7th International Conference on Innovations in Travel Modeling (ITM)

A Time-Dependent Intermodal A* Algorithm for Heterogeneous Travelers with Efficient Heuristics

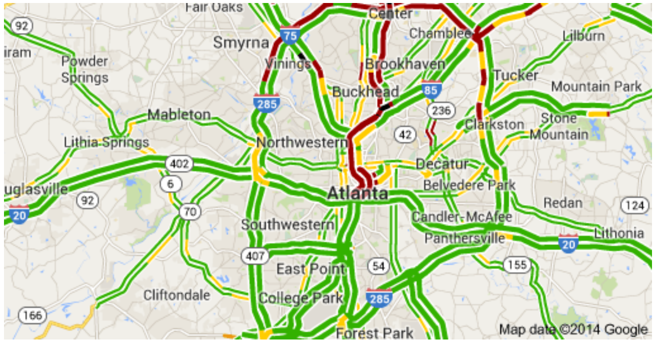
Ömer Verbas, Joshua Auld, Hubert Ley, Randy Weimer

Argonne National Laboratory

June 25th, 2018
Atlanta, GA

Motivation – Technology and User Experience

- New Mobility Alternatives
  - Car Sharing
  - Bike Sharing
  - Transportation Network Companies
  - Demand-Responsive Transit
- Real-Time Information
  - Traffic
  - Transit
- New Technologies
  - Connected and Autonomous Vehicles (CAV)
- New Business Models
  - Personally owned vs shared?
  - Compete vs. complement?
- User heterogeneity



Tree-Based vs. Point-to-Point (A*) Algorithms

- Tree-based algorithms find paths from a single origin to all destinations or from all destinations to a single destination
  - **Very efficient for homogenous travelers**
  - Since it is computationally challenging to find path to/from all nodes zonal aggregation is utilized.
  - Time discretization is used both for departure/arrival times and time-dependent attributes

- Point-to-point algorithms find paths from a single origin to a single destination
  - **Can reflect traveler heterogeneity**
  - Time discretization is used for time-dependent attributes
  - Issues of admissibility in time-dependent networks with FIFO violations

A Dijkstra-Based Estimated Cost Calculation

- Common practice is using the Euclidean distance divided by maximum speed.
- Not very practical for transit
- Khani, A., M. Hickman, and H. Noh. Trip-Based Path Algorithms Using the Transit Network Hierarchy. Networks and Spatial Economics, Vol. 15, No. 3, 2015, pp. 635–653. https://doi.org/10.1007/s11067-014-9249-3.
  - Obtain the minimum observed travel time on every link
  - Ignore waiting times, transfer penalties etc.
  - Solve tree-based algorithms from all nodes to all nodes and store the values
  - Admissible because actual travel times which also include waiting times can never be lower

Time-Dependent Intermodal A* (TDIMA*) Algorithm

- Multi-modal network
  - Transit links
  - Walking links
  - Vehicular links

- Intermodal routes
  - Cars
  - Walking
  - Walk-to-transit
  - Drive-to-transit (PNR, KNR)
  - Taxi/TNC/CAV before/after transit
  - Rail only versions

- Node switching penalties
  - Turn movements
  - Transfer penalties

- Individual upper bounds on:
  - Waiting
  - Walking
  - Driving

- Individual weights for:
  - Waiting
  - Walking
  - Driving
  - Transit vehicle
  - Transfer penalties

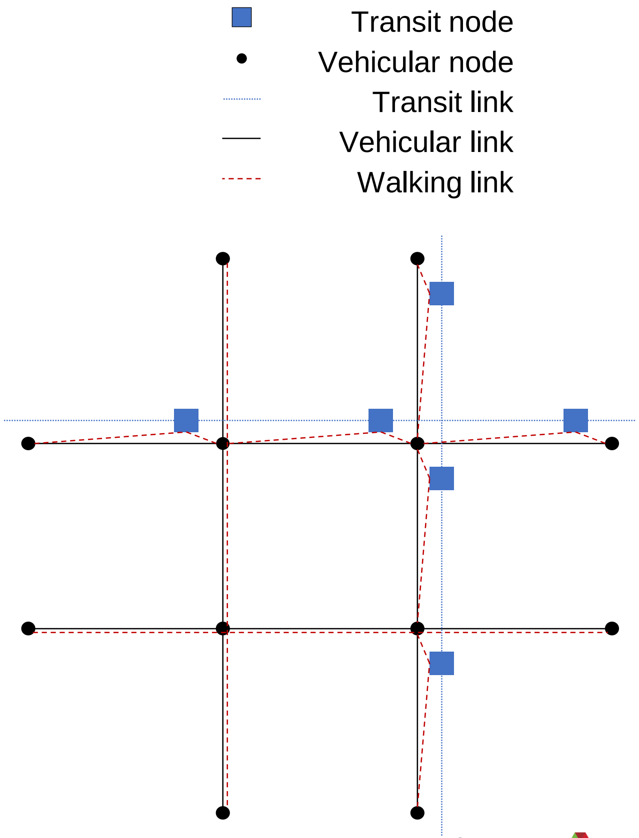
Network Representation

- Three layers
  - Transit links
  - Walking links
  - Vehicular links

- The transit layer is constructed using General Transit Feed Specification (GTFS)

- The vehicular layer is constructed using Chicago Regional network data designed for POLARIS

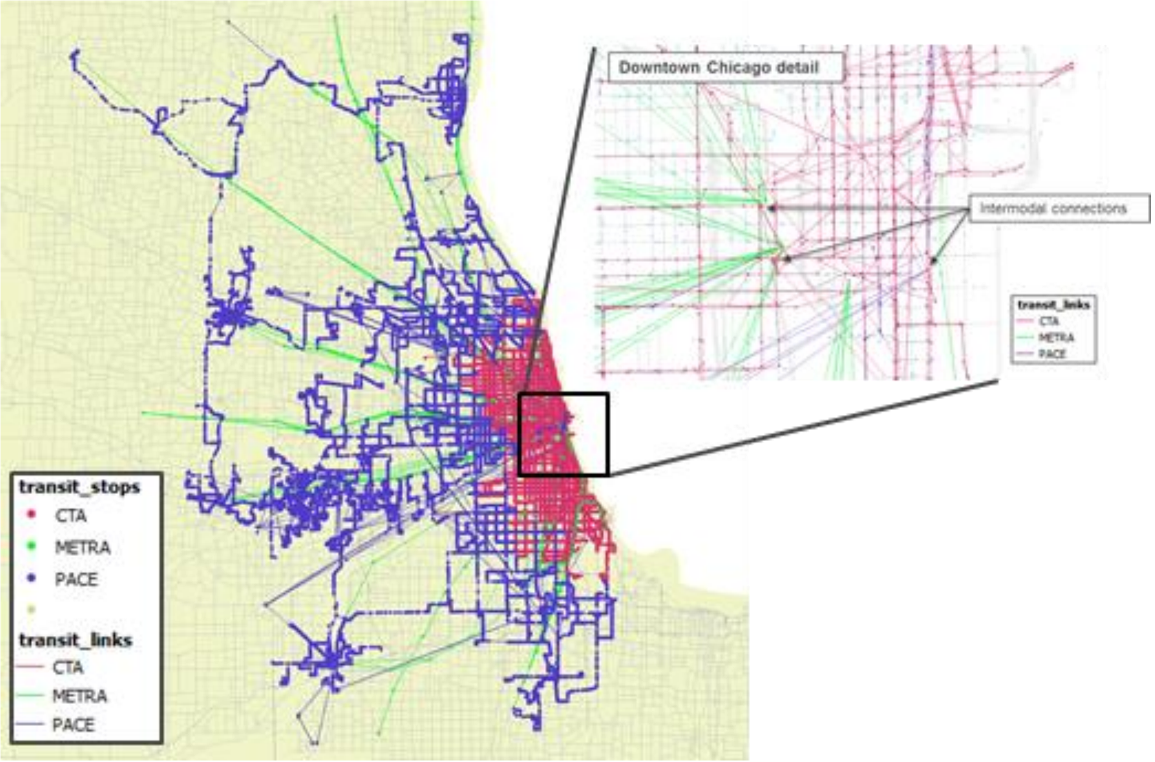
- The walking layer is derived from the vehicular layer
  - Vehicular links are segmented into smaller links wherever a transit stop/station is nearby



Argonne NATIONAL LABORATORY

Chicago Regional Network

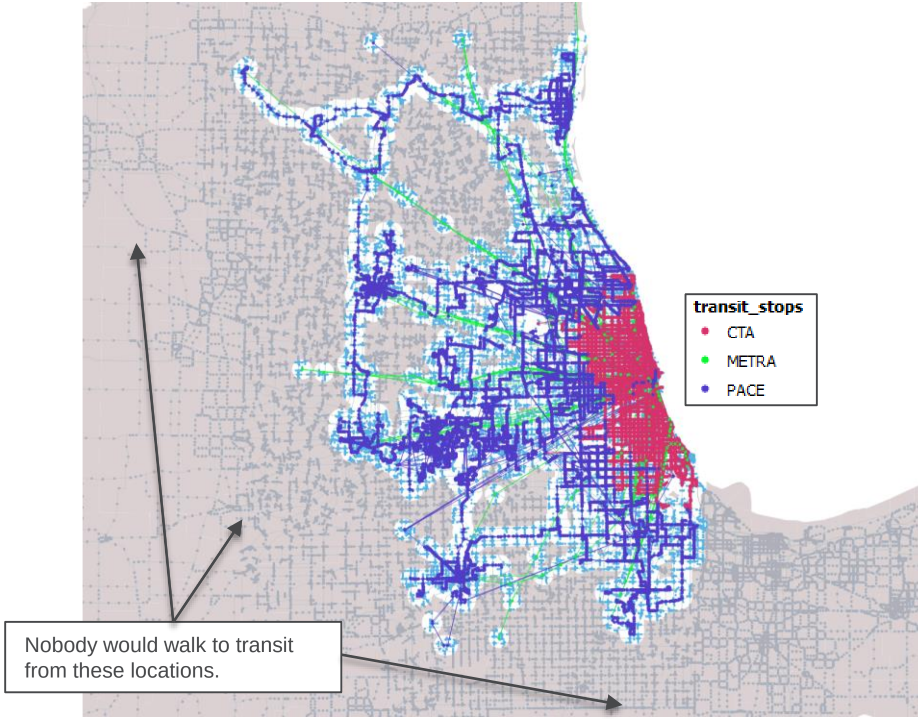
- 54,028 nodes
  - 35,077 transit
  - 18,951 vehicular
- 217,119 links
  - 37,642 transit
  - 123,000 walking
  - 56,477 vehicular
- 173,236 activity locations
- 1,961 traffic analysis zones
- 344 transit routes
- 2,098 transit patterns
- 28,138 transit trips



Mode Choice and Transit Routing Integration

Better filters eliminate infeasible walk-to-transit queries

- A Dijkstra-based one-to-all routing algorithm is written
- From every driving link in the multimodal network, we find the shortest driving time to a transit stop and store this as a link attribute.
- There are 123,000 walking links in the Chicago network. The overall process takes less than a minute.
- This gives the mode choice component to filter out the walk-to-transit option if the walking distance to the closest transit stop is more than 2 km, 5 km, or any selected value.
  - We check both walking distances at the origin and at the destination.
- The upper bound on walking is agent-specific, i.e. every traveler can have their own upper bound.

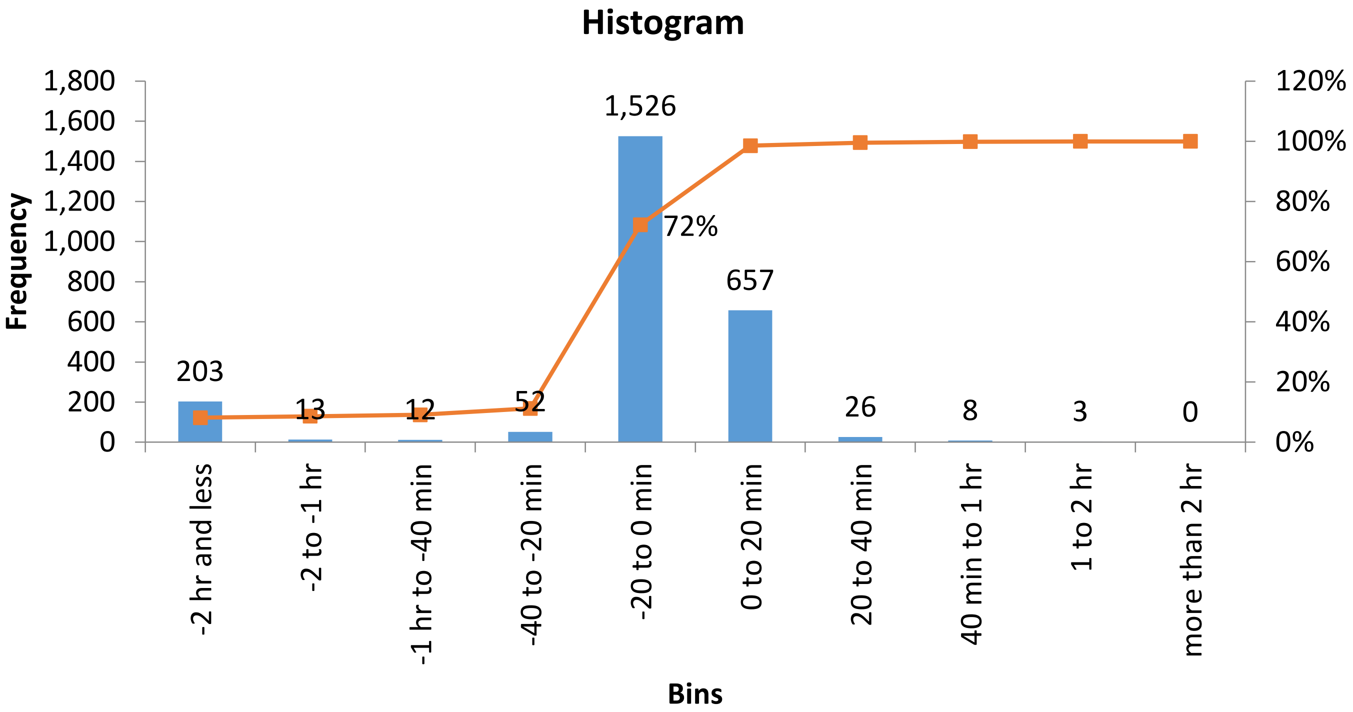


Mode Choice and Transit Routing Integration

Better filters eliminate infeasible drive-to-transit queries

- A Dijkstra-based one-to-all routing algorithm is written
- From every driving link in the multimodal network, we find the shortest distance to a transit stop and store this as a link attribute.
- There are around 56,000 driving links in the Chicago network. The overall process takes less than 30 seconds.
- This gives the mode choice component to filter out the drive-to-transit option if the driving time to the closest transit stop is “too close” to the overall duration of the trip.
  - If driving to the train station is 40 minutes, and the traveler would only travel in transit for 5 minutes, it is not reasonable to choose-park-and-ride. The person would drive for their entire trip.
- This relative upper bound on driving is agent-specific, hence we can allow the above behavior for certain travelers.

Validation of Walk-to-Transit Trips



Comparison with Google

- 2,500 random queries
- A **negative** value in the x-axis means that the **Argonne algorithm finds shorter paths**
- Argonne algorithm found shorter trips for 72% of the queries
- (1526+657)/2500 = 87% of the results are within ± 20 minutes.

Argonne NATIONAL LABORATORY

Validation of Park-and-Ride Trips

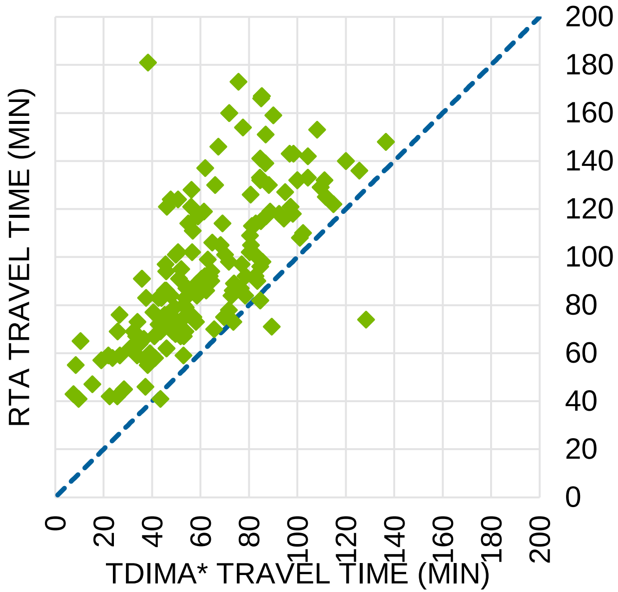
- Data
  - Chicago Metropolitan Agency for Planning (CMAP) 2008 Travel Tracker survey
  - 180 PNR trips out of 6,500 valid transit trips

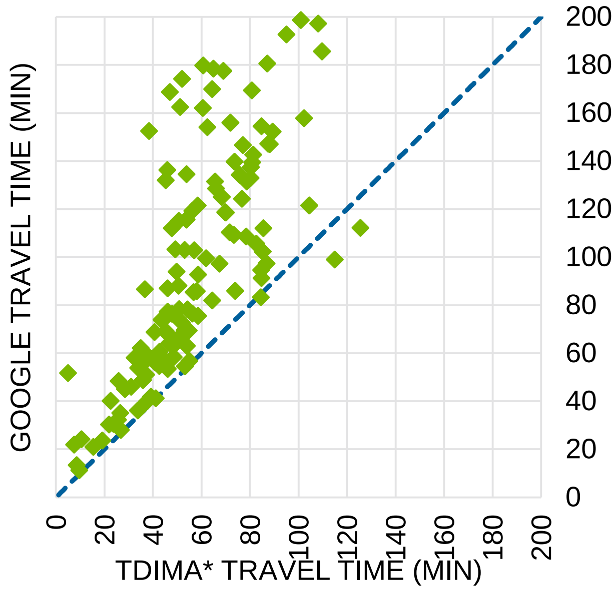
- Scenario Runs
  - TDIMA* algorithm using PNR mode
  - Google Directions API using transit mode
  - Regional Transit Authority (RTA) trip planner using PNR mode

Validation Runs

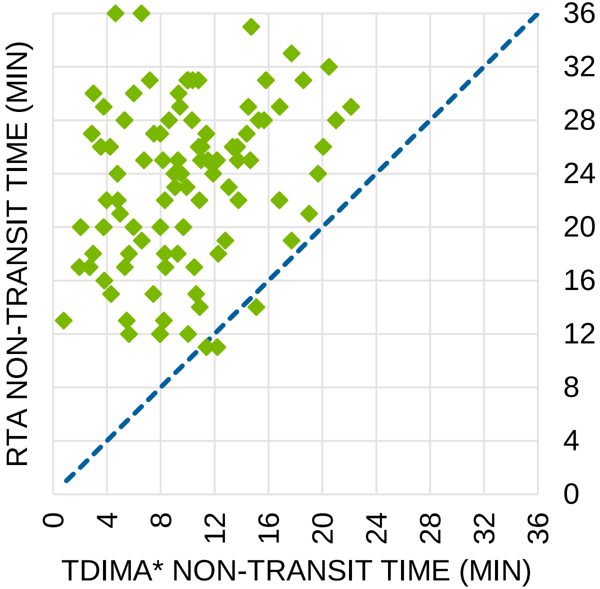
- TDIMA* Parameters
  - Base transfer penalty; $x = 5$ minutes,
  - Weight of time spent waiting; $w_\omega = 2$,
  - Weight of time spent walking; $w_\kappa = 2$,
  - Weight of time spent in a transit vehicle; $w_\lambda = 1$,
  - Weight of time spent in a traffic vehicle, $w_\mu = 3$,
  - Upper bound on individual waiting time; $U^\omega = 60$ minutes;
  - Upper bound on total walking time; $U^\kappa = 60$ minutes; with walking speed $v_\kappa = 1.39$ m/s; $U^\kappa = 5$ km in distance,
  - Upper bound on total time in a traffic vehicle: The remaining estimated cost

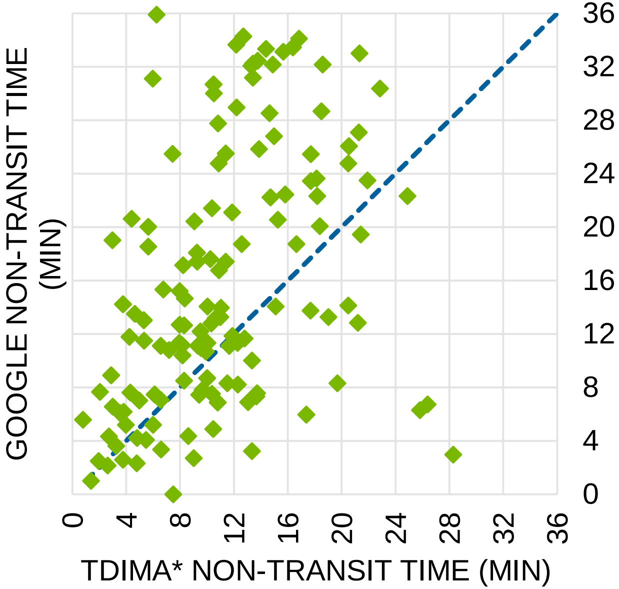
Total Travel Times



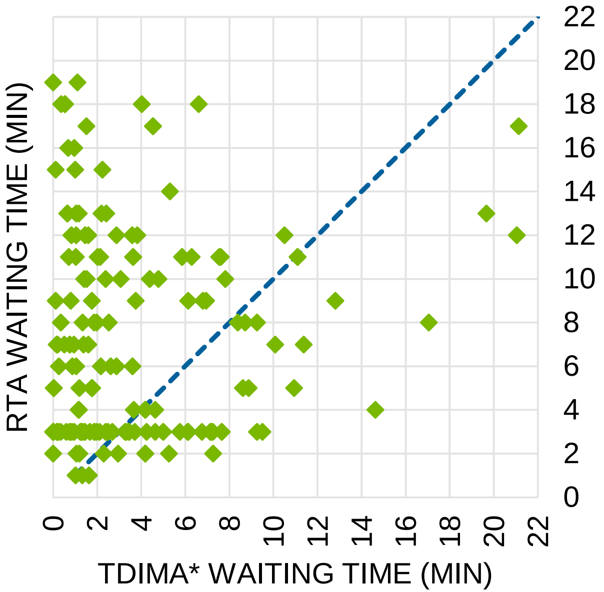


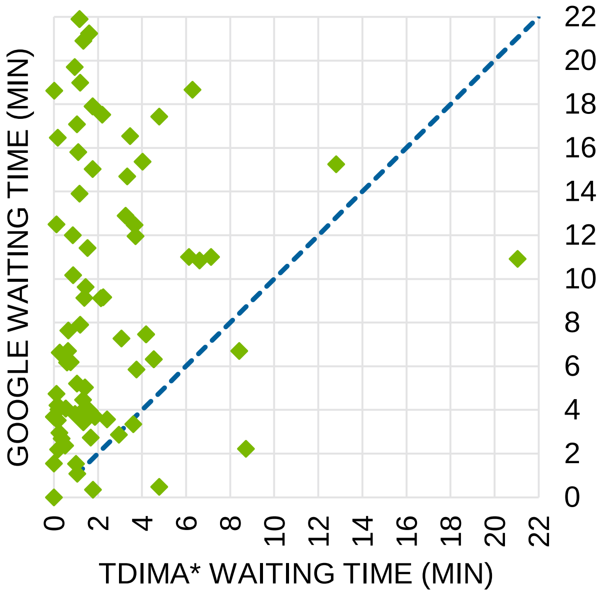
Total Non-Transit Time



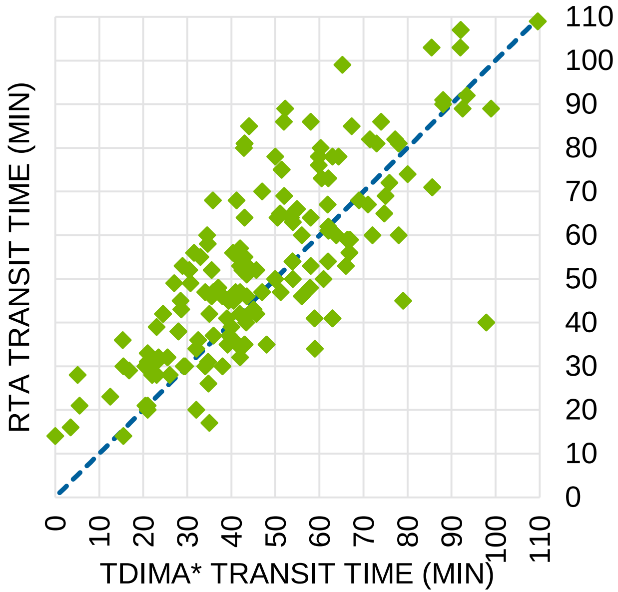


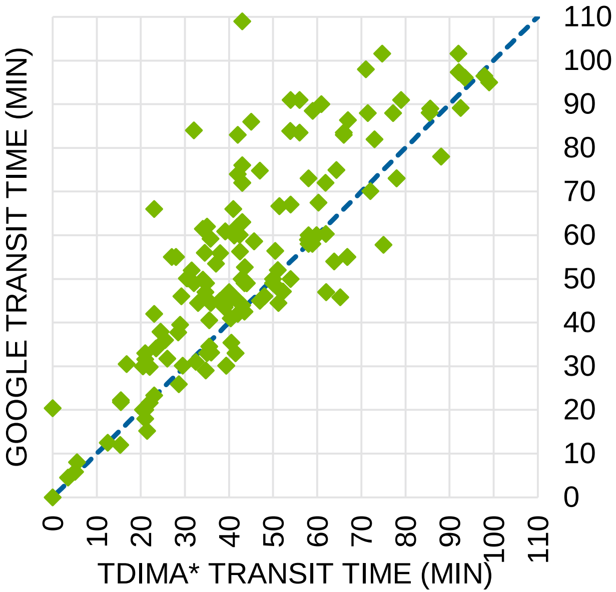
Total Waiting Time





Total In-Transit Vehicle Travel Time





Questions?

- Thank you very much for your time!
- https://github.com/anl-polaris/polaris
- https://polaris.es.anl.gov/