

KING'S COLLEGE LONDON

TRAFFIC SIMULATOR

GROUP PROJECT

Team Diversity

Members:

Balázs Kiss

Eddy Mukasa

Gabb Visessmit

Pongsakorn N. Riyamongkol

Snorri Hannesson



February 8, 2015

1 Project description

1.1 Background

Over the past few decades, the world's population has been continuously increasing. This increase in population has resulted in overwhelming traffic which is becoming a serious problem in most major cities around the world. Cities like London, Bangkok, and New York are faced with serious traffic congestion challenges. To be able to handle these challenges and make the traffic run smoother many different traffic management policies have been tried. Before these policies can be implemented in the real world they are tested on traffic simulators. These simulator are supposed to be abstract models of the real world, so if a traffic management policy works on the simulator it probably works in the real world.

Traffic Simulator project is a part of coursework in Group Project (7CCSMGPR) of King's College London. The aim of this course is to develop a software product: planning, designing, implementing, and reporting. Hence, the Traffic Simulator matches to the aim of Group Project Course.

1.2 Objectives

- To develop a traffic simulator program which has structure, as followed two types of vehicles, three types of drivers, functional road system with many roads and lanes, junctions, intersections and traffic lights.
- To compare with two different traffic management policies which are fixed time policy and congestion control policy.
- To examine how the system reacts in a time of emergency by simulated an ambulance to the system.

1.3 Scale

- Each road can have multiple lanes, which can be in the same or opposite direction.
- As in Britain the traffic is left-lane oriented.
- The system will have cars and buses.
- Driver behaviour can be cautious, reckless, or normal.

1.4 Outline

Description : The Traffic Simulator is a program that simulates traffic and provides two different traffic management policies to relief traffic congestion issues. The two policies are fixed time policy and congestion control policy. These policies can be compared by average time each vehicle is in the system. The system has both cars and buses and drivers behaviour can be cautious, reckless or normal. Moreover,

an emergency strategy is used to see how the system reacts when an ambulance is injected to the system. Therefore, the Traffic Simulator main goal is to reduce the traffic congestion on roads.

General Concept :

1. Vehicle Types: There are two types of vehicles: cars and buses. Cars go faster and have higher acceleration than buses. The shape and size of the bus is bigger than car, so it takes more space on the road.
2. Driver Behaviour: The drivers can behave in three different ways: cautious, reckless, and normal as illustrated in the table below. Reckless driver is more prone to change lanes and overtake other vehicles, if a fast vehicle is stuck behind a slow vehicle the fast vehicle will try to overtake the slow vehicle.

Type	Car (X pixels)	Bus (2X pixels)
Cautious	Acceleration: Y Top Speed: X	Acceleration: Y - 10 Top Speed: X - 20
Normal	Acceleration: Y + 10 Top Speed: X + 20	Acceleration: Y Top Speed: X
Reckless	Acceleration: Y + 20 Top Speed: X + 40	Acceleration: Y + 10 Top Speed: X + 20

3. Emergency Strategy: When an ambulance is injected to the system it gets highest priority. That means that cars on the right lane will change to the left lane if possible and the ambulance gets to pass at the intersection first.
4. Traffic Management Policy: There are two different traffic management policies:
 - Fixed Time
 - Congestion Control

With the **Fixed Time Policy** we have will have peak hours (when the traffic density is the highest) and off-peak hours (when the traffic density is normal). On off-peak hours the duration of the green light will be X seconds before it changes to red. On peak hours the duration of the green light will be 2X seconds before it changes to red.

The **Congestion Control Policy** will always have the duration of green light in each direction unless the system detects a congestion. When a congestion is detected in some direction the duration of the green light will be increased in that direction for one time to avoid traffic jam.

The two different policies will be compared by average time each vehicle is in the system. Each vehicle will have a timer that starts when it enters the system and gets written to a log when the vehicle exits the system. The

average time that a vehicle is in the system is calculated and the the policy that generates lower average time is considered better.

1.5 Methods and Strategies

- Software Development: developed in Java programming language.
- Graphical design: JavaFX
- Version control: GitHub.
- Documentation: LaTeX.
- Management: Trello.
- Assessment: graded by Burger's Algorithm and Peer Assessment.

1.6 Schedule

The time period is divided into three iterations:

Iteration 1:	16th of January - 12th of February
Iteration 2:	13th of February - 5th of March
Iteration 3:	6th of March - 26th of March

Then is each attribute to in the simulator either mandatory or optional. The colours represent in which iteration each attributed should be done:

Mandatory:	Roads	Lanes	Multiple vehicles	Different driver behaviour	Junctions and intersections	Traffic lights
Optional:	Different traffic management policies	Time logging	Emergency Strategy			

1.7 Expectation of Project Outcome

- Traffic simulator program is developed and implemented.
- Traffic simulator program may give the best traffic management policy (fixed time or congestion control).
- Traffic simulator program can also use in the emergency case.

2 Project organisation

2.1 Project management

A slightly adjusted Agile Methodology is used for this project. We have divided the time period for this course into iterations and at the end of each iteration we want to inching forward towards the final product.

2.2 Roles

Balázs Kiss: Lead programmer

Eddy Mukasa: Architect

Gabb Visessmit: Graphical designer

Pongsakorn N. Riyamongkol: Project Manager

Snorri Hannesson: Tester and Coordinator

Each member of the group has a responsibility to oversee one aspect of the project but is not expected to do all the work defined in his role. I.e. each member should/could do some programming but the lead programmer should oversee the code and make sure nothing is missing and everything is done properly. The same goes for the other roles.

2.3 Collaboration

We have physical meeting every Thursday at 10 o'clock. We have a log of all meetings on Github. Other means of collaborations are:

Facebook is used as a communication channel.

GitHub is used for version control.

Trello is used for project management..

2.4 Peer Assessment and Self Assessment

This involves members of our group to evaluate other members regarding the performance of their tasks. In addition, self-assessment is used to grade yourself in term of duties and accomplishment tasks. Peer and Self Assessment can be used for both formative and summative purposes. There are some techniques about peer and self assessment, as followed:

- Assessment Form
- Work done in GitHub and Trello.

2.5 Conflicts

If we have a conflict during this project, we have defined a simple method to resolve it:

- I. Realise conflict
- II. Handle conflict sooner rather than later.
- III. Find the solution together - democracy.
- IV. Apologise
- V. Appreciate