Intelligent Traffic Congestion Reduction System using Proximity Sensors

Palash Gandhi  
Department of Information Technology  
St. Francis Institute of Technology  
Mumbai, India

Aakash Chandrana  
Department of Information Technology  
St. Francis Institute of Technology  
Mumbai, India

Tushar Karia  
Department of Information Technology  
St. Francis Institute of Technology  
Mumbai, India

ABSTRACT
Traffic congestion is one of the major issues faced by many cities of the world. The causes for such congestion are the rapid growth in the number of vehicles and lack of proper infrastructure. In this study, we propose an intelligent traffic congestion reduction system using proximity sensors, which can be used at the crossroads of metropolitan cities for reducing traffic congestion. It can also be used for detecting vehicles and managing and controlling signaling system.

Keywords
Traffic congestion, signaling system, proximity sensors, traffic congestion reduction

1. INTRODUCTION
In India, the automobile industry has rapidly caught on to the growth of the transport sector and manufactures approximately 4.6 million vehicles every year. Experts predict that by 2020, there will be 450 million vehicles plying on Indian roads. Currently, the total number of vehicles in India stands at about 100 million. Most of the vehicles are owned in the urban areas with a maximum density in the metropolitan cities. [1]

In almost all cities, two wheelers constitute nearly 70% of the vehicle population. Due to higher income levels and flexibility in urban areas, more automobiles are owned and operated. Urban areas constitute more than 90% of automobiles. [2]

Although the government is trying its level best to overcome this issue, the results have been unsatisfactory when it comes to the real time implementation of traffic management-related projects as far as scope, cost, and feasibility are concerned.

This study aims to describe how intelligent traffic congestion reduction system (ITCRS) can be used to manage and control traffic in metropolitan cities. Studies indicate that drivers in the 68 largest urban areas in the U.S. experienced an increase in traffic delays due to congestion from 11 h/year in 1982 to 36 h/year in 1999. [3] The estimated cost of traffic congestion in these areas totaled $78 billion, representing a cost of 4.5 million additional hours of travel and 6.8 billion gallons (25.7 billion liters) of wasted fuel. [3] The average rush-hour trip takes 32% more time than the same trip taken during nonrush-hour conditions. Congested travel periods (rush hours) in major cities have doubled in less than 20 years, increasing from nearly 3 h (morning and evening combined) in 1982 to approximately 6 h in 1999. [3] Congestion is observed almost half of the daylight hours on workdays. [4]

ITCRS uses an array of proximity sensors at a certain threshold distance from the signals and detects vehicles as they halt at the signal. When these sensor arrays detect traffic at the signal, it uses a priority assigning algorithm to assign the priorities to all the signals by detecting and scheduling the sensor arrays. Vehicles need not be modified in this technique. The system uses artificial intelligence, which consists of sensors to sense the environment. In the proposed system, the array of proximity sensors are the sensors that are placed at a threshold distance from the signals, the knowledge base is the algorithm through which priorities are assigned to each signal, and the actuators are the signals and the control system which acts based on the perceived environment.

ITCRS is a simple yet effective tool to reduce this issue. It requires a sensor array of proximity sensors and a microprocessor-based system to control and assign the priorities to signals. Each intersection will have their own independent ITCRS, irrespective of the other intersections in that area. Detection of vehicles is performed on a real time basis and does not require any modifications to be made to the vehicles.

2. PROPOSED SYSTEM
2.1 Intelligent traffic congestion reduction system
Traffic in metropolitan cities can be benefit from several applications of the system. Some of the applications of the proposed system are as follows:

1) Intersection signal priority assigning can be used to detect traffic at each direction and assign priorities to the signals.
2) Intelligent traffic management of traffic flow can be achieved using this system, and accurate travel time can be estimated.

ITCRS does not require any alterations or additions to the vehicles as it simply detects the traffic using the sensor array placed at a threshold distance.

1) Sensor Array: The sensor array consists of a predefined number of proximity sensors, which varies for each intersection. When the number of sensors that are detecting vehicles increases more than a threshold value, the array transmits a command to the system. This command is in turn used to assign priorities to these signals.

The proposed sensor is a proximity sensor used to detect nearby objects without any physical contact. It transmits electromagnetic radiation and observes for changes in the field or return signal. Modulated infrared (IR) proximity

34
sensor is a combination of IR sensor and a 38-kHz square wave generator. Unlike the simple IR proximity sensor, this sensor provides high protection against normal sunrays and can be effectively used for all indoor/outdoor conditions. [5]

A polymer IR proximity sensor array can be used as the array of proximity sensors. A near-IR proximity sensor array is achieved by integrating a polymer light-emitting diode and a polymer photodetector. [6]

2) ITCRS Control System for the Signals: ITCRS control system is a microprocessor-based system, which uses the proximity sensor arrays to evaluate the traffic conditions. It decides the priorities to be assigned to the signals. Instead of the traditional fixed signal timings, this system will have variable timings for each signal based on the priorities. The command consists of binary values, i.e., 0s and 1s, that would indicate which direction the sensors detect traffic. For example, if the north direction at a certain crossroad has activated sensors, then the command sent to the system will be 1.

As we use a sensor array [0000] (default), the first element of the array represents the proximity sensors in the north direction, second for the east, third for the south, and fourth for the west.

The sensors will only sense when one signal turns green. This means that when north is on the remaining three, sensors arrays will start sensing and accordingly send a command to the system. There are three sensors arrays, and hence eight possible combinations of the sensor commands can be transmitted to the system.

Figure 1 indicates the sequence the system will follow to turn the lights green at an intersection that is in the clockwise cyclic manner. For example, when North and South detect traffic, the north signal turns green and then the south. After this, the system will not jump directly to West, but it will first turn East green and then West so as to complete one cycle.

In addition, a constraint has been added for the number of times a signal can turn green in a single cycle, which is two times. North cannot turn green more than two times unless West has turned green at least once. The west signal turning green indicates the end of one cycle. This applies to all the signals. This is performed using a count array, which keeps a count of the number of times each signal has turned green at a particular crossroad for one cycle. At the end of each cycle, this count array gets reset to 0. (countarray[0] = 0, countarray[1] = 0, countarray[2] = 0, and countarray[3] = 0).

Table 1. Possible combinations of the sensor commands that can be sent to the system

<table>
<thead>
<tr>
<th>Binary values scanned by the sensor arrays</th>
<th>Command interpretation by the system</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>This means that none of the signal sensors detect traffic</td>
</tr>
<tr>
<td>100</td>
<td>East detects traffic</td>
</tr>
<tr>
<td>010</td>
<td>South detects traffic</td>
</tr>
<tr>
<td>001</td>
<td>West detects traffic</td>
</tr>
<tr>
<td>110</td>
<td>East and South detect traffic</td>
</tr>
<tr>
<td>101</td>
<td>East and West detect traffic</td>
</tr>
<tr>
<td>011</td>
<td>South and West detect traffic</td>
</tr>
<tr>
<td>111</td>
<td>This means that East, West, and South detect traffic</td>
</tr>
</tbody>
</table>

Figure 2 represents an intersection with proximity sensor arrays placed at a threshold distance “d” from the signals in each direction. The four signals will also be connected to the ITCRS control system, which would assign priorities to each one of them.

3. ALGORITHM

1. Start

2. Declare an integer array for the sensor array that will detect which sensors are activated at a particular time.

3. Declare signal variables for each direction (North, South, East, and West), and initialize them to 0.

4. Declare flag array for counting the number of times each signal turns green. Flag[0] counts the number of times the north signal turns green. Flag[1] counts the number of times the east signal turns green. Flag[2] counts the number of times the south signal turns green. Flag[3] counts the number of times the west signal turns green.

5. Make signal_north = 1 and flag[0] = 1.

6. The signal_north = 1 checks the status of the other sensors from the sensor_array.
7. If all the other sensors apart from the north sensor are 0 or 1, then the signal functions normally.

Once the north signal turns red again, check the sensor array. Furthermore, check if flag[1] is less than or equal to 2, so that each signal opens not more than two times in one cycle.

8. The following table lists the actions to be performed for the different combinations that can be obtained by the sensor array.

<table>
<thead>
<tr>
<th>Combinations for (sensor_array[1], sensor_array[2], sensor_array[3])</th>
<th>Actions to be performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Make East green</td>
</tr>
<tr>
<td>010</td>
<td>Make south green</td>
</tr>
<tr>
<td>001</td>
<td>Make west green</td>
</tr>
<tr>
<td>110</td>
<td>Make east green, and then make south green</td>
</tr>
<tr>
<td>101</td>
<td>Make east green, and then make west green</td>
</tr>
<tr>
<td>011</td>
<td>Make south green, and then make west green</td>
</tr>
</tbody>
</table>

9. Before making a jump to any other signals using the above table if the value of flag[] at any of the signals exceeds the value 2, then skip that signal and go to the next signal in the order of N–E–S–W. Repeat the steps from 5 by making the next signal green for a new cycle by resetting all the elements of the flag array to 0, and also the combinations to be scanned will change according to the signal which is currently running.

4. RESULTS

For the result analysis, we assumed that there is an intersection where the north and south directions detect the most traffic, whereas the east and west do not. We assume that the arrival rate of vehicles at the north and south directions are 50 vehicles/minute. We place the sensor arrays at a distance that can detect a maximum of 50 vehicles between the array and the intersection. At the east and west directions, we assume that there are not more than 20 vehicles arriving per minute.

In an ideal scenario where ITCRS is not implemented, the signals are given an equal priority, and they are tuned green for the same time duration. But ITCRS will assign priorities based on the commands it receives from the sensor arrays. For the assumed scenario, north and south detects traffic, and thus the cycle in which the signals turn green will be N–S–N–E–S–W. The following steps indicate why this cycle is performed by ITCRS.

1. North turns green first as that is where the cycle begins. While north is green, ITCRS detects traffic at the south direction.

2. ITCRS assigns a priority to south, and thus south turns green. While south is green, ITCRS detects traffic at north.

3. ITCRS again assigns a priority to north, and thus north turns green.

4. As north has already turned green two times in the same cycle, ITCRS will not turn north green again regardless of whether it detects traffic or not.

5. Thus to complete the cycle, east is turned green.

6. While east is green, ITCRS detects traffic at both south and north. But as north cannot be turned green because of the constraint of one signal turning green not more than two times in one single cycle, ITCRS turns south green.

7. As south has also turned green two times now in one cycle, ITCRS will not turn south green again, and thus west is turned green, and the cycle is completed.

8. The entire process repeats itself at the completion of each cycle.

5. CONCLUSION

ITCRS proves to be an effective system for reducing traffic congestion. The implementation of this system will not only save time but also the congestion costs related to traffic. It can be used to efficiently manage traffic without much human intervention. Each system acts as an independent system at each intersection. All the problems related to traffic congestion such as performance of vehicles in terms of fuel intake are reduced, which results in low costs. Moreover, traffic can have behavioral effects on drivers who are stuck in traffic.

6. REFERENCES


