Smart Traffic Systems Using IoT

Travis Warling

Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

April 20, 2019
Issues with Traffic Congestion
Issues with Traffic Congestion

[Map of Minneapolis traffic congestion]

Travis Warling (Division of Science and Mathematics University of Minnesota, Morris Morris, Minnesota, USA)
Issues With Traffic Congestion

In the US:

- 1.9 billion gallons of fuel per year is wasted due to congestion
  - This is equivalent to 5 days’ worth of all fuel used in the US [4]
Issues With Traffic Congestion

In the US:

- 1.9 billion gallons of fuel per year is wasted due to congestion
  - This is equivalent to 5 days’ worth of all fuel used in the US [4]
- Average urban commuters are stuck in traffic congestion for 34 hours and waste $713 of fuel per year [4]
Issues with Traffic Congestion

In India:
- 600 billion Rupee (8.7 billion USD) is lost per year due to traffic congestion [7]
Issues with Traffic Congestion

In India:

- 600 billion Rupee (8.7 billion USD) is lost per year due to traffic congestion [7]
- The average speed of National Highway 44 is 20 kmph (12.43 mph) even though the speed limit is 100 kmph (62 mph) [7]
How can Smart Traffic Systems Help?

Dynamically change traffic light cycles in areas of congestion
How can Smart Traffic Systems Help?

Dynamically change traffic light cycles in areas of congestion
  - Longer green light times in the direction of traffic congestion
How can Smart Traffic Systems Help?

Dynamically change traffic light cycles in areas of congestion

- Longer green light times in the direction of traffic congestion
- Shorter green light times in directions leading into areas of congestion
How can Smart Traffic Systems Help?
Outline

1. Background
2. Raspberry Pi Based System
3. Sensor and Server Based System
4. Conclusion
1. Background
   - Internet of Things (IoT)
   - Raspberry Pi
   - General Purpose Input/Output (GPIO)

2. Raspberry Pi Based System

3. Sensor and Server Based System

4. Conclusion
Internet of Things (IoT)

- Basic concept: any device can be connected to networks or to other devices
Internet of Things (IoT)

- Basic concept: any device can be connected to networks or to other devices
- IoT allow smart traffic systems to:
Basic concept: any device can be connected to networks or to other devices

IoT allow smart traffic systems to:

- Have sensors that communicate with each other
Internet of Things (IoT)

- Basic concept: any device can be connected to networks or to other devices
- IoT allow smart traffic systems to:
  - Have sensors that communicate with each other
  - Have sensors that connect to and receive commands from networks
Raspberry Pi

- Single-board computer
Raspberry Pi

- Single-board computer
- Uses Linux operating system
Raspberry Pi

- Single-board computer
- Uses Linux operating system
- Can be configured to send and receive signals from the internet
General Purpose Input/Output (GPIO)

- Pins that are placed at the top edge of a Raspberry Pi
General Purpose Input/Output (GPIO)

- Pins that are placed at the top edge of a Raspberry Pi
- Do not serve a specific purpose - can be customized for specific needs
General Purpose Input/Output (GPIO)

- Pins that are placed at the top edge of a Raspberry Pi
- Do not serve a specific purpose- can be customized for specific needs
- Each pin outputs a voltage when set to high
Outline

1. Background

2. Raspberry Pi Based System
   - Adding Cameras to Traffic Lights
   - Raspberry Pi Implementation
   - Testing and Results

3. Sensor and Server Based System

4. Conclusion
Raspberry Pi Based System Overview

- Basuni et al.
- Requires humans to continuously interact with the system once installed
Raspberry Pi Based System Overview

- Basuni et al.
- Requires humans to continuously interact with the system once installed
- Uses Raspberry Pis and cameras to allow authorities to monitor traffic congestion and change traffic light times
Adding Cameras to Traffic Lights

- Cameras are placed on top of every traffic light within an intersection
Adding Cameras to Traffic Lights

- Cameras are placed on top of every traffic light within an intersection
- Pictures are taken and sent to a Raspberry Pi
Adding Cameras to Traffic Lights

- Cameras are placed on top of every traffic light within an intersection.
- Pictures are taken and sent to a Raspberry Pi.
- The length of time between pictures can be customized by authorities.
A Raspberry Pi placed on every traffic light within an intersection
A Raspberry Pi placed on every traffic light within an intersection

Has two main functions:
Raspberry Pi Implementation

- A Raspberry Pi placed on every traffic light within an intersection
- Has two main functions:
  1. Uses a wireless connection to a web server to transmit the camera feed.
A Raspberry Pi placed on every traffic light within an intersection

Has two main functions:

1. Uses a wireless connection to a web server to transmit the camera feed.
2. Changes the green light length of the traffic light duration
   - Two GPIO pins are used to allow for three different green light durations
Testing and Results

- System has not been tested
Testing and Results

- System has not been tested
- Basuni et al. strongly believe the system will reduce traffic congestion to some degree
Outline

1 Background

2 Raspberry Pi Based System

3 Sensor and Server Based System
   - Placing Sensors on Roads and Cars
   - Base Stations
   - Green Light Time Calculation Algorithm
   - Testing and Results

4 Conclusion
Sensor and Server Based System

- Chong et al.
- Doesn’t require human interaction once installed
Sensor and Server Based System

- Chong et al.
- Doesn’t require human interaction once installed
- Uses sensors placed within road surfaces and vehicles to detect traffic congestion and automatically change traffic light cycles
Sensors are placed within the road surface and on vehicles.
Placing Sensors on Roads and Cars

- Sensors are placed within the road surface and on vehicles
- Sensors communicate in an ad hoc system
Sensors are placed within the road surface and on vehicles.

Sensors communicate in an ad hoc system:
- Ad-hoc = components are able to connect to each other without the use of a router or base station.
Placing Sensors on Roads and Cars

- Sensors are placed within the road surface and on vehicles
- Sensors communicate in an ad hoc system
  - Ad-hoc = components are able to connect to each other without the use of a router or base station
- Three types of ad hoc sensor networks
Placing Sensors on Roads and Cars

Ad Hoc On-Road

Ad Hoc On-Vehicles

Ad Hoc Hybrid

wireless communication

sensors
vehicle
Placing Sensors on Roads and Cars

Ad Hoc On-Road  Ad Hoc On-Vehicles  Ad Hoc Hybrid

wireless communication  sensors vehicle
The data collected by the sensors in both the road surface and in the vehicles within the vicinity is sent to a base station via wireless connection.
Base Station

- The data collected by the sensors in both the road surface and in the vehicles within the vicinity is sent to a base station via wireless connection.
- The base station serves two functions:
Base Station

- The data collected by the sensors in both the road surface and in the vehicles within the vicinity is sent to a base station via wireless connection.
- The base station serves two functions:
  1. Compiles the data from the sensors and sends it to a cloud server to be analyzed.
Base Station

- The data collected by the sensors in both the road surface and in the vehicles within the vicinity is sent to a base station via wireless connection.

- The base station serves two functions:
  1. Compiles the data from the sensors and sends it to a cloud server to be analyzed.
  2. Receives commands from the cloud server and change green light times in traffic lights.
Data sent to the cloud server is analyzed for signs of traffic congestion.
Green Light Time Calculation Algorithm

- Data sent to the cloud server is analyzed for signs of traffic congestion
- The algorithm calculates the level of traffic congestion present and determines how long the green lights should be on in the direction of the congestion
Green Light Time Calculation Algorithm

- Data sent to the cloud server is analyzed for signs of traffic congestion.
- The algorithm calculates the level of traffic congestion present and determines how long the green lights should be on in the direction of the congestion.
- The algorithm calculates green light times for each traffic light in an intersection.
Green Light Time Calculation Algorithm
Current Traffic Volume ($CTV$) is calculated

$CTV$ is the total number of cars that can theoretically travel through the intersection within an hour

$$CTV = \frac{(n \times 3600)}{C}$$
Green Light Time Calculation Algorithm

- Current Traffic Volume ($CTV$) is calculated
- $CTV$ is the total number of cars that can theoretically travel through the intersection within an hour
- $n$ is the number of cars queued in a given direction

$$CTV = \frac{(n \times 3600)}{C}$$
Green Light Time Calculation Algorithm

- Current Traffic Volume (CTV) is calculated
- \( CTV \) is the total number of cars that can theoretically travel through the intersection within an hour
- \( C \) is the traffic light cycle duration

\[
CTV = \frac{(n \times 3600)}{C}
\]
Current Traffic Volume ($CTV$) is calculated

$CTV$ is the total number of cars that can theoretically travel through the intersection within an hour

3600 is the number of seconds in an hour

$$CTV = \frac{n \times 3600}{C}$$
Current Traffic Volume ($CTV$) is calculated

$CTV$ is the total number of cars that can theoretically travel through the intersection within an hour

\[ CTV = \frac{(n \times 3600)}{C} \]

\[ CTV = \frac{(4 \times 3600)}{30} \]

\[ CTV = 480 \]
Flow Ratio ($FR$) is calculated

$FR$ is the percent of traffic congestion present

$$FR = \frac{CTV}{MFR}$$
Green Light Time Calculation Algorithm

- Flow Ratio ($FR$) is calculated
- $FR$ is the percent of traffic congestion present
- $MFR$ is the maximum flow rate for the road in each direction

$$FR = \frac{CTV}{MFR}$$
Flow Ratio ($FR$) is calculated

$FR$ is the percent of traffic congestion present

$$FR = \frac{CTV}{MFR}$$

$$FR = \frac{480}{500}$$

$$FR = 0.96$$
CTV and FR is calculated for every direction going through the intersection
Green Light Time Calculation Algorithm

- $CTV$ and $FR$ is calculated for every direction going through the intersection
- Once completed, the results are ordered and given a priority
  - The direction with the highest $FR$ is given the highest priority.
Green Light Time Calculation Algorithm

Order of priority:

1. Traffic Light 1 \( (FR = 0.96) \)
2. Traffic Light 2 \( (FR = 0.48) \)
3. Traffic Light 3 \( (FR = 0.24) \)
4. Traffic Light 4 \( (FR = 0.24) \)
Green Light Calculation Algorithm

- Effective Green Time \((EGT)\) is calculated in every direction.
- \(EGT\) is the amount of time that is used by vehicles to drive through the intersection every time the light turns green.

\[
EGT = FR \times C
\]
Green Light Calculation Algorithm

- Effective Green Time ($EGT$) is calculated in every direction.
- $EGT$ is the amount of time that is used by vehicles to drive through the intersection every time the light turns green.

\[
EGT = FR \times C
\]

\[
EGT = 0.96 \times 30
\]

\[
EGT = 28.8
\]
The algorithm recursively calculates the remaining three equations for each traffic light, starting with the traffic light with the highest priority.
Green Light Calculation Algorithm

- Phase Green Time \((PGT)\) is calculated
- \(PGT\) is the amount of time that is used by vehicles to drive through the intersection every time the light turns green when outside factors are accounted for

\[
PGT = EGT + S - Y
\]
Green Light Calculation Algorithm

- Phase Green Time ($PGT$) is calculated
- $PGT$ is the amount of time that is used by vehicles to drive through the intersection every time the light turns green when outside factors are accounted for
- $S$ is the lost time due to cars needing to accelerate from a stop after a red light

\[
PGT = EGT + S - Y
\]
Phase Green Time ($PGT$) is calculated

$PGT$ is the amount of time that is used by vehicles to drive through the intersection every time the light turns green when outside factors are accounted for.

$Y$ is the length of the yellow light

$$PGT = EGT + S - Y$$
Green Light Calculation Algorithm

- Phase Green Time ($PGT$) is calculated
- $PGT$ is the amount of time that is used by vehicles to drive through the intersection every time the light turns green when outside factors are accounted for
- $Y$ is the length of the yellow light

\[
PGT = EGT + S - Y
\]

\[
PGT = 27.8 + 1 - 10
\]

\[
PGT = 19.8
\]
Green Light Calculation Algorithm

- Remaining Green Time ($RGT$) is calculated
- $RGT$ is the amount of time that is lost due to outside factors

$$RGT = EGT - PGT$$
Green Light Calculation Algorithm

- Remaining Green Time (\(RGT\)) is calculated
- \(RGT\) is the amount of time that is lost due to outside factors

\[
RGT = EGT - PGT
\]

\[
RGT = 28.8 - 19.8
\]

\[
RGT = 9
\]


- Remaining Effective Green Time (REGT) is calculated
- \( REGT \) calculates the optimal green light time for a given traffic light

\[
REGT = RGT \times \frac{CRT_{current}}{CRT_{remaining} - CRT_{previous}}
\]
Remaining Effective Green Time ($REGT$) is calculated. $REGT$ calculates the optimal green light time for a given traffic light. $CRT_{current}$ is the FR of the current traffic light being calculated for

$$REGT = RGT \times \frac{CRT_{current}}{CRT_{remaining} - CRT_{previous}}$$
Green Light Calculation Algorithm

- Remaining Effective Green Time ($REGT$) is calculated
- $REGT$ calculates the optimal green light time for a given traffic light
- $CRT_{remaining}$ is the sum of the $FR$ for every traffic light with a lesser priority than the one currently being looked

\[
REGT = RGT \times \frac{CRT_{current}}{CRT_{remaining} - CRT_{previous}}
\]
Remaining Effective Green Time (REGT) is calculated

REGT calculates the optimal green light time for a given traffic light

$CRT_{\text{previous}}$ is the FR for traffic light that was previously looked at

$$REGT = RGT \times \frac{CRT_{\text{current}}}{CRT_{\text{remaining}} - CRT_{\text{previous}}}$$
Green Light Time Calculation Algorithm

- Remaining Effective Green Time \((REGT)\) is calculated
- \(REGT\) calculates the optimal green light time for a given traffic light

\[
REGT = RGT \times \frac{CRT_{current}}{CRT_{remaining} - CRT_{previous}}
\]

\[
REGT = 9 \times \frac{0.96}{(0.48 + 0.24 + 0.24) - 0}
\]

\[
REGT = 9
\]
Once the algorithm finishes calculating $REGT$ the algorithm either:
Green Light Calculation Algorithm

- Once the algorithm finishes calculating \( \text{REGT} \) the algorithm either:
  - Calculates \( \text{PGT}, \text{RGT}, \) and \( \text{REGT} \) for the traffic light with next lowest priority
Green Light Calculation Algorithm

- Once the algorithm finishes calculating $REGT$ the algorithm either:
  - Calculates $PGT$, $RGT$, and $REGT$ for the traffic light with next lowest priority
  - Sets the traffic lights within the intersection to the calculated $REGT$
Green Light Time Calculation Algorithm
Testing and Results

- MATLAB software used to simulate a four-way intersection
Testing and Results

- MATLAB software used to simulate a four-way intersection
- Intersection consisted of cars traveling north, south, east, and west
Testing and Results

- MATLAB software used to simulate a four-way intersection
- Intersection consisted of cars traveling north, south, east, and west
- Maximum queue size of 75 standard vehicles
  - standard vehicle = 4.2 meters (13.78 feet) in length, 1.75 meters (5.74 feet) in width
Sensor and Server Based System
Testing and Results

Testing and Results

- MATLAB software used to simulate a four-way intersection
- Intersection consisted of cars traveling north, south, east, and west
- Maximum queue size of 75 standard vehicles
  - standard vehicle = 4.2 meters (13.78 feet) in length, 1.75 meters (5.74 feet) in width
- 100 light cycles were conducted, with 0 to 100 vehicles traveling in each direction every light cycle
## Testing and Results

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Dynamic Algorithm</th>
<th>Fixed Time Algorithm</th>
<th>Percentage of Improved</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Queue Length (meters)</td>
<td>198</td>
<td>620</td>
<td>68.06%</td>
</tr>
<tr>
<td>Average Waiting Time (seconds)</td>
<td>16</td>
<td>48</td>
<td>66.67%</td>
</tr>
</tbody>
</table>
Two smart traffic systems
- One that requires humans to maintain interaction with the system
- One that is able to run without human interaction
Conclusion

- Two smart traffic systems
  - One that requires humans to maintain interaction with the system
  - One that is able to run without human interaction
- The systems reduce traffic congestion
Conclusion

- Two smart traffic systems
  - One that requires humans to maintain interaction with the system
  - One that is able to run without human interaction
- The systems reduce traffic congestion
- Neither of the systems have been implemented
Acknowledgments

- Thank you to KK Lamberty and Elena Machkasova for their guidance and help throughout the entire process
References


7. P. Rizwan, K. Suresh, and R. M. Babu. Real-time smart traffic management system for smart cities by using internet of things and