Introduction to Dynamic Traffic Assignment

CE 392D

January 22, 2018

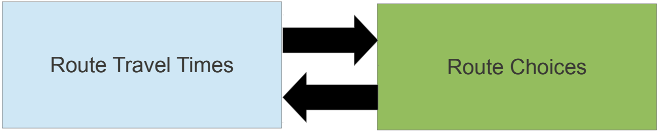
WHAT IS EQUILIBRIUM?

Transportation systems involve interactions among multiple agents. The basic facts are:

- Although travel choices are made individually, they impact others as well (congestion).
- The impacts of others' travel choices are important as you make *your* choices.

"It's the evening peak period, so the freweay will probably be congested. I'll take another route." (But isn't everyone else thinking the same way?)

Even with a relatively simple model of behavior (choosing the fastest route), we end up with mutual dependencies and circular relationships.



To resolve this circularity, we borrow some concepts from economics and game theory.

In an economic game, agents each choose *actions*; the joint actions of all agents results in an *outcome* and utility *payoffs* to each agent.

A *(Nash) equilibrium* is a strategy for all agents, where no agent can improve their utility by *unilaterally* changing their decision.

In dynamic traffic assignment, each traveler is an agent choosing a route; the joint actions of all travelers result in congestion patterns throughout the network, which determine the travel time each traveler faces.

At the equilibrium solution, no traveler can reduce their travel time by switching to another route.

Some examples of games...

		Bob	
		Cactus Cafe	Desert Drafthouse
Alice	Cactus Cafe	$(-1, -1)$	$(\mathbf{1}, \mathbf{1})$
	Desert Drafthouse	$(\mathbf{1}, \mathbf{1})$	$(-1, -1)$

Some games have more than one equilibrium solution.

Some examples of games...

		Fred	
		Testify	Remain silent
Erica	Testify	$(\mathbf{-14}, \mathbf{-14})$	$(0, -15)$
	Remain silent	$(-15, 0)$	$(-1, -1)$

In some games, all agents would be better off if they could coordinate their choices. (The invisible hand doesn't always work.)

Some examples of games...

		Harold: Heads	Harold: Tails
Ginger	Heads	$(+1, -1)$	$(-1, +1)$
	Tails	$(-1, +1)$	$(+1, -1)$

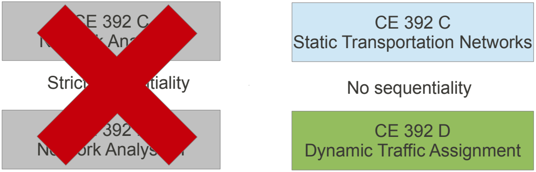
Some games have *no* equilibrium solution unless you allow agents to randomize their choices.

Important questions to keep in mind as we study dynamic traffic assignment: is there an equilibrium solution? Is there just one? Can a "prisoner's dilemma" occur?

Short answers: not always, not always, and yes.

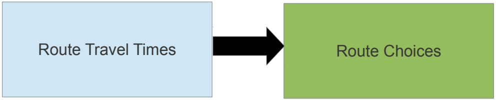
STATIC ASSIGNMENT

After the discussion about how different STA and DTA are, why review STA?



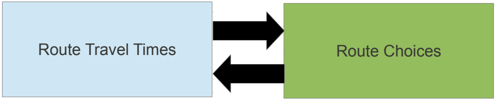
Three reasons: (1) there are some important parallels; but (2) where STA and DTA differ, they do so very intentionally.

STA and DTA make the same behavioral assumption: drivers want to reach their destination in the shortest time possible.



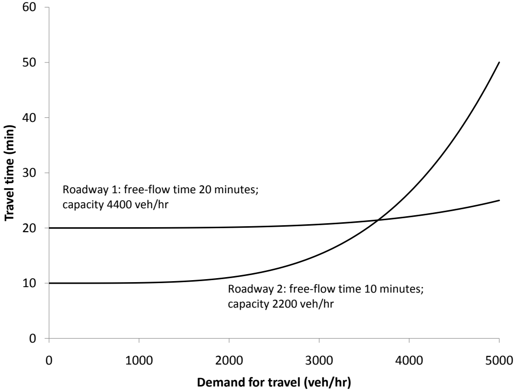
So, if I know the travel times on all of the routes, we can predict what route I will take.

However, there's a trick: travel times depend on the routes everyone else chooses.



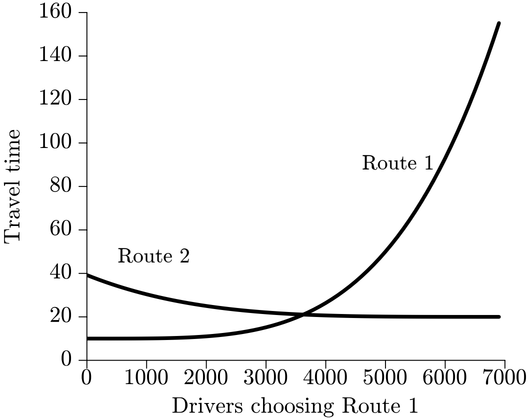
This mutual dependence is at the core of network transportation network modeling.

Static traffic assignment uses link performance functions to model congestion.



Each link performance function $t_{ij}(x_{ij})$ is an increasing, convex function of the flow x_{ij} on that link alone.

Since travelers want to minimize their travel times, they will switch to shorter routes from longer ones — which will decrease the travel time on the old route, and increase the travel time on the new route.



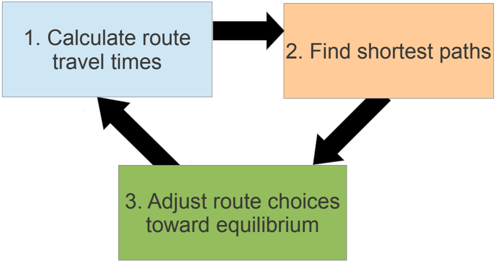
This will continue until all used routes between the same origin and destination have *equal* and *minimal* travel time. This is the *principle of user equilibrium*.

In other words, for any two routes connecting the same origin and destination, *either*

1. They have the same travel time; or
2. The route with the longer travel time is unused.

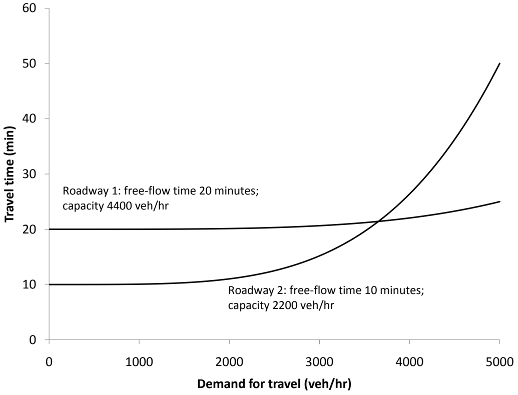
This contrasts with a system-optimal assignment, which minimizes the *total system travel time* $\sum_{ij} x_{ij} t_{ij}(x_{ij})$. In general, *these are not the same.*

Solving STA is an iterative process, involving three main steps



PROBLEMS WITH STATIC ASSIGNMENT

Link performance functions can be a bit problematic if you look at them too closely.



For example...

1. Not all vehicles will experience the same delay — in particular, traffic flow is *anisotropic* and obeys *causality*.
2. What does the "flow" x_{ij} mean in static assignment?
3. Where does congestion actually lie?
4. What about queue spillback?

What this means is that we need a fundamentally different congestion model, based in traffic flow theory rather than arbitrary functions.

Historically, there have been a few attempts at trying to build DTA models by running a sequence of static models, using link performance functions and tracking flow as it progresses through the network. This is no longer a serious line of inquiry — the fundamental problems with link performance functions must be addressed.

WHAT IS DTA?

The principle of *dynamic user equilibrium* is the following:

All routes used by travelers leaving the same origin **at the same time** for the same destination have *equal* and *minimal* travel time.

The sets of used paths can be different at different points in time.

The consensus among researchers is that dynamic traffic assignment *must* involve the following concepts:

1. A model for how congestion (travel times) varies over time.
2. A concept of equilibrium route choice.
3. Equilibration based on *experienced* travel times, not *instantaneous* travel times.

Despite what a software vendor may tell you, if it doesn't have these three things, it's not DTA. It is usually impossible to do all of these things with a "one shot" assignment.

Tradeoffs with DTA

The biggest strength of DTA models is a much more realistic traffic flow model.

However, this strength does not come without its own weaknesses.

Why doesn't Barbie have knees?

Tradeoffs with DTA

A perfect model is indeed realistic. But it would also be:

- Efficient in computational resources and running time
- Easy to obtain and calibrate data inputs
- Robust to errors in the input data
- Transparent and trusted among decision makers
- Amendable to mathematical analysis: do solutions always exist, are they unique, are there algorithms which converge

As a rule of thumb, DTA is best applied when the input data are known with high certainty, only a few scenarios are needed, and detailed congestion and queueing information are critical.

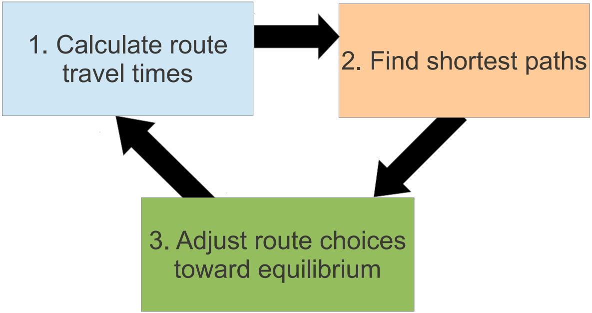
An outline for the next few weeks:

- Basic traffic flow theory and congestion models for DTA
- Time-dependent shortest path algorithms
- Finding equilibrium solutions
- Practical concerns and case studies

DYNAMIC TRAFFIC ASSIGNMENT

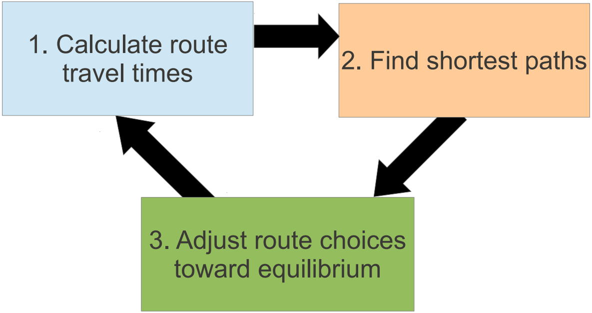
Framework

Because DTA is based on experienced travel times, rather than instantaneous ones, an iterative framework is needed.



Most DTA models can be placed into this general framework.

In this class, we will study each of these three steps in order.



Interestingly, the easiest step in static (updating path travel times) is the hardest one in dynamic.

We defined the principle of dynamic user equilibrium as follows:

All routes used by travelers leaving the same origin **at the same time** for the same destination have *equal* and *minimal* travel time.

We can devise iterative procedures in which dynamic user equilibrium solutions are "fixed points."

Equilibrium can be thought of in a behavioral way; this is the perspective of economic game theory.

		Bob: Cactus Cafe	Bob: Desert Drafthouse
Alice	Cactus Cafe	$(-1, -1)$	$(\mathbf{1}, \mathbf{1})$
Alice	Desert Drafthouse	$(\mathbf{1}, \mathbf{1})$	$(-1, -1)$

Alice and Bob have recently broken up, and do not want to see each other when they go out with friends.

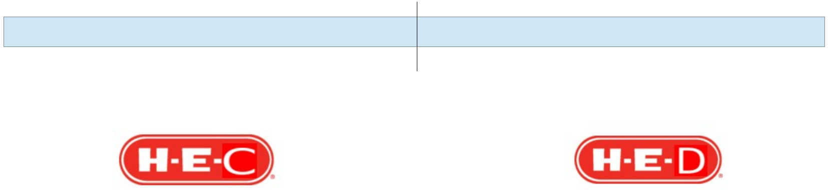
One of the insights of game theory is that *equilibrium solutions need not be socially-optimal solutions*.

		Fred: Testify	Fred: Remain silent
Erica	Testify	$(\mathbf{-14}, \mathbf{-14})$	$(0, -15)$
	Remain silent	$(-15, 0)$	$(-1, -1)$

This is due to the presence of externalities, where one person's choice has influence on another who has no say in the matter.

Tragedy of the commons

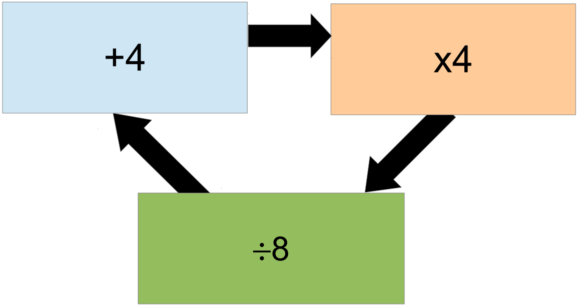
Where to locate your stores



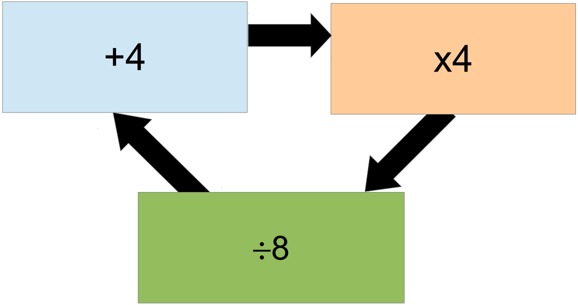
Turn prohibitions

Cut-through traffic

Consistency can be represented in a mathematical way:



A clever person can solve this type of problem by denoting the values in the boxes as x, y, and z.



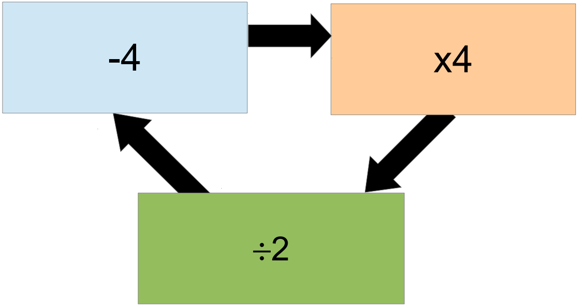
We then have $y = x + 4$, $z = 4y$, and $x = z/8$. Substituting them into each other, we have $x = \frac{1}{2}x + 2$ or $x = 4$.

A not-so-clever approach is to pick a starting value, then calculate through the loop iteratively.

$10 \to 7 \to 5.5 \to 4.75 \to 4.375 \to \ldots$ which converges to the correct answer (4).

Picking a different starting value: $1 \to 2.5 \to 3.25 \to 3.625 \to \ldots$ also converges to the same answer.

However, this simple approach won't always work.



Choosing the same starting value, we have $10 \to 12 \to 16 \to 24 \to 40 \to \ldots$ which diverges to $+\infty$.

The clever approach still works: $y = x - 4$, $z = 4y$, and $x = z/2$, so $x = 2x - 8$ and $x = 8$. If we used this as our starting value, the simple approach would work, but for any other value it will diverge.

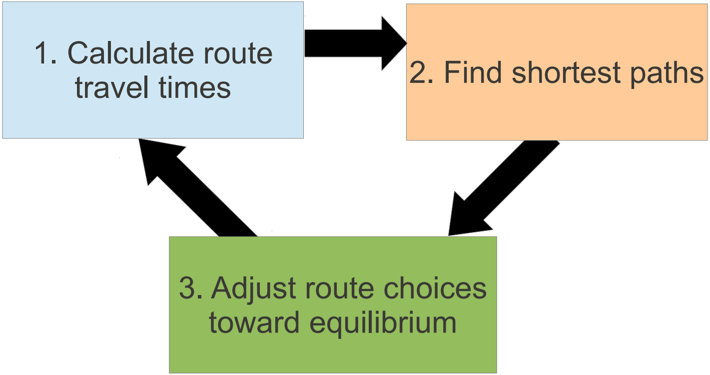
The moral of the story: if we are simply iterating between the three steps, we need to be careful to ensure that the solution will converge to a consistent answer.

Regional dynamic traffic assignment problems are complicated enough that we do not know of a faster "clever" method.

Both of these interpretations are useful ways to think about the solutions we are trying to find in dynamic traffic assignment.

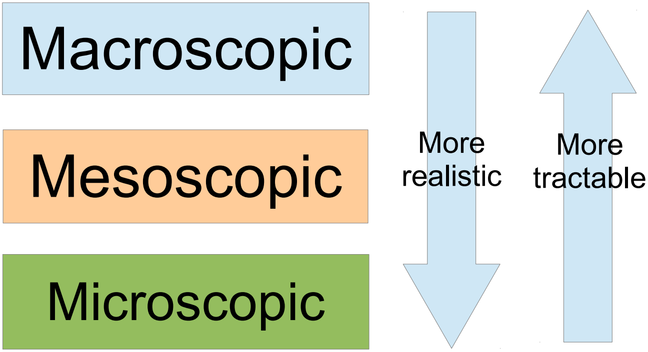
TRAFFIC FLOW MODELS

The first step in dynamic traffic assignment is to update the travel times on each path.



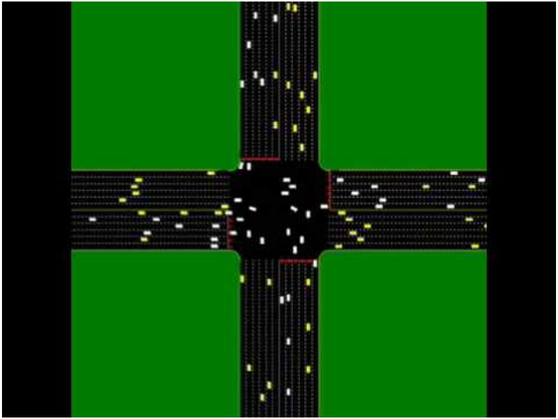
This is also known as the *network loading problem*: we know the paths and departure times for each traveler, and by "loading" them onto the network we can calculate the travel times they experience.

There are many different ways to do this, falling under the umbrella of traffic flow theory.



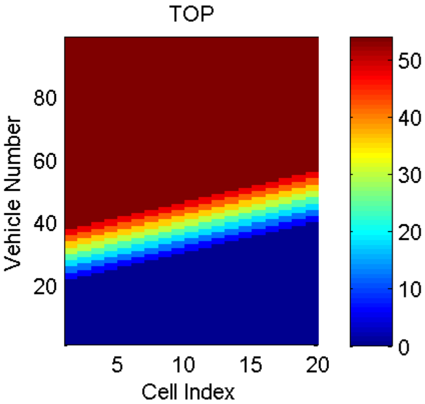
I personally dislike the macro/meso/micro distinctions because everyone uses them differently, but you will hear them and need to know them.

Microscopic models



Microscopic models track the trajectories of individual vehicles at a very fine time scale (≈ 0.1 seconds). They track the precise locations of each vehicles, account for variations in driver behavior, vehicle characteristics, and so on. This level of detail makes them highly realistic, but impractical for modeling large regions.

Mesoscopic models



Mesoscopic models are more aggregate, neglecting or greatly simplifying variations in behavior, and vehicle type. Time scale is on the order of 6 seconds. They are often based on fluid models of traffic flow. This class will focus on mesoscopic models