CISC 204 Modelling Project Report

Group 12: Michael Cassidy, Kieran Green, Cooper Moses, Mike Stefan

**Project Summary**

Our project aims to solve the possible different routes a vehicle could take to get to a destination given a grid of intersections which is randomly generated with red-lights, one-way roads, two-way roads, and busy pedestrian traffic. The grid of intersections will be given the number of rows and columns and the number of one-way roads per row and column. This number is easily changeable and will make it easy to test wildly different test cases. The one-way roads will randomly be given a direction based on a bool value given to them. For one-way roads going north and south, 1 will mean that vehicles can only travel north and 0 means that vehicles can only travel south. For one-way roads going east and west, 1 will mean that vehicles can only travel east and 0 means that vehicles can only travel west.

The rest of the roads will be assumed to be two-way roads. An 2D-array will then be made to fit the size of the grid. This will serve as the map the car will drive through. Each index in the array will serve as an intersection which has generated rules based on the roads that cross there. Some examples of these rules are a light blocking traffic going either North/South or East/West, as well if the intersection is at the edge of the grid the car will also not be able to go off the map.

**Propositions:**

* Mx, y: where x and y correspond to the grid location that car is at.
* Gx, y: position of the goal the car must get to.
* L: the car turns left at the intersection
* R: the car turns right at the intersection
* S: the car goes straight at the intersection
* E: the car has reached a dead end
* W: the car has reached the goal
* C: the light is red in front of the car
* Di: Where i is the direction the car is facing (N, E, S, W)

**Constraints:**

* (G2, 2 M2, 2) W
  + If the car is at the goal, then the car has reached the goal.
* (~C  **~** ((DN Mk,2) (DS Mk,0)( DE M2, k)( DW M0, k))) S
  + The car goes straight if the light is not red, and the direction of the car is not pointed off the map. K represents any value that correlates to the map. The example takes place in 3x3 grid with the bottom left corner being (0, 0) and the top left corner being (2,2), so k is a integer between 0 and 2. If it is at the bottom of the map then it’s position is Mk, 0 and if it is direction is DS (South). That means if it goes straight, it will go off the map so it is not allowed to go straight.
* (~S ~L ~R) E
  + If the car cannot move then it has reached a dead end.

**Model Exploration**

* How we made our projects and the steps we took

**Jape Proof Ideas:**

NOTE: jape proofs are just if the turns are possible, they do not take into account the path that needs to be taken or that one turn must be selected, they just use the Boolean of if the option is available.

Because we can’t use full words in jape, we define our variables with (some are not used in the examples):

* T = left
* R= right
* S = Straight
* G= green
* B = busy (cars)
* P = busy (people)
* A = left turn advance
* D = lights
* E = stop sign
* H = one-way (no crossroads)
* Q = four-way
* F = four-way (with one way across)
* C = T-shaped road

1. The idea is that: on a four-way road given that a left turn is possible, a right turn is possible and a straight-through is never possible, the light must be an advance on a non-busy street.

This means we get the following using only the necessary constraints:

¬S, T, Q, R, T→(A∨(¬B∧G)), G→S,(A∧B)→¬R ⊢ A∧¬B

Left implies an advance or it's not busy and green.

Green implies you can go straight.

advance and busy implies you cannot turn right as there will be cars turning into that lane.

1. The idea is that: if you can turn both straight and right but not left, there is no light, and it is not people busy, it must be a stop sign on a T-shaped road or a four-way with a one-way street.

The equation for this problem is: S, R, ¬T ,¬D , ¬P, R→¬H, (R∧¬D)→(E∧¬P), (E∧¬P∧¬T)→(C∨F∨H) ⊢ (C∧E)∨(F∧E) with the constraints as ordered:

right turn implies it is not a one-way street with no crossroads.

if you can go right and there are no streetlights, it must be a stop sign and not people busy.

if there’s a stop sign and it is not people busy, but you cannot turn left, it must be the street layout having no lane for left making it either a one-way (with crossroads) or a T-shaped road configuration.

1. The idea is that: at a four-way If you can go straight and right, the road must be car-busy, but not people busy. The light must also be green to go straight.

The Equation for this is: R, S, D, ¬T, (R∧G)→¬P, (¬T∧G)→(P∨B), (S∧D)→G ⊢ B∧¬P

The constraints are as follows:

If you can go right and it is green, it cannot be people busy

If you cannot go left and its green, it must be people or car busy

If you can go straight and there's lights, the light must be green

**First-Order Extension**

* What we want to improve on.

**Requested Feedback:**

* Are the Jape propositions correctly done formatting/idea wise?