably the important components of cost that is related to-- they're related to crew duties are going to be determined in data stage. So these are only proxies for cost, fleet size, and non-revenue time and mileages.

Input into the vehicles scheduling problem would be [INAUDIBLE] vehicle revenue trips is the timetable that we are to deliver. Each is characterized by trip stop time, and the end time and point. Then the other input would be possible layover arcs to account for reliability of these trips, and delay trips.

And deadhead arcs connecting depots to trip starting points-- we call those pull-out arcs from [? depots. ?] And then pull-ins back into depots from trip end points to depots. And then the deadhead, or balancing deficit arcs-- these are basically trips that bring vehicles from terminals with lower deficit than what is scheduled to terminals with higher deficit.

So what we try to do is that, from these arcs, we certainly need to choose the optimal combination of these arcs to deliver the service, and bring our vehicles out from depot and the same number back to people depot again, after the day is over.

So another important observation, or quick note, here, is that there may be many possible

number of deadhead and layover arcs. It's actually from a combinatorial nature. However, the designer, or the person who tries to solve this problem, needs to review these arcs, and come up with feasible and plausible arcs to start with.

And then, as I mentioned, the layover arcs are also important in terms of making sure we will have a certain level of reliability. There are variations to the problem that actually can make the problem more complicated, but we are going to stick to the simplest one, and show how we can formulate the problem, model it, and what are the algorithms that can solve it for us.

These variations, basically, are related to-- one possible variation is related to vehicles servicing single lines, or single routes, or inter-lining between different routes. The other variation would be related to possibility of having multiple depots in the network, big, larger transit agencies usually have multiple depots.

For example, for Boston, I know a few of those. I do not know what's the exact number of depots that exist in the network. Alfred Street is one--

AUDIENCE: Albany, Southhampton, two of them at Lynn, Fellsway, North Cambridge-- that's it.

PROFESSOR: That's, yeah, OK, so it's almost close to 10-- eight, nine. OK, so when it comes to different depots, and having multiple depots, there will be more complexities introduced to the problem. Because you may want to make sure that the exact number of vehicles who depart at a depot, come back to depot.

There are also some considerations related to the fuel that the fleet is using. You may have diesel buses, you may have CNG buses. If you have restriction in terms of fueling facilities in these depots, you may want to return back the vehicles to the particular depots that are consistent with the fueling system.

Then, there are different considerations related to vehicle types, related to parking spaces at the depots, and that actually makes the problem more complicated. But we can actually make some simplifying assumptions for now, and see how the problem looks like in the most basic form.

So this is the time space network representation of the vehicle scheduling problem-- we are trying, in the next three slides, we are trying to build on the time and space network structure, to realize what is the decision that we're trying to make, and how we can basically solve this.

So this is time space representation with x-axis being time of day, and y-axis being space. This is only one dimensional space here, but it could be two dimensional space, too.

So in this case, we have two routes that are represented here-- route one, or n routes-- route one to route n. As you can see there are trips from either direction of these routes, that are already timetabled, and we have already committed to deliver this service.

So the black arcs here are revenue arcs that are the arcs that we committed, and these have to be delivered between the two terminals of every route. Then we have-- in this problem, we have only one depot. There are two nodes here that are represented as depot, but it's basically the representation of depot at the beginning of the day when vehicles pull out, and the representation at the end of the day when vehicles pull in, basically.

There are also layover arcs, where you connect all possible alternatives that you may think would be plausible for connecting these trips to one another. So in this figure, we only have a few of those-- probably the closest one-- but you can work out any possible combination of bus arrival to a terminal, being connected to a different trip from that terminal.

AUDIENCE: Would you mind explaining what the [INAUDIBLE]? Maybe I'm missing something. Like, for example, when we start out at the beginning of the day, [INAUDIBLE]. We have two lines emanating from the very beginning of the day, so is that two buses leaving the depot at the same time? Maybe slightly different time?

PROFESSOR: Right, so let me go ahead and continue and finish all the arcs in the network-- that may clarify the situation. Because there are other arcs in the network, too. So there are arcs that are pull-out arcs, from the depot to start these trips, and arcs that return the vehicles back to the depot.

There's also arcs that are there to account for possibilities of deadheading trips, from one route to another route.

AUDIENCE: So now we have those two nodes going into route one, so two buses going into route one, right?

PROFESSOR: That's only a link. That link may include multiple vehicles.

AUDIENCE: Multiple vehicles at different times?

PROFESSOR: Multiple vehicles at different times-- well, it could be-- yeah, this is not the exact representation of time-dependent service. You can actually have connections, if you consider each one of these links to carry only one vehicle or not, then you need to add vehicles from depot to other trips along the time dimension, to vehicles that are basically joining that route [INAUDIBLE] time.

AUDIENCE: What is the top [INAUDIBLE]?

PROFESSOR: The top line, you can-- the top line is just-- [INAUDIBLE].

AUDIENCE: [INAUDIBLE].

PROFESSOR: It could be-- yeah, this is--

AUDIENCE: [INAUDIBLE].

PROFESSOR: This is, again, a simplified situation, but you can pull out at the middle of the route, too. Where you can, for example, just serve a vehicle from here. But you may want to make sure that the scheduled trip is served. So if you have services that's basically a start from the middle of the line, then you can model that, too.

This is just to show how you can model a situation like this. It may not be all comprehensive combinations, or situation.

AUDIENCE: I just wanted to make sure I understand what it says.

PROFESSOR: Doesn't id make sense now?

AUDIENCE: Yeah, I think so. I think one of the confusing things about this model, actually, is that you have that horizontal line. And it's saying [INAUDIBLE] revenue. Really what you should have is that layover arc shouldn't be an arc, it should just be a line connecting the end of-- [INAUDIBLE].

PROFESSOR: Exactly-- these horizontal lines, I should actually remove that. I will--

AUDIENCE: Yeah, just make the layover arcs lines.

PROFESSOR: Yeah, exactly, layover arcs-- that's exactly true. Actually, make the connection between the two directions of the route, and then the other arcs that are involved are the deadhead arc's points in the network.

AUDIENCE: But it could be noted that the layover arcs are not really-- [INAUDIBLE] there's no spatial dimension to them, because the bus isn't moving.

PROFESSOR: Right, yeah, exactly. Maybe that's-- the only spatial dimension to it would be just go ahead and turn around over the corner. And then come back, yeah.

AUDIENCE: It would still be the case that the bus that pulls into the terminal is the next one out. So that's when you might actually want to have the arc, to show that even though it's not really moving-- [INAUDIBLE] there is a reason why you might want to show it that way.

PROFESSOR: Exactly. And then the other point is that, this is, like even front from this node to nodes further down, the reason that we are [INAUDIBLE] think the optimization problem in the network structure, like many other optimization problems, is that with networks that consists of nodes and arcs, we can represent the constraints very efficiently. And there are efficient algorithms that use this representation to solve the optimization problem.

So whenever we can present an optimization problem in a network structure, we do that, because we have really elegant algorithms and network flows that can be used to solve optimization problems.

So let's see what's the optimization problem we are trying to solve here. One other link that we need to add here is basically from depot to depot, to make sure that the total number of vehicles that are going out of the depot, in the morning, are going to be equal to the total number of vehicles that are coming back to depot.

AUDIENCE: So we're assuming-- just to go back to that-- there are some lines there that don't have a pull-out, so we assume those are coming from other depots?

PROFESSOR: Yeah, in this example, we are only simplifying it into one depot situation, but some of the-- yeah, exactly-- some of the other routes could actually be--

AUDIENCE: I guess the question is, if a route ends, the first one that's going from the bottom to the top doesn't have a pull-out, and so there should be an arrow going back.

PROFESSOR: There should be an arrow, yeah, to that, too. But it's missing in this. These arrows are only representative, and they are not comprehensive. And we basically are defining integer capacities that would reflect a number of vehicles that are doing that duty, or are basically assigned to those duties.

These arcs are revenue, layover, and deadhead arcs. And these arcs, again, have costs. Like revenue arcs have given amount of cost, depending on the time that they take to make the trip, and so on and so forth. Layover trips, depending on how much time they take. And deadhead arcs, depending on how much time they take, and how much mileage they require to travel, the vehicles.

However, what's interesting is that [INAUDIBLE] arcs costs of zero for those, because it's not relevant. We eventually would have to deliver those revenue arc services. So it's a constant in the optimization problem. We might as well just remove them.

But what's important is that when we want to see what the layovers-- what's the optimal combination of layovers and deadheads in this problem? So again, the bounds that we have for revenue arcs is basically lower bound is one, upper bound is one. Why is it like this? Does anyone have any suggestion, why the lower bound and upper bound are both equal to one for revenue arcs?

AUDIENCE: Because you said we have to operate the trips, no matter what.

PROFESSOR: And we only have to operate to deliver them once, right? Remember, these are trips in the time space representation. So each trip is basically one arc-- we need to deliver it once. So it's basically the matter of choosing [INAUDIBLE] or configuration of layover arcs between zero and one, and deadhead arcs between zero and one.

A maximum block length is a guarantee-- this is usually not a problem. If the fuel situation, especially for electric vehicles, is restricting, we may want to put additional restrictions on maximum length of the vehicle. But with the existing diesel and CNG vehicles, these are usually fine to deliver the whole service day services without having to refuel. It's [INAUDIBLE] that is something that is already taken care of in the vehicle scheduling problem.

However, shifting heuristic needs to take place before we get to vehicle scheduling, at the level of timetable development, as we discussed after the deficit functions. So the matter would be the-- if we have multiple depot, the problem becomes a little bit more complicated.

For single depot situation, we solved the problem with network flow algorithms, minimum cost flow algorithm, that is very efficient and very, very fast. When it comes to [INAUDIBLE] flow with integer, [INAUDIBLE] which is [INAUDIBLE] in the problem that we have, that problem becomes really difficult. It becomes NP-hard, and it basically is very difficult to scale.

That basically means that we do not use the network flow algorithms for multi depot situation. We use some of the existing [INAUDIBLE] stakes. Are there are any questions about this so far?

So this is another network representation of the multiple vehicle scheduling problem. This is not a time space representation anymore, because it would become really complicated with time space representation. As you can see, there are different depots in the network, and there are different services that need to actually-- or revenue arcs-- that need to be served.

There may be some constraints on number of vehicles that pull out from each one of these depots, and the pull in each one of these depots. And again, if we didn't have the constraint of having integer solution, this could be solved in linear time, too. However, with integer constraints, it's not possible.

So is there any questions about this?

It basically-- in order to be able to implement this, we need to make a definition on trips, and identify compatible trips. Two compatible trips at a terminal, I and J, two trips I and J are compatible at the terminal, if J is arriving to terminal, I is departing from terminal, and the time difference between the two is actually greater than the layover that we want. And then, also, less than twice the time of traveling to depot. The solution may be only as good as just sending the vehicle back to depot, and then pull it out from depot to another part of the network.

So we only restrict the search for compatible trips to these two bounds. Like MK here is the minimum recovery time that we define as a lower bound of time difference between the arrival of trip I minus departure of trip J. And then, we use 2 times DK, which is the deadhead time from terminal to depot.

Then the algorithm is really simple-- it has only four steps. We start with the earliest arrival at terminal. If there is no earliest arrival, we go to step D. And basically, in step D, we need to pull out from depot to serve that service.

But if there is one earliest arrival, then we link that earliest arrival to compatible trip departure, based on the compatibility constraint, or definition that we had. Again, if there is no trips that is compatible with that, we need to go back to step A, and basically use the next arrival at

terminal.

If there is one trip that is compatible with the trip that arrives, then we will check the vehicle block length against the existing constraint. Again, that could actually be related to the fuel, fueling constraint, and so on and so forth. If it's constraining, then we need to do to cancel that and return the vehicle to depot. Otherwise, we would basically make the connection between the compatible trips, and consider the second trip a new arrival to the other [? extreme ?] [INAUDIBLE] basically.

After that, we serve, again, all remaining unlinked trips from depot. We need to work out in your mind these steps once. It's very easy to practice, to realize how this works. But it's basically using the strict first in, first out rule, based on compatibility of trips and over the compatible trips. Is there any question?

AUDIENCE: Kind of going back a little bit, [INAUDIBLE] the single depot [INAUDIBLE]?

PROFESSOR: For single depot situation, the existing algorithms that solve minimum cost flow, like successive shortest path algorithm, or some of the negative cycle canceling algorithms, can give you integer solution. However, when it comes to multi-depot, and when it comes to multi-commodity flow, that's not always true. So you may actually get a solution with a linear-- or with a network flow algorithm that is in linear time, but it's not integer.

So you may have you decimal, or partial vehicles going back to different depots. That's the difference between the two cases. So other questions?

All right, so after this, we will have just one single route of scheduling practice, with two nodes, A and B, or with two terminals, A and B. This is one route that we want to schedule for AM peak period, and base period. AM peak is defined from 6:00 to 9:00. And base period is after 9:00.

Headways are during peak period, 20 minutes. And scheduled trip times are 40 minutes for both directions. Layover is 10. During off-peak, headways are 30. Scheduled trip time is 35. And minimum layover is, again, 10.

Dominant direction of travel is AM peak. And it's in the direction of A to B. So then, what we do is that we will [INAUDIBLE] direction A to B, and given the headways and running times, we work out the [INAUDIBLE] for all the trips that are scheduled during peak from 6:00 to 9:00, and then after that up to 11:00, and the rest could be actually similar.

So as you can see, the headways from 6:00 to 9:00 are 20 minutes, after that it becomes 30 minutes. The running time is 40 minutes in the peak, and 35 [INAUDIBLE]. First thing that we do is that we tie the two directions together. This is the kind of coordination that we previously discussed with the comment that [? Aton ?] made in terms of shifting these trips together.

So this is one example of shifting some of the trips to make better efficiency in the next steps. In this case, we tie the trips in the first direction with trips in the reverse direction. So first trip that departs at 6:00 arrives at 6:40. Considering 10 minutes layover, we will schedule the first trip for the reverse direction from 6:50. And it's anticipated to arrive at 7:30.

And this is so on and so forth for all the timetable, given the travel times between the two ends, and the headways.

AUDIENCE: Which means we need five buses?

PROFESSOR: Yes, but you will work this out. Right, we need five buses, but some of them are partial only. They can pull in in the middle, and pull out. But we will see that in a second.

So we will start with first pull-out-- that is the bus that is going to serve the first round trip. After one cycle, it's going to be able to serve the 7:40 trip for us, after 10 minutes of layover. It will end this trip at 9:10. Given 10 minutes layover, it will be able to serve 9:30 for us, and also 11:00 again.

The next bus that we will have to pull out can serve the trip from 6:20, and from 8:00. After 8:00, once it does the cycle, it's going to be 9:30. And 9:30 is already-- trip is already taken. Then we will have to choose between sending this back to the depot, or using it for trip at 10:00.

This trip at 10:00 basically means like 30 more minutes, or 20 minutes of waiting, we decide to send it back to depot. We will have five blocks for five different buses. Block one basically gives the itinerary of duties for bus one, or for whichever bus that this is assigned to. Then so on and so forth for all the buses.

You mainly will have to work with this method for the homework. But it's valuable to know what are the [INAUDIBLE] approaches in practice, and what's going on in research that deals with problems like this.

Any questions? All right, OK. Thank you very much.