EAI Endorsed Transactions
on Ambient Systems
Research Article ICST.ORG

# On Movement of Emergency Services amidst Urban Traffic

Manoj Bode[1], Shashi Shekhar Jha[1], Shivashankar B. Nair[1,*]

[1] Department of Computer Science & Engineering,
Indian Institute of Technology Guwahati,
Guwahati-781039, Assam, INDIA

## Abstract

Managing traffic in urban areas is a complex affair. The same becomes more challenging when one needs to take into account the prioritized movement of emergency vehicles along with the normal flow of traffic. Although, mechanisms have been proposed to model intelligent traffic management systems, a concentrated effort to facilitate the movement of emergency services amongst urban traffic is yet to be formalized. This paper proposes a distributed multi-agent based mechanism to create partial green corridors for the movement of emergency service vehicles such as ambulances, fire brigade and police vans, amidst urban traffic. The proposed approach makes use of a digital network of traffic signal nodes equipped with traffic sensors and an agent framework to autonomously extend, maintain and manage *partial green corridors* for such emergency vehicles. The approach was emulated using *Tartarus*, an agent framework over a LAN. The results gathered under varying traffic conditions and also several emergency vehicles, validate the performance of this approach and its effects on the movement of normal traffic. Comparisons with the *non-prioritized* and *full green corridor* approaches indicate that the proposed *partial corridor* approach outperforms the rest.

Received on 27 February 2015; accepted on 02 April 2015; published on 01 December 2015

**Keywords:** Intelligent Traffic Management, Emergency Services, Mobile Agents, Multi-agent systems, Distributed Intelligence, Tartarus, Emulation



doi:10.4108/eai.1-12-2015.150713

## 1. Introduction

The ever increasing number of motorized vehicles has made commuting in urban areas an agonizing predicament. With the growth of cities and also the standards of living, management of such traffic has become a challenging task. Increase in traffic creates congestions especially at junctions which ultimately translate to delays in reaching respective destinations. Although such delays may be tolerated normally, the same become a serious impediment in case of an emergency. The delay in the movement of *emergency services* such as ambulances, fire brigade and police vehicles amidst Urban Traffic can result in loss of lives and property. The movement of such emergency services thus needs to be prioritized over the flow of normal traffic. This calls for an effective mechanism to facilitate free flow passage of emergency services in heavy traffic scenarios.

The most common method used to ensure a path for an emergency vehicle on the road is the use of a siren. This is a reactive mechanism which only caters to a local change in the traffic scenario within the immediate vicinity of the moving emergency vehicle. Although this mechanism is easy to implement, it does not guarantee a congestion free route. One naïve, effective and yet proactive mechanism to ensure a free flow of such emergency services is to create a *full green corridor* along the complete route of an emergency vehicle. A *full green corridor* essentially means blocking all other traffic that are orthogonal to the route of the emergency vehicle until its passage. The method has an adverse impact on the flow of adjoining traffic. In case of multiple emergency services moving in different directions, the task of making *full green corridors* for all such vehicles

* Corresponding author. Email: sbnair@iitg.ernet.in

translates to a multi-objective optimization problem. Hence, a mechanism for the movement of emergency services which is proactive and has least effect on the adjoining traffic forms the need of the day.

With advent of the digital era, advanced devices and smarter gadgets are being used to manage traffic in urban spaces. Cities are becoming smarter; using sensor networks to analyze traffic conditions and make quick and smart decisions on-the-go. State-of-art traffic management systems [17] use various approaches to manage traffic within the city. These approaches can be broadly classified into *centralized* and *distributed* approaches [20]. In a centralized approach, sensors collect local traffic information and send them to a central server. This information is used by the server to carry out massive computations to eventually manage the traffic. This approach calls for sophisticated infrastructure with high speed computational capability and bandwidth. The approach also implies a single point of failure. Further, making centralized decisions can be time consuming and may not be feasible in real-time scenarios.

Infrastructure of a traffic scenario is by nature distributed with a large number of concurrent and asynchronous processes and events occurring simultaneously. A distributed approach to manage traffic in urban areas thus seems more appropriate to realize a better traffic management system. Researchers have used multi-agent systems [9] to model the traffic management infrastructure effectively. These systems inherently provide enhanced features such as autonomy, adaptability, asynchronous processing, localized decisions, etc. which make them an ideal candidate for realizing distributed infrastructures. Although quite a few multi-agent based mechanisms have been proposed in the literature [5], to the best of our knowledge there seems to be no focussed work towards prioritizing the movement of emergency services within a traffic management system.

In this paper, we propose a multi-agent based mechanism to facilitate the movement of emergency services amidst urban traffic. The proposed mechanism creates partial green corridors *en route* the emergency vehicles with minimum impact on the movement of adjoining traffic. Extensions to our previous work [2] using a detailed and formal model of the proposed approach along with exhaustive experimentation and results have also been described.

The next section provides a background on the available multi-agent based approaches for traffic management. In succeeding sections, we describe the proposed approach for the movement of emergency services and its related dynamics. Further sections discuss the emulation experiments and the results obtained. The last section concludes the paper citing scope for future work.

## 2. Multi-agent based approaches for Traffic Management

Multi-agent technology, due to its inherent distributed characteristics, provides a natural solution to the highly distributed and dynamically changing traffic management and control scenarios. In [5], the authors have discussed the application of agents to different modes of transport by road, rail and air. They emphasize the power of agent based systems to regulate and improve the performance of traffic and transportation systems. Weyns et al. [19] present an agent-based approach using delegate multi-agent systems for anticipatory vehicle routing to avoid traffic congestions. They extend their approach in [6] with an environment-centric coordination model. In their approach, individual vehicles dispatch lightweight mobile agents for exploring alternate routes to find the shortest path to the destination, based on current traffic conditions. They further use intention agents to confirm the intended travel route on the road infrastructure. This information is used by the situated agents to estimate the future traffic so that they can alter the route of the approaching vehicles. In [15], the author proposes an intelligent travelling assistant based on a distributed model. They use personal agents for each individual traveller to communicate with the driver and the system to provide optimal advice to the former and update stored traffic information in the system. Katwijk and Koningsbruggen [18] present an agent-based model for the coordination of traffic-control and management of instruments. They modeled the traffic instruments as individual intelligent agents to tune their actions at a local level. They demonstrate that traffic management instruments can coordinate their actions to attain a common goal at the network-level using agent based concepts. Balaji et al. [1] attempt to exploit the advantages of evolutionary techniques for traffic management operations and congestion avoidance in Intelligent Transportation Systems. They propose a multi-agent based real-time centralized evolutionary optimization technique for urban traffic management using an evolutionary strategy for the control of traffic signals. Chen et al. [4] have proposed a model to integrate mobile agent technology with multi-agent systems. They have designed a model to enhance the ability of traffic management systems to deal with the uncertainty in dynamic environment. They use a system which facilitates mobility of agents (mobile agents) within a network called Mobile-C [3] to design an agent-based real-time traffic detection and management system. They argue that the use of mobile agents allows the deployment of new control algorithms and operations on-the-fly to respond to unforeseen events and conditions in urban traffic scenarios.

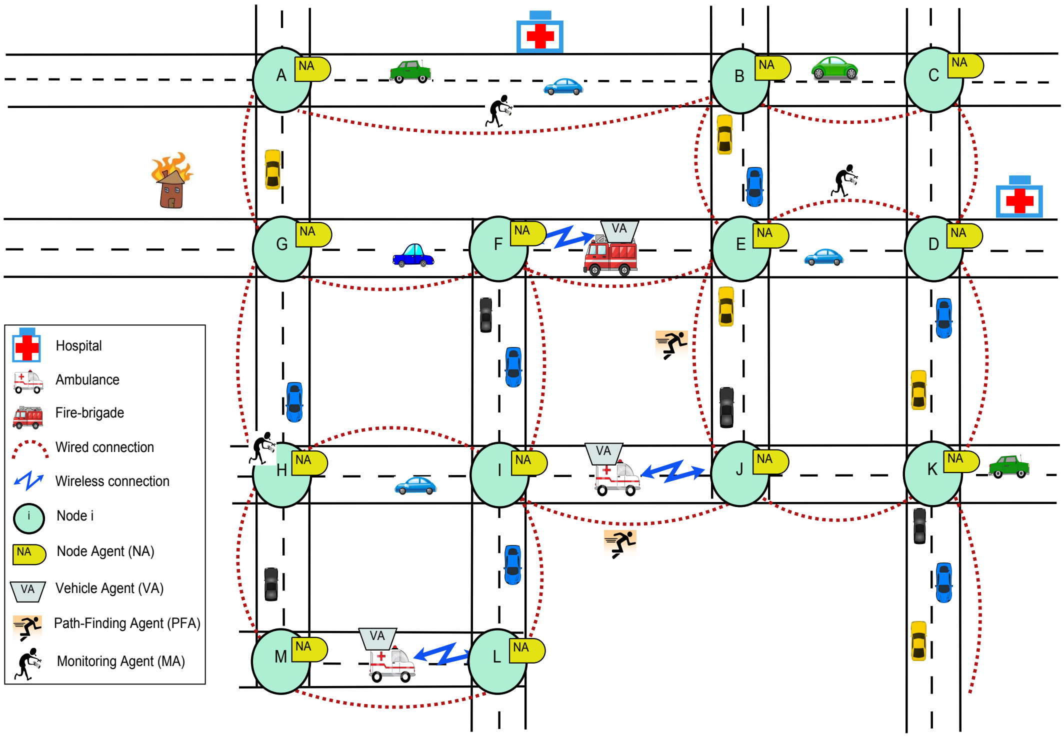



**Figure 1.** A Conceptual view of the Agent based Digital Traffic Infrastructure Network (DTIN)

The agent based traffic management and control approaches discussed so far focus on the city traffic as a whole. They seem to ignore the manner of movement of emergency services amidst general traffic. Handling the seamless flow of such emergency services constitutes a vital requirement in urban traffic. Discovering, managing and maintaining a *partial green corridor* towards the destination for such high priority services is mandatory in today's ever increasing traffic scenarios. In this paper, we present a concerted effort to prioritize the movement of emergency services within the general traffic flow using a mobile agent based multi-agent mechanism.

Mobile agents [8] have been used as an effective tool to realize various distributed applications. Their features such as autonomy, social ability and adaptability [8] along with the capability to migrate to other nodes of a network, carry their execution state and code and also clone provide for all the necessary characteristics of an intelligent distributed mechanism. Due to such features, mobile agents have a wide range of applications ranging from network management, electronic commerce, energy efficiency and metering, wireless sensors, grid computing, distributed data mining, human tracking, security, e-learning, etc. [13]. Martin-Campillo et al. [10] use mobile agents to collect medical data about patients related to allergies and infectious diseases in a mass emergency case, asynchronously. This avoids delay in deciding treatment once the patient reaches the hospital. Pan et al. [14] have used mobile agents to contact hospitals for emergency services for elderly people. These agents carry the health information of patients to notify an ambulance about necessary medicines and equipment required. In the proposed approach, mobile agents are used to disseminate the traffic information along with the creation of *partial green corridors*.

## 3. Generating Partial Green Corridors

The proposed multi-agent based approach uses a Digital Traffic Infrastructure Network (DTIN) constituting an Internet of Things (IoT) of traffic signals and sensors [7]. Figure 1 shows the conceptual DTIN. An agent framework that supports all agent related functionalities such as autonomy, mobility, cloning, asynchronous executions, etc. forms a crucial part of the DTIN. In addition, the DTIN consists of the following components:

1. *Nodes*: Traffic signals connected using wired links are called *Nodes*. Each node comprises an agent framework, the traffic signal, traffic sensors and a digital banner.

2. *Node Agents* (*NA*): The DTIN consists of a set of *Node Agents*. These agents are the static agents situated at every node within the DTIN. They gather local traffic information from traffic sensors and also control their respective traffic signals and the digital banner.

3. *Vehicle Agents* (*VA*): Every emergency vehicle is equipped with hardware running the agent framework. Hence, every emergency vehicle hosts a static agent called *Vehicle Agent*.

4. *Monitoring Agents* (*MA*): This set of mobile agents within the DTIN is responsible for acquiring and updating the flow of traffic information at the nodes. They move around in the network, collect information regarding the quantum of traffic and provide the same to all the nodes within the DTIN.

5. *Path-Finding Agents* (*PFA*): These mobile agents are spawned by the *VA*s. They migrate within the DTIN to construct and ensure a partial green corridor along the shortest path with least traffic load, leading towards the destination of the emergency vehicle.

The node to node communication within the DTIN is assumed to be wired whereas the communication between a *VA* and an *NA* is wireless.

## 3.1. Dynamics of the Monitoring Agents *(MA)*

In the proposed approach, as every *NA* has local traffic information ($Tm_{local}$), the *MA* constructs a traffic flow map ($\Psi$) using this information. The information within $\Psi$ comprises the location of nodes', their current traffic inflow and outflow, traffic load, timestamps, etc. Figure 2 shows a conceptual view of $\Psi$. An *MA* carries $\Psi$ as its payload and migrates within the DTIN to update the same at all the nodes. As a result, every node within the DTIN contains a copy of $\Psi$ termed $\Psi_{local}$. As there may be more than one *MA* within the DTIN, the $\Psi_{local}$ can be updated by different *MA*s at different times. Whenever an *MA* arrives at a node within the DTIN, it communicates with the *NA* at that node to collect the latest $Tm_{local}$ and $\Psi_{local}$. The *MA* then updates its $\Psi$ with the new information received from $Tm_{local}$ and $\Psi_{local}$. In addition, it also updates the $\Psi_{local}$ available with the *NA*. Algorithm 1 depicts the working of the *MA* within the DTIN. Hence, all the *NA*s contain the information about the overall traffic conditions of the DTIN. The total number of *MA*s required within the DTIN depends on the size of the DTIN and the frequency of updates needed at the nodes.

Due to the inherent flexibility of the agent framework, this approach is readily scalable. Hence, if new nodes are added to the DTIN, the *MA* automatically updates their information at other nodes without any reconfiguration cost. The same is true if some nodes crash or are removed from the DTIN. Even multiple *MA*s can communicate within a node to share their information with each other. This can hasten the exchange of information on any sudden change in the traffic flow such as an accident within the DTIN.

| Node | GPS Coordinates <lat, long> | Neighbouring Nodes (NN) | Distance to NN (in KM) | Traffic load towards NN | Timestamp HH:MM:SS |
|---|---|---|---|---|---|
| A | 26.0000° N, 91.0000° E | B | 8 | 3 | 12:01:02 |
| | | G | 3 | 2 | 12:02:26 |
| B | 26.0000° N, 91.0481° E | A | 8 | 4 | 12:01:02 |
| | | C | 4 | 2 | 11:59:34 |
| | | E | 5 | 6 | 12:01:53 |
| C | 26.0000° N, 91.0730° E | B | 4 | 2 | 12:01:22 |
| | | D | 2 | 1 | 12:02:17 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

**Figure 2.** A sample Traffic flow map ($\Psi$)

**Algorithm 1:** Monitoring Agent (*MA*)

**while** *true* **do**
  $Tm_{local} \leftarrow$ get local traffic information;
  $\Psi_{local} \leftarrow$ get local traffic flow map;
  update $\Psi$ with $Tm_{local}$;
  **foreach** $Tm_i$ *collected from node* $i \in \Psi$ **do**
    **if** ***exists*** $local\_Tm_i \in \Psi_{local}$ **then**
      **if** $timestamp\_Tm_i > local\_timestamp\_Tm_i$ **then**
        update $Tm_i$ to $\Psi_{local}$;
      **end**
      **else**
        update $local\_Tm_i$ to $\Psi$;
      **end**
    **end**
    **else**
      add $Tm_i$ to $\Psi_{local}$;
    **end**
  **end**
  set node as visited;
  move to next node;
**end**

## 3.2. Dynamics of the Path-Finding Agent *(PFA)*

As soon as an emergency vehicle decides a destination, the *VA* situated within this vehicle communicates with an *NA* in its vicinity. The *VA* then spawns a *PFA* with the knowledge of the destination location into the DTIN. The *PFA* uses the $\Psi_{local}$ available with the *NA* at that node and constructs the shortest and least crowded path from that node to the destination. The algorithm used by the *PFA* for the construction of such a path is provided by the system administrator. Such algorithms may depend upon the type of emergency vehicle or the complexity of the DTIN.

After calculating the path, the *PFA* informs the subsequent nodes within a distance $\lambda$ (partial window) along the intended path about the arrival of the emergency vehicle. Hence, $\lambda$ decides the length of the *partial green corridor*. The *PFA* then moves from the starting node (where it calculated the path) along the next $\eta$ nodes thus traveling a distance $d$. If $d < \lambda$ then it

**Algorithm 2:** Path-Finding Agent (*PFA*) algorithm

```
Input: η - Minimum number of nodes in partial
       green corridor;
       λ - Partial window;
while local node ≠ destination node do
    if Path R = null then
        calculate path R;
    end
    if not alerted about arrival of emergency
    vehicle then
        alert local node;
        add local node to corridor node list;
    end
    if number of nodes in corridor < η OR corridor
    length < λ then
        move to next node from path list R;
    end
    else
        if change in traffic detected then
            recalculate path R_new from local
            node;
            modify corridor from local node with
            R_new;
        end
        else
            if emergency vehicle passed local node
            then
                extend corridor forward from last
                node in corridor list;
            end
            else
                move to next node in corridor
                node list;
            end
        end
    end
end
```

moves forward to the next node along the path to extend the *partial green corridor*. Once $d \geq \lambda$, then the *PFA* ceases further migration and commences monitoring for possible road jams, sudden blockages, etc. along the stretch of the corridor to ensure that it remains green. Since the *MA* continuously updates the current traffic at all nodes, if the *PFA* finds some blockage or increase in traffic load at a node within the *partial green corridor*, it recalculates the path from that node and modifies the *partial green corridor* accordingly.

The *NA* after receiving the information of the approaching emergency vehicle from the *PFA*, starts broadcasting the route to be taken by the vehicle. The *NA* controls the traffic signals so as to make a *green corridor* for the incoming emergency vehicle. It displays this message on a digital banner at the node along the route thereby alerting nearby traffic. Once the emergency vehicle comes in the vicinity of the *NA*, the *VA* within, receives the intended path from it and informs the *NA* about its transit from that location. Once the *NA* receives this information from the *VA*, it changes traffic signal state and allows other vehicles to move on. Once the *PFA* patrolling along the *partial green corridor* receives the information about the passage of the emergency vehicle along the node, it extends the corridor further. Algorithm 2 portrays the working of the *PFA*.

For illustration, consider an emergency vehicle such as an ambulance shown in Figure 1 which is in the vicinity of node $L$. Assume the destination of the ambulance is node $D$ which is near to the hospital. The *VA* on the ambulance releases a *PFA* into the DTIN via node $L$ using the wireless connection. The *PFA* calculates the route to the node $D$ using the $\Psi_{local}$ with the *NA* on node $L$. The *PFA* finds a route $R$ viz. $L \rightarrow I \rightarrow J \rightarrow K \rightarrow D$. It first moves a distance $d$ up to node $J$ (initial $\eta = 3$) to alert these nodes about the approaching ambulance. If the $d \geq \lambda$, the *PFA* then starts moving to and fro along $L \rightarrow I \rightarrow J$ nodes in that order to ensure a *partial green corridor* along these nodes. While oscillating, it checks for any sudden change in traffic conditions using the $\Psi_{local}$ available at each of these nodes. Once the ambulance passes node $L$, the *PFA* moves forward from $J$ to the subsequent nodes till the $\lambda$ and $\eta$ conditions are satisfied. Suppose the ambulance crosses node $L$ and the traffic in between node $J$ and node $K$ increases suddenly, the *PFA* then recalculates the path from node $J$ to the destination node $D$. Suppose the new path from $J$ is $J \rightarrow E \rightarrow D$, the *PFA* now starts alerting nodes on this new path. The *PFA* expands the *partial green corridor* to $E$ and then moves to and fro to monitor the traffic in the path $I \rightarrow J \rightarrow E$.

## 4. Implementation

To implement the proposed approach for creating multiple *partial green corridors*, we emulated the same using the SWI-Prolog based agent framework nicknamed *Tartarus* developed at the Robotics Lab. of the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, India. *Tartarus* provides all agent based functionalities such as mobility, autonomy, cloning, asynchronous execution and payload carrying capability. *Tartarus* is an advanced version of *Typhon* [11] with enhanced payload carrying capability, thread based execution support and lower hop times. It has been developed over open source SWI-Prolog and can run on heterogeneous systems including Windows, Linux and embedded systems like the *Raspberry Pi* and

*Intel's Galileo*. *Tartarus* also has an interface to control LEGO® MINDSTORM® NXT robots. These make *Tartarus* suitable for realizing applications based on an Internet of Things (IoT) or Cyber Physical Systems (CPS).

An instantiation of *Tartarus* can emulate a node in a network. Such instantiations can be created on either a single computer or different computers to form various overlay network topologies. For our emulation experiments, we created a 500-node *Tartarus* based network over a LAN. These nodes were connected to each other in a grid topology. Thus, the *Tartarus* based grid network emulated the proposed DTIN. The distances between a pair of nodes within the emulated DTIN was initialized randomly. In addition, the traffic flowing within the DTIN was also initialized randomly during the creation of the network. Vehicles (mobile agents) within the DTIN move from different sources to different destinations. The *NA* constituted the static agent on each *Tartarus* instantiation. The *MA* used the *conscientious migration strategy* [12] to migrate from one node to another. The *conscientious migration strategy* evenly distributes the frequency of visits of the *MA* at all the nodes. In the current implementation, the movement of emergency vehicles as also the rest of the traffic was emulated using mobile agents.

We equipped the *PFA* with an $A^*$ [16] based algorithm to calculate plausible routes to the destination using the $\Psi$s available with the *NA*s. Evidently there can be $k$ different paths to reach a destination. To minimize the search space, we have used a heuristic approach. For calculating a path $R$ to destination $D$ from a node $n$, the *PFA* evaluates all possible routes and selects the one with minimum time to traverse. Hence, $R$ is chosen as per Equation 1.

$$R = \min_{\forall i \in k}(\xi(i) + \rho(i)) \tag{1}$$

where $\xi(i)$ is the estimated time to travel from a node $n$ to the next node $i$. $\rho(i)$ is the heuristic to calculate the expected time of traversal from node $i$ to the destination $D$.

The heuristic function, $\rho(.)$, is calculated as the Euclidean distance between the immediate neighbour nodes and the destination $D$ based on their GPS locations multiplied by the average traffic load within the DTIN. For instance, suppose the emergency vehicle has to move from a node $I$ to the destination node $S$, and node $I$ has 3 routes leading to nodes $J$, $K$ and $L$ along the paths to $S$. As per the current traffic estimates at node $I$, suppose it takes $\xi(J)$, $\xi(K)$ and $\xi(L)$ units of time to reach the nodes $J$, $K$ and $L$ respectively from the node $I$. Let $\tau$ be the average time to travel a unit distance within the DTIN and $\alpha(;)$ be the function which returns the distance between two nodes based on GPS coordinates then,

$$R = min\{\xi(J) + \rho(J), \xi(K) + \rho(K), \xi(L) + \rho(L)\}$$

where,

$$\rho(j) = \alpha(j, S)\tau$$

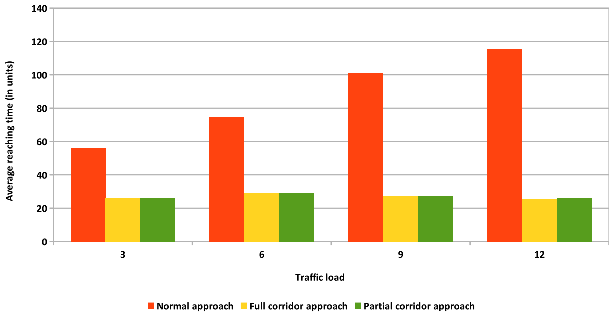
## 5. Results and Discussions

Experiments were performed by varying the number of emergency vehicles and the traffic conditions. Along with the proposed *partial corridor* approach, two more approaches were considered for the experimentation viz.

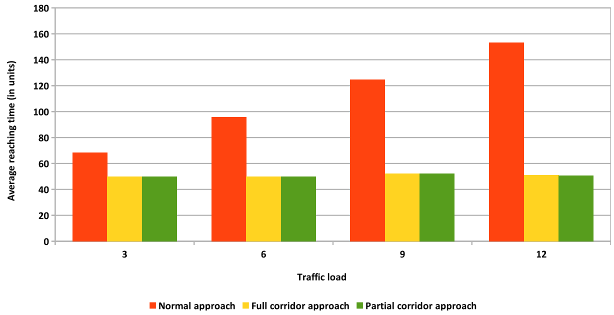
- *Full corridor approach*: In this approach, the *PFA* finds the complete path to the destination and alerts all nodes on that path to create a *full green corridor*. Other vehicles crossing the *green corridor* are forced to wait till the emergency vehicle passes that node.
- *Normal approach*: In this approach, no method is used for alerting the traffic in advance for the movement of emergency vehicle. Hence, the emergency vehicle moves just like any other vehicle in the traffic in a *non-prioritized* manner.

Experiments were performed under different traffic loads. The traffic load is a numeric quantity which signifies the delay overhead in traveling the distance between two nodes within the DTIN. A higher numeric value denotes a higher traffic load and vice-versa. Each experiment was carried out at least 5 times, with different initializations to discard any stochastic effects. The graphs have been plotted by taking the average of the readings gathered from multiple experiments.

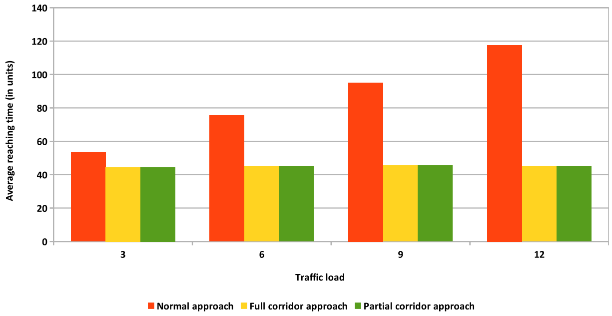
The graphs in Figure 3 show the average time required for the emergency vehicles to reach their respective destinations (*reaching time*) with varying traffic loads ranging from 3 to 12 and number of emergency vehicles ranging from 1 to 15 within the 500-node DTIN. The *reaching time* is defined as the time taken by a vehicle to reach its destination from its starting location. The number of other vehicles that constituted the general traffic in this case was 50. As can be observed, the *reaching time* is least in case of the proposed *partial corridor* and the *full corridor* based approaches. It is always high in case of the *normal* approach and increases with increase in traffic load. The graphs clearly depict that the proposed approach is at par with the *full corridor* based approach facilitating a seamless movement of emergency vehicles. Further, the performance of the proposed approach does not deteriorate with increasing traffic load and number of emergency vehicles.



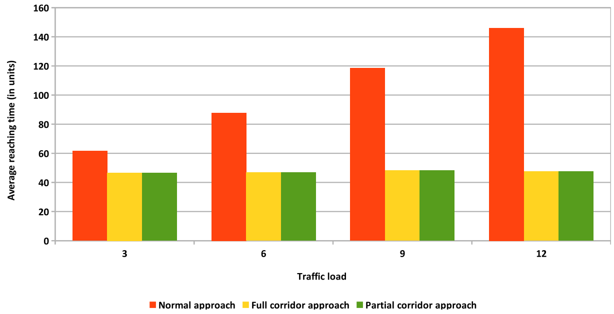

(a) Emergency Vehicle = 1




(b) Emergency Vehicle = 5

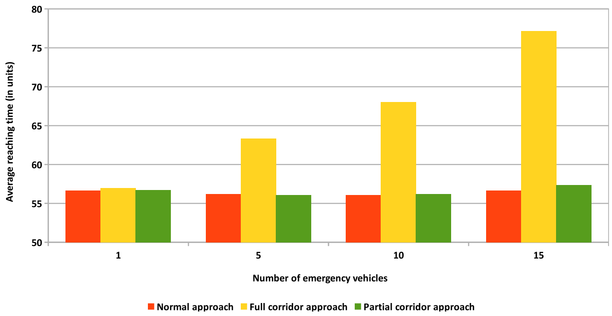



(c) Emergency Vehicle = 10

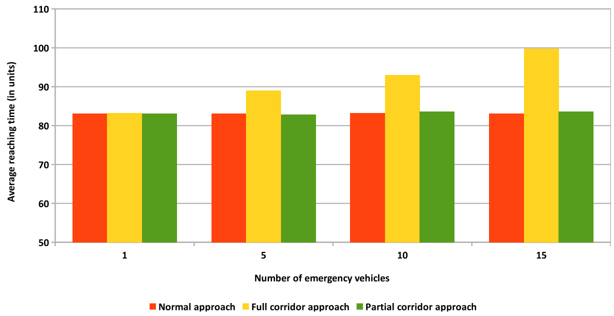



(d) Emergency Vehicle = 15

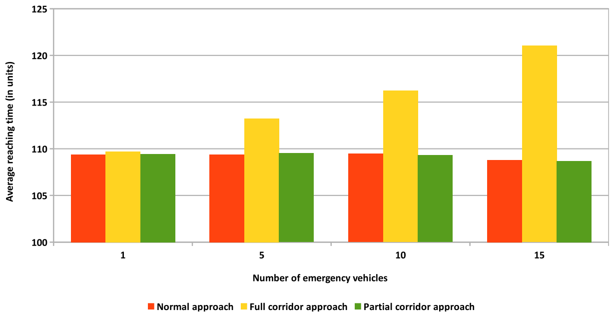
**Figure 3.** The average *reaching time* for varying number of emergency vehicles under different traffic loads in case of *Normal*, *Full Corridor* and the proposed *Partial Corridor* based approaches



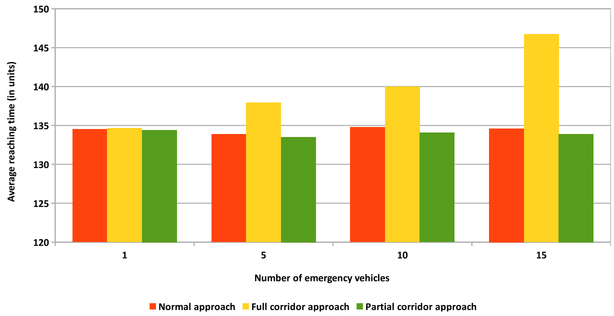

(a) Traffic Load = 3




(b) Traffic Load = 6



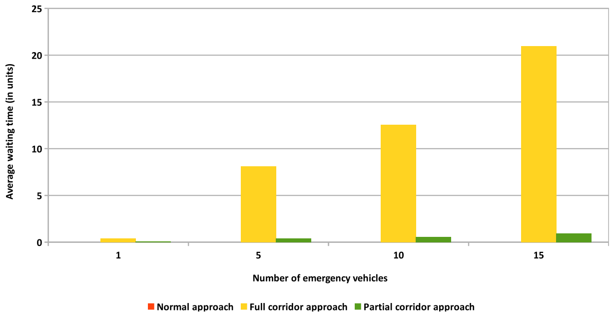

(c) Traffic Load = 9



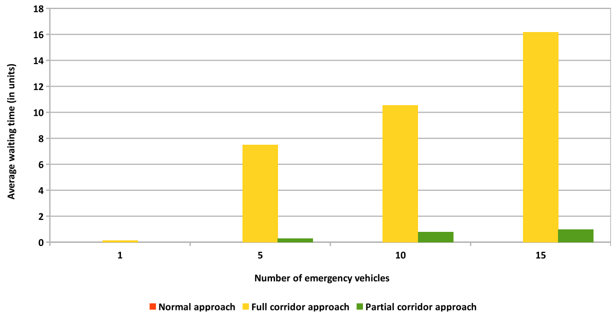

(d) Traffic Load = 12

**Figure 4.** The average *reaching time* of other vehicles with varying number of emergency vehicles under different traffic loads with 50 other vehicles within a 500-node DTIN

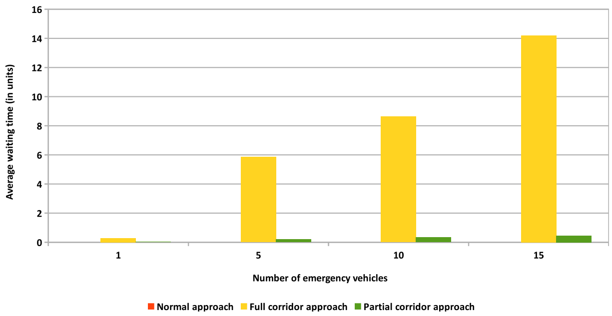
The graphs in Figure 4 depict the *reaching times* of other vehicles in case of multiple emergency vehicles moving towards different destinations, under different traffic loads. It can be observed that the *reaching times*



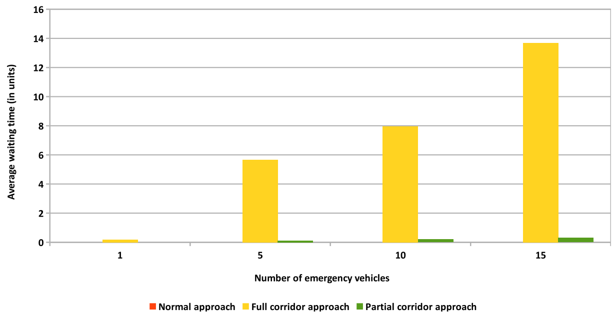

(a) Traffic Load = 3




(b) Traffic Load = 6




(c) Traffic Load = 9




(d) Traffic Load = 12

**Figure 5.** The average *waiting time* of other vehicles with varying number of emergency vehicles under different traffic loads with 50 other vehicles within the 500-node DTIN

of the other vehicles are almost same for all the three approaches (*Normal*, *Full corridor* and *Partial Corridor*) when there is only one emergency vehicle for all the cases of traffic loads. The point to be noted is that as the number of emergency vehicles increases, the *reaching times* of the rest of the vehicles increase for the *full corridor* approach. This is due to the fact that multiple emergency vehicles block large portions of the traffic flow hindering their hassle free passage. Further, the *reaching times* of the other vehicles in case of the *normal* and *partial corridor* approaches are similar. This shows that the use of the proposed *partial green corridor* based approach has least impact on the movement of the adjoining traffic flow.

The graphs in Figure 5 show the average *waiting times* of the other vehicles with varying number of emergency vehicles under different traffic loads. The *waiting time* of a vehicle is defined as the total time a vehicle has to wait whenever it has to cross *green corridors* within the DTIN. The graphs depict the usefulness of the proposed *partial green corridor* approach. While the waiting times of the other vehicles are always high in case of the *full corridor* based approach, the same is very low in case of the *partial corridor* approach. Apparently, the *waiting times* in case of the *normal* approach will be zero as there is no distinction or priority given to the emergency service vehicles.

It may also be seen that when the traffic load is low (equal to 3 in Figure 5(a)) the *waiting times* for the *full corridor* approach is higher than those in cases when this load is high (compared to Figure 5(b) and 5(c)). This is contrary to the general intuitive deduction that as the traffic load increases the *waiting times* also increase. When the traffic loads are low the other vehicles reach the *green corridor* in lesser times making them wait a longer time till the emergency vehicle clears the *corridor*, thereby increasing their *waiting times*.

Hence, the proposed approached not only prioritizes the movement of emergency vehicles by significantly reducing their *reaching times* (as shown in Figure 3), it also causes least impact on the movement of the normal traffic as shown in the graphs in Figures 4 and 5. Further, one can also observe that the proposed *partial green corridor* based approach shows similar performances across different traffic loads in all the graphs reported in the Figures 3, 4 and 5. Even, increasing the number of emergency vehicles does not affect the *reaching* and *waiting* times of the other vehicles as shown in graphs in Figures 4 and 5. Similar performances were observed when the number of other vehicles were increased to 60 and 70 in case of 5 emergency vehicles under different traffic loads. Since, these results do not aid in portraying any new information, the same have not been depicted here.

Figure 6 shows the variation of *waiting times* on part of other vehicles with increasing corridor lengths, $\lambda$. It can be seen that these times keep progressively increasing with $\lambda$ and eventually peak when the

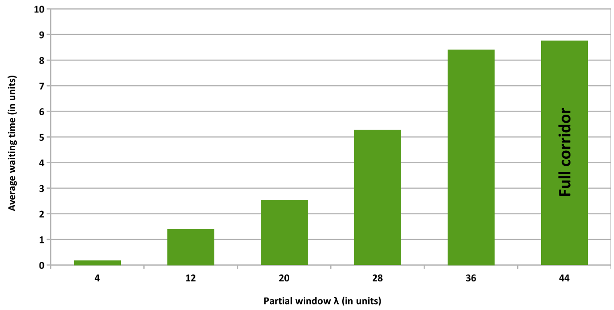



**Figure 6.** Comparative graph of *waiting times* for other vehicles with increasing corridor length, $\lambda$

complete path from the source to the destination is made the *green corridor* (*full corridor* approach) at $\lambda = 44$ units. The graph provides insights into the manner in which one may determine the value of $\lambda$ which may be based on the gravity of the emergency situation. The value of $\lambda$ thus provides an indication of the impact on the *waiting times* of the other vehicles due to the creation of *green corridors*.

## 6. Conclusions and Future work

Traffic management in urban spaces has become a crucial part of an urban infrastructure due to the ever increasing inflow of affordable motor vehicles. The problems in traffic management can be addressed by evolving intelligent and smart solutions. Although many mechanisms have been proposed to model an intelligent traffic management system, most of them do not segregate traffic based on the priority or need. Thus, a concentrated effort to facilitate the seamless movement of *emergency services* with minimal effect on the adjoining traffic flow amidst urban traffic is grossly missing. This paper proposes a multi-agent based approach to create and manage *partial green corridors* for prioritizing the movement of *emergency services* within urban traffic with minimal effect on the movement of normal traffic. The proposed approach has been implemented using *Tartarus*, an agent framework. Various emulation experiments, with varying number of emergency vehicles and traffic loads were performed. The results validate that the proposed approach allows the movement of the emergency services with minimum delays and causes least impact on the flow of adjoining vehicles.

In future, we endeavour to enhance this approach by using real traffic data and sensory information such as traffic speeds, the length and number of parallel lanes and their capacities, speed limits, traffic regulations, etc. There is also a need to look into the priorities of different emergency vehicles so that they can be segregated based on their current requirements and importance. The proposed *partial green corridor* based approach is completely distributed and autonomous with no human intervention. A human in the loop is however a major requirement so that at times of need some aspects of the agent based system could be overruled. A hierarchical system could thus be the best suited one wherein, at the top level there are humans who are centrally monitoring the overall DTIN by periodically collecting data from all the nodes while at the bottom, the proposed agent based *partial green corridor* approach carries out operations autonomously. Such a hierarchical system not only enhances the control of the complete system but also provisions to alter or prioritize decisions on-the-go which may not be possible with a completely automated system. Evidently, the *Tartarus* agent framework can be used to create such hierarchical systems. In addition, the traffic load considered for experimentation does not change while the vehicles move within the DTIN. Dynamically changing the traffic load, while the vehicles are moving within the DTIN, can provide more insights on how the proposed approach tackles dynamically evolving scenarios.

## Acknowledgements

The first and second authors would like to acknowledge the Ministry of Human Resource Development, Govt. of India for providing support during the work reported in this paper. The second author would also like to acknowledge Tata Consultancy Services, India for rendering support under their TCS RSP scheme.

## References

[1] Balaji, P., Sachdeva, G., Srinivasan, D. and Tham, C.K. (2007) Multi-agent System based Urban Traffic Management. In *IEEE Congress on Evolutionary Computation*: 1740–1747. doi:10.1109/CEC.2007.4424683.

[2] Bode, M., Jha, S.S. and Nair, S.B. (2014) A mobile agent based autonomous partial green corridor discovery and maintenance mechanism for emergency services amidst urban traffic. In *Proceedings of the First International Conference on IoT in Urban Space* (ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)): 13–18.

[3] Chen, B., Cheng, H.H. and Palen, J. (2006) Mobile-c: a mobile agent platform for mobile c/c++ agents. *Software: Practice and Experience* **36**(15): 1711–1733.

[4] Chen, B., Cheng, H.H. and Palen, J. (2009) Integrating mobile agent technology with multi-agent systems for distributed traffic detection and management systems. *Transportation Research Part C: Emerging Technologies* **17**(1): 1 – 10. doi:http://dx.doi.org/10.1016/j.trc.2008.04.003, URL `http://www.sciencedirect.com/science/article/pii/S0968090X08000314`.

[5] Chen, B. and Cheng, H. (2010) A Review of the Applications of Agent Technology in Traffic and Transportation Systems. *IEEE Transactions on*

*Intelligent Transportation Systems* **11**(2): 485–497. doi:10.1109/TITS.2010.2048313.

[6] Claes, R., Holvoet, T. and Weyns, D. (2011) A Decentralized Approach for Anticipatory Vehicle Routing Using Delegate Multiagent Systems. *IEEE Transactions on Intelligent Transportation Systems* **12**(2): 364–373. doi:10.1109/TITS.2011.2105867.

[7] Godfrey, W.W., Jha, S.S. and Nair, S.B. (2013) On a Mobile Agent Framework for an Internet of Things. In *International Conference on Communication Systems and Network Technologies*: 345–350. doi:10.1109/CSNT.2013.79.

[8] Harrison, C.G., Chess, D.M. and Kershenbaum, A. (1995) *Mobile Agents: Are they a good idea?* (IBM TJ Watson Research Center Yorktown Heights, New York).

[9] HORLING, B. and LESSER, V. (2004) A survey of multi-agent organizational paradigms. *The Knowledge Engineering Review* **19**: 281–316. doi:10.1017/S0269888905000317, URL http://journals.cambridge.org/article_S0269888905000317.

[10] Martín-Campillo, A., Martí, R., Robles, S. and Martínez-García, C. (2009) Mobile agents for critical medical information retrieving from the emergency scene. In *7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009)* (Springer): 30–39.

[11] Matani, J. and Nair, S.B. (2011) Typhon - A Mobile Agents Framework for Real World Emulation in Prolog. In *Multi-disciplinary Trends in Artificial Intelligence* (Springer Berlin Heidelberg), *Lecture Notes in Computer Science* **7080**, 261–273.

[12] Minar, N., Kramer, K. and Maes, P. (1999) Cooperating Mobile Agents for Dynamic Network Routing. In Hayzelden, A. and Bigham, J. [eds.] *Software Agents for Future Communication Systems* (Springer Berlin Heidelberg), 287–304. doi:10.1007/978-3-642-58418-3_12, URL http://dx.doi.org/10.1007/978-3-642-58418-3_12.

[13] Outtagarts, A. (2009) Mobile Agent-based Applications : A Survey. *International Journal of Computer Science and Network Security* **9**: 331–339.

[14] Pan, J.I., Yung, C.J., Liang, C.C. and Lai, L.F. (2007) An intelligent homecare emergency service system for elder falling. In *World Congress on Medical Physics and Biomedical Engineering 2006* (Springer): 424–428.

[15] Rothkrantz, L.J.M. (2009) Dynamic Routing Using the Network of Car Drivers. In *Proceedings of the 2009 Euro American Conference on Telematics and Information Systems: New Opportunities to Increase Digital Citizenship*, EATIS '09 (ACM): 11:1–11:8. doi:10.1145/1551722.1551733, URL http://doi.acm.org/10.1145/1551722.1551733.

[16] Russell, S. and Norvig, P. (2010) *Artificial Intelligence: A Modern Approach*, Prentice Hall series in artificial intelligence (Prentice Hall), 3rd ed.

[17] Vaa, T. (2007) Intelligent transport systems and effects on road traffic accidents: state of the art. *IET Intelligent Transport Systems* **1**: 81–88(7). URL http://digital-library.theiet.org/content/journals/10.1049/iet-its_20060081.

[18] van Katwijk, R. and van Koningsbruggen, P. (2002) Coordination of traffic management instruments using agent technology. *Transportation Research Part C: Emerging Technologies* **10**(5 - 6): 455 – 471. doi:http://dx.doi.org/10.1016/S0968-090X(02)00034-7, URL http://www.sciencedirect.com/science/article/pii/S0968090X02000347.

[19] Weyns, D., Holvoet, T. and Helleboogh, A. (2007) Anticipatory Vehicle Routing using Delegate Multi-Agent Systems. In *IEEE Intelligent Transportation Systems Conference*: 87–93. doi:10.1109/ITSC.2007.4357809.

[20] Wunderlich, K., Kaufman, D. and Smith, R. (2000) Link travel time prediction for decentralized route guidance architectures. *IEEE Transactions on Intelligent Transportation Systems* **1**(1): 4–14. doi:10.1109/6979.869017.