**Uber Navigation**

The whole road network is modeled as a graph. Nodes represent intersections, and edges represent road segments. The edge weights represent a metric of interest: often either the road segment distance or the time it takes to travel through it. Concepts such as one-way streets, turn restrictions, turn costs, and speed limits are modeled in the graph as well.

Once decided on the data structure, shortest path routing algorithm is used to find a route. Algorithms based on contraction hierarchies achieve fast performance. Uber uses an open source algorithm called OSRM. As new traffic information comes in with real time data, graph edge weights will be updated dynamically.

The graph will be divided into layers of small cells which are relatively independent of each other to make the processing even faster.

**Driver-Rider Matching and ETA**

Consider a network where the nodes are Uber app users, particularly of nearby drivers and the riders. This would be a linear programming problem where the objective function would be to assign a driver to a rider minimizing the wait time.

Linear constraint, for example, could be that one driver can only pick up one rider and one rider can only go with one driver. Cost matrix will be populated based on the network graph discussed earlier.

The system would also consider several real time data, like the rider’s current location, where the driver is headed, traffic, if it's a shared ride, etc. to connect drivers and riders with the shortest possible ETA. It would also consider the filters that user applied like the type of Uber; for example, UberX, Uber Black, Uber Pool etc. Drivers with higher ratings and with lesser complaints will get a higher priority while doing this matching.

**Uber Surge Pricing (GeoSurge)**

In situations when the demand outstrips the supply, the pricing algorithm increases the prices to bring the market to equilibrium. But this is not permanent, as with time the demand decreases and the price falls back to normal. Uber has filed a patent for its secret surge pricing algorithm.

The exact details of the algorithm aren’t known but what the general idea of the “surge pricing algorithm” is that this algorithm is ran every 5–10 minutes to determine the prices of rides for places of high demand. When prices go high it would also motivate more drivers to move into the area of surge. The algorithm is very effective in keeping a balance between demand and supply.

With the massive troves of user data that Uber has collected, they most likely have tweaked the algorithms for each city to adjust for the varying sensitivities to surge pricing like income, traffic patterns, car ownership, etc. They also inform drivers in advance about the expected surges based on the historical data that they collected on a specific city.