A Distributed Simulation of a Road Traffic System Applying HLA

Pichaya Tandayya, Totsaporn Keawmanee, Kaned Seubpradit, Jedsada Pengsuwan, Latthaphon Phorphon
Department of Computer Engineering, Faculty of Engineering
Prince of Songkla University, Thailand
Overview

Two main points:

• Design and implementation
• Performance findings
Outline

• Introduction
• Road Vehicle Simulation
• Object Model Template for a Vehicle
• Map Structure and Database
• Database of Traffic Rules and Situations
Outline (Cont.)

• Wrapper Class for System Interface
• Performance Analysis
• Reducing Bandwidth with Threads
• Conclusion and Future Work
Introduction

• High Level Architecture (HLA)

• Road Traffic Simulation
HLA

- A common framework for distributed modelling and simulation where
  - a simulation node is called a federate.
  - a federation is group of federates that participates in the same exercise.
HLA (Cont.)

HLA is defined by:

- Run-time Infrastructure (RTI)
- An Object Model Template (OMT)
- Rules
Other Works

• Most works on HLA are military-based and run on dedicated networks
• Most traffic simulations concentrate on the macroscopic level
What’s about this work?

- Microscopic road traffic simulation
- Run on a non-dedicated network

What are the differences?
Motivation

• Test a driver’s reactions under many different situations.
• ‘Learn by doing’ in simulated situations
What do we need to implement?

- 3-D graphical models
- map database
- vehicle simulation models
- database of traffic rules and situations
## Structure of a Road

<table>
<thead>
<tr>
<th>Road ID</th>
<th>Starting &amp; End Points</th>
<th>Width</th>
<th>No of Lanes</th>
<th>Equation</th>
<th>Connected Paths</th>
</tr>
</thead>
</table>

Figure 1 Structure of a road
Map Structures and Database

Road map:
• vectors of connections
• created and destroyed on the fly
• support the visualisation of multiple maps
# Road Paths and Connections

<table>
<thead>
<tr>
<th>Map No</th>
<th>Path No</th>
<th>Path Type</th>
<th>Lane No</th>
<th>Location</th>
<th>Direction</th>
</tr>
</thead>
</table>

**Figure 2 Data structure for the Road Paths**

<table>
<thead>
<tr>
<th>Map No</th>
<th>Connector No</th>
<th>Connector Type</th>
<th>Path No</th>
<th>Path No</th>
<th>…</th>
</tr>
</thead>
</table>

**Figure 3 Data Structure for Road Connections**
Vehicle Behaviours

Two types of behaviours:

• parameter-driven behaviours
  – basic equations of a vehicle's movement
• rule-and-situation-based behaviours
Object Model Template of a Vehicle

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Integer</td>
<td>Vehicle’s identification</td>
</tr>
<tr>
<td>Type</td>
<td>Integer</td>
<td>Type of vehicle</td>
</tr>
<tr>
<td>Color</td>
<td>Integer</td>
<td>Color of vehicle</td>
</tr>
<tr>
<td>Width</td>
<td>Double</td>
<td>Width of vehicle</td>
</tr>
<tr>
<td>Length</td>
<td>Double</td>
<td>Length of vehicle</td>
</tr>
<tr>
<td>Height</td>
<td>Double</td>
<td>Height of vehicle</td>
</tr>
<tr>
<td>SpeedLimit</td>
<td>Double</td>
<td>Speed limit of vehicle</td>
</tr>
<tr>
<td>Acceleration</td>
<td>Double</td>
<td>Acceleration of vehicle</td>
</tr>
<tr>
<td>Speed</td>
<td>Double</td>
<td>Speed of vehicle</td>
</tr>
<tr>
<td>Radius</td>
<td>Double</td>
<td>Radius of vehicle for collision checking</td>
</tr>
<tr>
<td>Position</td>
<td>2D vector</td>
<td>Position of vehicle</td>
</tr>
<tr>
<td>Orientation</td>
<td>2D vector</td>
<td>Orientation of vehicle</td>
</tr>
<tr>
<td>Turn Angle</td>
<td>Double</td>
<td>Turning angle of vehicle</td>
</tr>
<tr>
<td>RoadID</td>
<td>Integer</td>
<td>ID of the road that the vehicle is located</td>
</tr>
<tr>
<td>NextRoadID</td>
<td>Integer</td>
<td>ID of next road</td>
</tr>
<tr>
<td>RoadID</td>
<td>Integer</td>
<td>ID of road</td>
</tr>
</tbody>
</table>

Table 1 Object Model Template for a Vehicle
Traffic Rules and Situations

Traffic rules

Rule-based / event-driven behaviours

Traffic light

Traffic signs

Road map database

Driver’s decision

signal & location

Other vehicles

Figure 4 Information that Affects a Driver’s Decision
Road Vehicle Simulation

- Detecting and avoiding collision
- Vehicle following
- Following the road, changing lanes and turning
- Overtaking
Detecting Collision

- A collision occurs when $R_1 + R_2 \leq D$

Figure 5 Bounding Spheres
Bounding Spheres

- no collision without being detected
- not all intersections mean collision
- therefore, further detection is needed
Avoiding Collision

Collisions are avoided by:

• examine larger circular regions to check if the two objects are close to each other.

• then, use options such as
  – stopping
  – reducing or increasing speed
  – lane changing
Vehicle Behaviours

- Vehicle following
  - speed synchronisation
- Orientation changing
  - changing road and lanes, and turning
- Overtaking
  - based on speed limits, speeds and locations of vehicle, and space
Other Behaviour Handlings

Traffic elements:
- Traffic Light
- Traffic rules
- Situations
System Interface

Figure 6 CSimulator Interface Wrapper Class

A Distributed Simulation of a Road Traffic System Applying HLA
Test Bed

- RTI-NG 1.3
- Windows 2000
- Network
  - non-dedicated 100 Mbps (public)
Performance Analysis

- attribute size varies but latency is stable
- no. of objects linearly affects latency
- throughput:
  - no update message loss
  - updating and receiving rates change in opposite to the size of the attribute; small data size suggested for frequently updates
Performance Analysis (Cont.)

• No. of objects affects the updating and receiving rates
  – rates drop to about 1/5 after second object
• stable ownership transfer
• transmitting and receiving
  TimeAdvanceGrant:
  – stable at 19 Grants/Second
Reducing Bandwidth with Threads

- RTI perceives updates of a single registered object
- involved federates must agree
- applied to objects of the same registered class
Figure 7 The Look of the Road Traffic Simulation
Conclusion

• Design and implementation
  – road map database
  – vehicle's model template
  – driver decision making
• one simulated object per federate
• Use of threads to reduce bandwidth usage
Future Work

- Full 3D
- more traffic elements and situations