Automated Synchronized Traffic Control System

Introduction:

Traffic Control System is one of the major application of AI. For different situations of the rode different traffic control systems were made. But most of them are not applicable in real life situation. Till now we need human resources to control our traffic system. In our project we try to build a system that will control traffic lights of each junction from traffic data from each road of that junction.

In our system we use simple graph technique to represent the model. Each node represents a junction and each edge represent a rode. For simplicity we assume our graph is like a 2D grid. Each rode is associated with two values, i) Number of vehicle in that road, ii) Importance of that road. If the importance is higher then probability of giving that particular rode green signal is higher. In each junction maximum one rode can have green signal. Our system opens one road from each intersection such that the sum of congestion of open roads across all intersection and sum of importance of open roads is maximized. We take special care so that no road can monopolized a junction. We do this by means of keeping track of which road was opened how many times in last 100 turns and if one road is kept open for too long, its kept shut even if it has more importance or more congestion.

We use a python program to take input of the roads traffic and importance data. Then we save those data in a data file (.dzn format). Using this data file the Minizinc model finds the optimal solution and gives it back to python program . And again using python we show which road have green light and which have red light for every junction. Our demonstration has a simple simulation of how keeping a road open affects its surrounding road traffics. This is only for demonstration purposes and in a real deployment this would have to be replaced with real world data taken through sensors / cameras.

Language and Software requirement:

- Python 3
- Minizinc
- PyMzn

Algorithm:

Our approach was to model the road systems using graphs. The nodes are intersections and the edges are roads. Each road has a congestion value, which can be thought of as a number of car to length or road ratio. Each road also has an importance value associated with it. We try to open only the roads that are most congested and also the most important. For that we have used Minizinc. Our main.py file builds a dzn (data) file named "datafrompython.dzn" and then sends the data to a Minizinc model called "main_solver.man". The Minizinc model sees a adjacency matrix representation of the road graph. It tries to find the optimum combination of open roads with a few constraints applied (i.e. only one road per intersection can be open at once, the sum of open road congestion and sum of open road importance had to be minimized together, and no road can monopolize any intersection etc.) . After finding the optimum decision, it sends this decision back to "main.py" , which uses this decision to show which road in each intersection needs to be kept open (green) and which are kept closed (red).

The Minizinc model "main_solver.mzn" requires its search space to be made small so that it can be implemented in a real time system. The model "maximum_calculator.mzn" does just that. It limits the search space for a few variables and sends them back to "main.py" which uses these values when calling "main_solver.mzn" to limit the search space.

How to run our Minizinc model:

1) Unzip the attached zip file

   Or

   Get the project source code by running this command in git bash: *git clone* *https://github.com/Farabi-shafkat/automated-synchronized-traffic-control-system-using-minizinc.git*

2) Install PyMzn : *pip install PyMzn*
3) Run the code using: python "main.py"
   Important: **navigate to the directory of the project before running it.**
4) In python editor console give input number of intersections you have in your model.
5) Then you have to build the graph. You have to give which junction is directly connected with other junction using road.
6) Along you have to give importance value and congestion value of each road. sample input:

```
9
2 11 1 4 10 2
1 10 3 3 4 4 5 4 5
2 3 6 6 26 7
1 2 8 7 13 9 5 15 10
2 5 11 4 14 12 8 30 15 6 17 16
3 25 17 5 16 18 9 18 19
4 12 20 8 21 21
5 31 22 7 22 23 9 4 24
8 2 25 6 19 26
10
```

7) Then you have to give after how many second you want to change the lights of each junction. Then after each period of time your output will be shown. If you want to stop the system then press Ctrl+C
8) The output should look like this:

```
for intersection 1 ============================>>>>>>>>>
  road towards intersection 2 congestion:11 importance: 1 decision :## RED ##
  road towards intersection 4 congestion:10 importance: 2 decision :## GREEN ##
-------------------


for intersection 2 ============================>>>>>>>>>
  road towards intersection 1 congestion:10 importance: 3 decision :## RED ##
  road towards intersection 3 congestion:4 importance: 4 decision :## RED ##
  road towards intersection 5 congestion:4 importance: 5 decision :## GREEN ##
-------------------


for intersection 3 ============================>>>>>>>>>
  road towards intersection 2 congestion:3 importance: 6 decision :## RED ##
  road towards intersection 6 congestion:26 importance: 7 decision :## GREEN ##
-------------------


for intersection 4 ============================>>>>>>>>>
  road towards intersection 1 congestion:2 importance: 8 decision :## RED ##
  road towards intersection 7 congestion:13 importance: 9 decision :## GREEN ##
  road towards intersection 5 congestion:15 importance: 10 decision :## RED ##
-------------------


for intersection 5 ============================>>>>>>>>>
  road towards intersection 2 congestion:5 importance: 11 decision :## RED ##
  road towards intersection 4 congestion:14 importance: 12 decision :## RED ##
  road towards intersection 8 congestion:30 importance: 15 decision :## RED ##
  road towards intersection 6 congestion:17 importance: 16 decision :## GREEN ##
-------------------


for intersection 6 ============================>>>>>>>>>
  road towards intersection 3 congestion:25 importance: 17 decision :## RED ##
  road towards intersection 5 congestion:16 importance: 18 decision :## RED ##
  road towards intersection 9 congestion:18 importance: 19 decision :## GREEN ##
-------------------
```

Conclusion:

This is a prototype model of how traffic lights can be controlled using AI. If we want to implement this in real life scenario then we need to scale up our system. We also need to take input from different sensors that will give condition of each road in real time. There is whole lot of improvement possible but we could not do due to lack of time.