Approximately Orchestrated Routing and Transportation Analyzer (AORTA)

Large-scale Traffic Simulation for Autonomous Vehicles

Dustin Carlino, Mike Depinet, Piyush Khandelwal, and Peter Stone
Department of Computer Science
The University of Texas at Austin
{dcarlino,msd775,piyushk,pstone}@cs.utexas.edu
AORTA

- In **5 minutes**, start simulating human and autonomous traffic in **any** city
- Investigate interactions between intersection policies, routing, and agent behavior
- Extend the simulator with new modules

Open source: code.google.com/p/road-rage
Implementation

- Scala, a language on top of the JVM with functional programming constructs and a more powerful type system
- Java compatibility
- Modular design: implement a stop sign policy in 60 lines
Maps

- AORTA's map model: a digraph with lanes and turns
- OpenStreetMap: a moderated, user-editable map of the world
Results from importing OSM

Number of lanes inferred from OSM way “type”
Results from importing OSM

Ultimately, OSM could encode turns at intersections
Results from importing OSM

Ultimately, OSM could encode turns at intersections
Simulation Overview

- Microscopic (each agent has its own route and behavior)
- Discrete time (fixed time-steps)
- Continuous space (no tiling of roads)
- Deterministically replay any scenario
- Intersections locked at the granularity of sets of compatible turns
Introducing Agents

- In the UI, select the source and destination regions
- Specify the rates of introduction
- Demand files easy to incorporate
Using AORTA for your work

- Agent behaviors: react every time-step
- Routing strategies: decide where to go
- Intersection policies: control when agents may turn
Agent Behaviors

- Implement a driver model
- The default behavior is stateless and performs worst-case analysis, always avoiding collisions and obeying intersections
  - Extend it to simulate human drivers
  - Tune it to optimize fuel consumption or hypermile
Route Strategies

- Pathfind from start to goal
- Can delegate heavier computation to a thread pool
- Static A*
- Drunken walk (for fun)
- Hierarchical planning
- Future idea: consult a more centralized manager for coordination, dynamically re-plan
Intersection Policies

- Agents will statelessly poll as they approach, sending desired turn and current state (position, velocity, acceleration)
- Policies must decide STOP or GO in any way that avoids collisions
- Currently: stop signs, traffic signals, greedy autonomous reservations
Demo: How to use

1) Install Scala and AORTA: code.google.com/p/road-rage
2) Export your favorite city from openstreetmap.org to XML
3) Import the OSM map into AORTA
4) Set up scenarios with the GUI and experiment
   Or run headless and visualize later
5) Post-process trip time and intersection throughput CSV data
Future Work

- **Simulator changes**
  - Distribute workloads over a cluster
  - Investigate vehicle-to-vehicle protocols for agent behaviors

- **Applications built on AORTA**
  - Coordinated routing
  - Contraflow [Hausknecht et al]
Related Work

- Other agent-based microscopic simulators based on OSM
  - MATSim [Balmer et al]
  - SUMO [Behrisch et al]
- AIM [Dresner et al]
Thanks!

- Contact me: dabreegster@gmail.com