SE 450: Object-Oriented Software Development
Traffic Flow Simulation Final Project
Due 17 March 2004

Introduction

The Final Project gives you the opportunity to apply the object-oriented software development skills you have acquired during the course to a significant but moderately-sized application. The following sections describe the general behavior of the application, the simulation, and the deliverable requirements for the project.

The final project for this class is a simple, flow-based traffic simulation. The O-O paradigm was founded in simulation and is an excellent means for exercising its strengths. The Simula programming language, followed by the Smalltalk environment, together pioneered the development and dissemination of O-O languages, and were first and foremost designed as simulation languages.

Project Description

The traffic simulation application should have the following features:

- The program should be started and run from the command line.
- After the initial program banner is displayed, the program should sequentially display each model parameter (see the following subsection for a list of these) with a default value. The program should then allow the user to enter a new, validated value for any of these parameters.
- When all model parameters have been displayed and optionally modified, the program should provide the user with the option to save the set of currently entered model parameter values as the default values.
- The program should then prompt the user with the option to start the simulation or to exit.
- If the user chooses to exit, the program terminates.
- If the user chooses to run the simulation, a graphical display window should appear, showing the map view of the simulation. (See Figure 1 for a sample layout.) The map view should show the following:
    - The road structure and directions of travel on each road segment.
    - The location of traffic signals.
    - The locations of all vehicles currently within the boundaries of the simulation.
    - The current time step, size of a time step Δt (in seconds), and the number of vehicles within the boundaries of the simulation map.
- At this point, the simulation begins and the map view of the simulation should graphically show the following:
    - The position of all vehicles within the simulation boundaries during the current time step. Vehicle icons should be to scale.
    - The state of each traffic signal (red, yellow or green) during the current time step.
    - The current values of the current time step, Δt, and the number of vehicles within the boundaries of the simulation map.

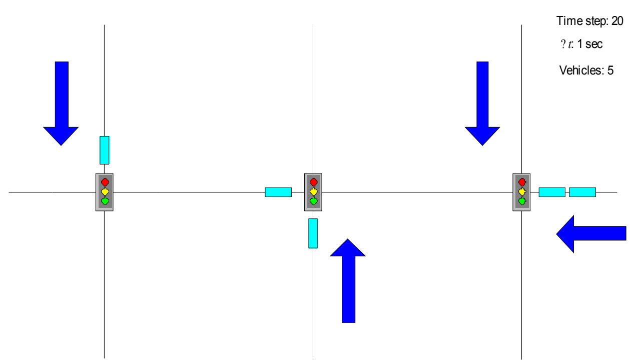

Figure 1. Sample simulation map view

- The simulation should then run to completion, updating the map view at each time step. When the last time step is reached, the text showing the time step should change from black to red to indicate the simulation has completed.

Simulation Constraints, Behavior, and Parameters

- The simulation must observe the constraint of mass conservation. That is, any vehicle that enters the boundaries of the simulation must eventually leave the boundaries of the simulation. No vehicle may spontaneously appear or disappear within the boundaries of the simulation.
- The simulation must observe the exclusion principle. That is, two vehicles cannot occupy the same space within the simulation boundaries. This means that collisions are not modeled.
- The simulation must use MKS (meter, kilogram, second) units.
- All roads are single-lane, one-way thoroughfares.
- A vehicle stays on the road upon which it enters the simulation. This means there are no vehicle turns.
- The following are parameters that should be defined in your model and adjustable in the initial program text dialog:
  - *Frequency of vehicle arrival*. This should be adjustable for all four simulation entry points.
  - *Vehicle length*. Use 5 m as the default.
  - *Maximum vehicle velocity*. This is the parameter c_{limit} discussed in the next major bullet item. Use a value of 50 km/h (≈ 31 mph) ≈ 13.9 m/s.
  - *Maximum obstruction influence distance.* This is the parameter D discussed in the next major bullet item. Use a value of 15 m.
  - *Minimum distance from an obstruction.* This is the parameter D_{min} discussed in the next major bullet item. Use a value of 1.5 m.
  - *Signal timing*. A *timing profile* is simply a triplet of numbers that specify the length of time that a signal remains in a particular color state. For example, the triplet (30, 5, 45) indicates that the signal will be in the red state for 30 s, in the yellow state for 5 s, and in the green state for 45 s. Be sure the corresponding cross street signal has a consistent profile, in this case (50, 5, 25). The three signals (left, center, and right) should have independently adjustable timing profiles.
  - *Simulation time step*. Use 1 s as the default.
  - *Number of simulation steps*. This represents the total number of simulation time steps that will be executed. When this number is reached, the simulation has completed.
  - *Road segment length*. This represents the length in meters of the individual road segments. Use 100 m as the default.
- A vehicle i has a velocity v_i governed by the following equations:

$$v_i = c_{limit} \qquad \text{for } x \geq D \tag{1}$$

$$v_i = \mathbf{m}x + \mathbf{b} \qquad \text{for } D_{min} < x < D \tag{2}$$

$$v_i = 0 \qquad \text{for } x \leq D_{min} \tag{3}$$

  where c_{limit} is a constant vehicle velocity (speed limit), x is the distance from vehicle i to any obstruction (such as another vehicle or a red traffic signal) in front of it, D is the maximum distance at which an obstruction has an influence on vehicle i, and D_{min} is the distance at which vehicle i stops for the obstruction.

By means of the boundary conditions at $x = D_{min}$ and $x = D$, we solve for the slope $\mathbf{m}$ and intercept $\mathbf{b}$ and derive a single linear equation to govern the velocity of the vehicle. Skipping the details of the derivation, we get:

$$v_i = (c_{limit} / (D - D_{min}))(x - 1) \qquad (4)$$

Thus, a vehicle travels at a constant speed c_{limit} until it encounters an obstruction at a distance D, at which point its velocity decreases linearly until it stops at a distance D_{min} from the obstruction. Conversely, when the obstruction is removed (such as a traffic signal turning green) or the obstruction increases its distance by moving away, the vehicle increases its speed using the same governing equations. The velocity profile corresponding to equation (2) is shown in Figure 2.

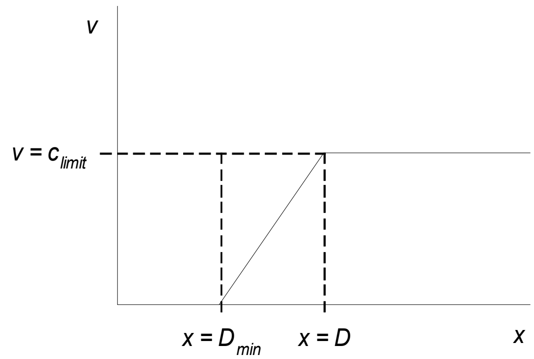

Figure 2. Vehicle velocity as a function of distance from obstruction.

- Traffic signals have three states: red (stop) which is treated as an obstruction, green (go) which is treated as a non-obstruction, and yellow, which is treated as an obstruction if the vehicle is at a distance $D_{min} < x \leq D$ and which is treated as a non-obstruction if the vehicle is at a distance of $x > D$ or $x \leq D_{min}$.