Sensor-based Computing Techniques For Real-time Traffic Evacuation Management

Georgiana Hamza-Lup

University of Central Florida

Part of the Computer Sciences Commons, and the Engineering Commons

Find similar works at: https://stars.library.ucf.edu/etd

University of Central Florida Libraries http://library.ucf.edu

This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2004-2019 by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation

https://stars.library.ucf.edu/etd/944
SENSOR-BASED COMPUTING TECHNIQUES FOR REAL-TIME TRAFFIC EVACUATION MANAGEMENT

by

GEORGIANA LUCIA HAMZA-LUP
B.S. Technical University of Cluj-Napoca, 2002
M.S. University of Central Florida, 2004

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the School of Electrical Engineering and Computer Science in the College of Engineering and Computer Science at the University of Central Florida Orlando, Florida

Summer Term
2006

Major Professor: Kien A. Hua
ABSTRACT

The threat of terrorist incidents is higher than ever before and devastating acts, such as the terrorist attacks on the World Trade Center and the Pentagon, have left many concerns about the possibility of future incidents and their potential impact. Unlike some natural disasters that can be anticipated, terrorist attacks are sudden and unexpected. Even if sometimes we do have partial information about a possible attack, it is generally not known exactly where, when, or how an attack will occur. This lack of information posses great challenges on those responsible for security, specifically, on their ability to respond fast, whenever necessary with flexibility and coordination.

The surface transportation system plays a critical role in responding to terrorist attacks or other unpredictable human-caused disasters. In particular, existing Intelligent Transportation Systems (ITS) can be enhanced to improve the ability of the surface transportation system to efficiently respond to emergencies and recover from disasters. This research proposes the development of new information technologies to enhance today’s ITS with capabilities to improve the crisis response capabilities of the surface transportation system. The objective of this research is to develop a Smart Traffic Evacuation Management System (STEMS) that responds rapidly and effectively to terrorist threats or other unpredictable disasters, by creating dynamic evacuation plans adaptable to continuously changing traffic conditions based on real-time information.
The intellectual merit of this research is that the proposed STEMS will possess capabilities to support both the unexpected and unpredictable aspects of a terrorist attack and the dynamic aspect of the traffic network environment. Studies of related work indicate that STEMS is the first system that automatically generates evacuation plans, given the location and scope of an incident and the current traffic network conditions, and dynamically adjusts the plans based on real-time information received from sensors and other surveillance technologies. Refining the plans to keep them consistent with the current conditions significantly improves evacuation effectiveness. The changes that STEMS can handle range from slow, steady variations in traffic conditions, to more sudden variations caused by secondary accidents or other stochastic factors (e.g., high visibility events that determine a sudden increase in the density of the traffic). Being especially designed to handle evacuation in case of terrorist-caused disasters, STEMS can also handle multiple coordinated attacks targeting some strategic area over a short time frame. These are frequently encountered in terrorist acts as they are intended to create panic and terror.

Due to the nature of the proposed work, an important component of this project is the development of a simulation environment to support the design and test of STEMS. Developing analytical patterns for modeling traffic dynamics has been explored in the literature at different levels of resolution and realism. Most of the proposed approaches are either too limited in representing reality, or too complex for handling large networks. The contribution of this work consists of investigating and developing traffic models and evacuation algorithms that overcome both of the above limitations. Two of the greatest impacts of this research in terms of science are as follows. First, the new simulation environment developed for this project provides a test bed
to facilitate future work on traffic evacuation systems. Secondly, although the models and algorithms developed for STEMS are targeted towards traffic environments and evacuation, their applicability can be extended to other environments (e.g., building evacuation) and other traffic related problems (e.g., real-time route diversion in case of accidents).

One of the broader impacts of this research would be the deployment of STEMS in a real environment. This research provides a fundamental tool for handling emergency evacuation for a full range of unpredictable incidents, regardless of cause, origin and scope. Wider and swifter deployment of STEMS will support Homeland Security in general, and will also enhance the surface transportation system on which so many Homeland Security stakeholders depend.
To my family
ACKNOWLEDGMENTS

I would like to take this opportunity to thank, from all my heart, the following honorable people for having supported me in the pursuit of achieving my ideals.

My deepest gratitude is now directed to:

• My advisor, Professor Kien A. Hua, who has guided me throughout my research and challenged me to expand my point of view and see “the bigger” picture.

• My committee members: Charles Hughes, Ron Dutton, Essam Radwan and Annie Wu for their comments, suggestions and timely advice.

• My parents, Geta and Lucian, who deserve a vast amount of credit for supporting me both emotionally and financially throughout my education and life, and most of all for believing in my intellectual capacity. Their encouragement has helped me to find the path in the difficult moments when "hope was a lost friend".

• My brother and my sister-in-law, Felix and Nico, for their unconditional support, care, and help.

• My boyfriend, TJ, for his encouragement, support and understanding.

• My family, my grandparents Mami, Tati, Buni and Bunu, my uncles Ghita and Nelu, my aunts Ica and Jeni, my cousins Marius, Adelina and Monica, my nephew Horatiu , who have all surrounded me with their love.

• My DSG fellows, Angel, Danny, Tai, etc. for their friendship and constructive discussions.
• My undergraduate advisor, professor Ioan Salomie for his advice and guidance during my undergraduate research at the Technical University of Cluj-Napoca.

• To anybody I missed who deserves a mention.
# TABLE OF CONTENTS

LIST OF FIGURES ...................................................................................................................... xii
LIST OF TABLES ........................................................................................................................ xv
LIST OF ACRONYMS ............................................................................................................... xvi

1. INTRODUCTION ................................................................................................................ ... 1

2. MOTIVATION AND GOALS ................................................................................................ 6

3. INITIAL STEMS FRAMEWORK ........................................................................................ 13
   3.1. Modeling the Evacuation Problem ............................................................................ 14
   3.2. Data Structures .......................................................................................................... 15
   3.3. Evacuation Graph Construction Algorithm .............................................................. 17
   3.4. Automatic Evacuation Plan Generation Algorithms ................................................. 18
       3.4.1. All-Links Approach ...................................................................................... 19
       3.4.2. Fastest-Links Approach ................................................................................ 20
   3.5. Real-Time Evacuation Management ......................................................................... 21

4. ENHANCING STEMS WITH ALGORITHMS FOR HANDLING MULTIPLE
   INCIDENTS ................................................................................................................................. 25
   4.1. Handling Overlapping Evacuation Zones ................................................................. 25
   4.2. Congestion Avoidance Using Secondary Zone Evacuation ..................................... 29

5. SIMULATIONS STUDIES ................................................................................................... 31
   5.1. Simulation Environment ........................................................................................... 31
   5.2. Computation Efficiency ............................................................................................. 32
5.3. Evacuation Effectiveness Study................................................................................ 34
  5.3.1. Single Incident Scenarios ............................................................................. 35
  5.3.2. Multiple Incidents Scenarios ........................................................................ 41
6. ADAPTIVE ALGORITHMS FOR DYNAMIC EVACUATION PLAN UPDATING ...... 49
  6.1. Network Flow-based Approach ........................................................................ 51
  6.2. Simulation Results ............................................................................................. 56
7. ENHANCED EVACUATION ROUTES CONSTRUCTION ALGORITHMS ............. 60
  7.1. Hybrid Evacuation Route Construction Approach ........................................... 60
  7.2. Exit-Point based Evacuation Route Construction Approaches ....................... 61
    7.2.1. The Hop-based Approach ..................................................................... 62
    7.2.2. The Hop-Away Approach ..................................................................... 64
    7.2.3. The Time-based Approach ..................................................................... 65
  7.3. Simulation Results ............................................................................................. 67
  7.4. Influence of Driver Behavior on Evacuation Efficiency .................................... 76
8. NEW EVACUATION PLAN UPDATING APPROACH............................................. 81
  8.1. Analysis of Evacuation Routes Construction Approaches that are Sensitive to Traffic
       Information ........................................................................................................... 81
  8.2. The Effect of Updating the Evacuation Routes Periodically on Evacuation
       Efficiency ............................................................................................................ 88
  8.3. Improved Evacuation Plan Updating Approach .............................................. 89
9. SYSTEM PROTOTYPING....................................................................................... 94
  9.1. Obtaining the Input Data for the STEMS Prototype ........................................ 95
9.1.1. Traffic Network Topology ................................................................. 98
9.1.2. Traffic Network Conditions ............................................................. 100
9.2. STEMS Implementation ................................................................. 104
9.3. Testing STEMS Performance ............................................................ 107
9.4. Proposed Environment for Field Testing of the STEMS Prototype .......... 115
10. CONCLUSIONS ................................................................................. 116
11. FUTURE RESEARCH DIRECTIONS .............................................. 118
APPENDIX A: LIST OF PUBLICATIONS ............................................ 121
APPENDIX B: SELECTED SOURCE CODE ......................................... 125
LIST OF REFERENCES ............................................................................ 132
LIST OF FIGURES

Figure 1-1. A one-dimensional view of research ................................................................. 1
Figure 1-2. A two-dimensional view of research ................................................................. 1
Figure 1-3. Use-driven research .......................................................................................... 2
Figure 1-4. Interdisciplinary research .................................................................................. 3
Figure 3-1. The Fastest-Links approach ............................................................................. 15
Figure 3-2. Evacuation graph construction algorithm ......................................................... 18
Figure 3-3. Evacuation routes maintenance ....................................................................... 19
Figure 3-4. The All-Links approach .................................................................................... 20
Figure 3-5. Evacuation signals control ................................................................................ 24
Figure 4-1. All-Links evacuation for two incidents with overlapping EZs ....................... 27
Figure 4-2. Secondary zone construction .......................................................................... 30
Figure 5-1. EE comparison for Scenario 2 in an active actuated environment .................. 37
Figure 5-2. EE comparison for the Fastest-Links approach for Scenario 2....................... 38
Figure 5-3. EU comparison for Scenario 2 in an active actuated environment .................. 40
Figure 5-4. EU comparison for Scenario 2 in an active pre-timed environment ............... 40
Figure 5-5. Scenario 3 ....................................................................................................... 41
Figure 5-6. Scenario 4 ....................................................................................................... 41
Figure 5-7. EE comparison for Scenario 3 ......................................................................... 42
Figure 5-8. EE comparison for Scenario 3 with and without SZ evacuation .................... 43
Figure 5-9. The effect of traffic density on performance improvement when evacuating the SZ for Scenario 4 under (a) lower density, and (b) higher density......................................................... 44

Figure 5-10. Scenario 3: All-Links vs. Fastest-Links (a) without SZ, (b) with SZ ....................... 46

Figure 5-11. Scenario 4: All-Links vs. Fastest-Links (a) without SZ, (b) with SZ .......................... 46

Figure 5-12. Evacuation throughput at (a) EEP1, (b) EEP2, and (c) EEP3 ................................... 47

Figure 6-1. Dynamic STEMS simulation ..................................................................................... 50

Figure 6-2. Flow restrictions in normal conditions....................................................................... 54

Figure 6-3. Flow restrictions in the initial stage of the evacuation.............................................. 55

Figure 6-4. Flow restrictions during the evacuation..................................................................... 56

Figure 6-5. EE comparison for the static approaches ................................................................. 58

Figure 6-6. EE comparison: static (dotted lines) vs. dynamic (solid lines) approaches ............. 59

Figure 7-1. The Hybrid approach................................................................................................ 61

Figure 7-2. The Hop-based approach .......................................................................................... 63

Figure 7-3. The Hop-Away approach........................................................................................... 65

Figure 7-4. The Time-based approach ....................................................................................... 67

Figure 7-5. EE comparison (pre-timed control)......................................................................... 70

Figure 7-6. EE comparison (actuated control).......................................................................... 71

Figure 7-7. EU comparison for the exit-point based approaches............................................... 73

Figure 7-8. EE comparison: static (dotted-lines) vs. dynamic (solid lines) approaches .......... 75

Figure 7-9. Influence of driver behavior on the TET................................................................. 77

Figure 7-10. EE comparison for different % of drivers not following the evacuation plans ...... 80

Figure 8-1. The Fastest-Links approach ..................................................................................... 82
Figure 8-2. The Hybrid approach................................................................. 85
Figure 8-3. The Time-based approach .......................................................... 87
Figure 8-4. EE comparison for the new evacuation plan updating approach ............... 91
Figure 9-1. Environment for designing, testing & deployment of STEMS......................... 94
Figure 9-2. Sensor data conversion.......................................................................... 96
Figure 9-3. Sensor emulation software ..................................................................... 97
Figure 9-4. District 5 real-time data format .............................................................. 98
Figure 9-5. Data format for the traffic network topology .............................................. 100
Figure 9-6. Configuration table describing the association between District 5 and TIGER road
segments.............................................................................................................. 101
Figure 9-7. Data conversion module......................................................................... 102
Figure 9-8. Real-time traffic information in CDF ....................................................... 103
Figure 9-9. STEMS GUI : Fastest-Links based evacuation plan................................. 104
Figure 9-10. STEMS GUI: Evacuation plans for multiple incidents............................... 105
Figure 9-11. Evacuation plan generation process ...................................................... 106
Figure 9-12. STEMS GUI: The zoom-in option......................................................... 107
Figure 9-13. Simulation environment for testing the STEMS algorithms....................... 108
Figure 9-14. Interaction of simulation environment components..................................... 109
Figure 9-15. TSIS output file format ................................................................. 110
Figure 9-16. Turn movements example ............................................................... 112
Figure 9-17. Computing the relative turning volumes................................................ 114
LIST OF TABLES

Table 5-1. Computation times in milliseconds ................................................................. 34
Table 5-2. Computation time per road segment in milliseconds ................................. 34
Table 5-3. TET comparison for Scenario 1 (small evacuation area) ......................... 36
Table 5-4. TET comparison for Scenario 2 (large evacuation area) ......................... 36
Table 5-5. EU comparison for Scenario 2 ................................................................. 39
Table 6-1. TET comparison ............................................................................. 57
Table 7-1. TET comparison for the exit-point based approaches ....................... 68
Table 7-2. EU comparison for the exit-point based approaches ......................... 72
Table 7-3. TET comparison ............................................................................. 74
Table 8-1. Comparison of the evacuation routes construction approaches .......... 88
Table 8-2. TET comparison for the new evacuation plan updating approach .......... 90
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDF</td>
<td>Canonical Data Format</td>
</tr>
<tr>
<td>EE</td>
<td>Evacuation Effectiveness</td>
</tr>
<tr>
<td>EEP</td>
<td>Evacuation Exit Point</td>
</tr>
<tr>
<td>EU</td>
<td>Evacuation Uniformity</td>
</tr>
<tr>
<td>EZ</td>
<td>Evacuation Zone</td>
</tr>
<tr>
<td>FDoT</td>
<td>Florida Department of Transportation</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transportation Systems</td>
</tr>
<tr>
<td>MBR</td>
<td>Minimum Bounding Rectangle</td>
</tr>
<tr>
<td>STEMS</td>
<td>Smart Traffic Evacuation Management System</td>
</tr>
<tr>
<td>SZ</td>
<td>Secondary Zone</td>
</tr>
<tr>
<td>TET</td>
<td>Total Evacuation Time</td>
</tr>
<tr>
<td>TMC</td>
<td>Traffic Management Center</td>
</tr>
<tr>
<td>TIGER</td>
<td>Topologically Integrated Geographic Encoding and Referencing</td>
</tr>
<tr>
<td>TSIS</td>
<td>Transportation System Integrated Software</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

After the Second World War American research agenda has been characterized as a linear continuum from basic research to applied research (Figure 1-1). In his book “Pasteur’s Quadrant”, Donald Stokes suggests a two dimensional view (Figure 1-2), in which basic research, such as the work of Niels Bohr on the atomic structure, can be thought of as a vertical axis of the research space, while the strictly applied research of Thomas Edison can be visualized as a horizontal axis. Stokes calls the area in between these axes “Pasteur’s Quadrant”, because the work of Pasteur exemplifies clearly the interwoven nature of basic and applied research. The concept is that research like Pasteur’s, is often applied, practical and basic at the same time.

Figure 1-1. A one-dimensional view of research

Figure 1-2. A two-dimensional view of research
Research in the coming years will be considered superior, if it produces practical results and promotes basic understanding. Such research is called use-driven research and is illustrated in Figure 1-3. While the pure basic research takes existing understanding and provides improved understanding, and the pure applied research takes existing technology and provides improved technology, the use-inspired research is able to both improve existing understanding and to create improved technology.

Figure 1-3. Use-driven research

A particular case of use-inspired research is interdisciplinary research. Figure 1-4 shows that “true” interdisciplinary research lays in the intersection area between applied computing research in the “service” of discipline “X” and the basic research on the discipline “X” in the context of computing. Sometimes researchers will start in one of the other quadrants to get acquainted with the other discipline and problem, but the target is also the upper-right quadrant.
This is where neither computing nor the other discipline can alone create new knowledge and researchers can recognize new problems that neither discipline alone would recognize.

My interdisciplinary research area, *Intelligent Transportation Systems* (ITS), emerged from the lower right quadrant (in Figure 1-4) by applying computing in the “service” of transportation and traffic engineering. I believe that embedding computer technology in the traditional transportation system can bring tremendous benefits. Therefore, I have focused my research in three areas: web-based sensor data dissemination with applications to Internet-scale traffic information systems, wireless communication protocols for intelligent vehicle applications, and evacuation/incident management systems.

**Figure 1-4. Interdisciplinary research**
Recent advancements in sensor technologies and their widespread deployment within the transportation network have enabled the development of intelligent infrastructure systems, able to provide traffic data with better spatial and temporal resolution, accuracy, precision and consistency. A common platform that would allow easy access of raw traffic data, to all potential users, would bring great benefits. Together with another PhD student from my research group, I have proposed and developed an Internet framework, called iSEE (Internet Sensor Exploration Environment), for finding and sharing sensor data and sensor-based applications. Any user of this environment can search for an advertised sensor and visualize the raw data on the screen. Software developers can also use this platform to advertise and sell sensor-based software.

In terms of intelligent vehicles, I have been studying vehicle-to-vehicle communication. The ability of vehicles to “talk” among themselves and/or with the ITS infrastructure has a great potential of improving transportation efficiency. Wireless communication protocols are of critical importance for these applications. My collaboration with two other PhD students resulted in the development of a Connectionless Approach to Mobile Communication (CAM). Instead of using the traditional fixed-path routing, CAM adopts a less restrictive approach by allowing any intermediate vehicles, along the general direction toward the destination vehicle, to relay the data packets without having to establish a hop-by-hop connection. The performance of this new solution is essentially unaffected by the high mobility of the vehicles.

Although current ITS research is mostly concerned with improving the operational efficiency of daily transportation, I believe that ITS can also play an important role in expanding the crisis
response capabilities of the surface transportation system. My research in the area of traffic
evacuation/incident management is the focus of my dissertation and is targeted towards
developing sensor-based computing techniques for real-time traffic evacuation management, in
case of terrorist attacks or other unpredictable incidents. The next section emphasizes the
motivation for my decision and the established goals.
2. MOTIVATION AND GOALS

The surface transportation system plays a crucial role in responding to natural disasters and other catastrophic incidents. The tragic events of September 11, 2001 have raised the nation’s consciousness about the need for better crisis management, disaster planning and prevention, as well as effective detection and response, particularly in the case of deliberate terrorist attacks. Intelligent Transportation Systems (ITS) can play an important role in enhancing the ability of the surface transportation system to respond efficiently to emergencies and recover from natural or human-caused disasters. This research focuses on investigating new ITS technologies to enhance the crisis response capabilities of the surface transportation system, with improved support for evacuation.

ITS have attracted great research interest in recent years. The Intelligent Transportation Society of America, in conjunction with the U.S. Department of Transportation, has published the “National ITS Program Plan: A Ten-Year Vision” in January 2002. This Plan prescribed a broad set of research activities, enabled by ITS, to advance the safety and efficiency of the surface transportation system. However, the current stage of this national initiative focuses mainly on managing daily normal traffic by providing real-time information and trip advisories to system users, including ITS service providers, in order to avoid congestion and to allow travelers to make timely and informed travel decisions.
We believe that ITS can also play an important role in enabling and advancing the surface transportation aspects of homeland security, by enhancing the surface transportation system use for responding to natural or man-made disasters. The basic ITS capabilities for disaster response and evacuation are essentially the same in all types of disasters scenarios, but specific disasters do have unique characteristics (e.g., the amount of warning available, responder risks, chances for secondary events, recovery operations required, and scope and scale of the damage). This research focuses on national security emergencies, such as terrorism and terrorist acts like, chemical, biological, and radiological weapons attacks. The unexpected and stochastic character of terrorist attacks poses unique challenges to those responsible for security. Unlike some natural disasters that can be anticipated, terrorist attacks are sudden and unpredictable. Even if we have some information on a possible attack, we will generally not know exactly where, when, or how an attack will occur. Without this information the most effective strategy is to plan in advance, to prevent and mitigate risks where possible and to respond whenever necessary with flexibility, coordination and speed. Since the threat of terrorism is obscure and security measures are costly, it is hard to justify the expenditures before an attack; and even though there have been great successes in identifying and deterring terrorist threats, it is practically impossible to prevent all acts of terrorism. Therefore, security against terrorism tends to be reactive and techniques for responding to attacks, after they occur, are of great interest and of crucial importance.

The goal of this research is to develop a Smart Traffic Evacuation Management System (STEMS) that can effectively respond to terrorist attacks and other unpredictable incidents, by automatically managing an evacuation operation. STEMS will automatically generate dynamic
evacuation plans, adaptable to the continuously changing traffic conditions, as captured by various sensing technologies, and will subsequently reroute traffic effectively out of the disaster areas based on the generated plans. To the best of our knowledge, this is the first study of dynamic traffic control in conjunction with traffic management techniques for real-time evacuation in case of unpredictable incidents. The significant advancement brought by STEMS is that it possesses capabilities to support both the unexpected and unpredictable aspects of a terrorist attack and the dynamic aspect of the traffic network environment.

Currently available approaches for addressing evacuation are based on proactive planning. This involves developing in advance, different plans for different scenarios and finding among the available plans the most suitable one to be used whenever an incident occurs. These static predefined schemes are effective only in situations when we know precisely the location of the incident, and thus the area to be evacuated and the typical conditions in that area (e.g., building evacuation) [1]. In contrast, a traffic network environment raises more challenges, due to its larger and more dynamic structure, making planning ahead less effective. Currently available tools for proactive evacuation planning and analysis include: the PBS&J model developed by PBS&J Inc.[2], the Oak Ridge Evacuation Modeling System (OREMS) developed by Oak Ridge National Laboratory (ORNL) [3] and the Dynamic Network Evacuation Planning System (DYNEV) developed by KLD Associates Inc. [4]. Currently, these tools are mostly used in the pre-planning analysis (pre-evacuation stage) [5, 6] and in the post-analysis procedure (post-evacuation stage) [7, 8]. During a real-time evacuation operation (in-evacuation stage) these tools can only assist evacuation staff in decision making [9]. In contrast to the existing tools, our
STEMS is targeted towards the real-time evacuation operation. Given an incident location and scope, STEMS will automatically manage an evacuation operation, by employing a set of intelligent algorithms to first generate the evacuation plans and then subsequently guide the evacuees according to the generated plans.

The significance of this research lies in the challenge of developing a set of highly adaptive algorithms to address a very dynamic problem: traffic evacuation management. Due to the dynamic nature of our application, the algorithms required by STEMS must address the following two challenges:

1. The algorithms must generate the initial evacuation plan quickly to ensure rapid and effective emergency response to crises.
2. The algorithms must adapt to various changes in the road and traffic conditions during the course of evacuation.

Several algorithms have been developed for building evacuation planning [10-14]. They are ill-suited for traffic evacuation management for two reasons. First, they do not satisfy the above two STEMS requirements; and secondly, street evacuation is significantly more complex than evacuating a building. Additional challenges arising in traffic environments include managing a higher density of evacuees (increased potential for congestion), handling reverse traffic flowing opposite to the evacuation direction, and conflicting traffic streams competing for the right of way at intersections.
By generating the evacuation plans dynamically, whenever an incident occurs, STEMS will be able to adapt to the traffic dynamics, by leveraging various traffic information, ranging from historical data specific to the location and time of the incident, to more accurate real-time traffic information obtained from sensors and other surveillance technologies. The ubiquitous deployment of sensors in the transportation network, along expressways [15-17] and at intersections of local streets [18-20], is one of the motivating factors for investigating techniques to make STEMS adaptable to the continuously changing traffic conditions. The second important factor is the increased accessibility of the collected real-time and historical traffic data over the web. Several approaches for traffic data dissemination over the Internet have been recently proposed in the literature. As an example, a browsing environment providing access to real-time and historical traffic sensor data over the web is the Freeway Performance Measurement System (PeMS) [21-23], currently deployed in California. An example of a more generic web-based architecture for sharing real-time raw sensor data is the Internet Sensor Exploration Environment (iSEE) [24, 25], developed at University of Central Florida. This framework provides a new environment for finding and sharing live sensor data and applications over the Internet, and encourages a more spread deployment of sensor data, facilitating the development of more and more transportation applications adaptable to real-time traffic conditions. The advancements in sensor technologies and in the field of web-based sensor data dissemination have motivated us to explore new techniques to make STEMS adaptable to changes in traffic conditions, and consequently to improve the evacuation efficiency. To the best of our knowledge we are the first to develop dynamic algorithms for real-time traffic evacuation management.
Our research problem is more complex than the traditional *incident management* problem in traffic engineering [26-28], in the sense that the latter only handles detouring traffic from a certain damaged or congested road segment, while *evacuation* implies routing traffic away from all road segments inside the affected areas.

Another important requirement is to augment STEMS with capabilities to handle more complex scenarios such as multiple coordinated incidents targeting a strategic area in a short time frame. One specific characteristic of terrorist attacks is that they may occur close in time and proximity to create panic and instability. In this situation not only do we need to direct traffic to avoid a particular incident location, but we also need to revise the evacuation plans currently under execution, to reroute traffic away from all incidents.

Our final goal is to develop a complete STEMS system, ready to operate in a real production environment, e.g., the SunGuideSM [29] environment, a statewide transportation management system available from *Florida Department of Transportation* (FDoT). Developing STEMS based on FDoT standards and procedures will later facilitate the integration of STEMS into their traffic management systems, for production use.

Our research is consistent with the national security interests and initiatives, by addressing several concerns published in 2002, by the *Intelligent Transportation Society of America* in a document entitled “Homeland Security and ITS” [30]. In particular, this document calls for the development of tools and technologies for minimizing the consequences of attacks, increasing
the effectiveness of agencies responsible for public safety in time of crisis, and providing the ability to restore the transportation system after a disaster. The STEMS system addresses several of these issues. One of the broader impacts of this research is its support for homeland security in general. Wider and swifter deployment of STEMS will enhance the surface transportation system on which so many homeland security stakeholders depend.

The remainder of this dissertation is organized as follows. We present the challenges and proposed approaches for developing an initial STEMS framework in Section 3. In Section 4 we explain how we have extended the STEMS framework with algorithms that can handle multiple coordinated incidents. The simulation studies and results are discussed in Section 5, in order to facilitate the discussion on how STEMS can handle traffic dynamics, which is explained in Section 6. Then Section 7 presents a set of enhanced evacuation algorithms and compares their performance with the previously proposed ones. Section 8 describes a more intelligent approach for handling traffic dynamics and presents the simulation results that show how this approach improves the performance of the dynamic STEMS proposed in Section 6. The system prototype is presented and discussed in Section 9. Finally, Section 10 concludes the dissertation and Section 11 discusses some future research directions.
3. INITIAL STEMS FRAMEWORK

The first task in our research was to identify the functionality required by STEMS, as follows:

1. Generate evacuation routes that efficiently leverage all available road capacity through load balancing, in order to maximize the effectiveness of the evacuation operation. The system should minimize congestions and risks to the evacuees.

2. Effectively control evacuation signals to manage traffic at intersections and reroute it according to the generated evacuation plans.

3. Provide contingency evacuation plans for multiple coordinated attacks that may occur before and during an evacuation operation.

4. Dynamically respond to incidents that hamper evacuation efficiency (e.g., congestion caused by vehicle breakdowns or secondary accidents) by adapting the evacuation in execution and generating new plans as necessary.

Starting from the above objectives we have developed the initial STEMS framework by targeting the first two of the above requirements, which represent the minimal functionality that any real-time evacuation management system should provide: first, to dynamically generate evacuation plans given an incident specified by its intensity and location within the traffic network; and secondly, to guide the evacuees on the proper routes according to the generated plans. The challenges and proposed solutions for developing this initial framework are presented in the following sub-sections.
3.1. **Modeling the Evacuation Problem**

Our first requirement in developing the STEMS framework was to define a model of the road network environment on which to formulate the evacuation problem. We model the road network as a graph, with nodes representing intersections, and links corresponding to the road segments between two intersections. Given the location of an incident and its scope $R$, we define the *incident node* as the node $IN$ closest to the incident location, and the *evacuation zone* $EZ$ as the circular area centered at $IN$ with radius $R$. Then we define the *evacuation exit points* (EEP) as nodes that fall within a certain threshold outside the EZ boundary and are connected to nodes inside the EZ. If we define the *evacuation graph* as the sub-graph of the road network induced on the set of nodes inside the EZ and having the EEPs as leaves, then the evacuation problem becomes directing traffic from all the links of the evacuation graph (i.e., links inside or intersecting the EZ), to flow away from the incident and exit the affected area via the designated EEPs.

The above concepts are illustrated in Figure 3-1. It shows potential evacuation routes radiating from the incident node at the center of two circles. The inner circle indicates the boundary of the EZ and the area between the two circles is the threshold area. Nodes that fall within this area and are connected to nodes inside the EZ are chosen as the EEPs and at these points traffic from outside will be blocked from entering the EZ.
3.2. Data Structures

Given a representation model for the road network environment, the next requirement was to choose an efficient data structure suitable for developing algorithms to solve the evacuation problem. Without loss of generality, we have stored the road network and the related traffic data in two relational tables as follows:

- **Nodes table**: Each record represents an intersection characterized by the NodeID, and the X and Y coordinates of the location.

- **Links table**: Each record represents a road segment characterized by its two end points (NodeID1 and NodeID2), and the average speed values in each direction (Speed_{12} and Speed_{21}).
The above data structure contains two types of data: static and dynamic. Static data consists of the topology of the road network, that is, the road segments and the intersections, and their geographic location. While this data does not change frequently in time, the average speed values for individual road segments is much more dynamic, varying in time according to several parameters (e.g., the time of the day, weather conditions, etc.). Average (or space mean) speed is defined as the arithmetic mean of the speed of vehicles occupying a given road segment. Some of the widely used sensing devices for measuring speed include: inductive loops, pulsed ultrasound, single detection zone passive infrared, and magnetometers. These sensors are used in pairs and the speed is calculated as the ratio of the distance between the sensors and the time it takes a vehicle to traverse that distance. Another way to measure speed is with multi-detection zone sensors such as a video image processor, multi-zone passive infrared, microwave radars, or acoustic arrays. For example, a video processing technique can measure speed based on vehicle movement across a calibrated distance in the field of view. The space mean speed used by the STEMS algorithms can be either historical/statistical data specific to the location and time of the incident, or more accurate real-time traffic information, obtained from sensors and other surveillance technologies.

To facilitate efficient retrieval of certain nodes in the Nodes table, we consider maintaining a hierarchical index structure, such as the quadtree. A quadtree divides the entire map area into four even quadrants, and divides those four quadrants into four smaller subquadrants, and so on. It stops dividing the area when it reaches a predefined minimal size. This set size can be chosen so that the data tuples corresponding to nodes located in the subspace fit in a single disk page.
As part of this data partitioning process, we also create an index tree with the tuples stored in the leaf nodes. Each of the internal index nodes represents a disjoint subspace in the street map, and has four pointers to its own four quadrants in the next level of the hierarchy. With this access structure, we can descend the tree starting from the root node until we reach an index node representing the smallest region that fully encloses the query area. Thus, using this file organization, we can efficiently retrieve nodes falling within any given rectangular subspace. Obviously, we can also use any other multi-dimensional indexing technique such as D-trees [31].

3.3. **Evacuation Graph Construction Algorithm**

Given an incident specified by its intensity and location within the traffic network, the first algorithm we have developed was to identify the *evacuation graph*, which will be the input for the *evacuation routes* construction algorithms. The pseudo-code (including SQL-like statements) for the *evacuation graph* construction is given in Figure 3-2.

The *distance* function computes the distance between the two nodes represented by the two parameters. As a result of this algorithm, the leaf nodes (EEPs), the non-leaf nodes and the links of the *evacuation graph* are stored in three temporary tables: *EEPNodes*, *EZNodes* and *EZLinks*, respectively. Given the *evacuation graph* the second objective was to develop algorithms for constructing the *evacuation routes*. 

17
Once we identify the evacuation graph corresponding to an incident, the second step is the automatic evacuation plan generation. We define the evacuation routes (or an evacuation plan) as a directed sub-graph of the evacuation graph, having an evacuation direction assigned on each link, which would indicate the desired traffic direction on the corresponding road segment. The evacuation plan will be stored in the EZLinks table by augmenting it with an additional
column: *EvacuationDirection*. The algorithm for capturing the evacuation plan in the *EZLinks* table is given in Figure 3-3. For each directed link $L$ part of the *evacuation routes*, the tuple $(NodeID1, NodeID2, EvacuationDirection)$ is identified in the *EZLinks* table, and the value of the *EvacuationDirection* is set to “1” if the direction of $L$ is from $NodeID1$ to $NodeID2$, and to “2” otherwise. We have investigated two approaches for our route construction algorithms: one for *active environments*, in which we have real-time information about the traffic conditions, and the other for *passive environments*, in which we have no information about the current traffic conditions. The propose *evacuation routes* construction algorithms are presented in the next subsections.

1. Augment *EZLinks* with a column: *EvacuationDirection*.
   
   ```sql
   ALTER TABLE EZLinks ADD EvacuationDirection tinyint NOT NULL DEFAULT 0
   ```

2. For all directed links $L$ (from $Node1$ to $Node2$) on the *evacuation routes*, do:
   
   ```sql
   UPDATE EZLinks
   SET EvacuationDirection = (NodeID1=Node1) ? 1 : 2),
   WHERE (NodeID1 = Node1 AND NodeID2 = Node2)
   OR (NodeID1 = Node2 AND NodeID2 = Node1)
   EndFor
   ```

**Figure 3-3. Evacuation routes maintenance**

### 3.4.1. All-Links Approach

The *All-Links* approach, illustrated in Figure 3-4, is designed for passive environments. It uses the entire road network as evacuation routes, by assigning an *evacuation direction* on each link of the *evacuation graph*. This approach is based on a breadth-first graph traversal; starting from
the incident node \( IN \), we traverse the road network in a breadth-first order such that each link is visited only once. The traversal direction for each link indicates the evacuation direction that traffic will be forced to follow when reaching an intersection, in order to flow away from the incident and eventually exit the EZ via the designated EEPs.

![Diagram of road network with incident node and evacuation zones]

**Figure 3-4. The All-Links approach**

### 3.4.2. Fastest-Links Approach

The *Fastest-Links* approach, illustrated in Figure 3-1, is designed for active environments and is based on multicast routing. The idea is to construct the *evacuation routes* as a multicast tree having the incident node as the source and the EEPs as receivers. Due to nature of our problem the tree construction reduces to overlapping the fastest paths from the incident node to each
EEPs. The multicast tree would constitute the *evacuation routes* that all traffic would follow, to leave the EZ via one of the EEPs. This tree structure provides us with the *evacuation routes*, indicating the desired direction for the traffic flow (from the root to the leaves), and allowing us to assign an evacuation direction to each link in the tree. The performance metrics employed for constructing a shortest path is the *aggregated time*, which represents the time required to traverse all its links, given that each link is characterized by a certain average speed. Links that are not part of the multicast tree do not have an evacuation direction and they will have bi-directional traffic during evacuation. Note that initially, in both *All-Links* and *Fastest-Links* approaches, all the links of the *evacuation graph* have bi-directional traffic, but eventually, those that are part of the *evacuation routes* will only have traffic flowing in the evacuation direction. This is explained in detail in the following section, in which we describe how we direct traffic on the *evacuation routes*.

### 3.5. Real-Time Evacuation Management

After computing the *evacuation routes*, the third problem to address was how to guide the evacuees according to the generated evacuation plan. In our environment, we use an individual evacuation signal for each approaching direction of an intersection to guide traffic in the proper outgoing directions. The design of this signal consists of an arrow pointing in each of the possible outgoing directions. These arrows can take on any one of the three traditional traffic lights colors, namely red, yellow, and green, at different times. A red arrow is equivalent to a traditional “do not enter” sign; and incoming traffic should halt in the corresponding direction.
Similarly, a green arrow gives the incoming traffic the right of way to proceed in that direction. A yellow arrow specifies a transitional stage; and the incoming traffic should slow down and prepare to stop. If there is a concern for color-blind motorists, we can consider the following alternative design - turning on the light instead of using the green color and turning off the light in lieu of red.

Although initially, we likely have traffic flowing opposite to the evacuation directions, such traffic can be forced in the evacuation directions when it reaches an intersection. This is illustrated in Figure 3-5. It shows that vehicle #4 is initially in the evacuation direction. Although vehicles #2 and #3 are not, they can be directed by the evacuation lights, to follow the evacuation direction at the next intersection. Recall that in the Fastest-Links approach, links that are not part of the evacuation routes may have bi-directional traffic that will gradually be assimilated by the evacuation routes. Thus, only the traffic entering the nodes that are part of the evacuation routes has to be forced on certain links to follow the evacuation directions. In the All-Links approach, all nodes must be controlled by forcing incoming traffic in the evacuation directions. Thus, we have two rules:

**Rule 1:** At all nodes on the evacuation routes, the incoming traffic is guided to pick one of the outgoing links such that to flow in the evacuation direction. If such a choice is not available than traffic may pick any outgoing direction.

**Rule 2:** At all nodes not on the evacuation routes, incoming traffic may pick any outgoing direction.
The above rules are only for enforcing directions for the traffic flow. We also need certain timing schemes to give the right of way to approaching traffic in an orderly manner. In our work we employ a simple timing scheme, which allows equal usage of the intersection among all incoming streams. As for traditional traffic lights, either pre-timed or actuated signal control strategies, can be employed for controlling our evacuation signals. While the pre-timed controllers use fixed predetermined timing plans, the actuated controllers are sensitive to traffic volumes. Actuated controllers are connected to traffic detectors to sense traffic demand and adjust signal timing accordingly. Thus, if there is no traffic entering an intersection from a certain direction, the right of way is passed to the next incoming traffic stream.

We note that in the *Fastest-Links* approach traffic flowing opposite to the evacuation direction may occur only in the early stage of the evacuation. Eventually, all the traffic will be flowing through the multicast tree, only in the evacuation direction. Since each node in the multicast tree can have only one parent node, there is only one incoming traffic flow at each intersection in the tree. In this case, if actuated evacuation signals are used, they will detect the absence of traffic from all the other directions, and maintain the right of way to the sole approaching traffic flow. Effectively, timing schemes are no longer necessary in this stage and all traffic flows freely along the multicast tree towards the EEPs. In both *All-Links* and *Fastest-Links* approach, we rely on drivers’ discretion to choose among feasible links while flowing through the *evacuation routes* (e.g., in Figure 3-5 there are two feasible directions for vehicle #4 to follow: either straight or to the left).
Our initial STEMS framework presented so far was targeted towards the first two basic requirements that an evacuation management system should provide: to automatically construct evacuation routes, given an incident location and scope, and subsequently, to control the evacuation signals such that to direct traffic out of the affected areas, according to the generated plans. As mentioned at the begin of Section 3, another important requirement of a traffic evacuation management system is to provide contingency evacuation plans for multiple coordinated attacks that may occur before or during an evacuation operation. We address this requirement in the next section.
4. ENHANCING STEMS WITH ALGORITHMS FOR HANDLING MULTIPLE INCIDENTS

One specific characteristic of terrorist attacks is that they may occur close in time and proximity to create panic and instability. In this situation not only do we need to direct traffic to avoid a particular incident location, but we also need to revise the evacuation plans currently under execution, to reroute traffic away from all incidents. In this section we propose evacuation algorithms that enhance STEMS with capabilities to handle incidents occurring in a short time frame and having overlapping or neighboring evacuation zones. For handling multiple incidents targeting a specific area in a short time frame, our evacuation algorithms have to address two additional challenges: first, to be able to construct evacuation routes for incidents with overlapping evacuation zones, such that to direct traffic away from all incidents; and secondly, to prohibit congestion in the areas around the evacuation zones from impeding the evacuation execution. This second problem occurs in areas that are in close proximity with more than one incident, and it may drastically affect the evacuation times.

4.1. Handling Overlapping Evacuation Zones

Using the EvacuationDirection column for capturing evacuation routes (as in Figure 3-3), the Links table will maintain a global evacuation plan for all incidents, and update it with new plans whenever new incidents occur. The procedure is as follows. When an incident occurs, the corresponding evacuation graph is computed as in a single-incident scenario (presented in
Section 3.3). Recall that the evacuation graph construction algorithm identifies the sub-graph of the street network graph, affected by the incident, and stores its leaf nodes, non-leaf nodes and links in three temporary tables: EEPNodes, EZNodes and EZLinks, respectively. The evacuation graph is used as input by the evacuation plan construction algorithm to generate evacuation routes, which are then captured in the EZLinks table (as in Figure 3-3).

In case of multiple incidents, the evacuation routes of a new incident may overlap with existing plans generated by previous incidents and may result in conflicting evacuations directions on some of the street segments. These conflicts occur in areas where the EZs of two or more incidents overlap. However, not all street segments in the overlapping areas will experience conflicts. Given several overlapping incidents, if their corresponding evacuation plans assign the same evacuation direction on a street segment, then traffic following that direction will flow away from all incidents. This is because an evacuation plan is constructed such that to direct traffic away from the given incident. Conversely, if a street segment is assigned different evacuation directions, neither of those directions will direct traffic away from all incidents. As evacuation should not direct traffic towards any incident, but away from all of them, we do not enforce an evacuation direction on street segments that experience conflicts. In other words, we remove those street segments from the evacuation plans of both current and previous incidents. We illustrate this with the example in Figure 4-1. The initial evacuation plan for the smaller incident (shown in Figure 3-4) changes when the larger incident occurs; the red-highlighted street segments are the ones assigned conflicting evacuation directions and are removed from the evacuation routes of both incidents. This change assures that whenever a new incident occurs,
the previous evacuation plans, currently under execution, are updated such that they do not direct traffic towards the new incident.

Figure 4-1. *All-Links* evacuation for two incidents with overlapping EZs

To differentiate between street segments that were never assigned an evacuation direction (because they were not part of any evacuation plan, so their *EvacuationDirection* is “0”) and the ones that were assigned a direction that was later removed due to conflicts, we set the *EvacuationDirection* of the latter ones to ”3”. The evacuation routes for the currently handled incident, captured in the *EZLinks* table, are overlaid on the global evacuation plan resulting from previous incidents (stored in the *Links* table) according to the following rule: if the currently computed evacuation direction for a street segment (from the *EZLinks* table) conflicts with the one already assigned as a result of previous incidents (captured by the *Links* table) the
EvacuationDirection in the Links table is set to “3”; otherwise it is set to the newly computed value (from the EZLinks table). We note that once the EvacuationDirection of a street segment becomes “3” it will not change anymore, as future assignments will always result in conflicts.

To guide traffic according to the generated global evacuation plan, we control the evacuation signals at the intersections that are part of the evacuation routes, by imposing some restrictions. More exactly, the two rules defined in Section 3.5 become:

**Rule 1:** At all nodes that are part of the evacuation routes incoming traffic is only allowed to pick one of the outgoing links such that to flow in the evacuation direction. If this choice is not available, traffic is guided to pick one of the links with no evacuation direction (such that not to flow opposite to an evacuation direction). If such a choice is not available either, then traffic is allowed to pick any of the outgoing links and will be guided in the evacuation direction when it reaches the next intersection.

**Rule 2:** At nodes that are not on the evacuation routes traffic proceeds as in normal conditions, that is, drivers may pick any outgoing direction.

The above restrictions assure that traffic will not leave an evacuation route until it ends. Figure 4-1 shows that all the traffic flowing towards the overlapping area will be stopped by the blue-highlighted street segments and directed along them, away from both incidents. Similarly, traffic inside the overlapping area will be blocked from flowing towards the incidents, being directed along the blue-highlighted streets and thus away from both incidents.
4.2. **Congestion Avoidance Using Secondary Zone Evacuation**

As stated previously, another problem associated with multiple incidents, is when the EZs of the incidents are not necessarily overlapping, but they are close enough that the area between them becomes congested and impedes the evacuation execution. To handle this problem we define a *secondary zone* (SZ) around the existing EZs and evacuate traffic from this area in order to facilitate the evacuation of the traffic in danger from inside the evacuation zones. A SZ is defined as follows. Whenever an incident occurs we first check whether its evacuation area is in close proximity with any of the previous incidents. If the difference between the distance between the two incidents and the sum of their radii is higher than a certain threshold value than we consider the incidents independent; otherwise, they are considered in close proximity (even though they are not necessarily overlapping). If two or more incidents are in close proximity then the *minimum bounding rectangle* (MBR) enclosing their EZs is first computed and then the secondary zone is defined as the disk having the same center as the MBR and diameter equal to the larger side of the MBR. In Figure 4-2, the black disks represent the EZs and the gray one represents the secondary zone. The SZ captures the areas around the EZs that are susceptible to congestion. The more critical street segments are the ones in the dark grey area, as they receive traffic from more than one incident. To avoid congestion and facilitate the evacuation of the traffic in danger from inside the EZs we also evacuate the traffic from the SZ. Either of the two evacuation routes construction approaches (*All-Links* or *Fastest-Links*) can be applied for the SZ evacuation assuming that we have a virtual incident in the center of the SZ.
We note here that a naïve approach would be to simply replace several incidents with their corresponding virtual incident and then evacuate traffic by constructing the *evacuation routes* for the virtual incident. However, in this situation, by guiding traffic away from the virtual incident we will be directing it towards the actual incidents. This undesirable effect does not allow us to ignore the EZs of the actual incidents. Our approach is to assign an evacuation direction only on streets that are not part of any EZ (the gray areas in Figure 4-2), when constructing the *evacuation routes* for the SZ. Thus, streets that belong to a certain EZ will maintain their previously assigned evacuation directions, assuring that traffic in each EZ will flow away from its corresponding incident. This approach decreases the evacuation times significantly.

![Figure 4-2. Secondary zone construction](image)

Figure 4-2. Secondary zone construction
5. SIMULATIONS STUDIES

Due to the nature of this research, modeling and simulation methods are essential elements in the design and evaluation of STEMS. For our simulation studies we have used the TSIS (Transportation System Integrated Software [32]) traffic simulator, build as a shell around the well-known microscopic simulator Corsim [33]. Although several other traffic simulators exist (e.g., TRANSIMS [34], PARAMICS [35]), TSIS was the one that better matched the requirements of our studies. There are few other simulators [36, 37] suitable for our study, garnered from research projects at various universities and institutions, but they are still under prototyping and not yet available.

5.1. Simulation Environment

In our performance study, we considered several evacuation areas, from downtown Orlando in Florida. Hereafter, we will present only a relevant subset of the simulation results in order to facilitate our discussion. In order to emphasize the performance improvement of our algorithms when provided real-time traffic information from various sensing devices, we have performed simulations on two types of traffic network environments: active and passive. The passive environment accounts for the situations in which real-time information about current traffic conditions is not available, thus, all road segments are assumed to have an average speed equal to the speed limit. In the active environment real-time average speeds for individual road segments are considered available. To emphasize the significance of the actuated signal
controllers, we present simulations results for both pre-timed and actuated traffic signal control strategies.

We present our simulation results from four of our simulation scenarios. To analyze and compare the performance of our algorithms in handling single incidents we have investigated two single incident scenarios: (1) a small-area incident with 50 intersections, 65 road segments and approximately 1,000 vehicles to be evacuated (with some small deviation for each simulation run due to the randomness in the initial setup), and (2) a large-area incident with 218 intersections, 344 road segments and approximately 3,500 vehicles. To illustrate STEMS performance in handling multiple incidents we present two more scenarios: (3) three incidents having overlapping evacuation zones, with 79, 78 and 73 intersections, and 109, 110 and 105 road segments, respectively; and (4) three incidents occurring in close proximity (but without overlapping evacuation zones), having 76, 68, 75 intersections and 105, 95, 107, street segments, respectively. The third scenario is used for investigating the effectiveness of our algorithms in handling overlapping, and the fourth scenario is used for assessing the role and efficiency of the secondary zone evacuation.

5.2. Computation Efficiency

Due to the requirement of our STEMS to ensure rapid and efficient emergency response to crisis, the evacuation delay defined as the time to compute the evacuation plan must be very short. To
investigate this property we have first analyzed the complexity of our evacuation routes construction algorithms, and then examined their actual execution times on a typical computer.

The total computation time $T(n,m)$ represents the time required by STEMS to generate an evacuation plan, given an incident scope and location within a street network environment with $n$ intersections and $m$ road segments. $T(n,m)$ consists of two components: the time $T_1(n,m)$ necessary to identify and extract from the road network database the information relevant to evacuation (i.e., the evacuation graph construction) and the time $T_2(n,m)$ necessary to generate the evacuation routes. Thus,

$$T(n,m) = T_1(n,m) + T_2(n,m).$$

In case of the Fastest-Links approach the time $T_2$ to generate evacuation routes is the time necessary to compute the shortest paths from the incident node to the EEPs. We have used the well-known Dijkstra’s algorithm [38] with the time requirement of $O(m\log n + n\log n)$. Dijkstra can be implemented more efficiently by storing the road network graph in the form of adjacency lists and using heaps. If binary heaps are used than the running time is $O(m\log n)$; and it is $O(m + n\log n)$ if Fibonacci heaps are used assuming that the comparisons of links weights take constant time. In case of the All-Links approach the time $T_2$ to generate evacuation routes is the time necessary to perform a breadth first traversal of the road network graph. If the graph is represented in the form of an adjacency list the running time is $O(m + n)$, assuming that the visiting time is constant.
Table 5-1 contains the running times in milliseconds, obtained when the algorithms were executed on a 1.6GHz Intel Pentium 4 processor.

<table>
<thead>
<tr>
<th>Scenario 1: n = 50, m = 65</th>
<th>Scenario 2: n = 218, m = 344</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>T2</td>
</tr>
<tr>
<td>All-Links</td>
<td>308</td>
</tr>
<tr>
<td>Fastest-Links</td>
<td>308</td>
</tr>
</tbody>
</table>

To have a better view of the behavior of the algorithms from the execution time point of view, we investigated the average computation times per road segment, computed in Table 5-2 for the two evacuation techniques under two different scenarios. Since average computation times per road segment are negligible, we can safely conclude that the proposed techniques are suitable for time-critical traffic evacuation.

<table>
<thead>
<tr>
<th>Scenario 1: m = 65</th>
<th>Scenario 2: m = 344</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>T2</td>
</tr>
<tr>
<td>All-Links</td>
<td>4.74</td>
</tr>
<tr>
<td>Fastest-Links</td>
<td>4.74</td>
</tr>
</tbody>
</table>

5.3. **Evacuation Effectiveness Study**

We compare our proposed evacuation algorithms based on three performance metrics, namely *Total Evacuation Time* (TET), *Evacuation Effectiveness* (EE), and *Evacuation Uniformity* (EU). Although TET is commonly used in the literature to study evacuation operations, we feel that
this metric alone is sometimes misleading, and further assessments can be done using also EE and EU. We explain these metrics in the following sub-sections in parallel with presenting the results.

5.3.1. Single Incident Scenarios

We first compare the All-Links and Fastest-Links techniques in terms of TET. This is the time needed for all the vehicles to exit the evacuation zone via one of the available EEPs. In order to have a better feel for the effectiveness of our evacuation techniques, we have defined a base-case scenario, in which the drivers are not aware of the incident, and will continue to drive normally until they exit the evacuation area at one of the EEPs. At this point, we assume that such drivers become aware of the incident and will not return to the evacuation zone. We summarize the performance of the three schemes in Table 5-3 and Table 5-4, for Scenario 1 and Scenario 2, respectively. We note that No-Information and All-Links perform the same in both passive and active environments, because these evacuation schemes do not take into account the available traffic information.

The simulation results show that both All-Links and Fastest-Links algorithms are very efficient, generating evacuation times much smaller than in the case when there is no information about the incident. We can draw two conclusions. First, as expected, the actuated control performs better, resulting in evacuation times smaller than for the pre-timed scheme. Secondly, the Fastest-Links approach performs better than the All-Links approach in the active environment.
(i.e., when it has information about the traffic conditions on the roads, so the multicast tree can choose the fastest evacuation routes); However, in the passive environment, the Fastest-Links approach might concentrate traffic on a multicast tree which might contain very slow routes, so depending on the topology of the road network and current traffic conditions, it might perform better (see Table 5-4) or worse (see Table 5-3) than the All-Links approach.

### Table 5-3. TET comparison for Scenario 1 (small evacuation area)

<table>
<thead>
<tr>
<th>TET [minutes]</th>
<th>No Info.</th>
<th>All-Links</th>
<th>Fastest-Links</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Passive</td>
<td>Active</td>
</tr>
<tr>
<td>Pre-timed</td>
<td>25.33</td>
<td>11.00</td>
<td>11.33</td>
</tr>
<tr>
<td>Actuated</td>
<td>20.66</td>
<td>7.33</td>
<td>7.33</td>
</tr>
</tbody>
</table>

### Table 5-4. TET comparison for Scenario 2 (large evacuation area)

<table>
<thead>
<tr>
<th>TET [minutes]</th>
<th>No Info.</th>
<th>All-Links</th>
<th>Fastest-Links</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Passive</td>
<td>Active</td>
</tr>
<tr>
<td>Pre-timed</td>
<td>112.00</td>
<td>25.66</td>
<td>38.33</td>
</tr>
<tr>
<td>Actuated</td>
<td>79.33</td>
<td>12.00</td>
<td>12.66</td>
</tr>
</tbody>
</table>

Although the benefit of the All-Links and the Fastest-Links approaches over the No-Information case is very clear from the TET performance study, it is not obvious which of the two techniques is better, given that their evacuation times are fairly similar. Thus, we also compare them in terms of EE in order to better differentiate between the two. It is worth mentioning here that due to the nature of evacuation, performance improvement has a different meaning than in other computer related applications. Since we are dealing with people’s safety and human lives, even a small increase in performance has an extremely significant value.
Evacuation Effectiveness (EE) provides an answer to the question “How fast is the traffic evacuated?” If we plot the number of evacuees still in the EZ over time, then the EE can be interpreted as the area under the curve. Obviously, a smaller EE means that the slope of the curve is sharper indicating greater speed in guiding evacuees out of the danger zone. The simulation results in terms of EE, for Scenario 2 in an active environment, are plotted in Figure 5-1. We notice that the Fastest-Links approach can evacuate significantly faster. For instance, it evacuates about 500 more vehicles than All-Links after 5 minutes.

![Figure 5-1. EE comparison for Scenario 2 in an active actuated environment](image)

To get insight into the benefit of the availability of traffic data in evacuation, we investigated the plot shown in Figure 5-2. It shows the EE of the Fastest-Links approach for Scenario 2 in the
four different environments considered in Table 5-4. This study indicates that regardless of the signal control scheme (pre-timed or actuated), *Fastest-Links* performs significantly better in the active street network environment. In other words, the *Fastest-Links* approach is able to leverage real-time traffic information to improve its performance.

![Figure 5-2. EE comparison for the Fastest-Links approach for Scenario 2](image)

*Evacuation Uniformity* (EU) measures how well the algorithms leverage the EEPs in evacuation. It reflects how uniform the traffic is distributed among the EEPs. High EU means that traffic is not uniformly distributed among the EEPs, and this may lead to congestions around the EZ. EU is computed as the standard deviation of the throughputs at the individual EEPs as follows:
where $n$ is the number of EEPs, $\text{throughput}(\text{EEP}_i)$ is the number of vehicles exiting the EZ at EEP$_i$ and $\overline{\text{throughput}}$ is the average throughput of all EEPs.

The results of this study for Scenario 2 are summarized in Table 5-5. It indicates that the _Fastest-Links_ approach outperforms the _All-Links_ approach by almost 50%, from the EU point of view. This explains the performance advantage of the _Fastest-Links_ approach over the _All-Links_ approach as observed in the EE comparison. The small difference in EU between the actuated and the pre-timed environment is due to the stochastic character of the traffic simulator.

| Table 5-5. EU comparison for Scenario 2 |
|-------------------------------|---------------------|
| EU                            | Pre-timed control  | Actuated control   |
| All-Links                     | 90.67               | 83.04              |
| Fastest-Links                 | 50.19               | 51.29              |

Figure 5-3 and Figure 5-4 show the total number of evacuees exiting at each EEP, for the actuated mode and pre-timed environment, respectively. They show once again that traffic exits at the EEPs more uniformly under the _Fastest-Link_ approach.
Figure 5-3. EU comparison for Scenario 2 in an active actuated environment

Figure 5-4. EU comparison for Scenario 2 in an active pre-timed environment
5.3.2. Multiple Incidents Scenarios

We evaluate the efficiency of our evacuation algorithms in handling multiple coordinating incidents by investigating two scenarios: incidents with overlapping evacuation zones (Figure 5-5) and incidents that occur in close proximity, but do not overlap (Figure 5-6).

The performance metric used to compare the *All-Links* and *Fastest-Links* approaches (with and without SZ evacuation) is *Evacuation Effectiveness* (EE). We first investigate the overlapping incidents scenario to assess the effectiveness of our algorithms in handling overlapping. We present the EE for each individual evacuation zone and then the global EE for all three EZ in Figure 5-7. The three peaks in the global EE correspond to the occurrence times of the three incidents, each increasing the evacuation zone and thus the number of evacuees. The first incident occurs at time 0, the second incident occurs 80 seconds later, and the third incident
occurs 120 seconds after the second. We notice that in all four plots from Figure 5-7 the number of evacuees in each EZ is continually decreasing, meaning that later incidents do not cause more traffic to enter the EZ of previous incidents.

Figure 5-7. EE comparison for Scenario 3
Figure 5-7 also shows that by evacuating the SZ, the evacuation times are significantly reduced. This is emphasized in Figure 5-8. For the All-Links approach after 32 minutes the evacuation is almost completed when SZ evacuation is used, while there are approximately 500 more vehicles to be evacuated when SZ is not used. Similarly, for the Fastest-Links approach, the difference is about 600 vehicles.

![Graph](image)

**(a)** (b)

Figure 5-8. EE comparison for Scenario 3 with and without SZ evacuation

To further assess the role and effectiveness of the secondary evacuation zone in multiple incidents evacuation we perform simulations on our fourth scenario (Figure 5-6). We summarize the performance of the All-Links and Fastest-Links techniques, for the non-overlapping incidents scenario, with and without SZ evacuation, in Figure 5-9(a). We observe the same behavior as for our third scenario: without evacuating the SZ, both All-Links and Fastest-Links approaches result in evacuation times significantly higher than when the SZ is evacuated. In the All-Links approach, after 32 minutes, the three EZ are empty if SZ evacuation is enforced, while there are
about 400 vehicles still left, if the SZ is not evacuated. However, in our third scenario, the performance improvement achieved by the *Fastest-Links* approach is higher than that of the *All-Links* approach and in our fourth scenario the *Fastest-Links* approach exhibits lower performance gain.

![Graphs showing the effect of traffic density on performance improvement when evacuating the SZ for Scenario 4 under (a) lower density, and (b) higher density](image)

**Figure 5-9.** The effect of traffic density on performance improvement when evacuating the SZ for Scenario 4 under (a) lower density, and (b) higher density
We also investigated the effect of the traffic density in the areas to be evacuated. Figure 5-9(b) shows the global EE of the *All-Links* and *Fastest-Links* approaches for the fourth scenario, in a slightly higher traffic density environment. We observe that when SZ evacuation is used the evacuation times are almost unaffected by this slight increase in traffic density, while the approaches that do not use SZ evacuation are drastically affected. Thus, we conclude that in higher traffic density environments, evacuating the SZ is significantly more important.

We also compare the performance of *All-Links* and *Fastest-Links* in Figure 5-10 and Figure 5-11, for Scenario 3 and Scenario 4, respectively. The graphs show that in Scenario 3, *Fastest-Links* performs almost similar to the *All-Links* approach when SZ is evacuated and slightly worse when SZ evacuation is not used. But in Scenario 4 (Figure 5-11), the *Fastest-Links* approach performs slightly better than the *All-Links* approach, especially when SZ evacuation is not applied. Thus, we can conclude that the performance of the two techniques is dependent on the topology of the network. We note that the results presented here are for a passive environment, thus, all street-segments are assumed to have an average speed equal to the speed limit. This may negatively affect the performance of the *Fastest-Links*, by causing it to direct traffic on routes that are not the fastest.

To have a better in-site into the role of SZ evacuation we investigated the throughput of one EEP from each evacuation zone. The chosen EEPs are shown in Figure 5-6.
Figure 5-10. Scenario 3: *All-Links vs. Fastest-Links* (a) without SZ, (b) with SZ

Figure 5-11. Scenario 4: *All-Links vs. Fastest-Links* (a) without SZ, (b) with SZ
The evacuation throughput at these points, measured as vehicles/second, is shown in Figure 5-12 (for the *All-Links* approach). We observe that if the SZ is not evacuated the throughputs at the EEP₁ and EEP₂ become close to 0 after 6 minutes, and at the EEP₃ after 18 minutes. When SZ evacuation is applied the throughput at each point continues to have high values until evacuation of the corresponding zone is completed.

![Figure 5-12](image-url)

*Figure 5-12. Evacuation throughput at (a) EEP₁, (b) EEP₂, and (c) EEP₃.*

Our simulation studies emphasize the efficiency and effectiveness of our proposed STEMS in handling single or multiple incidents scenarios. The algorithms proposed so far, are able to generate evacuation routes when given an incident location and scope and the current traffic...
conditions (if available) and to efficiently control the evacuation signals such that to guide the evacuees out of the disaster area. Furthermore, in case of multiple incidents, STEMS algorithms are able to revise the current evacuation plans under execution in order to safely reroute traffic away from all incidents. Although there are other requirements that STEMS has to assure we have presented our simulation studies in order to provide the reader with a better in-sight into our research and proposed algorithms and to facilitate our discussion on the next STEMS functionality that we address.

As identified in Section 3, the fourth requirement that STEMS has to assure is to adapt to the traffic dynamics. This involves updating the evacuation plans during their execution in order to respond to changes in traffic conditions or to incidents that may hamper evacuation efficiency (e.g., congestion caused by vehicle breakdowns or accidents). We address this requirement in the next section.
The advancements in sensor technologies and the increased availability of the traffic data allow us to enhance STEMS evacuation algorithms with capabilities to handle traffic dynamics. We have considered an iterative approach, which involves executing the evacuation plan generation algorithm repeatedly, with a new set of input data. The new input data consists of the latest average link speed values collected from sensors or other surveillance technologies, or input manually by human operators. This iterative approach would account for variations in the input data that arise either as consequences of accumulated queuing delays that trigger moderate changes in the link speed values, or as consequences of other stochastic factors (e.g., car accidents or congestion due to high visibility events) that drastically affect the traffic network characteristics. Updating the evacuation routes is essential and brings the following two advantages; first, it prevents congestion, by creating new evacuation routes using the fastest road segments and thus avoiding the already crowded areas; and secondly, it decreases congestion by directing traffic on currently congested routes to disperse on alternative routes.

To simulate our real-time traffic information obtained from sensors or other surveillance technologies, we used TSIS’s capability to extract and provide traffic information at the end of pre-defined time periods. Thus, during simulation, at the end of each time frame we extract the average link speed values and use them to generate a new plan for the next time period. The pseudo-code for the dynamic STEMS simulation is given in Figure 6-1. In Step 1, the evacuation
plan generation algorithm extracts the information relevant to the given incidents from the street network database (including the current link speed values) and constructs the evacuation routes. In Step 2 the evacuation signal control algorithm creates a simulation file, based on the evacuation plan generated in the previous step. In Step 3 we run Corsim for a pre-defined time-period on the generated simulation file; in Step 4, we extract the average link speed values from the Corsim’s output file. Then in Step 5 we update the street network database with the new link speed values, and then go back to Step 1, to generate a new evacuation plan based on the new link speed values.

For each pre-defined time period, do:

1. Execute the evacuation plan generation algorithm
2. Execute the evacuation signals control algorithm to generate a simulation file
3. Execute the Corsim simulation on the generated file for the pre-defined time period
4. Extract the link speed values from Corsim’s output file
5. Update the street network database with the new link speed values and go to step 1.

**Figure 6-1. Dynamic STEMS simulation**

We note that the *All-Links* approach does not take into account the traffic characteristics, and therefore, the evacuation routes will not change when the dynamic evacuation algorithms are applied. On the other side, the *Fastest-Links* approach updates the evacuation routes according to the current traffic conditions, but it suffers from the following drawback: when it regenerates the evacuation routes it tends to choose links that were not part of the previous routes, as those
would be emptier; thus, after each iteration, traffic that was originally following a route will be flowing at the drivers’ discretion, until eventually it will reach one of the new routes to follow; this incurs delays in evacuation and thus, diminishes the advantages of the dynamic Fastest-Links approach (as it will be shown by the simulation result in Section 6.2). To overcome this problem we propose a new evacuation routes construction approach that combines the advantages of the All-Links and the Fastest-Links approaches. We present this approach in detail in the next section.

6.1. **Network Flow-based Approach**

The main disadvantages of the two evacuation routes construction approaches, presented in the previous sub-sections, are the following: the Fastest-Links approach may create congestion, by concentrating traffic on the evacuation tree and leaving the other road segments unutilized; and the All-Links approach, even though it utilizes the entire road network, does not take into account the different traffic condition on the links. To overcome these limitations we propose a new approach, in which we use the entire traffic network (as in All-Links), but we direct more traffic on faster, less congested routes (as in Fastest-Links). In this approach, based on network flows, the evacuation routes construction is formulated as a minimum-cost flow problem [39]. We discuss this strategy in detail.

Our first objective was to model our evacuation problem as a minimum-cost flow problem. This requires the construction of a directed graph with a cost and a capacity associated to each link.
As the *evacuation graph* is an undirected graph we cannot use it directly in our model. To obtain a directed graph we use the *All-Links* approach, and traverse the *evacuation graph* in a breadth-first order, starting from the incident node, such that each link is visited only once. The traversal direction indicates the direction for each link. In order to completely define our model we also need to assign on each directed link a capacity and a cost. For our problem, as all street segments are part of the urban road network, we specify capacity as a function of the length of the street segment and the maximum density, assuming the same maximum density for all street-segments. As our goal is to achieve low evacuation times, we define the cost associated to each link as the time needed to traverse that link. This time can be computed as the ratio of the length and the average (or space mean) speed associated to each road segment.

Once we have our evacuation environment modeled we define our evacuation problem as finding the optimal traffic flow to be assigned on each directed link such that to minimize the overall evacuation time. The direction assigned on each link, while modeling evacuation as minimum-cost flow problem, will represent the *evacuation routes*, i.e., the directions that traffic would follow to exit the EZ. On the other hand, the flow assignment will provide us information about how to direct traffic at intersections, on the *evacuation routes*, such that to better take advantage of the different traffic conditions in various parts of the EZ and minimize the evacuation time.

The flow assigned for each road segment has to satisfy two constraints: first, it has to be positive and less than or equal to the capacity of that link; and secondly, the sum of the flows on all incoming links on a node plus the traffic volumes on these incoming links has to be equal to the
sum of the flows on the outgoing links (that is, the amount of traffic entering an intersection has
to be the same as the amount of traffic exiting it). We define the traffic volume on a link as the
traffic density times the length of that link. We assume an inverse linear relationship between
density and the average speed associated with each road segment. The rationale is that the lower
the traffic density on a road segment, the higher the average speed is; and on congested road
segments the average speeds are lower. When a street segment is empty, i.e., the speed
associated with it is 0, its corresponding volume is 0. The two constraints that the flow is
subjected to, can be defined formally as follows:

\[ 0 \leq flow_{ij} \leq capacity_{ij}, \text{ for all links } ij \]

\[ \sum_j (flow_{ji} + volume_{ji}) = \sum_j flow_{ij}, \text{ for all nodes } i \]

We explain the notation used and the justification for the second restriction in details. The traffic
flow entering an intersection from a particular road segment is equal to the traffic flow entering
that road segment plus the traffic volume already on that road segment. This is illustrated with a
3-way intersection in Figure 6-2. It shows traffic entering an intersection as thick white arrows
resulting from the thin black-contour arrows, which represent the traffic flow entering a road
segment. The difference in thickness represents the traffic volume already on that road segment.
Traffic flowing out of the intersection is represented as gray arrows and is equal to the traffic
entering that intersection (the white arrows). Even though this equality refers to intersections we
can express it in terms of the flow associated with each road segment, as follows: 

\[ (flow_{21} + volume_{21}) + (flow_{31} + volume_{31}) + (flow_{41} + volume_{41}) = flow_{12} + flow_{13} + flow_{14}. \]
The above equality will change once evacuation starts, as traffic at intersections will be directed such that to follow the evacuation directions. For example, given the evacuation directions from Figure 6-3 (assigned using the All-Links approach), we can see that \( \text{flow}_{12} \), \( \text{flow}_{41} \) and \( \text{flow}_{31} \), in Figure 6-2, will become 0 as those flows are opposite to the evacuation directions. Therefore, the above equality becomes: 

\[
(\text{flow}_{21} + \text{volume}_{21}) + (\text{volume}_{31}) + (\text{volume}_{41}) = \text{flow}_{13} + \text{flow}_{14},
\]

and Figure 6-2 becomes Figure 6-3. Moreover, traffic flowing opposite to the evacuation direction (\( \text{volume}_{41} \) and \( \text{volume}_{31} \)) will appear only in the initial stage of evacuation. This is because traffic is directed in the evacuation direction when it reaches an intersection. Thus, after the initial stage, traffic will be flowing only in the evacuation direction. This is illustrated in Figure 6-4. In this situation, the equality becomes: 

\[
(\text{flow}_{21} + \text{volume}_{21}) = \text{flow}_{13} + \text{flow}_{14}.
\]
This represents the second restriction that we use in calculating the flow assignment. Our evacuation problem can be formulated as finding the flow assignment that minimizes the time required by each flow of vehicles to traverse its corresponding link. Formally, our objective can be defined as:

\[
\text{minimize } \sum_{ij} \text{flow}_{ij} * \text{length}_{ij} / \text{speed}_{ij}
\]

given the constraints: \( 0 \leq \text{flow}_{ij} \leq \text{capacity}_{ij} \), for all links \( ij \)

\[
\sum_{j} (\text{flow}_{ji} + \text{volume}_{ji}) = \sum_{j} \text{flow}_{ij}, \text{ for all nodes } i
\]

The optimal solution to the flow assignment problem can be obtained using linear programming methodologies.
This *Flow-based* approach brings the following two advantages: first, it uses the entire street network as *evacuation routes*, thus avoiding the congestion that appears in the *Fastest-Links* approach; and secondly, it leverages information about the traffic conditions, when directing traffic along the *evacuation routes*, by distributing more traffic flow on those paths that have better travel times. In order to direct traffic according to the flow assigned on each link, we use a simple timing scheme in which we divide the time among the possible outgoing links proportional to the flow on those links.

### 6.2. Simulation Results

Our simulation environment consists of an area of the road network of downtown Orlando, in Florida. We considered three incidents, with overlapping EZs, with 100, 124 and 81
intersections; and 141, 186 and 122 road segments, respectively. We have investigated the performance of our dynamic STEMS, using the *Total Evacuation Time (TET)* and the *Evacuation Effectiveness (EE)*.

Table 6-1. TET comparison

<table>
<thead>
<tr>
<th>TET [minutes]</th>
<th>All-Links</th>
<th>Fastest-Links</th>
<th>Flow-based</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Static/Dynamic</td>
<td>Static</td>
<td>Dynamic</td>
</tr>
<tr>
<td>EZ 1</td>
<td>21.00</td>
<td>20.33</td>
<td>20.00</td>
</tr>
<tr>
<td>EZ 2</td>
<td>46.33</td>
<td>30.66</td>
<td>22.00</td>
</tr>
<tr>
<td>EZ 3</td>
<td>54.00</td>
<td>37.66</td>
<td>35.33</td>
</tr>
<tr>
<td>All EZs</td>
<td>54.00</td>
<td>37.66</td>
<td>35.33</td>
</tr>
</tbody>
</table>

Table 6-1 shows the TET for the dynamic approach for the *All-Links*, *Fastest-Links* and *Flow-based* algorithms in comparison with the static approach (in which the evacuation plans are not updated periodically). Note that the performance of the static and dynamic *All-Links* approach is the same, as this approach is insensitive to traffic information. The first three rows correspond to the time necessary to evacuate each of the three evacuation zones; and the last row corresponds to the time necessary to evacuate all three incident areas.

The evacuation effectiveness comparison is shown in Figure 6-5 and Figure 6-6. The results from Figure 6-5 show that when evacuation plans are not updated dynamically, the *Fastest-Links* approach performs much better than the *All-Links* approach and almost identical to the *Flow-based* approach.
However, Figure 6-6 shows that when evacuation plans are updated dynamically *Fastest-Links* approach exhibits a small performance improvement, while the *Flow-based* approach performs significantly better. This is explained by the fact that updating the evacuation routes in *Fastest-Links* approach, may cause traffic that was originally following a route to flow aimlessly, until eventually it will reach one of the new routes to follow. But by the time it will reach it the routes might change again. In contrast, in the *Flow-based* approach the evacuation routes do not change, but the traffic distribution on the evacuation routes will change. This will ensure that when changes occur traffic will still be directed based on the same evacuation plan but more traffic will be directed on those routes that are faster or less congested.
Figure 6-6. EE comparison: static (dotted lines) vs. dynamic (solid lines) approaches
7. ENHANCED EVACUATION ROUTES CONSTRUCTION
ALGORITHMS

The three evacuation construction algorithms presented so far have the following weaknesses: while the *All-Links* approach utilizes the entire street network as evacuation routes, it does not take into account the different traffic densities of the road segments. On the other hand the *Fastest-Links* approach is sensitive to traffic information, but it inherently facilitates congestion by utilizing only a small percentage of the road network as evacuation routes and leaving the other road segments unutilized. The *Flow-based* approach solves the above problem by utilizing the entire road network and directing traffic on the less congested road segments, however, it is based on linear programming techniques and therefore, it is only suitable for small traffic networks. In the next sub-sections we propose a set of evacuation route construction algorithms that overcome all the above limitations.

7.1. **Hybrid Evacuation Route Construction Approach**

The *Hybrid* approach is similar to the *Fastest-Links* approach but combines the advantages of all the previously proposed techniques. This approach, illustrated in Figure 7-1, takes into account the different traffic densities of the links (as the *Fastest-Links* approach) and utilizes a higher percentage of the road network as evacuation routes (like the *All-Links* approach). Moreover, the execution time will be much faster than for the *Flow-based* approach, which is based on linear programming techniques.
Instead of constructing a tree containing the fastest routes from the incident node to each EEP (as the Fastest-Links approach) the Hybrid approach constructs the tree containing the fastest paths from the incident node to all the other nodes in the evacuation zone. As for the Fastest-Links approach, the performance metrics employed for constructing the shortest path tree is the aggregated time, and an evacuation direction is assigned on each link in the tree (from the root to the leaves). Simulation studies to assess the performance of this new approach and compare it with the other approaches will be presented in Section 7.3.

Figure 7-1. The Hybrid approach

7.2. Exit-Point based Evacuation Route Construction Approaches

The main limitation of the previously proposed evacuation routes construction approaches is that they are only suitable for uniform, grid-based road network environments. The All-Links
approach directs traffic away from the incident, but for irregular networks, away from the incident might not necessarily be towards or closer to an EEP. The same problem applies to the Hybrid approach. On the other hand, the Fastest-Links approach does route traffic towards the EEPs, however, if these EEPs are not dense enough or not uniformly distributed around the EZ then a large part of the evacuation area will not be covered by the evacuation routes. Therefore, it will take longer for the traffic on those links to be assimilated by the evacuation routes and directed towards the EEPs.

To overcome the limitations of the previously proposed algorithms, we propose a reversed evacuation routes construction approach, in which we build the evacuation routes starting from the EEPs and ending at the incident node. We propose three algorithms based on this approach: the Hop-based, the Hop-Away, and the Time-based algorithms.

7.2.1. The Hop-based Approach

In the Hop-based approach we traverse the road network starting from the EEPs and compute for each node the distance in terms of number of nodes from itself to the closest EEP. The algorithm is as follows. In the first step the EEPs are all assigned the value “0”. In the second step the nodes inside EZ directly connected to the EEPs will be assigned the value “1”. In the next step, the nodes directly connected with those having the value “1”, will be assigned the value “2”, if they have not already been assigned the value “0”, and so on. At each step, if the incident node is reached it will be ignored, i.e., it will not be assigned a value. After all the nodes have been
visited the incident node is assigned a value greater with "1" than the highest value assigned to any node in the network. An evacuation direction is assigned to each link, from the node with the higher value to the node with the lower value. If two end nodes have the same value their corresponding link will not be assigned an evacuation direction. Figure 7-2 illustrates the Hop-based approach and the values assigned to each node, representing the distance (in terms of numbers of nodes to be traversed) from that node to the closest EEP.

![Figure 7-2. The Hop-based approach](image)

The rationale of the Hop-based approach is as follows. By directing traffic from the nodes with higher values towards those with lower values we ensure that traffic is directed towards the EEPs, passing through the minimum number of intersections. As the intersection delay is the
main source of delay on arterials, by routing traffic out of the evacuation area through the minimum number of intersections we can achieve a minimum delay. In certain topology scenarios this approach may result in some traffic actually getting closer to the actual incident location in its way to the closest EEP. Although this would allow that traffic to exit the evacuation zone faster, getting closer to the actual incident location is most likely undesirable. Therefore, when assigning an evacuation direction to a link we also check if the node with the lower value is further away from the incident than the node with the higher value. If this is true than the evacuation direction will be assigned to that link, otherwise that link will have no evacuation direction (see the dotted line in Figure 7-2).

7.2.2. The Hop-Away Approach

The *Hop-Away* approach is designed to assure that when traffic is directed towards the EEPs it is also directed away from the incident. Each node is assigned two values. The first value is determined as in the *Hop-based* approach. The second value is determined by applying the same algorithm used by the *Hop-based* approach, but starting from the incident node instead of the EEPs. In other words, in step one the incident node is assigned the value “0”. Then in step two all the nodes connected to the incident node are assigned the value “1”. In the next step all the nodes connected to those with value “1” are assigned the value “2”, and so on. Therefore, the two values assigned to each node represent the distances (in terms of number of nodes) from that node to the closest EEP and to the incident node, respectively. We then assigned evacuation directions to all links from the node with the higher first value to the one with the lower first
value. We then check the second values for each link. If the assigned evacuation direction of a certain link is from a node with the higher second value to the one with the lower second value than the evacuation direction for that link is removed. This assures that only evacuation directions that direct traffic towards an EEP and away from the incident node are assigned to the links. The \textit{Hop-Away} approach is illustrated in Figure 7-3.

![Figure 7-3. The Hop-Away approach](image)

\textbf{7.2.3. The Time-based Approach}

The above two approaches are designed for situations in which real-time traffic information is limited or not available. On the other hand, the \textit{Time-based} approach is based on real-time traffic
information. It assumes that each link is characterized by a certain space mean speed and therefore, each link can be assigned the average time required to traverse it. Then each node is assigned a value equal to the minimum travel time required to reach an EEP starting from that node. As for the *Hop-based* approach, the evacuation route computation algorithms starts from the EEPs and the nodes are visited in the same order. In the first step all EEPs are assigned the value “0”. In the next step each node directly connected to an EEP is assigned the value of the EEP node it is connected to, plus the travel time corresponding to that link. If the value to be assigned to a certain node is higher than the value previously assigned, then the node keeps its old (smaller) value. Therefore, the value of each node will represent the minimum travel time required to reach an EEP starting at that node. As for the *Hop-based* approach the incident node will be ignored until all nodes have been visited, when it will be assigned a very large value to assure that no other nodes will direct traffic towards it. Then, at each node traffic will be directed towards the one neighboring node for which the sum of its assigned value and the travel time of the corresponding link is the lowest. This will ensure that traffic entering each node will flow on the fastest route towards an EEP.

The *Time-based* approach is illustrated in Figure 7-4. Note that each node has exactly one outgoing arc, except the nodes that are connected to more than one EEP. These nodes will have outgoing arcs towards all connected EEPs and not only towards the one that is closer in terms of travel time. This is done in order to assure that traffic will not enter the evacuation zone.
As in our previous simulation studies, our environment consists of an area of the surface transportation network of downtown Orlando in Florida. We present the results obtained for an incident with 362 nodes and 573 links, under two scenarios, with actuated and with pre-timed control. We mention here that for large network size, like the above scenario, the response time of the Flow-based approach is relatively high, making this approach not feasible for real-time traffic evacuation. Therefore, we only compare the performance of the remaining six approaches: the All-Links, Fastest-Links, Hybrid and the three Exit-Point based approaches. We have
investigated the performance of our evacuation algorithms, based on the three performance metrics used in our previous studies: Total Evacuation Time (TET), Evacuation Effectiveness (EE), and Evacuation Eniformity (EU). Table 7-1 shows the TET for all six evacuation algorithms.

<table>
<thead>
<tr>
<th>TET [minutes]</th>
<th>Pre-timed</th>
<th>Actuated</th>
</tr>
</thead>
<tbody>
<tr>
<td>All-Links</td>
<td>78.33</td>
<td>67.66</td>
</tr>
<tr>
<td>Hybrid</td>
<td>48.00</td>
<td>38.33</td>
</tr>
<tr>
<td>Fastest-Links</td>
<td>36.33</td>
<td>20.33</td>
</tr>
<tr>
<td>Hop-based</td>
<td>27.33</td>
<td>16.66</td>
</tr>
<tr>
<td>Hop-away</td>
<td>23.00</td>
<td>15.00</td>
</tr>
<tr>
<td>Time-based</td>
<td>23.66</td>
<td>13.33</td>
</tr>
</tbody>
</table>

As expected, for the actuated environment evacuation times are significantly smaller. The poor performance of the *All-Links* approach is attributed to the irregular nature of the road network surrounding the chosen incident. Recall that the *All-Links* approach is based on breadth first graph traversal, and when applied on regular, grid-like environments, it does direct traffic away from the incident and towards the EEPs. However, in a more irregular road network environment (as the one chosen for this scenario) the *All-Links* approach does not guarantee that traffic is directed away from the incident node and towards the EEPs. In contrast, the *Fastest-Links* approach always directs traffic towards the EEP, as each evacuation route ends in an EEP. Therefore, the *Fastest-Links* approach is not as affected by the irregularity of the road network as the *All-Links* approach. This explains why the performance difference between the *All-Links* and the *Fastest-Links* applied in irregular network environments is significantly higher than their
performance difference in grid-like environments, observed throughout Section 5. The *Hybrid* approach suffers from the same weakness as the *All-Links* approach. Although when applied on regular, grid-like road networks it performs better than the *Fastest-Links* approach (as it uses a higher percentage of the road network as evacuation routes), in the chosen scenario it performs worse, because it directs traffic away from the incident but not necessarily towards the EEPs. Although the *Fastest-Links* approach is not as sensitive to the road network irregularities as the *All-Links* and *Hybrid* approaches, it is instead sensitive to the distribution of the EEPs around the EZ. If the EEPs are not uniformly distributed around the EZ, a large percentage of the road network will not be covered by evacuation routes and it will take longer for traffic in this area to be assimilated by the evacuation routes. This is the reason why the performance of the *Fastest-Links* approach is lower than that of the exit-point based approaches.

The evacuation effectiveness, plotted as the number of vehicles inside the EZ over time, is shown in Figure 7-5 and Figure 7-6, for the pre-timed and actuated control respectively. In the pre-timed environment the number of vehicles in EZ decreases at almost the same rate for all approaches (except *All-Links*) until minute 15 of the evacuation. At that point the *Time-based* approach takes the lead, resulting in the smallest TET.

In the actuated environment, the number of vehicles in EZ decreases at more diverse rates for the different approaches. This shows that the delays inherent to the pre-timed environment conceal or diminish the variation in performance of the evacuation routes construction algorithms. The difference in terms of EE between different algorithms is more noticeable in the
actuated environment. Notice that Fastest-Links approach has the best evacuation effectiveness until minute 11, after which its performance decreases, allowing the exit-point based approaches to take the lead again.

![Figure 7-5. EE comparison (pre-timed control)](image-url)
Evacuation uniformity reflects how uniform the traffic is distributed among the EEPs. High EU means that traffic is not uniformly distributed among the EEPs and this may lead to congestions around the EZ. The results of this study are summarized in Table 7-2. The total number of evacuees exiting at each EEP is plotted in Figure 7-7. We observe from Table 7-2 that the EU is
higher for the exit-point based approaches, therefore traffic exits at the EEPs more uniformly under the incident-based approaches.

Table 7-2. EU comparison for the exit-point based approaches

<table>
<thead>
<tr>
<th>Method</th>
<th>EU Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td>All-Links</td>
<td>48.89</td>
</tr>
<tr>
<td>Hop-based</td>
<td>65.46</td>
</tr>
<tr>
<td>Hybrid</td>
<td>53.19</td>
</tr>
<tr>
<td>Hop-away</td>
<td>64.72</td>
</tr>
<tr>
<td>Fastest-Links</td>
<td>47.43</td>
</tr>
<tr>
<td>Time-based</td>
<td>66.98</td>
</tr>
</tbody>
</table>
Figure 7-7. EU comparison for the exit-point based approaches

We have also investigated the performance of our new evacuation routes construction algorithms when deployed in a dynamic environment, in which evacuation plans are periodically updated during the evacuation operation, according to the changes in traffic conditions. We note that the All-Links approach, the Hop-based and Hop-Away approaches do not take into account traffic information and therefore, they behave similarly in a dynamic and static environment. Table 7-3
shows a comparison of the total evacuation times of the three approaches that are sensitive to traffic conditions, when applied in a static and dynamic environment.

<table>
<thead>
<tr>
<th>TET [minutes]</th>
<th>Static</th>
<th>Dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fastest-Links</td>
<td>36.33</td>
<td>35.33</td>
</tr>
<tr>
<td>Hybrid</td>
<td>48.00</td>
<td>16.33</td>
</tr>
<tr>
<td>Time-based</td>
<td>23.66</td>
<td>15.00</td>
</tr>
</tbody>
</table>

From TET perspective the *Fastest-Links* and the *Time-based* approaches perform better the Hybrid approach in the static environment (i.e., evacuation plans are not updated dynamically). However, in the dynamic environment the *Hybrid* and the *Time-based* approaches perform the best. As explained in Section 6.2 the poor performance of the dynamic *Fastest-Links* approach is determined by the fact that updating the evacuation routes may cause traffic that was originally following a route to flow aimlessly, until eventually it will reach one of the new routes to follow. But by the time it will reach it, the routes might have changed again. In contrast, the *Hybrid* and the *Time-based* approaches use a higher percentage of the road network as evacuation routes and therefore, when the evacuation plans are updated, the new routes are reached faster. Although from the TET point of view the *Time-based* approach has the best performance, the EE comparison (Figure 7-8) shows that in the dynamic environment the *Hybrid* approach evacuates more traffic faster than the *Time-based* approach (i.e., the area under the Dynamic Hybrid curve is the smallest among all approaches). However, the *Time-based* approach is the only one that performs well in both a static and dynamic environment.
Figure 7-8. EE comparison: static (dotted-lines) vs. dynamic (solid lines) approaches
7.4. Influence of Driver Behavior on Evacuation Efficiency

Besides the relative performance of the proposed evacuation route construction algorithms, we have also investigated the influence of the driver behavior on the efficiency of our algorithms. In our preliminary simulation studies, we assumed that the drivers behave in a safe manner, obeying traffic rules without making mistakes or misjudgments and following the evacuation signals directions. This, however, is true only in an ideal world and it might be very inadequate in emergency situations when panic and fear might push drivers to lose their self-control and behave in unpredictable ways [40]. Our objective was to investigate the changes in effectiveness of our evacuation algorithms when applied in a more realistic environment that better emulates real driver behaviors in crisis situations. For this purpose, we have introduced different percentages of forced misbehavior in our simulations and evaluate the variation in the total evacuation time of our algorithms. We have used the same simulation environment as in Section 7.3 with a slightly higher vehicle density. Figure 7-9 shows how sensitive the evacuation routes construction algorithms are to different percentages of drivers that do not follow the evacuation directions.

As expected, when forced misbehavior is introduced the All-Links approach shows a small variation in terms of TET. This is because in this approach the entire road network is used as evacuation routes and therefore, drivers are directed in the evacuation direction at each intersection. The same is true for the Hybrid approach, in which drivers are directed in the evacuation direction at all non-leaf nodes of the shortest-path tree, which represent a high
percentage of the nodes in the road network. Similarly, in the *Time-based* approach there is one outgoing evacuation direction at all nodes in the road network and therefore, drivers will be directed in the evacuation direction at all nodes (as in the *All-Links* approach). In contrast, in the *Fastest-Links* approach the evacuation routes are sparser and once a driver disobeys the evacuation directions and leaves the evacuation routes it will take longer for him/her to reach another route and be directed out of the disaster area. The same reasoning applies to the *Hop-based* and the *Hop-away* approaches in which the evacuation routes cover a smaller percentage of the road network and therefore, drivers are directed to follow the evacuation direction less often.

![Figure 7-9. Influence of driver behavior on the TET](image)

Figure 7-9. Influence of driver behavior on the TET
To have a better insight into the behavior of our algorithms when drivers disobey the evacuation directions we have also plotted the evacuation effectiveness for each algorithm in Figure 7-10. As the TET comparison, Figure 7-10 shows that for the All-Links, Hybrid and Time-based approaches the evacuation effectiveness is relatively similar for 0%, 5%, 10% and 20% of disobeying drivers, and starts to decrease slightly only when the number of drivers not following the evacuation directions reaches 30%. For the Fastest-Links approach, we notice that although the TET is the same for 0% and 5% of disobeying drivers, the EE for the 5% scenario is better than for the 0% scenario. This can be explained as follows: when all drivers follow the evacuation directions, the evacuation routes (which cover only a small percentage of the street network) become congested and this slows down the evacuation execution. When 5% of drivers are disobeying the evacuation direction, the routes become less congested and this allows faster evacuation.

Similar effects exist but are less noticeable for the Hop-based and Hop-Away approaches. For example, in Hop-based approach when 10% of the traffic is disobeying the evacuation directions traffic is evacuated slightly faster than when only 5% of the traffic is misbehaving. Similarly, in the Hop-away approach although the TET is the same for 0% and 5% of misbehaving traffic, the EE is higher when 5% of the traffic is disobeying the evacuation routes than when all traffic follows them. This positive effect, of reducing congestion, that misbehaving traffic has on the evacuation is more noticeable for the Fastest-Links approach than for the Hop-based and Hop-away approaches because the latter ones use as evacuation routes a higher percentage of the road network than the Fastest-Links approach, and therefore they are less susceptible to congestion.
For all approaches, we notice that the TET and EE become significantly worse only when the percentage of traffic disobeying the evacuation directions reaches 30%. Therefore, the effectiveness of our evacuation algorithms is negatively affected only when the percentage of traffic disobeying the evacuation directions is higher than 30%.
Figure 7-10. EE comparison for different % of drivers not following the evacuation plans
8. NEW EVACUATION PLAN UPDATING APPROACH

We have noticed during our research that updating the evacuation plans periodically might negatively affect evacuation performance in some situations, especially towards the end of the evacuation execution. This effect was noticed in the Fastest-Links approach, when changing the evacuation routes has caused traffic that was originally following a route to flow aimlessly, until eventually it would reach one of the new routes to follow. To better understand the effect of the evacuation routes updating on the performance of our algorithms, we have performed a detailed analysis and comparison of the behavior of the three proposed evacuation routes construction approaches that are sensitive to traffic information: the Fastest-Links, the Hybrid and the Time-based approaches.

8.1. **Analysis of Evacuation Routes Construction Approaches that are Sensitive to Traffic Information**

Figure 8-1 shows an evacuation plan constructed using the Fastest-Links approach. The evacuation routes are highlighted in red and the evacuation directions on the links that are part of the evacuation routes are indicated by arrows. As expected (and exemplified by the scenario in Figure 8-1) the evacuation routes for the Fastest-Links approach, cover a small percentage of the road network (in this particular case, approximately 40%). The sparseness of the evacuation routes makes them susceptible to congestion, as all the traffic inside the evacuation area will be directed out through one of these evacuation routes. Traffic that is not on the evacuation routes is gradually being assimilated by the evacuation routes. Once traffic reaches an evacuation route it
will follow it until it leaves the evacuation area via one of the EEPs. The fact that this assimilation process is gradual allows the evacuation routes to better handle the high amount of traffic that is flows through them.

Figure 8-1. The Fastest-Links approach
In other words, the negative effect of the sparseness of the evacuation routes is ameliorated by the gradual way in which they absorb the traffic inside the evacuation zone. This is one of the reasons why the performance of the Fastest-Links approach is consistently good throughout our simulation studies.

Another important reason that justifies the good performance of this approach is that the evacuation routes are constructed such that they guarantee to direct traffic towards the EEPs. This feature is essential for irregular street network environments, where directing traffic away from the incidents does not necessarily mean that traffic is directed towards the EEPs. In the Fastest-Links approach, each evacuation route ends in an EEP and that makes this technique behave well in irregular street network environments.

One negative aspect of the Fastest-Links approach is that it is heavily influenced by the distribution of the EEPs around the evacuation area. In those areas where EEPs are denser the evacuation routes will cover a higher percentage of the street network. Similarly, in those areas where the EEPs are sparser the evacuation routes will be sparser and therefore, it will take longer for them to assimilate the traffic that is not on the evacuation routes.

Another good feature of the Fastest-Links approach is related to its deployment in an actuated environment. An evacuation plan constructed with this approach has the structure of a tree. The fact that in a tree each node has only one parent assures that each node on the evacuation routes has only one incoming traffic stream, once all the traffic is assimilated by the evacuation routes.
At this point, if actuated traffic control is used, at each node the right of way will be maintained mainly by that incoming traffic stream and therefore, traffic will flow almost uninterrupted, through the evacuation routes towards the EEPs. This represents another factor that ameliorates the negative effect of the sparseness of the evacuation routes, and justifies the high performance of the *Fastest-Links* when deployed in actuated environments, as observed through our simulation studies.

In the *Hybrid* approach (illustrated in Figure 8-2) the evacuation routes cover a much higher percentage of the road network. Therefore, this approach is less susceptible to congestion than the *Fastest-Links* approach. However, the evacuation routes only guarantee that traffic is directed away from the incident and not necessarily towards an EEP. For example, the areas within the blue circles in Figure 8-2 identify regions of the evacuation routes where traffic is directed away from the incident but into areas that are further away from any EEP. Traffic in these areas will eventually be directed towards the EEPs because of our rule for directing traffic such that not to flow against the evacuation direction. However, the fact that traffic is directed into these areas first and then out of these areas results in significant delay and makes the performance of the *Hybrid* approach decrease in irregular street network environments, as seen throughout our simulation studies.

Another weakness of the *Hybrid* approach is related to the fact that although the evacuation routes form a tree structure, not all leaves of the tree represent EEPs. Consequently, there is no guarantee that there will be an outgoing evacuation direction from each node on the evacuation
routes. More exactly, those leaves of the evacuation tree that are not EEPs will not have an outgoing evacuation direction. At all these nodes traffic will be allowed to follow any direction as long as it does not flow against the evacuation directions.

Figure 8-2. The Hybrid approach
In other words, there is no guarantee that once traffic reaches an evacuation route it will be directed towards an EEP. The areas within the blue circles in Figure 8-2, show that in irregular network topologies, directing traffic towards leaf nodes that are not EEPs may incur additional delays that increase the total evacuation time. When the street network has a more regular grid-based topology (the upper-left quadrant in Figure 8-2) directing traffic towards leaf nodes that are not EEPs does not affect the evacuation performance, as this traffic will sink immediately onto routes that do lead towards EEPs.

In the *Time-based* approach illustrated in Figure 8-3, the evacuation routes cover almost the same percentage of the road network as the *Hybrid* approach. While the *Hybrid* approach guarantees exactly one incoming evacuation direction in each node in the street network (each node has exactly one parent), the *Time-based* approach guarantees exactly one outgoing evacuation direction from each node in the street network. (except those nodes that are connected to more than one EEP). This property does not only assure high coverage of the street network but it also guarantees that there is guidance for traffic at each node/intersection inside the evacuation area. Recall that in the *Hybrid* approach there is no guidance once drivers reach leaf nodes that are not EEPs and in the *Fastest-Links* approach there is no guidance at the nodes that are not part of the evacuation routes. Therefore, the *Time-based* based approach is the only technique among the three, which guarantees traffic guidance at each intersection. Moreover, this guidance also assures that traffic entering each intersection is directed towards the closest EEP. This allows the *Time-based* approach to effectively handle irregular street network topologies and non-uniform EEPs distributions around the evacuation zone.
Table 8-1 summarizes the comparison of the three evacuation routes construction approaches detailed above. In the next section we examine the behavior of these approaches in a dynamic environment, when evacuation routes are updated periodically according to the traffic conditions.
Table 8-1. Comparison of the evacuation routes construction approaches

<table>
<thead>
<tr>
<th></th>
<th>Fastest-Links</th>
<th>Hybrid</th>
<th>Time-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>High evacuation routes coverage</td>
<td>NO</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>(handle uneven distributions of EEPs)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provide guidance at each node</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Directs traffic towards EEPs</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>(can handle irregular networks)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8.2. The Effect of Updating the Evacuation Routes Periodically on Evacuation Efficiency

Our simulation studies have shown that updating the evacuation routes periodically, when the Fastest-Links approach is used, does not result in a significant evacuation performance improvement. The reason for this behavior is the sparseness of the evacuation routes which translates into long distances between the old and new evacuation routes. Moreover, the fact that there is no guidance at intersections that are not part of the evacuation routes makes it even harder for traffic on the old evacuation route to reach one of the new routes. These delays, in transferring traffic from congested routes to new less congested routes, diminish the advantage of updating the evacuation routes periodically.

The Hybrid approach suffers from the same drawback described above. However, it is affected much less than the Fastest-Links approach because the evacuation routes are denser and therefore, traffic from congested routes can reach the newly generated routes much faster.
The *Time-based* approach is the one that benefits most from the periodic evacuation plan updating. However, if the evacuation plans are updated too frequently, it might cause traffic to oscillate back and forth in the attempt to reach the closest EEP, instead of just flowing towards one of the EEPs. Therefore, we have proposed a new evacuation plan updating approach in which the frequency of updating the evacuation routes is dependent on the significance of the changes in the traffic conditions. This improved evacuation plan updating approach is presented in the next section.

### 8.3. Improved Evacuation Plan Updating Approach

While real-time traffic conditions are pulled periodically, we propose to update the evacuation plan only in those time periods in which there are significant changes in the traffic conditions. To quantify these changes we investigate separately the links that are part of the evacuation routes and those that are not. Given the fact that real-time traffic information suffers from various degrees of inaccuracy we consider that the traffic conditions on a link have changed only if the difference between the old and new reported average speeds is higher than a certain threshold. For our simulation studies, we choose a 5 mph threshold to identify changes in traffic conditions on individual links.

To quantify the overall change in the traffic conditions in the evacuation area, we first look at those links on the evacuation routes for which the new reported speeds are lower than the previous ones. In other words we would like to reconstruct the evacuation routes when the
speeds on the current routes become smaller. Conversely, we could also look at the links that are not part of the evacuation routes which exhibit a speed increase. However, all of the links that are not part of the evacuation routes will exhibit a speed increase during the evacuation, because the traffic will be absorbed by the evacuation routes. Therefore, we do not consider the changes on the links that are not on the evacuation routes, as triggers for evacuation plan reconstruction.

Once we identify the links of the evacuation routes that exhibit a decrease in speed, we update the evacuation routes only if the number of such links is higher than a certain threshold. This threshold is defined as a function of the proportion of the number of links on the evacuation plan out of the total number of links in the evacuation area. In our simulation studies we consider this threshold as 4 times the ratio of the number links in the evacuation area over the number of links that are part of the evacuation plan.

We present the results of the proposed evacuation plan updating approach, in terms of total evacuation time and evacuation effectiveness, for the scenario from Figure 8-1. The total evacuation time is given in Table 8-2 and the evacuation effectiveness comparison is plotted in Figure 8-4.

<table>
<thead>
<tr>
<th>TET [minutes]</th>
<th>Static</th>
<th>Old Dynamic Approach</th>
<th>New Dynamic Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fastest-Links</td>
<td>30.33</td>
<td>29.33</td>
<td>20.33</td>
</tr>
<tr>
<td>Hybrid</td>
<td>53.00</td>
<td>21.00</td>
<td>18.66</td>
</tr>
<tr>
<td>Time-based</td>
<td>27.66</td>
<td>19.33</td>
<td>17.33</td>
</tr>
</tbody>
</table>
Figure 8-4. EE comparison for the new evacuation plan updating approach
Table 8-2 compares the TET for the three considered evacuation plan construction approaches in three environments: *static* (when evacuation plans are not updated dynamically), *old dynamic* (when evacuation plans are updated every predefined time-period) and *new dynamic* (when evacuation plans are updated only when the traffic conditions on the evacuation routes deteriorate significantly). As expected, the *Time-based* approach performs better than the *Hybrid* and *Fastest-Links* approaches in all three environments. However, in the new dynamic environment the difference in performance between the *Time-based* and the other two approaches is smaller. In other words, the new evacuation plan updating approach has the highest positive effect on the *Fastest-Links* approach.

The three dotted lines in Figure 8-4 represent the EE of the three considered techniques, when applied in a static environment. Their corresponding solid lines represent the EE of the three techniques when applied in a dynamic environment, in which evacuation plans are updated periodically. The remaining three solid lines represent the EE of the three approaches when applied in a dynamic environment in which evacuation plans are only updated when the traffic conditions on the evacuation routes deteriorate significantly. As the TET results, the EE plots show that the new evacuation plan updating approach performs significantly better than the old one. As expected, the most significant performance improvement is observed for the *Fastest-Links* approach. Moreover, the EE comparison plot shows that the new evacuation plan updating approach equalizes the behavior of the three evacuation routes construction approaches. In other words, this updating approach manages to compensate for the various weaknesses (summarized
in Table 8-1) of the different evacuation routes construction approaches, and brings their performance up to the optimal level.
9. SYSTEM PROTOTYPING

Our final objective was to develop a system prototype ready to be deployed in a real production environment, e.g., the SunGuide℠ environment, a statewide transportation management system available from Florida Department of Transportation (FDoT). This approach would allow STEMS to be easily integrated into the Florida transportation system, and thus would facilitate its deployment. The design environment for the STEMS study is shown in Figure 9-1. It contains two major software systems, namely SunGuide℠ and the traffic simulation package, TSIS.

![Figure 9-1. Environment for designing, testing & deployment of STEMS](image)
The simulator provides a platform to test and evaluate the STEMS algorithms. During the development stage, STEMS only interacts with the traffic simulator and a street database through a set of interfaces. During its deployment STEMS interacts with the existing SunGuideSM software to control the various emergency deployed control devices, i.e., the evacuation signals.

9.1. **Obtaining the Input Data for the STEMS Prototype**

Our first requirement in designing the prototype was to obtain input from the Traffic Management Center (TMC) managers and operators, and the Statewide Traffic Incident Management Teams. This input included all available real-time ITS data (i.e., speed, volume, occupancy, travel times) and the incident detection and management processes, etc. During this phase we have identified that not all FDoT Districts are deploying SunGuideSM at their TMCs. Moreover, the format of the collected sensor data and the processed traffic information, as well as, the traffic data management software differs from District to District. Therefore, to facilitate the integration of our STEMS with any traffic management software, we have proposed a Canonical Data Format (CDF) for the sensor traffic data, which our STEMS algorithms will assume as input. Specific data conversion modules can be developed for each FDoT District as necessary, to convert their real-time data on the fly from their specific format into the CDF required by STEMS. This process is illustrated in Figure 9-2.
During the data acquisition stage we have also encountered the problem of not having direct access to the real-time feeds of traffic data from the FDoT Traffic Data Servers, mainly because their systems were still under development or under deployment testing. However, archived data for various periods of time was available upon request. Therefore, to be able to test our prototype, we have proposed the development of software modules to emulate real-time sensor data from the archived data. Based on the timestamps of the archived data, the sensor emulator software feeds the traffic data into the STEMS database in real-time. From here our STEMS algorithms can pull the available data as frequently as necessary. As illustrated in Figure 9-3, the archived data from a certain District is first passed through its corresponding data conversion module, to be transformed into the CDF and then it is fed into STEMS by the sensor emulator.
For testing our prototype, we have used archived data from FDoT District 5. The traffic management system used by District 5 (specifically, the Center-to-Center component) allows different subscribers to register and receive real-time traffic information in the form of .xml files. An .xml file is sent to each subscriber every minute. Each .xml file contains various traffic parameters (average speed, average travel time, etc.) for a set of predefined road segments. The format of this .xml file is shown in Figure 9-4. Although we could not register as a subscriber to receive these .xml files in real-time, we were provided a set of .xml files corresponding to a couple of weeks worth of data. To be able to use this data in our prototype we had to perform the following two steps (illustrated in Figure 9-3): first, to convert this data into the canonical data format used by STEMS and secondly, to use the <timestamp> field (from the .xml format) to feed the data into our STEMS database in real-time. We have used Microsoft SQL Server for our STEMS database, JAVA to develop the application part of the prototype and the JDBC/ODBC bridge to communicate between the two.
In addition to the real-time traffic information the STEMS database also stores the traffic network topology. The following sub-sections describe in detail the STEMS database.

### 9.1.1. Traffic Network Topology

The traffic network topology used by STEMS was obtained from the Census 2000 TIGER/Line Data Set, which consists of shapefiles created from the Topologically Integrated Geographic Encoding and Referencing (TIGER) database of the United States Census Bureau. The shapefiles
contain data about several features: line features: (roads, railroads, hydrography, and transportation and utility lines), boundary features: statistical (e.g., census tracts and blocks), government (e.g., places and counties), and administrative (e.g., congressional and school districts) and landmark features: point (e.g., schools and churches), area (e.g., parks and cemeteries), etc. The Census 2000 TIGER/Line data is available for download at: “http://arcdata.esri.com/data/tiger2000/tiger_download.cfm”. As we were only interested in the road network topology we have used only the roads layer from the 2000 TIGER/Line Data Set.

Each shapefile consists of a main file, an index file, and a dBASE table. The main file is a direct access, variable-record-length file in which each record describes a shape with a list of its vertices. For our roads layer, each record describes a line (representing a road segment) with its two end nodes. In the index file, each record contains the offset of the corresponding main file record from the beginning of the main file. The dBASE table contains feature attributes with one record per feature. The one-to-one relationship between geometry and attributes is based on record number. The dBASE table for our road layer contained various attributes, including road ID, the IDs of the end nodes, road name, road type, etc. We have used the ArcMap 9.1 software to combine the geometry and attributes of our roads layer into the dBASE file. Specifically, we extracted the projected coordinates of the end nodes of each road (line feature) from the main shapefile into the dBASE table. This table was then imported into MSQL. The screenshot in Figure 9-5 shows the relevant fields of this table that are used by the STEMS algorithms.
The **TLID** field represents a unique identifier for each link/road segment. The **FNODE** and **TNODE** represent the IDs of the end nodes of each link, and **(FEAST, FNORTH)** and **(TEAST, TNORTH)** represent their projected geographic coordinates (based on NAD83 coordinate system). **FENAME**, **FETYPE** and **LENGTH** represent the road segment name, type and length respectively.

### 9.1.2. Traffic Network Conditions

In addition to the above fields that represent the road network topology, we have added two more fields to store the traffic conditions. Specifically, the two additional fields are **Speed12** and
and represent the average speed values in each direction of a road segment. We populate these two fields periodically from the .xml files (Figure 9-4) containing archived traffic information from District 5. To facilitate this process we use a semi-automatically generated configuration table that contains the associations between the road segments from the TIGER data files (that define our network topology) and the predefined road segments used by District 5 for reporting the traffic conditions. More exactly, each predefined District 5 road segment spans over a set of road segments from the TIGER data set. This association is illustrated in Figure 9-6. The **D5Link** column stores the IDs of all the District 5 links (i.e., the id field from the .xml file). The **TigerLinks** column stores the IDs of the road segments (i.e., the **TLID** field) from the TIGER data set that are associated with each District 5 road segment. Therefore, the translation of the data from District 5 format into the CDF (used by our STEMS algorithms) can be done automatically on-the-fly based on this predefined configuration table.

**Figure 9-6. Configuration table describing the association between District 5 and TIGER road segments**

This process is illustrated in Figure 9-7. It shows the XML Parser extracting from the .xml file the **id** and the **averageTravelSpeed** for each District 5 link. Then based on the pre-defined
configuration table the matching process generates pairs of TIGER links and their corresponding average speed. Each pair is generated in the form of an SQL update statement and associated with the timestamp of the .xml file.

Recall that real-time traffic information is generated in the form of an .xml file every minute. Therefore, all SQL statements generated from one .xml file have the same timestamp. Each SQL update statement is fed into the sensor emulation module that will execute it based on its timestamp. As a result the Speed12 and Speed21 fields for those TIGER links that are associated with a District 5 link, will be updated accordingly (Figure 9-8).

**Figure 9-7. Data conversion module**

![Data Conversion Module Diagram]

Recall that real-time traffic information is generated in the form of an .xml file every minute. Therefore, all SQL statements generated from one .xml file have the same timestamp. Each SQL update statement is fed into the sensor emulation module that will execute it based on its timestamp. As a result the Speed12 and Speed21 fields for those TIGER links that are associated with a District 5 link, will be updated accordingly (Figure 9-8).
Another challenge in proving STEMS with real-time traffic information was related to the limited coverage of the existing traffic data collection systems. Real-time traffic information in the form of travel times and average speeds is currently provided for most of the highways but only for a limited number of arterial roadways. Therefore, our system was designed to use a set of default values for the locations where there is no traffic data collected and for the situations when real-time traffic information failed to be generated by the software deployed by each District at their regional TMCs. Current default speeds are set to the speed limits, however, more accurate default values, specific to the time of the day and/or the day of the week, can be used if available.
9.2. **STEMS Implementation**

The STEMS prototype was developed entirely in JAVA. The *Graphical User Interface* (GUI) allows the operator to load a map, add incidents and generate evacuation plans using the proposed evacuation routes construction approaches. Figure 9-9 shows the GUI and the evacuation plan generated with the *Fastest-Links* algorithm for a given incident.

![Figure 9-9. STEMS GUI: Fastest-Links based evacuation plan](image-url)
The user can add an incident by clicking on the map on a desired location (which will automatically populate the X and Y fields in the GUI) and then specifying the radius of the incident (in feet). When multiple incidents are simulated the user is also allowed to specify the occurrence time of each incident (relative to the occurrence time of the previous incident). The evacuation plan for two overlapping incidents is illustrated in Figure 9-10.

![STEMS GUI: Evacuation plans for multiple incidents](image)

**Figure 9-10. STEMS GUI: Evacuation plans for multiple incidents**
The user can choose any of the proposed evacuation routes construction algorithms to be used for generating an evacuation plan. The evacuation plan generation process is illustrated in Figure 9-11. The *Links* table stores the traffic network topology and the traffic conditions described in Section 9.1. The user can visualize the network topology stored in the *Links* table (or a subset of it) through the GUI, by pressing the “Load Map” button. The GUI also provides the user with the ability to zoom in and out on the map to reach the desired level of detail. This is illustrated in Figure 9-12. Once the user adds an incident and selects an evacuation routes construction approach, the evacuation plan generation process is triggered by pressing the “Generate Plan” button.

![Figure 9-11. Evacuation plan generation process](image)

As shown in Figure 9-11, the first step in the evacuation plan generation process is the identification of the evacuation area (presented in Section 3.3). Recall that this step identifies the sub-graph of the street network graph, affected by the incident and stores its leaf nodes, non-leaf nodes and links in three temporary tables: *EEPNodes, EZNodes* and *EZLinks*, respectively. These are used as input by the evacuation plan construction algorithm to generate *evacuation routes*, which are then captured in the *Links* table (as explained in Section 4.1) and displayed to the user.
through the GUI. The evacuation plan generation process, shown in Figure 9-11, is repeated periodically as explained in Section 8.3.

![STEMS GUI: The zoom-in option](image)

**Figure 9-12. STEMS GUI: The zoom-in option**

9.3. **Testing STEMS Performance**

In order to test the performance of our STEMS algorithms we have proposed the development of the simulation environment from Figure 9-13. It shows that the only manual input (e.g., from the TMC operator) into STEMS is the incident information (i.e., incident location and effect radius). Based on this information STEMS generates an evacuation plan that is then translated into a simulation file, which is used as input by the traffic simulator. TSIS then executes the simulation and generates an output file containing various measures of effectiveness. From this file, “real-time” traffic information is extracted, then transformed in CDF format, and used to update the
database that feeds STEMS with the real-time information. Consequently, STEMS will generate a new evacuation plan according to the new traffic conditions as captured by the simulator. This process, in which STEMS and the traffic simulator drive each other automatically, is repeated periodically until the evacuation is completed.

![Diagram](image)

**Figure 9-13. Simulation environment for testing the STEMS algorithms**

The detailed diagram showing the components of the STEMS simulation environment and their interaction is presented in Figure 9-14. The process in which STEMS and the TSIS simulator drive each other automatically is represented by the red numbered arrows. The numbers represent the order in which the steps of this processed are performed. In the first step the evacuation plan generation algorithm extracts the traffic network information from the STEMS database and generates an evacuation plan based on the incident information (entered through the GUI). Then the generated evacuation routes are stored in the database (step 2). In step 3 the *DB to TSIS converter* extracts the evacuation plan information from the database and generates a simulation file *FileX.trf* (step 4).
As shown in Figure 9-14 the DB to TSIS Converter also takes as input the simulation file generated in the previous iteration (i.e., File X-1.trf). Basically, in step 4 the converter augments the previously generated simulation file based on the new evacuation plan, generating a new simulation file. Note that for the first iteration, the file File0.trf is used as input. This file is generated from the Links table before the evacuation plan generation process is initiated. Specifically, the Trf Generator module constructs an initial simulation file tFile.trf which
contains the road network topology. Then the *Initialization* module adds turn movement records generating the *initFile.trf*. Pre-timed or actuated traffic lights are then added at all intersection by the *Add Pre-timed Control* module, or *Add Actuated Control* module respectively. The output of this process is the *File0.trf* simulation file.

Once the simulation file *FileX.trf* is generated, it is fed into TSIS (step 5) which executes the simulation and produces an output file (*FileX.out*) containing various measures of effectiveness (step 6). The format of that section of the output file which contains road network statistics is shown in Figure 9-15. It shows the road segments, represented by the IDs of their end nodes, in the first column, and their corresponding average speeds in the last column.

![Figure 9-15. TSIS output file format](image)

From the simulation output file (*FileX.out*), the *Speed Extractor* module (i.e., the *Driver* in Figure 9-13) pulls out the average speed values for all road segments (step 7) and updates the
traffic conditions in the database, accordingly (step 8). At this point one simulation cycle is completed and step 1, the evacuation plan generation, is executed again based on the new traffic conditions. During each simulation cycle, the evacuation routes are also displayed to the user through the STEMS GUI. The user can also visualize the simulation through TRAFED, a simulation visualization tool provided by TSIS.

In what follows we will explain how the DB to TSIS converter generates a simulation file given an evacuation plan. For that purpose, we will briefly describe the format and content of a simulation file. A Corsim simulation file contains the input data used to define the road network and to drive the simulation. This data is represented by 84 record types. The two most important record types for our evacuation problem are record type 11 and record type 21.

Record type 11, mainly describes the geometry of a link, in terms of the turn movements allowed from the downstream node of that link. Four turn movements can be specified: left-turn, through, right-turn and diagonal-turn. For each link in our Links table (storing the road network topology), we will have two records type 11, one record for each direction. The allowed turn movements for a directed link are determined by the neighboring nodes of the downstream node of that link. For example, for the directed link (J, I) from Figure 9-16, three turn movements are allowed: left-turn (towards node L), through (towards node T) and right-turn (towards node R).
Record type 21, is used to specify the relative turn volumes, in other words, the percentage of traffic following each of the allowed turn movements specified by record type 11. This is the most important record type, as it will allow us to direct traffic according to an evacuation plan. At each intersection, in order to force traffic in the evacuation direction, we will specify non-zero volumes for those turn movements that result in traffic flowing in the evacuation direction and zero volumes for all the other turn movements. For example, for the directed link \((J, I)\) from Figure 9-16, although there are three possible turn movements (left-turn, through and right-turn) only two of them (through and right-turn) will cause traffic to flow in the evacuation direction (indicated by the red arrows). Therefore, to enforce traffic to flow in the evacuation direction, the percentage of left-turning traffic will be set to 0 and the percentage of through traffic and right-turning traffic will both be set to 50%. In other words 50% of traffic from link \((J,I)\) will turn right towards node R and the other 50% will go through towards node T, thus following the
evacuation directions. In conclusion, the translation from an evacuation plan to a simulation file consists of generating relative turn volumes (records type 21) specific to that evacuation plan.

Note that relative turn volumes can be specified either as percentages or as unitless numbers that have the desired ratios across all movements. For example, for the directed link \((J, I)\) from Figure 9-16, we can specify the left-turning, through and right-turning volumes as 0, 1 and 1 respectively. Then the percentage of through and right-turning traffic will be computed as: 
\[
\frac{1}{(0+1+1)} = 50\%
\]
and the percentage of left-turning traffic will be computed as: 
\[
\frac{0}{(0+1+1)} = 0\%.
\]

The pseudo-code for computing the relative turn volumes given an evacuation plan is shown in Figure 9-17. This algorithm is based on our rules, proposed in Section 4.1, for controlling the evacuation signals. Recall that traffic entering each intersection is guided to pick only those outgoing links that would cause it to flow in the evacuation direction. In the pseudo-code from Figure 9-17 this restriction is represented in Step 3. If this choice is not available, traffic is guided to pick one of the links with no evacuation direction (such that not to flow opposite to an evacuation direction). This restriction is represented by Step 4 in Figure 9-17. If such a choice is not available either, then traffic is allowed to pick any of the outgoing links (Step 5 in Figure 9-17) and will be guided in the evacuation direction when it reaches the next intersection. As the relative turning volumes depend on the evacuation plans, they need to be regenerated during each simulation cycle (illustrated with red arrows in Figure 9-14). As the other record types remain unchanged, the \textit{DB to TSIS Converter} will take the simulation file generated in the previous iteration (i.e., \textit{File X-1.trf}) and augment it with the new turn volumes corresponding to the current evacuation plan. The resulting simulation file is then fed into Corsim for execution.
For each direction (from NodeID1 to NodeID2) of each link from the Links table, do:

1. Set the relative turn volumes to 0 for all turn movements
   \[ \text{LeftVolume} = \text{ThroughVolume} = \text{RightVolume} = \text{DiagonalVolume} = 0 \]

2. Identify the downstream nodes receiving left, through, right and diagonal-turning traffic
   \[ \text{LeftID} = \text{NodeID} \text{ of the downstream node receiving left-turning traffic from NodeID2} \]
   \[ \text{ThroughID} = \text{NodeID} \text{ of the downstream node receiving through traffic from NodeID2} \]
   \[ \text{RightID} = \text{NodeID} \text{ of the downstream node receiving right-turning traffic from NodeID2} \]
   \[ \text{DiagonalID} = \text{NodeID} \text{ of the downstream node receiving diagonal-turning traffic from NodeID2} \]

3. Set the traffic volume to 1 for those turn movements that would result in traffic flowing in the evacuation direction
   \[ \text{If LeftID} \neq \text{Null and EvacuationDirection for link (NodeID2, LeftID)) is from NodeID2 to LeftID} \]
   Then \[ \text{LeftVolume} = 1 \]
   \[ \text{If ThroughID} \neq \text{Null and EvacuationDirection for link (NodeID2, ThroughID)) is from NodeID2 to ThroughID} \]
   Then \[ \text{ThroughVolume} = 1 \]
   \[ \text{If RightID} \neq \text{Null and EvacuationDirection for link (NodeID2, RightID)) is from NodeID2 to RightID} \]
   Then \[ \text{RightVolume} = 1 \]
   \[ \text{If DiagonalID} \neq \text{Null and EvacuationDirection for link (NodeID2, DiagonalID)) is from NodeID2 to DiagonalID} \]
   Then \[ \text{DiagonalVolume} = 1 \]

4. If no turn movements satisfy the requirement from step 3, then set the traffic volume to 1 for those turn movements that would cause traffic to flow on links without an evacuation direction
   \[ \text{If LeftVolume} + \text{ThroughVolume} + \text{RightVolume} + \text{DiagonalVolume} = 0 \]
   \[ \text{If LeftID} \neq \text{Null and link (NodeID2, LeftID) has no EvacuationDirection Then LeftVolume} = 1 \]
   \[ \text{If ThroughID} \neq \text{Null and (NodeID2, ThroughID) has no EvacuationDirection Then ThroughVolume} = 1 \]
   \[ \text{If RightID} \neq \text{Null and (NodeID2, RightID) has no EvacuationDirection Then RightVolume} = 1 \]
   \[ \text{If DiagonalID} \neq \text{Null and (NodeID2, DiagonalID) has no EvacuationDirection Then DiagonalVolume} = 1 \]

5. If no turn movements satisfy the requirements from step 3 or 4, then set the volume to 1 for all allowed turn movements
   \[ \text{If LeftVolume} + \text{ThroughVolume} + \text{RightVolume} + \text{DiagonalVolume} = 0 \]
   \[ \text{If LeftID} \neq \text{Null Then LeftVolume} = 1 \]
   \[ \text{If ThroughID} \neq \text{Null Then ThroughVolume} = 1 \]
   \[ \text{If RightID} \neq \text{Null Then RightVolume} = 1 \]
   \[ \text{If DiagonalID} \neq \text{Null Then DiagonalVolume} = 1 \]

6. Set relative turn volumes for link (NodeID1, NodeID2) to LeftVolume, ThroughVolume, RightVolume, DiagonalVolume.

End for

Figure 9-17. Computing the relative turning volumes

114
9.4. Proposed Environment for Field Testing of the STEMS Prototype

As mentioned previously, one challenge of STEMS, when deployed in a real traffic environment, is how to inform traffic participants to follow the evacuation plans. While the use of evacuation signals allows STEMS to accurately inform the drivers about the evacuation routes, it has the disadvantage of requiring new device infrastructure with additional costs. An alternative method has been proposed by Aved et al. [41] that achieves a compromise between accuracy and cost. Based on this approach STEMS will not directly inform the traffic participants about the evacuation routes through the evacuation signals, but instead it will inform the police officers dispatched within the incident area, through their PDA or other hand held devices. In turn police officers will direct traffic according to the evacuation plans received through their PDAs. Although this approach requires no additional field devices to be installed, it has the disadvantage that the accuracy of informing the traffic about the evacuation plan is diminished by the fact that we cannot assume to have an police officer at each intersection within the evacuation zone. Strategies for allocating the available police officers to the intersections within the incident area are of critical importance for the evacuation performance and are available in [41].
10. CONCLUSIONS

Emergency evacuation is a critical component of life safety. Current approaches for addressing evacuation are based on proactive planning. This involves developing in advance different plans for different scenarios and then choosing among the available plans the most suitable one to be used whenever an incident occurs. This approach, “if scenario X then follow plan X”, cannot predict real world evacuation scenarios very well and therefore, there is a strong need for real-time traffic management under emergency evacuation.

Our proposed STEMS comprises of a set of intelligent algorithms that can automatically manage a real-time evacuation operation. STEMS assures the following functionality:

1. it can dynamically generate evacuation plans given an incident location within the traffic network and effect radius.
2. it can guide traffic participants out of the disaster areas based on the generated evacuation plans.
3. it can handle multiple incidents occurring in close proximity in a short time frame, by regenerating the evacuation routes such that to direct traffic away from all incidents.
4. it can capture the dynamic aspect of the traffic environment by periodically updating the evacuation plans during their execution to keep them consistent with the continuously changing traffic conditions, as captured by various sensors or surveillance technologies.
The relative performance of the proposed evacuation routes construction algorithms is highly dependent on the traffic network topology. The exit-point based approaches are less sensitive to the network topology as they construct the evacuation routes starting from the EEPs. On the other hand, the incident-based approaches (except the *Fastest-Links* approach) only guarantee that traffic is directed away from the incident, but necessarily towards the EEPs. Therefore, when applied in irregular networks their performance declines, the performance decrease being dependent of the level of irregularity of the network. However, when applied in more regular, grid-like network topologies the performance of the incident-based approach can be as good as that of the exit-point based approaches.

The performance of the approaches that are sensitive to traffic information is also influenced by the type of the environment that they are applied in. In an *active* environment (when information about the traffic conditions is available) these approaches always perform better than the approaches that are insensitive to traffic information. However, when applied in passive environments (when traffic information is not available and all streets are assumed to have an average speed equal to the speed limit) these approaches might perform worse than the ones that are insensitive to traffic information, as they might concentrate traffic on slower, more congested links. Our simulation studies indicate that the *Time-based* approach is the least sensitive to the type of the environment it is applied in and therefore, has the best performance among all evacuation routes construction approaches.
11. FUTURE RESEARCH DIRECTIONS

There are two critical problems facing the automobile transportation system today: first, an awful number of people die or are injured daily due to vehicular crashes; secondly, increasing travel demand causes more severe traffic congestion and delays. Despite a slow growth in jobs and travel in 2003, congestion caused 3.7 billion hours of travel delay and 2.3 billion gallons of wasted fuel, an increase of 79 million hours and 69 million gallons from 2002 to a total cost of more than $63 billion [42]. The traffic accidents and congestion issues have spurred a great deal of research interest in transportation in recent years. Consequently, innovative technologies known as Intelligent Transportation Systems (ITS) have emerged to reduce the surface transportation fatalities, injuries, and delays by effectively integrating hard and soft information systems technologies into the transportation system infrastructure and in vehicles themselves.

During my research I have investigated and proposed information technologies for developing both intelligent infrastructure and intelligent vehicle systems. These technologies were mainly targeted towards providing real-time traffic information to reduce congestion and expand personal travel choices. In addition, I have also investigated expanding the scope and reach of ITS technology to applications supporting homeland security and emergency management.

In the near future I would like to further investigate intelligent vehicle applications. Current research on embedding computer technologies into vehicles can be divided into two categories: autonomous vehicles and intelligent vehicles. Although the DARPA Grand Challenge in 2004
has proved that *autonomous vehicles* is a promising research area, it is not for another couple of decades that this technology will reach the deployment stage. In contrast to *autonomous vehicles* technologies which completely eliminate the driver and attempt to “guess” the environment (i.e., road infrastructure and the other vehicles), *intelligent vehicles* focus on assisting the driver by collaborating with the infrastructure and with the other vehicles in the environment. As I am more drawn to practical applications, I would like to focus my research on *intelligent vehicles* technologies, as they are most likely to be deployed in the very near future.

As part of my research on *intelligent vehicles*, I have so far investigated wireless communication protocols and my collaboration with two other PhD students resulted in the development of a *Connectionless Approach to Mobile Communication* (CAM). Instead of using the traditional fixed-path routing, this communication protocol adopts a less restrictive approach by allowing any intermediate vehicles, along the general direction toward the destination vehicle, to relay the data packets without having to establish a hop-by-hop connection. The performance of this solution is essentially unaffected by the high mobility of the vehicles. One potential application of this communication protocol, that would reduce transportation fatalities and injuries, is a system that would assist drivers in merging with the existing traffic when entering a highway. This would involve cooperation between the vehicles entering the highway at a particular ramp and the vehicles currently approaching that ramp. This collaboration would result in adjusting the speeds of all vehicles involved, to avoid crashes and facilitate merging of traffic.
Another significant source of accidents is sudden braking, usually in response to an unpredictable incident (e.g., a previous accident). In a collaborative environment, the effect of sudden braking could be alleviated, if for example, the stopping vehicle would be able to communicate with the vehicles upstream, to inform them that it has just suddenly braked. This information would allow the upstream vehicles to break before their drivers notice the sudden slow down and react accordingly.

As part as my future research I would also like to continue to explore ITS applications that support emergency management systems. Current microscopic traffic simulators are based on driver behavior models that assume drivers behave in a safe manner, obeying traffic rules without making mistakes or misjudgments. However, this might be inadequate in emergency situations when panic and fear might push drivers to lose their self-control and behave in unpredictable ways. Also, the inherent agitation in a crisis situation might distract drivers, leading to crash occurrences and increased tension. Therefore, I would like to investigate driver behavior models that incorporate the panic and tension inherent in crisis situations. Traffic simulation models that incorporate “panicked” driver behaviors would help in evaluating strategies for incident and emergency management systems.
APPENDIX A: LIST OF PUBLICATIONS
Journal Papers


Conference Papers


1. **DB to TSIS Converter** (Figure 9-14)

```java
import java.io.*;

public class DBtoTSISConverter {
    public static void updateTrfFile(int index, int time) {
        updateTrfFile(index, time, null, -1);
    }

    public static void updateTrfFile(int index, int time, int[] controlNodes, int last) {
        // index = the index of the incident (index=1 for the first incident, index=2 for the second)
        // time = the time (in seconds) between the current incident and the previous incident
        String[] entries = new String[1000];
        int noEntries = 0;
        try {
            PrintWriter pw = new PrintWriter(new FileWriter("File"+index+".trf", false), true);
            BufferedReader br = new BufferedReader(new FileReader("File"+(index-1)+".trf"));

            pw.println(br.readLine()); pw.println(br.readLine());
            pw.println(br.readLine()); pw.println(br.readLine());
            //------------------------------------------------------------------
            // update record type 3, i.e. time periods
            //------------------------------------------------------------------
            String s = br.readLine();
            String temp = s.substring(0,4*index-4) + temp + s.substring(4*index-4,4*index)
            +s.substring(4*(index+1));

            //------------------------------------------------------------------
            // print to file the part of the trf file that remains unchanged (time period 1= no incident)
            //------------------------------------------------------------------
            while (!(s = br.readLine()).substring(76,80).equals(" 210")) {
                if (s.substring(76,80).equals(" 50")) entries[noEntries++]=s;
                pw.println(s);
            }

            //------------------------------------------------------------------
            // print to file the next existing "index" time periods
            //------------------------------------------------------------------
            for (int i=0;i<index-1;i++) {
                pw.println(s);
                while (!(s = br.readLine()).substring(76,80).equals(" 210")) pw.println(s);
            }
            // print the modified record type 210
            pw.println("   0   3                                                                     210");
            br.close();
        }
    }
}
```
for each link in the Link table create a new record type 21

```java
Graph sn = new Graph("getAllLinksFullInfo");
for (int i=0;i<sn.noLinks;i++) {
    if (sn.Links[i].evac%10==1) {
        updateTRFfile(sn.Links[i].node1, sn.Links[i].node2, index,pw, sn, sn.Links[i].flow);
        updateTRFfile(sn.Links[i].node2, sn.Links[i].node1, index, pw, sn, 0);
    } else if (sn.Links[i].evac%10==2) {
        updateTRFfile(sn.Links[i].node1, sn.Links[i].node2, index, pw, sn, 0);
        updateTRFfile(sn.Links[i].node2, sn.Links[i].node1, index, pw, sn, sn.Links[i].flow);
    } else {
        updateTRFfile(sn.Links[i].node1, sn.Links[i].node2, index, pw, sn, 0);
        updateTRFfile(sn.Links[i].node2, sn.Links[i].node1, index, pw, sn, 0);
    }
}
```

set the entry volumes to 0

```java
for (int i=0;i<noEntries;i++)
    pw.println(entries[i].substring(0,8) + "   0" + entries[i].substring(12));
```

print to file the remaining of the trf file

```java
pw.println("   0                                                                         170");
pw.println("   1   0   0                                                                 210");
pw.close();
} catch (Exception e) {System.out.println(e);} 
```

private static void updateTRFfile(int n1, int n2, int index, PrintWriter pw, Graph sn, int flow) {
```java
try {
    String node1 = s(n1, 4);
    String node2 = s(n2, 4);

    BufferedReader br = new BufferedReader(new FileReader("File0.trf");
    int left=0,right=0,thru=0,diag=0; //these are the neighbors of node2
    String s="";
    while (! (s=br.readLine()).substring(0,8).equals(node1+node2) ) {}  
    String sleft = s.substring(36,40), sthru = s.substring(40,44);
    String sright = s.substring(44,48), sdiag = s.substring(48,52);
    try { left=Integer.parseInt(sleft.trim());} catch (Exception ex){}
    try { thru=Integer.parseInt(sthru.trim());} catch (Exception ex){}
    try { right=Integer.parseInt(sright.trim());} catch (Exception ex){
    try { diag=Integer.parseInt(sdiag.trim());} catch (Exception ex){
    if (diag<0) {
```

```
```java
String speedleft=" 33", speedthru=" 33", speedright=" 33", speeddiag=" 33";

// Step 2: Check the Links for (Node 2, left/thru/right/diag) direction 1
// or (left/thru/right/diag, Node2) direction 2

String dleft="   0", dthru="   0", dright="   0", ddiag="   0";
String tleft="   0", tthru="   0", tright="   0", tdiag="   0";

for (int i=0;i<sn.noLinks;i++) {
    if (sn.Links[i].node1==n2) {
        if ((sn.Links[i].evac%10==1)) {
            if (sn.Links[i].node2==left) dleft= speedleft;
            if (sn.Links[i].node2==thru) dthru= speedthru;
            if (sn.Links[i].node2==right) dright= speedright;
            if (sn.Links[i].node2==diag) ddiag= speeddiag;
        } else if ((sn.Links[i].node2==n2) {
            if ((sn.Links[i].evac%10==2)) {
                if (sn.Links[i].node1==left) dleft= speedleft;
                if (sn.Links[i].node1==thru) dthru= speedthru;
                if (sn.Links[i].node1==right) dright= speedright;
                if (sn.Links[i].node1==diag) ddiag= speeddiag;
            }
        }
    } else if (sn.Links[i].node2==n2) {
        if ((sn.Links[i].evac%10==1)) {
            if (sn.Links[i].node2==left) dleft= speedleft;
            if (sn.Links[i].node2==thru) dthru= speedthru;
            if (sn.Links[i].node2==right) dright= speedright;
            if (sn.Links[i].node2==diag) ddiag= speeddiag;
        } else if ((sn.Links[i].evac%10==2)) {
            if (sn.Links[i].node2==left) dleft= speedleft;
            if (sn.Links[i].node2==thru) dthru= speedthru;
            if (sn.Links[i].node2==right) dright= speedright;
            if (sn.Links[i].node2==diag) ddiag= speeddiag;
        }
    }
}

if ((dleft+dthru+dright+ddiag).equals("   0   0   0   0")) {
    // there are no outgoing links flowing into the evacuation direction
    dleft = tleft; dthru = tthru;
    dright = tright; ddiag = tdiag;
}

if ((dleft+dthru+dright+ddiag).equals("   0   0   0   0")) {
    // when all links are incoming
    if (left!=0) dleft= speedleft;
    if (thru!=0) dthru= speedthru;
    if (right!=0) dright= speedright;
    if (diag!=0) ddiag= speeddiag;
}
```
// Step 3: Optional - Introduce different % of drivers disobeying the evacuation direction

int found = 0;
for (int i = 0; i < sn.noNodes; i++)
    if ((sn.Nodes[i] == n1) || (sn.Nodes[i] == n2)) found++;
if (found == 2) {
    int ln = Integer.parseInt(dleft.trim());
    int tn = Integer.parseInt(dthru.trim());
    int rn = Integer.parseInt(dright.trim());
    int dn = Integer.parseInt(ddiag.trim());
    double nmisbehaved = percentage * (ln + tn + rn + dn);

    int v = 0; // v is the number of directions I can introduce misbehavior on
    if ((left != 0) && (ln == 0)) v++;
    if ((thru != 0) && (tn == 0)) v++;
    if ((right != 0) && (rn == 0)) v++;
    if ((diag != 0) && (dn == 0)) v++;

    if (v != 0) {
        int imisbehaved = (int) Math.round(nmisbehaved / v);
        if ((left != 0) && (ln == 0)) ln = imisbehaved;
        if ((thru != 0) && (tn == 0)) tn = imisbehaved;
        if ((right != 0) && (rn == 0)) rn = imisbehaved;
        if ((diag != 0) && (dn == 0)) dn = imisbehaved;
        dleft = s(ln, 4);
        dthru = s(tn, 4);
        driight = s(rn, 4);
        ddiag = s(dn, 4);
    }
}

// add new record type 21
pw.println(node1 + node2 + dleft + dthru + driight + ddiag + " 21");
} catch (Exception e) {System.out.println(e);}

// translates an integer n into a string of length l
private String s(int n, int l) {
    String str = String.valueOf(n);
    while (str.length() < l) str = " +" + str;
    return str;
}
} // end of class DBtoTSISConverter
public class DataConverter extends Thread {
    public DataConverter(String D5xmlFile) {
        try {
            parseXMLFile(D5xmlFile);
        } catch (Exception e) { System.out.println(e); }
    }

    public void parseXMLFile(String file) {
        try {
            int avgSpeed = 0;
            String timestamp = "", linked = "";

            DocumentBuilderFactory docBuilderFactory = DocumentBuilderFactory.newInstance();
            DocumentBuilder docBuilder = docBuilderFactory.newDocumentBuilder();
            Document doc = docBuilder.parse(new File(file));
            doc.getDocumentElement().normalize();

            NodeList listOfLinks = doc.getElementsByTagName("vehicleProbe");
            for (int s = 0; s < listOfLinks.getLength(); s++) {
                org.w3c.dom.Node linkNode = listOfLinks.item(s);
                if (linkNode.getNodeType() == org.w3c.dom.Node.ELEMENT_NODE) {
                    Element linkElement = (Element) linkNode;
                    linkID = linkElement.getAttribute("id");

                    NodeList avgSpeedList = linkElement.getElementsByTagName("averageTravelSpeed");
                    Element avgSpeedElement = (Element) avgSpeedList.item(0);
                    NodeList textASList = avgSpeedElement.getChildNodes();
                    avgSpeed = (int) Double.parseDouble(((org.w3c.dom.Node) textASList.item(0)).getNodeValue().trim());

                    NodeList timestampList = linkElement.getElementsByTagName("timestamp");
                    Element timestampElement = (Element) timestampList.item(0);
                    NodeList textTSList = timestampElement.getChildNodes();
                    timestamp = ((org.w3c.dom.Node) textTSList.item(0)).getNodeValue().trim();

                    // System.out.println(linkID + " " + avgSpeed + " " + timestamp);
                    generateUpdate(linkID, avgSpeed);
                }
            }
        } catch (Exception e) { System.out.println(e); }
    }
}
public void generateUpdate(String D5Link, int speed) {
    try {
        Connection con = DBconnection.create();
        Statement stm = con.createStatement();
        ResultSet rs = stm.executeQuery("SELECT Query1, Query2 FROM config WHERE D5Link = "+D5Link+");
        String query1 = "UPDATE Links SET Speed12 = " + speed + " " ;
        String query2 = "UPDATE Links SET Speed21 = " + speed + " " ;
        if (rs.next() != false) {
            query1 = query1 + rs.getString("Query1");
            query2 = query2 + rs.getString("Query2");
            stm.executeUpdate(query1);
            stm.executeUpdate(query2);
        }
    } catch (Exception e) {System.out.println(e);}
}

3. **Speed Extractor Module (Figure 9-14)**

    public class SpeedExtractor {
        public static void Extract(int trf) {
            Graph StreetNetwork = new Graph("getAllLinks");
            try {
                BufferedReader br = new BufferedReader(new FileReader("File"+(trf)+"out");
                String s = " " ;
                int n1, n2, speed = 0;
                String is = ""+(trf+1);
                while (is.length() < 2) is = " "+is;
                while(true) {
                    while (! (s=br.readLine()).equals("1 TIME PERIOD "+is+ " SPECIFIC NETSIM STATISTICS")) {}
                    for (int i=0;i<9;i++) br.readLine();
                    for (int j=0;j<45;j++) {
                        s = br.readLine();
                        n1 = Integer.parseInt(s.substring(2,6).trim());
                        n2 = Integer.parseInt(s.substring(7,11).trim());
                        try {speed = (int) Float.parseFloat(s.substring(124).trim());}
                            catch (Exception ex) {speed = 0;}
                        int k = 0 ;
                        for (k=0;k<StreetNetwork.noLinks;k++)
                            if (((StreetNetwork.Links[k].node1==n1)&&(StreetNetwork.Links[k].node2==n2))
                                {StreetNetwork.Links[k].speed12 = speed;break;}
                            else if (((StreetNetwork.Links[k].node1==n2)&&(StreetNetwork.Links[k].node2==n1))
                                {StreetNetwork.Links[k].speed21 = speed;break;}
                    }
                }
            } catch (Exception ex) {/*System.out.println(ex);*/}
            update = StreetNetwork.updateSpeedsDB();
        }
    } //end of class SpeedExtractor
LIST OF REFERENCES


