UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING AND THE ENVIRONMENT

Modelling Traffic Incidents to Support Dynamic Bus Fleet Management for Sustainable Transport

by

Polyvios Polyviou

Thesis for the degree of Doctor of Philosophy

June 2011

Abstract

The continuous implementation of highly technological functions and specifically intelligent transport systems in public transport highlights the need of highly efficient, accurate and reliable bus operations network. Intelligent transport systems can support a variety of functions, including dynamic bus fleet management which has yet to be established in most bus fleets in the UK in a systematic way. In order to support dynamic bus fleet management by detecting the fundamental role of bus and traffic incidents in bus-based public transport, a microscopic simulation model capable of modelling the impact of the individual incidents' characteristics on bus operations has been developed and applied to a variety of scenarios.

This research draws on a review of existing literature on bus fleet management and available computer software in this field. It investigates research gaps in modelling the impact of traffic incidents on overall bus performance; it describes the design and development of the new simulation model, SIBUFEM (Simulating Incidents for Bus Fleet Management) for modelling bus operations during whole day periods in which incidents of different types can occur. The model simulates a high frequency bus service using existing field data and incorporates the continuous circulation of buses along the bus route. It uses journey time profiles, passenger-dependent bus stop dwell times and deterministic time-dependent queuing theory to model traffic incidents and the impact of their characteristics on the bus performance parameters.

The model results, presented in this thesis, focus on performance measures including but not limited to bus journey times, passenger waiting times and bus delays resulting from various bus and traffic incidents. Incidents vary from bus breakdowns, to traffic incidents such as road-works, traffic accidents, burst water mains, disabled vehicles and illegal parking; in SIBUFEM they are specified in terms of their location, duration and severity (i.e. loss of capacity). The model has been applied to a main bus corridor in Southampton, UK, with a base case of 'normal' operations established, for comparison with results from 24 different incident scenarios, and using key model performance parameters of average bus journey time, bus speed and excess waiting time.

This PhD demonstrates the functionality of SIBUFEM with model results demonstrating the extent to which passenger waiting times increase with increasing incident severity and duration. The overall comparison of the simulation results showed that the more severe the level of severity or the longer the duration of an incident, the higher the expected impact of the event on the overall bus performance was. In terms of the incident location parameter, the effect is greater when the incident occurs in the middle of the bus route than when it occurs at the end. The effect of incident location is especially evident in the case of traffic incidents such as roadworks, traffic accidents and illegal parking. Findings from this research also demonstrated that these incidents are usually more severely affected by a change in an incident parameter than by bus breakdown incidents. The thesis concludes with a discussion on potential dynamic bus fleet management strategies and how SIBUFEM can be further developed to allow these strategies to be evaluated.

SIBUFEM is capable of modelling traffic incidents to support dynamic bus fleet management and, thus, encourage the use of intelligent transport systems applications in bus operations. This offers great potential in the field of bus-based public transport as part of a guidance tool for bus operators, as well as the way to increase bus level service thereby increasing customer satisfaction and thus the development of a sustainable transport system.

Table of Contents

CHAPTER 6

CHAPTER 7

List of Figures

List of Tables

DECLARATION OF AUTHORSHIP

I, Polyvios Polyviou, declare that the thesis entitled:

'Modelling Traffic Incidents to Support Dynamic Bus Fleet Management for Sustainable Transport'

and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- **this work was done wholly or mainly while in candidature for a research degree at this University;**
- **where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;**
- **where I have consulted the published work of others, this is always clearly attributed;**
- **where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;**
- **I have acknowledged all main sources of help;**
- **where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;**
- **none of this work has been published before submission.**

Signed: ……………………………………………………………………..

Date:……………………………………………………………………….

Acknowledgments

I would like to thank my supervisors for their patience and support throughout my PhD. Dr Nick Hounsell for providing me with continuous feedback on my research; for being always available for meetings and happy to answer all my reasonable or stupid questions; for helping me see the bigger picture of my work; and for all his contribution during the writing of my thesis. And Dr Birendra Shrestha, for his continuous support and help in my research and especially in computer modelling issues; thank you for being there whenever I was stuck in between thousands lines of source code looking for tiny errors, and for encouraging me to keep thinking positive and give myself credit for my work.

I would also like to say a thank you to all the staff in Transportation Research Group and the School of Civil Engineering and the Environment for their support and help with all issues that came up during my PhD. A big thank you goes to bus operators of ‘FIRST’ in Southampton, ‘London United’ in London and ‘Brighton and Hove Buses’ in Hove, UK, for their time, support, and invaluable feedback.

I need to say a big thank you to my friends for being there when I needed them and for trying not to disturb when I had to isolate myself and try to minimise my social life to devote all my time to work. Thank you also for the company, for the support and for being around to share my happiness and to take my frustration and stress away.

A huge sincere thank you would not be enough for White, who never stopped supporting me, encouraging me and believing in me more than anyone, even more than myself sometimes. Thank you; you kept me going and helped me find the inner strength to overcome all challenges and obstacles throughout this tough journey.

Finally, a special thank you goes to my family. I would not have been able to be at this stage if it was not for the support and encouragement of my parents, Olympia and Andreas. Mum and dad, you provided me with two weapons: always think positive aiming for the best and keep my feet on the ground. I would like to thank my brothers, Evagelos and Stavros, for being there for me and making me feel that they are both so proud of me; thank you for confirming that the bond between siblings is the most powerful connection that may exist. I mostly thank God for giving me all the above.

List of Acronyms

APC	Automatic Passenger Counting
ATC	Area Traffic Control
ATMS	Advanced Traffic Management Systems
AVL	Automatic Vehicle Location
BFM	Bus Fleet Management
BFMC	Bus Fleet Management Centre
BRT	Busway Rapid Transit
DBFM	Dynamic Bus Fleet Management
DfT	Department for Transport
DSS	Decision Support System
ETC	Electronic Toll Collection
EWT	Excess Waiting Time
GIS	Geographic Information System
GPS	Global Positioning System
ITS	Intelligent Transport Systems
O-D	Origin-Destination
OR/MS	Operations Research/Management Science
PhD	Doctor of Philosophy
QSI	Quality of Service Indicators
RAD	Rapid Application Development
SIBUFEM	Simulating Incidents for BUs FlEet Management
TCRP	Transit Cooperative Research Program
TfL	Transport for London
TIM	Traffic Incident Management
TIMS	Traffic and Incident Management System
UK	United Kingdom
US	United States
VCSP	Vehicle and Crew Scheduling Problem
VRSP	Vehicle Routing and Scheduling Problem
VSP	Vehicle Scheduling Problem

Contributions

This PhD Research has contributed to the following:

Polyviou P., 2011, *The 'SIBUFEM' model, a new micro-simulation approach capable of modelling bus and traffic incidents for bus fleet management purposes,* **European Transport Conference (ETC) 2011** (to be presented)**,** Glasgow, Scotland, UK. 10th-12th October 2011.

Polyviou P., 2011, *How can Modelling of Traffic Incidents support Intelligent Transport Systems and Reinforce Bus-Based Public Transport?, Poster,* **4th NEARCTIS Workshop** 'Towards an Integrated European Community in Advanced Road Cooperative Traffic Management', Bron, France. 10th January 2011.

Polyviou P., Hounsell N.B., 2011, *How can Modelling of Traffic Incidents support Intelligent Transport Systems and Reinforce Bus-Based Public Transport?,* **8th Intelligent Transport Systems (ITS) European Congress**, Lyon, France. 6th-10th January 2011.

Polyviou P., Hounsell N.B., Shrestha B.P., 2011, *Modelling Bus and Traffic Incidents for Bus Fleet Management Purposes using Intelligent Transport Systems,* **43rd Annual Universities' Transport Study Group (UTSG) Conference**, Milton Keynes, London, UK. 5rd-7th January 2011.

Polyviou P., Hounsell N.B., Shrestha B.P., *Intelligent Transport Systems and Simulation Modelling of Dynamic Bus Fleet Management,* **The Young European Arena of Research (YEAR) 2010**, Brussels, Belgium. 7th-11th June 2010.

Polyviou P., 2008, *Evaluation of New Bus Fleet Management Options using Intelligent Transport Systems,* **40th Universities' Transport Study Group (UTSG) Conference**, Marriot Hotel, Portsmouth, UK. 3rd-5th January 2008.

Dedicated to my parents…

-Chapter 1-

Introduction

1.1 Background and Motivation

Time, particularly nowadays is considered to be the most precious resource of modern life; Theophrastus (300 B.C.) stated that: "*Time is the most valuable thing a man can spend*" (Diogenes, 250-300 A.D.). In this regard, human activities related to every field of modern life focus on the attempt of spending time wisely. This parameter, equally, is regarded to be a significant determinant of any decision making within the interesting, promising and demanding research field of public transport. Within public transport systems, time is arguably the most crucial parameter, both for operators and passengers. Public transportation planners and operators face increasing pressure to stimulate patronage through the provision of an efficient and friendly service, which is valued, estimated and analysed in terms of time.

The Public Service Agreement of the Department for Transport (DfT) on local public transport is to, by the end of 2010, increase the use of public transport (bus and light rail) by more than 12 per cent in England compared with 2000 levels, with growth in every region. In the seven years leading up to 2007/08, public transport patronage increased by 19 per cent (DfT, 2008), focus now remains on increasing the efficiency of this service.

Buses are the most dominant mode of public transport, representing 64% of total passenger journeys on public transport in England (DfT, 2009). Between 2007/08 and 2008/09 annual bus passenger journeys on local buses in England were estimated to have increased by 1.6% to 4,783 million journeys, according to the National Statistics published by the DfT (DfT, 2009). Considering that the sustainability of the environment is one of the main targets of current UK transport policies, achieving significant improvements in bus operations is a challenging task. In recent years, this direction has increased in desirability due to rapidly increasing problems induced by traffic congestion in towns and cities around the world.

According to a report by the UK Sustainable Development Commission (SDC), 84% of all trips in the UK are journeys of less than 10 miles, which indicates the huge potential for public transport use (UK SDC, 2010). One of the most interesting and promising ideas to support and improve bus operations, is dynamical operation: incorporating static and dynamic information and investigating the optimum way of obtaining and collating accurate information. Static bus scheduling has been widely used and improved all over the world by using static information such as bus schedule information, historical information about recurrent traffic conditions and average dwell time at bus stops (Lin and Bertini, 2004). Recently, dynamic scheduling is being increasingly introduced to support responses to unexpected public transport and traffic incidents and demand changes; dynamic information includes real-time bus location data, delay at stops and current weather and traffic conditions (Lin and Bertini, 2004). Dynamic bus scheduling application consists of two parts: off-line scheduling of vehicles and crews, and online re-scheduling in response to incidents (McDonald *et al.*, 2006). Evaluating and calculating the impact of such bus and traffic incidents on bus operations is of high interest and a great challenge towards the dynamic bus fleet management domain. Dynamic scheduling is key to research in bus operations and is increasingly supported by rapid developments in technologies.

Rapid technological developments in the fields of communication, information, and control provide ways to better achieve current transport policy objectives and to enable the evolution of new policies; Intelligent Transport Systems (ITS) is the collective title given to such technologies (McDonald *et al.*, 2006). ITS can improve road transport, driver support and mobility (Zhicai *et al.,* 2006), traffic conditions and reinforce the efficient use of public transport. The main objective of ITS is to enable people to become “Smart”, to supply them with the opportunity of doing the right thing in the right place at the right time (Lam, 2001). One of the main implemented ITS technologies is Automatic Vehicle Location (AVL) Systems, which is of major significance for any bus operation. Although AVL provides valuable information to bus operators during the decision making process, it seems that the limited guidance that operators have on how to use AVL information for best results, contradicts their attempt to implement dynamic fleet management to improve bus operations (Hounsell, 2004). Therefore, a critical and complete evaluation of the use of ITS in bus systems, with emphasis given on incident management investigating bus and traffic incidents towards suggesting dynamic bus fleet management strategies, is required.

1.2 Research Objectives

The main aim of this research is to explore ways of evaluating the impact of bus and traffic incidents on bus operations, and specifically on high frequency bus services, in order to put forward new bus fleet management options to address these incidents. More specifically, the individual objectives of this research work are:

- ✓ to review and understand the current state of art of bus fleet management and ITS used in this field, with particular reference to AVL systems,
- ✓ to review the existing state of art of commercially available computer software used for modelling of bus and traffic incidents for bus fleet management,
- ✓ to propose and develop a simulation model capable of modelling the details of bus and traffic incidents and their impact on bus operations,
- ✓ to apply the simulation model to a variety of incident scenarios in order to assess the impact of incidents on overall bus performance and hence suggest potential fleet management strategies for improved efficiency, and
- ✓ to investigate and propose any further options for development in this field.

The following section describes the core of the methodology followed to meet the objectives described. The complete description and development of the methodology are presented in Chapter 3, where the research methodology is analyzed with specific focus on the key components of this work.

1.3 Summary of Methodology

The methodology of this research consists of necessary steps required to satisfy the objectives. The first of these steps involves the statement and comprehensive understanding of the issues, which focuses on presenting the research field parameters and illustrating the necessity of this specific research. The statement of the problem is followed by the review of literature in the field of bus and traffic incidents, understanding of the bus fleet management and possible uses of ITS and especially AVL systems. In order for this review to be complete, a further investigation of the commercially available computer software is required. This is of crucial significance in order to decide whether this software is capable of achieving the research targets or there is a need of a new model.

The literature review illustrates the lack of computer software capable of modelling the bus and traffic incidents impact for bus fleet management purposes. Therefore, this PhD thesis aims to cover this research gap and the methodology describes the general specifications, the key requirements and the expectations of the development of a new microscopic simulation model. The new simulation model, called SIBUFEM, is developed and applied to various bus and traffic incidents. Various key parameters of the incidents and their impacts on bus operations are modelled and analyzed. The model is applied to a variety of options, parameters and scenarios, providing useful visualised and computational output. The results obtained from the performance of the model and the effectiveness of this research on incident modelling to support bus fleet management are presented and critically analysed; recommendations for future work are also suggested. The outline of the research process is illustrated in Figure 1.1.

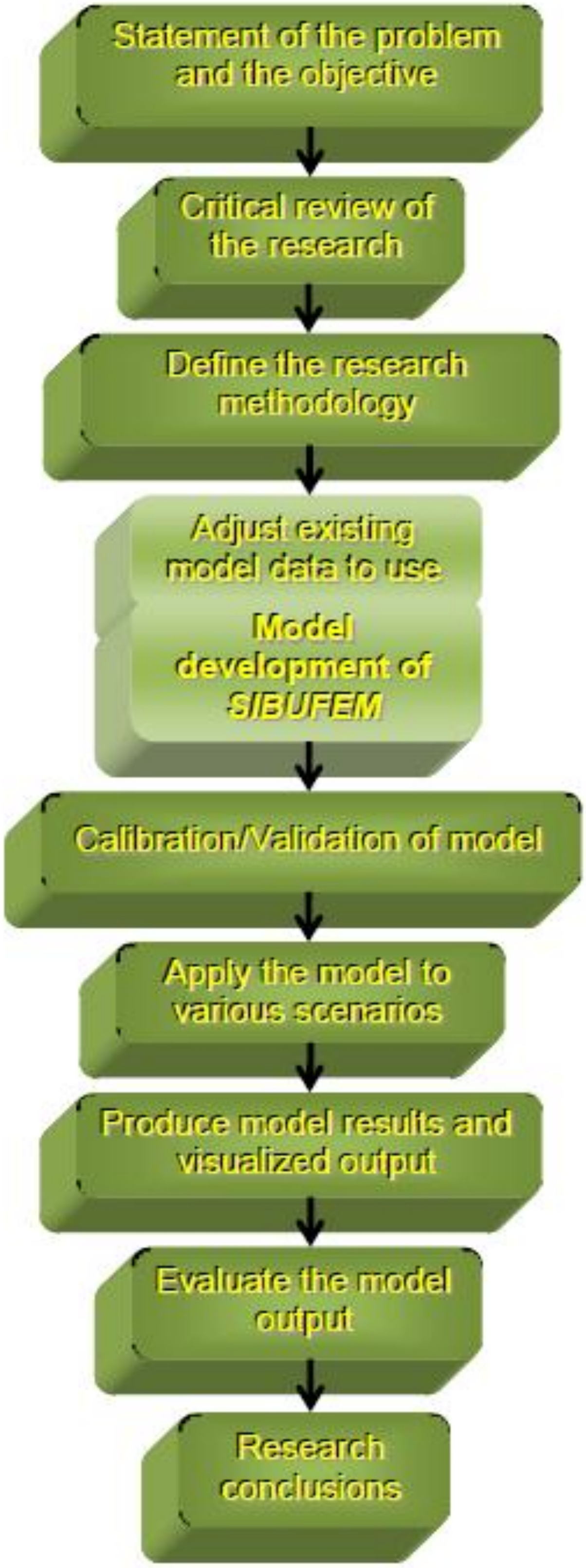


Figure 1.1: Structure of the research

-Chapter 2-

Literature Review

2.1 Introduction

In order to achieve the research targets set in Section 1.2 a complete and critical review of the field is required. The main objective is to investigate the sector of bus and traffic incidents and the use of Intelligent Transport Systems (ITS) in bus operations. This research area is part of the wider field of dynamic bus fleet management and therefore a background review of this is provided. In this Chapter, applications of ITS are reviewed and the contribution of this towards bus operations enhancement is portrayed. A particular focus on one of these technologies, Automatic Vehicle Location systems, is the subject of the next section, which is followed by a review of bus and traffic incidents, the key parameters describing them and the process of categorizing each incident according to their characteristics. Finally the bus fleet management domain and it's dynamic aspect are described, highlighting the research gaps in it's implementation. The last section of this Chapter summarizes key and related findings of this research area.

2.2 Intelligent Transport Systems

2.2.1 Definition and applications

With the rapid increase of road traffic congestion in recent years, a great variety of Intelligent Transport Systems (ITS) has been developed and applied throughout the world (Clowes, 2000), (Polk, 2000). Attention has been given recently to the development of ITS to improve traffic conditions and reinforce the efficient provision of transport. ITS are combinations of technological developments as described by Department for Transport (DfT) in Table 2.1. If the general objectives of ITS were summarized then this would be to enable people to become “Smart”, to allow them to do the right thing in the right place at the right time (Lam, 2001). ITS technologies retain the ability of ensuring a better bus service by ensuring that bus operations are fast, reliable and safe; that buses run on time, their performance are monitored and, in case they are required, schedule adjustments should be made swiftly and accurately. The boarding at stations should be convenient and passengers should be informed of the exact times of bus arrivals. All these objectives can be supported by ITS and bus operations can therefore be significantly enhanced (TCRP, 2003a).

Table 2.1: Intelligent Transport Systems (DfT, 2005).

What are Intelligent Transport Systems?
Combinations of information processing, maps, databases, communications and real time data from a range of sensors, to produce solutions that enable:
■ infrastructure owners and operators to improve the quality, safety and management of transport networks;
■ individual travellers, drivers, transport operators and authorities to make better informed, more ‘intelligent’ journey decisions;
■ network operators and ‘third party’ service providers to supply advanced information services, increasingly on a multi-modal basis, to all types of traveller;
■ road users to drive safer, ‘smarter’ vehicles.

Applications of information technology are now expanding rapidly across all modes of transport, under the heading of ITS. A wide range of applications for bus-based public transport have been developed. This direction can help improve the efficiency of bus operations, making a further step towards providing a real transport alternative to the private car (Hounsell and Graham, 2001). ITS consist of several key systems, such as Advanced Traffic Management Systems (ATMS), Area Traffic Control (ATC), Electronic Toll Collection (ETC) and Automatic Vehicle Location (AVL) (Lam, 2001). Of the available ITS applications there are several that can be identified as possibly noteworthy and comprise a core set of applications that should be pursued. Some of the most promising ones (Lam, 2001) are listed and presented in Table 2.2.

Table 2.2: List of the most promising ITS applications

Intelligent Transport Systems (ITS) applications	
1. Transport Information System (TIS)	16. Traffic Signal Priorities
2. Passenger Information Systems (PIS)	17. Pedestrian Assistance
3. Bus Fleet Management	18. Cyclist Assistance
4. Traffic Control and Surveillance	19. Emergency Notification
5. Automatic Passenger Counting (APC)	20. Freight Management
6. Area Traffic Control (ATC)	21. Driving Assistance
7. Fleet Management (e.g. GPS)	22. Emission Control
8. Public Transport Priority	23. Traffic Signal Equipment
9. Automatic Cruise (e.g. AVL)	24. Ramp Metering
10. Vehicle Guidance and Control	25. Damage Mitigation
11. Electronic Payment (e.g. smart card)	26. Co-ordinated Logistics
12. Variable Message Signing (VMS)	27. Road Status Monitoring
13. Emergency Vehicle Dispatching	28. Road Works Management
14. Data Logger (in-vehicle speed device)	29. Vehicle Permit
15. Traffic Management & Information Centre	30. Licensing

A series of research projects and field trials carried out in both the United States and Europe have provided good insight into the ITS applications over the last decade (McDonald *et al.*, 2006). The majority of bus operation systems have some ITS applications, especially in places where ITS have been most successfully applied to bus operations, such as in Los Angeles, where ITS key systems are part of a geographically larger and functionally comprehensive ITS system (TCRP, 2003a). In public transport, applications of the 'O-Bahn' system in Australia (Adelaide) and in Europe (Essen, Leeds, Ipswich, and Edinburgh) and Busway Rapid Transit (BRT) applications such as in Brisbane, Australia and in Luton, UK are a few examples of ITS implementations around the world (Luke, 2006).

2.2.2 Contribution

ITS achieve the integration of modern telecommunications, control and information processing technology to transport systems. The term 'Transport Telematics', as they are known in Europe, refers to the actual technologies used to deliver such intelligent systems (Firmin, 2006). The efficient management of traffic is assisted by the adoption of these technologies, which also provide better information to travellers using the various systems.

The public is expected to derive enormous benefits from the deployment of the ITS applications. The increase in transport activities along with the need to reduce vehicle emissions as underlined in the 'White Paper-European transport policy for 2010: time to decide' (European Commission, 2001) imply the crucial role of ITS in transport. In traveller services, ITS can facilitate ticketing and intermodal transfer for passengers, as well as providing them with information and confidence throughout a trip (McDonald *et al.*, 2006). Furthermore, ITS also have huge potential to improve the operation and performance of individual transport modes, ensure optimum use of individual freight modes, improve the safety of travel and minimise the disruption from traffic incidents (ROSETTA, 2004). If the above benefits are achieved, congestion is to be reduced and a considerable scope for reducing air pollution impact will exist. The area in which ITS can be most effective though is "real time" applications, such as dynamic scheduling and bus fleet management. Research has shown that dynamic bus fleet management should be encouraged and supported with the use of ITS applications (Lam, 2001).

A set of criteria is required for the selection of suitable and cost effective ITS applications among the many existing systems. The selection criteria consist of being able to achieve reasonable levels of efficiency, effectiveness, economy, environment, safety and finance for the bus operation system. A further essential role for ITS is to cover the issues on whether the ITS applications are affordable, whether technology is available/stable and whether the public will accept the applications with regards to costs, time, convenience, ease of use, comfort and equity (Lam, 2001).

An ITS Master Plan, like the one proposed for the development of ITS in Hong Kong, (Lam, 2001), should be prepared to cover essential tasks, such as:

- to evaluate the cost-effectiveness of various options and recommend an integrated ITS strategy,
- to derive the associated operational plan including partial responsibilities and integrated requirements,
- to design a definition plan of the scope of application and an estimation of cost,
- to develop an implementation plan with Cost-Benefit Analysis to support the scope of short, medium and long-term implementation.

The application of ITS in a variety of forms is one of the numerous initiatives, that City Authorities are investing in, towards the direction of encouraging increased use of buses. There are many examples of ITS applications existing worldwide, as mentioned earlier, with the fastest growing application in Europe, with a range of options for systems and architectures in the field of bus fleet management; one of the most widely implemented refers to the Automatic Vehicle Location (AVL) systems (Hounsell and Graham, 2001).

2.3 Automatic Vehicle Location Systems

An Automatic Vehicle Location (AVL) system is a computer-based vehicle tracking system capable of determining a vehicle's location in real time (Lee *et al*., 2001). Initially AVL systems were used for military purposes, but now among other uses they are used to monitor transit and trucking fleets and police cars and ambulances. In transit operations, an AVL system measures the "real time" position of each vehicle. The "real time" position can be compared with the expected one and the potential of improving time bus scheduling is evident (Hwang *et al.*, 2006). AVL allows a dispatcher from a control centre to track vehicle movement and communicate with the vehicle's operator. The use of AVL systems offers a wide range of benefits, of which the most obvious are: (i) improved schedule adherence, (ii) real time information, (iii) data availability, (iv) improved emergency response and (v) more efficient fleet management (Lee et al, 2001). Several ITS measures are used to support reliability and efficiency of bus services, including AVL technologies that (a) enable local highway authorities and bus operators to work closely together to give buses priority at traffic controls and (b) bring significant fleet management benefits to bus operators (DfT, 2005).

2.3.1 Applications

The use of AVL systems has generated a wide range of opportunities within public transport sector, such as identifying late running buses and calculating levels of headway irregularity in high frequency services (Hounsell and Shrestha, 2005). AVL applications also include fleet management (real-time scheduling), real-time passenger information and public transport priority. The key components of AVL systems are often categorised in the following five functional components (Hounsell and McLeod, 1999):

i. on-vehicle equipment, used to track the position of the vehicle in real-time and provide a variety of on-bus functions,
ii. roadside equipment (beacons) for vehicle location and data transfer,
iii. the control centre, where the location of the bus fleet is monitored in real-time and the optimisation-management of the fleet takes place,
iv. communication systems, for the communication between the AVL centre and the vehicles, and

v. bus stop equipment, "driven" by the AVL system, supplying information to passengers.

Much of the focus of AVL systems deployment in recent years has been on increasing the overall capabilities, sophistication and degree of integration involved driving a remarkable progress in the degree of functionality and reliability (TCRP, 2008). The techniques that can be used for establishing vehicle location include (Lobo, 1998):

- Dead reckoning, such as odometers
- Beacon-based, such as microwave and inductive loops
- Radio triangulation, such as Loran-C, Datatrack and Omega
- Satellites based, such as Global Positioning System (GPS)

Commercially available AVL systems have increasingly been incorporating rapid advances that have become available in overall communications, computing and networking technologies (TCRP, 2008). These technologies each have their own advantages and disadvantages and, hence, the suitability of any of these AVL techniques depends on the functional requirement. Recognizing the importance of AVL systems, transport departments around the world are continuously implementing a variety of applications to support public transport. Among these applications, a recent addition is the implementation of a combined use of AVL technologies, the 'iBus' of London (D'Souza *et al.*, 2010), (ITS UK, 2010).

2.3.2 AVL benefits

The general objective of an AVL system in a transit operation environment is to improve services to customers and make them more efficient. The main benefits are found in the areas of (Greenfeld, 2000):

- Schedule adherence
- Improved fleet management
- Safety and security
- Performance monitoring
- Public information
- Improved management system

The use of AVL systems offer great potential in the field of public transport and may be beneficial in key urban transport issues such as the estimation of road traffic conditions using location AVL data (D'Acierno *et al.*, 2009). At 31 March 2008, 36% of buses in Great Britain were fitted with GPS and other electronic devices measuring bus punctuality (DfT, 2008). The objective of improving bus operations, using AVL systems, is not to make a system-wide plan but to adjust to on-street development and respond to demands that special circumstances require (Greenfeld, 2000). Such special circumstances may involve bus or traffic incidents with direct impact on the bus service level and the bus operation in general.

2.4 Bus and Traffic Incidents

2.4.1 Definition

As long as the role of incidents affecting bus operations is key element for the bus fleet management domain, incident management is vital for this research. Before making any reference to the incorporation of incident management in the bus fleet management procedure, a clear definition of the term “incident”, as it is used in the field of transportation, is required.

The 2000 Traffic Incident Management (TIM) Handbook defined an ‘incident’ as any non-recurring event that causes a reduction of roadway capacity or an abnormal increase in demand (Farradyne, 2000). However major weather events, such as Hurricane Katrina which took place in U.S. in 2005, caused the redefinition of TIM within a blueprint for incident response, highlighting the critical role of TIM in national preparedness (U.S. Department of Transportation, 2010). Hence, transportation agencies are recognizing that TIM is more than just a tool for increasing mobility and reducing congestion. However, the issue of major weather events incorporating responder and motorist safety is beyond the scopes of this research. More widely, incident events include traffic crashes, disabled vehicles, spilled cargo, highway maintenance and reconstruction projects, and special non-emergency events such as ball games, concerts, or any other event that significantly affects roadway operations (Zhu *et al.*, 2009). Another definition of the term would identify as incident any planned or unplanned event resulting in disruption of normal travel during a particular time and at a specific location along the transportation network (Zhao *et al.*, 1997).

In most metropolitan areas, incident-related delays account for 50%-60% of total congestion delay, however, in smaller urban areas, this can account for an even larger proportion (Lobo, 1998). For the top ten most congested urban areas in the US in 1998, the amount of incident-related congestion delays ranged from 218,000 to 1,295,000 person-hours, as reported by the Texas Transportation Institute (Hounsell *et al.*, 1996).

2.4.2 Incident key parameters

Identifying incident impacts is critical in improving traveller and responder safety, transportation system efficiency, and the nation’s economic competitiveness in general. In order to fully comprehend how to minimize incident impacts, an understanding of different types of incidents is also helpful. Incidents are categorized according to their size, severity, duration, impacts and locations. The issues that are regarded as essential in identifying problems to be addressed with respect to incident management are (Farradyne, 2000):

i. the frequency of incidents by location, time weather, and special circumstances,
ii. the duration of the incidents,
iii. the severity of the incidents, and
iv. the traffic impacts of the incidents.

In collecting data related to incident occurrence, the following information has been shown to be useful (Koehne, 1995), (Farradyne, 2000):

- frequency of occurrence by location,
- frequency of occurrence by month,
- frequency of occurrence by season of the year,
- frequency of occurrence in rain and non-rain,
- frequency of occurrence by day of week,
- frequency of occurrence by hour of day, and
- frequency of occurrence during special events

Measuring the effect of incidents on traffic is one of the most difficult and demanding issues in the field of incident management. Video monitoring and loop detectors may contribute to assessing traffic impacts and simulation modelling is carried out to determine incidents' traffic impacts. This research aims to address the issue of estimation of the impact of traffic incidents on the overall performance of bus operation and the suggested methodology focuses on this direction.

2.4.3 Incident management process

Once the definition for the incidents has been given the incident management program development process can be addressed. Incident management consists of a centralized effort focused on detecting, responding to, and clearing incidents to recover traffic flow (Fang, 2006). Efficient and coordinated management of incidents reduces their adverse impacts on public safety, traffic conditions, the local economy and the environment (U.S. Department of Transportation, 2000). Systems development theory provides the logical framework around which to organize this complex process. The logical steps provided (Figure 2.1) follow the same process that is used in many types of transportation planning exercises (Farradyne, 2000):

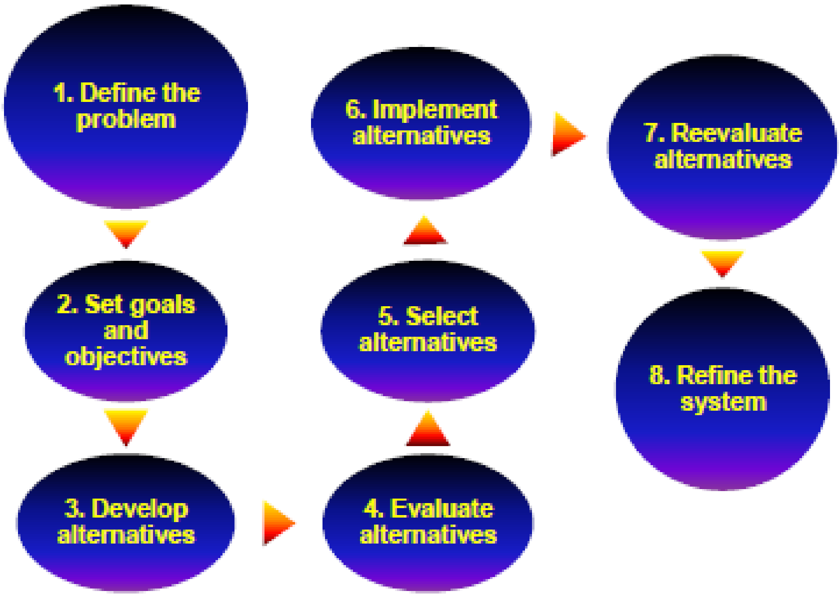


Figure 2.1: The Incident Management Process.

It should be highlighted that no consistent standard has been identified that can be applied to evaluate the quantifiable benefits of an effective incident management program. Nevertheless, there is a variety of examples of quantifiable benefits that have been attributed successful operation of incident management programs provided; the initial step of the employment of such incident programs is the identification and categorization of the incidents.

2.4.4 Incident categories

For bus fleet management, incidents could be detected by comparing a bus' schedule with real-time location data and taking actions to re-establish an acceptable quality of service. There are numerous incidents covered by bus fleet management which fall mainly within the following general categories depending on their nature (Kontaratos *et al*., 1996):

- Individual delay (one of the buses in the line is delayed);
- Generalized delay (several buses of same line are delayed);

- Multi-line delay (delays on several lines);
- Advance (a bus arrives at a stop before expected time);
- Saturation (people can not take the bus because it is full);
- Services together (two buses of same line are too close to each other).

Each bus-related incident which occurs falls into one of the above categories. Such incidents, according to information collected through direct liaising with operators of bus companies (i.e. Operations Directors of ‘London United’ in London and ‘First Bus’ in Southampton) (Polyviou, 2007), would mainly involve bus breakdowns, road works, traffic incidents, diversion, burst water main, road traffic collision, police incidents, illegal parking vehicles, disabled vehicles, sport events and other special events. In the attempt to decrease the occurrence of such incidents, bus operators have been upgrading their operation systems and the overall quality of the vehicles. However, the average age of the bus fleet in Great Britain at 31 March 2008 was 8.3 years, just slightly less than the average age of 8.4 years in 2005 (DfT, 2008). Nevertheless, the common strategy followed by bus operators is to address each traffic incident and its impact on the level of bus service using control actions that their invaluable experience in the sector of bus-based public transport suggests.

These are some of the key issues incorporated in the incident management procedure which is therefore related to the bus fleet management domain, taking into consideration that the occurrence of traffic incidents may affect the overall bus performance; the latter requires accurate and efficient control strategies which should be part of an extensive incident management program.

2.5 Dynamic Bus Fleet Management

2.5.1 Bus fleet management

Fleet management is the most significant function for any bus-based public transport organization and involves the scheduling and planning of routes simultaneously ensuring that buses run according to the schedule (Pradeep *et al.,* 2000), (Polyviou, 2008). The purpose of fleet management is to improve the overall efficiency on the vehicle fleet operations and therefore to improve the quality of public transport services (UITP, 2003). Management of a bus fleet primarily involves ensuring the timely arrival and dispatch of buses. In larger cities however, where the number of buses is significantly high and buses perform repetitive trips, this can become exceptionally difficult. In order to protect the essence of bus fleet management, which is to ensure that each bus reaches the enroute points on schedule, an integration of two or more modern technologies, such as Global Positioning System (GPS) and Geographic Information Systems (GIS), shows promise (Pradeep *et al.*, 2000).

In modern buses, fleet vehicles are equipped with devices, such as radio and GPS, which provide useful information to control operators in a Bus Fleet Management Centre (BFMC). This information enables them to determine the current location of buses and estimate the time of arrival at bus stops, hence enabling operators to take the appropriate actions to maintain the desired quality of the public transport service (Fernandez *et al*., 2004).

The main aim of Bus Fleet Management (BFM) is to enable operators to detect incidents by comparing a bus' schedule with real-time location data and taking necessary actions so as to address the disruption caused by these incidents. As mentioned in section 2.4, there are numerous incidents and, in case of such an incident, operators have to devise a management plan, in order to minimise the effect of the incident on the overall quality of the bus service. According to the type of the incident, different control actions can be identified as in the example in Table 2.3.

Table 2.3: List of main bus fleet management control actions (Belmonte and Fernandez, 2005)

Main control actions for bus fleet management strategies
• ***change*** **(from a timetable)** ***to frequency regulation*** **(all buses of a line)**
• ***change*** **(from frequency)** ***to timetable regulation***
• ***change frequency*** **of regulation**
• ***increase/decrease speed*** **of an individual bus**
• ***jump stops*** **(to set a delayed bus on time by going straight to the convenient stop)**
• ***help between services*** **(two buses interleave the stops they arrive at)**
• ***advance following service*** **(a bus must overtake a bus that is saturated)**
• ***advance head service start*** **(the bus at head of line must start before it is scheduled)**
• ***line head retention*** **(when two buses arrive at head stop at the same time, one of them must await on it)**
• ***timetable rotation*** **(each bus in a line adopts the scheduled timetable of its successor)**
• ***reinforce service*** **(an additional bus is included in a line)**

Time being the crucial parameter in transport field, the faster the operators make such management decisions the less the impact of the incident on the level of the bus service will be. A real-time response of sufficient quality, though, requires a Decision Support System (DSS) (Cuena and Hernandez, 1997) able to deal with the quantity of data that operators receive and the complexity of their reasoning.

At this stage, it is essential to highlight that a DSS is not a substitute for operators, but a useful tool that helps them explore the potential consequences of their control actions. The final decision and the responsibility for it remain with the bus operators (Fernandez et al, 2004). For bus operators, though, to respond promptly and accurately to bus and traffic incidents, strong, reliable and accurate research on incident modelling is required, and thus this research work is a necessity.

The main target of a fleet management policy is to ensure timely arrival and departure of buses as per scheduled. Research has investigated innovative bus fleet management options, such as an application which has been proposed and involves artificial intelligence, which utilizes data from the vehicle tracking system in order to enforce the schedule monitoring of the bus (Peripimeno et al, 2004). An application of artificial intelligence (transductive inference) for signal estimation to evaluate timing points arrivals given the tracking data of a bus through its journey is proposed. The aim of this work was to apply the proposed algorithm to the bus management field and analyse its performance to test if it permitted more accurate computations of the timely arrivals of buses at the timing points. This was achieved by estimating the arrival and departure times at certain points of the trip (bus stops, interchange points, crossroads) which are considered as crucial decision points for the fleet management aspect. To make a comparison of the timetable with the current position of the vehicle, operators should be provided with timely arrivals of the bus at the timing points. Even though this mathematical approach provided more accurate estimates compared to the traditional methods based on linear interpolation or Ridge regression, one of the main disadvantages is the fact that the computational load which is much higher The proposed method is part of the concept towards the direction of accurate timing points detection and the predictions' making on the condition of the network. The latter is very important task for maintaining the agreed standard of service and enable operators use that in the dynamic BFM procedure, which, though in early stages, has received high interest by both bus operators and research community.

2.5.2 Dynamic operation

Dynamic Bus Fleet Management (DBFM) is one of the main elements of dynamic bus scheduling. The objective of dynamic scheduling application is to make the public transport scheduler more effective in producing efficient schedules while providing a high quality of service to the travelling public. The problem of dynamic scheduling can be divided into two main parts, that of off-line scheduling and that of on-line scheduling. Off-line scheduling involves bus trips scheduling and crew scheduling for the various bus lines, this bus and crew schedule covers all day types across the whole year. On-line scheduling involves the re-scheduling that follows when the off-line schedule is put into practice. On-line scheduling attempts to bring the current schedule of buses and crews back into order resembling the original schedule (Marques *et al*., 1999).

Dynamic fleet management for bus operations is in its early development stages, taking into consideration the promising potential of applying ITS in the fleet management application. DBFM would employ Automatic Vehicle Location (AVL), Automatic Passenger Counting (APC) and information display technologies, but it would also incorporate advanced and widely deployed information communication systems, advanced user interfaces and computerized fleet management software. The ultimate requirements for such a system would be adequate information for transit users and for transit operators (UTCA, 2003).

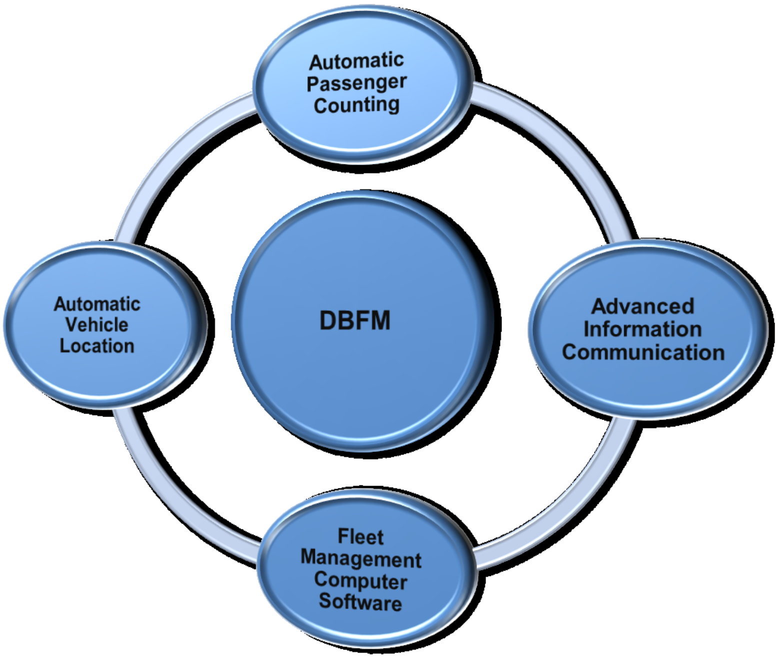


Figure 2.2: The concept of Dynamic Bus Fleet Management (DBFM)

2.5.3 Gaps in implementation

Real-time knowledge of bus locations and performance, as provided by AVL, enables bus operators to implement DBFM to improve situations which may have been affected by incidents. Adding 'spare' buses to routes to regain regularity or removing buses from congested sections are examples which could be included in such a DBFM. However, operators should be provided with better guidance on how to use ITS, like AVL, towards this direction, as it appears they have limited guidelines and hence insufficient knowledge (Hounsell, 2004).

An investigation has illustrated that the maximum capacity of the bus, the number of bus stops and the stopping time at each bus stop have important effects on the dynamical behaviour of the bus on routes (Nagatani and Yoshimura, 2002). As mentioned, the response to unexpected incidents and demand changes is a key element of the DBFM. Real-time operation scheduling methods to respond to unexpected incidents and demand changes have been developed and tested, such as the Block-Constraint Search Method (BCSM) (Aisu *et al.*, 2000). Another method of investigating an incident situation is data mining, whose aim is to mine the data for any hidden ‘treasures’, here in the form of new knowledge, which is not readily feasible to accomplish while employing other tools for investigation, such as simulation (Der-Hong *et al*., 2002).

2.6 Chapter Summary

A wide range of Intelligent Transport Systems (ITS) applications for bus-based public transport has recently been developed to improve the reliability and efficiency of bus operations thereby supporting buses as a real transport alternative to the private car. Dynamic Bus Fleet Management (DBFM) takes into consideration the promising potential of applying ITS to fleet management applications and addressing the impact of bus and traffic incidents on overall bus performance.

However, the lack of sufficient guidance on how bus operators can use ITS and how each characteristic of a traffic incident influence the overall bus performance highlight the need for further research into bus and traffic incidents and the way they affect bus operations. In order to carry out specific research on traffic incidents modelling to support DBFM and identify the key parameters that are involved in it, the need for a review of the available approaches related to vehicle routing and scheduling problem and the existing computer packages was carried out, as described in Chapter 3.

-Chapter 3-

Problem Description, Approaches and Gaps

3.1 Vehicle Routing and Scheduling Problem

3.1.1 Introduction

In recent years, it has become increasingly important for public transport companies to provide an adequate service level to their customers, due to the privatization and the growing competition in the public transport market (Huisman *et al*., 2004). A large number of software packages which aid the planning of routes and schedules for vehicle fleets has been developed for this purpose. The aim of this Section is to illustrate the key features, the components and the potential of existing vehicle routing and scheduling software packages.

It is crucial at this stage to present a review of the Vehicle Routing and Scheduling Problem (VRSP), since the bus fleet management is highly connected to it, with particular focus on dynamic Vehicle Scheduling Problem (VSP), before presenting the existing approaches to these problems and available software packages. Therefore, the first part of the Chapter introduces the VRSP and dynamic VSP, while the second part describes available vehicle routing and scheduling software.

3.1.2 Background

Over the past 30 years, vehicle routing and scheduling has grown to be an important area in urban mass transit systems and transportation. Consequently, successes have been documented and commercial software packages have been produced.

The domain of problems considered to be vehicle routing and scheduling has rapidly expanded.

Vehicle routing and scheduling applications consist of a great variety of problems including node routing and scheduling with precedence constraints (e.g. pickup-and-delivery problems), arc routing and scheduling (e.g. scheduling household-refuse collection and meter readers) and pure scheduling (e.g. scheduling vehicles and drivers for mass transit systems) (Bodin, 1990).

Practical vehicle and scheduling systems have three important components: the algorithmic approaches that have been designed for solving these problems in practice, the computer environment for practical vehicle routing systems and the role of geographic information systems in vehicle routing and scheduling systems. The computational complexity in solving vehicle routing and scheduling problems to optimality led to the employment of heuristic methods.

3.1.3 Heuristic algorithms

This term derives from the Greek '*heuriskein'* meaning to find or discover. The word has been used in Artificial Intelligence circles with quite a different connotation, for example it may include methods which find globally optimal solutions. Heuristic optimization techniques are defined by Reeves (1995) as the techniques which seek good (i.e. near optimal) solutions at a reasonable computational cost without being able to guarantee either feasibility or optimality, or even in many cases to state how close to optimality a particular feasible solution is (Reeves, 1995).

The interest for heuristics has been increased rapidly in the last few years. Since heuristics obtain approximate solutions the following issue has arisen: the problem that we are actually optimizing is a model of a real world problem and there is no guarantee that the best solution of the problem is also the best solution to the underlying real-world problem. In other words the question whether we should favour an exact solution of an approximate model or an approximate solution of an exact model arises. It is very difficult to formulate an exact model for a real-world problem, but heuristics are flexible and are capable of coping with more complicated and realistic objective functions and/or constraints than exact algorithms. Thus, it is possible to model a real-world problem rather more accurately than is possible if an exact algorithm is used (Lawler, 2001).

Heuristic algorithms can be classified to three main groups, although usually combinations of these algorithms are used (Kirannoudis, 2004):

- Constructive algorithms
- Local Search algorithms
- Metaheuristics

There are two methods that in the last few years have received considerable coverage, Simulated Annealing and Tabu Search. Both of them have been widely used as solution approaches for the Vehicle Routing and Scheduling Problem (VRSP) (Kirannoudis, 2004). TABUROUTE is such an example of a Tabu Search algorithm for the VRP that often produces satisfying solutions; according to results, Tabu Search outperforms the best existing heuristics (Gendreau et al, 1994), (Laporte et al, 2000).

3.1.4 Dynamic vehicle scheduling

3.1.4.1 Vehicle scheduling problem

There is no doubt that the Vehicle Scheduling Problem (VSP) arises in the management of every transit agency. The VSP is considered to be one of the main scheduling problems of a public transport company. Figure 3.1 illustrates the relation between the four operational planning problems in the planning process at a public transport company.

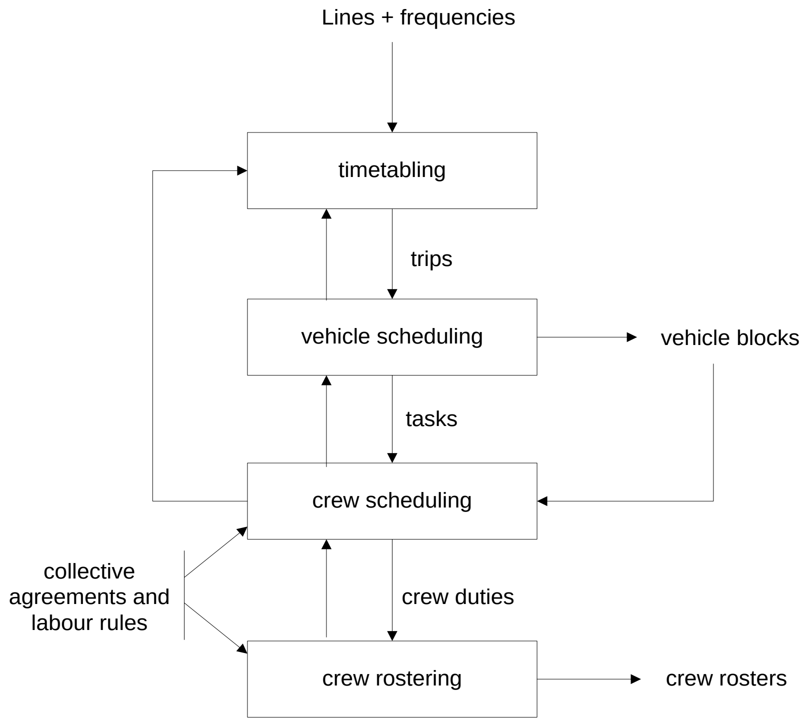


Figure 3.1: Traditional Planning Process (Huisman et al., 2004)

In the Multiple Depot VSP, the total vehicle costs have to be minimized subject to constraints, which among others involve depot capacity constraints that specify for every depot a maximum number of vehicles (Huisman *et al*., 2004). This constraint of fleet size plays a crucial role in bus-based public transport operation, thus, this parameter is particularly important for the current research. Furthermore, the vehicle costs consist of a fixed component for every vehicle and variable costs for idle and travel time. Another significant term is the deadhead, which is the period that a vehicle is moving to or from the depot, or the period between two trips where a vehicle is outside of the depot (possibly moving without passengers) (Huisman *et al*., 2004).

The usual planning procedure consists of solving the vehicle and the crew scheduling problems sequentially. However it is thought to be useful to address crew scheduling at the same time as vehicle scheduling. The combination of these two problems is called the Vehicle and Crew Scheduling Problem (VCSP) (Haase *et al*., 2001). While the single-depot, homogenous fleet vehicle scheduling problem is a polynomially solvable minimum cost flow problem, simultaneously scheduling vehicles and crews is an NP-hard (non-deterministic polynomial-time hard) problem, independently of the number of depots (Haase *et al*., 2001). An exact approach for solving the VCSP in urban mass transit systems has been presented via a computational study which has proved that this approach out-performs previous methods (Haase et al, 2001).

3.1.4.2 Dynamic vehicle scheduling problem

The fact that the VSP is traditionally solved a few months before the new timetable starts and will not be changed for the whole period that the timetable is valid highlights a weakness: when there is a delay, during the execution of the schedules, at a certain moment, the trip following the delayed trip may start late (Huisman *et al*., 2004), (Huisman *et al.,* 2006). If the VSP is solved dynamically, which indicates that we reschedule a few times a day, that is, we reassign vehicles to trips, we may be able to prevent the delays at the start of a trip in many cases. However, it is obvious that this approach requires a fast and reliable communication and information system between the vehicles and the planners (Huisman *et al*., 2004), (Huisman *et al.,* 2006).

The dynamic VSP is an innovative approach and general surveys about dynamic and stochastic models explain why it may be useful to use dynamic models instead of static ones for many problems in the field of transportation and logistics (Powell et al, 1995). Some of the key features commonly found in dynamic transportation problems are presented below (Sadeh and Kott, 1996):

- There are multiple demands to transport commodities or entities from/to origin or destination points,
- multiple vehicle resources are to be routed and scheduled to meet the demands,
- each demand is to be assigned to one or more vehicles,
- the destination and/or origin points may have to be determined dynamically,
- there are time window constraints,
- there are constraints on the vehicle capacity,
- there are multiple other constraints of varied nature, such as constraints on duration of tours associated with vehicles and/or with particular requests,
- the demands can change dynamically while the schedule is executing, and
- the resources may also change dynamically.

A robust solution approach to the dynamic VSP has been recently presented and the results reported show that the number of trips starting late and the delay costs can be reduced by using only a few vehicles more, if the dynamic method produced is used instead of the traditional static one (Huisman *et al*., 2004). The following section is dedicated to the software relative to the vehicle routing and scheduling and describes the available microcomputer-based routing and scheduling systems.

3.2 Vehicle Routing and Scheduling Software

3.2.1 Background

In the past 35 years, the most impressive change in vehicle routing and scheduling systems is the computer environment in which the algorithms are embedded. Vehicle routing and scheduling systems were primarily batch algorithms operating on a large computer. These systems developed in the late 1970s were called 'computerized routing and scheduling systems' (Bodin, 1990). The introduction of the microcomputer allowed the user to make manual interventions in the process and give a graphical display of the solution. The original routing and scheduling systems on microcomputers had these characteristics:

- They had low computational speed;
- Most of them were weak algorithmically;
- Only some of them had reasonable graphics and manual intervention abilities.

The second generation systems which followed were the microcomputer-based vehicle routing and scheduling systems, these were developed in the mid 1980s. However, with the rapid pace of the computer technology development, most of those systems have similarly been replaced by the next generation's microcomputer-based systems. Nowadays, microcomputers have become powerful and the user has become more demanding. Therefore, the ability to integrate the vehicle routing system with systems such as order entry and vehicle location is becoming a requirement of most users. Undoubtedly the software technologies, except for the algorithms, appear to have finally caught up with the users' needs (Bodin, 1990).

3.2.2 Review of available software

A large number of commercially available software packages aid the planning of routes and schedules for vehicle fleets. Most of these packages include the following features:

i. Vehicles (numbers, locations, capacities)
ii. Staff (numbers, hours worked, pay scales)
iii. Depots (locations)

iv. Collections and/or deliveries to be made (numbers, volume, locations)
v. Road network (geometry, speed limits, distances)
vi. 'Optimisation' according to cost or distance or time, although optimal solutions cannot be guaranteed for large problems

Surveys of routing and scheduling software manufacturers (OR/MS Today, 2010) quantified the operating characteristics and the key areas of investigation of the most commonly used systems on the market:

- Routing functions: node routing, arc routing, real-time routing, daily routing, route planning and analysis
- Maximum size of problem solvable by the system: number of stops, vehicles, depots
- GIS capabilities: mapping facilities provided
- Performance: computation speed, *what types of algorithms are employed,* are approximations used?
- System requirements: computer operating system, memory, processor speed

The response to the question about the types of algorithms employed turned out to be of particular interest, hence the answers are summarised below with the software product shown in brackets (OR/MS Today, 2010):

- Genetic algorithm (A.MAZE)
- Mix of local and global optimisation techniques (Cube Route Platform)
- Linear programming and combinatorial methods (EDGAR)
- First solution heuristics are provided, including savings and variants of nearest addition. Improvement methods include Tabu Search (TU) and guided local search with predefined neighbourhoods (ILOG Dispatcher)
- Combination of heuristics and optimizations methods (Optrak 4, Roadnet)
- Heuristic algorithm designed for optimising road-based transportation operations. With many routing parameters that enable

to handle the wide range of transportation constraints. Includes an optimising 'shortest path' algorithm (Paragon)

- Heuristic methods and some integer programming (STARS)
- Proprietary and confidential (DirectRoute, DISC, REACT, Roadshow Enterprise, RouteSmart, Shortrec, TruckStops)

The surveys led to the conclusion that there are many vehicle routing and scheduling packages available, which employ different algorithms and techniques, and that different packages are likely to produce different results. Users should therefore take into consideration that packages are not guaranteed to provide a global optimum result for every application. In addition, there is no evidence that one technique is clearly the best to use. Other existing vehicle routing and scheduling packages involve: 'LogiX' (vehicle scheduling model developed by DPS International), 'RouteLogiX' (route journey planner tool), 'RoundBuilder' and 'LogisticsOptimiser' (vehicle trip planning tools designed by Entec for waste collection purposes), 'FleetRoute' and 'SIMULINK'. The last two were investigated more thoroughly in order to explore their features and potential for use in bus fleet management issues.

FleetRoute

FleetRoute is developed by Civix Software and is a routing and scheduling package that has been specifically designed for waste collection (Polyviou, 2006). The developers claim that the package is particularly useful for creating practical routes as the user can modify routes easily and specify certain conditions, which hinted that it could have the potential of being used as a tool for bus operations. Therefore, some more detailed information is presented below, after contacting the developers of this product. According to the product developers, then, Fleetroute enables the user to (Polyviou, 2006):

- ✓ Maximize the productivity of each vehicle
- ✓ Specify the maximum and minimum number of hours, stops, capacity, drum trips and distance desired for each vehicle.
- ✓ Balance the service days and districts.
- ✓ Specify the general area of vehicle before route creation.
- ✓ Customize computer generated route configurations to optimize existing routes.

- ✓ Find the closest transfer stations/landfills for each vehicle for each dump trip and allocate vehicles to the optimal depot.
- ✓ Find optimal locations for new facilities.
- ✓ Control the types of turns made such as minimizing u-turns

However, the response of the developers of FleetRoute to the author's question, whether FleetRoute could offer solutions, as routing and scheduling software, at the field of bus fleet management, was (Polyviou, 2006):

'*FleetRoute (tm) specializes in high density (arc routing) and low density (point-to-point) routing which would be somewhat applicable to school bus or para-transit (on-demand) routing bus not to scheduled municipal bus service routing. Fleetroute (tm) could offer some tools for analysis (such as optimizing bus stop locations in relation to resident walk-times), but does not currently have a formalized approach to the problem (in other words there would be consulting and development required).'*

SIMULINK

Another existing computer package which has been used during the development of a simulation support tool for real-time dispatching control in public transport and seems to have the potential of being useful in bus fleet management is SIMULINK (Adamski and Turnau, 1998). The main purpose of dispatching control is the elimination of the deviations from schedule of the vehicles on the route with not too high a level or rate of change of interventions. The great advantage, that SIMULINK offers, of using graphical windows which are automatically scaled, makes this package a conventional visual tool for the human operator (MathWorks, 2010). Every change of the controller or model parameters can be done interactively. The library procedures of MATLAB and SIMULINK toolboxes can be easily included into the model in a simply way.

SIMULINK (version 1.2c and later versions) equipped with the Control System Toolbox, the program from the MathWorks for simulating dynamic systems, contains a number of improvements (Adamski and Turnau, 1998). SIMULINK is a platform for multidomain simulation and Model-based Design of dynamic systems. It provides an interactive graphical environment and a customizable set of block libraries that let user accurately design, simulate, implement, and test control, signal processing, communications, and other time-varying systems. The key features of this computer package are the following (MathWorks, 2010):

- ✓ Extensive and expandable libraries of predefined blocks
- ✓ Interactive graphical editor for managing intuitive block diagrams
- ✓ Ability to manage complex designs by segmenting models into hierarchies of design components
- ✓ Model explorer to navigate, create, configure, and search all signals, parameters, and properties of the model
- ✓ Ability to interface with other simulation programs and incorporate hand-written code, including MATLAB algorithms
- ✓ Option to run fixed- or variable-step simulations of time-varying systems interactively or through batch simulation
- ✓ Functions for interactively defining inputs and viewing outputs to evaluate model behaviour
- ✓ Graphical debugger to examine simulation results
- ✓ Model analysis and diagnostics tools to ensure model consistency and identify modelling errors

3.2.3 Conclusions on software review

Computer scheduling of public transport has been the subject of many international conferences and research focus. There are several commercial systems, as highlighed above, whose methods have been published and others whose methods have not been fully published. The most widely exposed of the latter have been the HASTUS system from Montreal and the BUSMAN system developed through 1980s (Wren and Kwan, 1999). More recent work including a bus scheduling system is the BOOST system, an object-oriented bus vehicle scheduling system presented in 1999.

Commercially available vehicle routing and scheduling systems have spectacularly increased in functionality and sophistication in the last decade. The systems have significantly better graphics and user interfaces and the classes of applications have increased. Vehicle routing and scheduling software is considered to be a necessary part of every transport company’s logistics/management system.

On the other hand, algorithmic development for solving practical problems has not kept pace with these other developments and there is still a tremendous opportunity to develop effective computational procedures for solving practical vehicle routing and scheduling problems, such as the one of bus fleet management. The challenge to the researcher is to discover these procedures and for the vendor to commercialize them.

3.3 The Modelling Approach

3.3.1 Introduction

A review of the VRSP domain is required to fully understand the nature of bus fleet management and identify the lack of a relevant computer packages capable of modelling bus and traffic incidents for bus fleet management purposes. Further investigation into the availability of software, examining both aggregate methods and simulation modelling approaches is required at this stage. A significant number of computer programs is available nowadays due to the recent rapid growth in computer technology. The aim of this Section is to investigate the features and the capabilities of each computer program, so as to choose the one capable of modelling incidents for evaluation of bus fleet management strategies. There are two main categories of computer programs, depending on the modelling approach (Shrestha, 2003), the aggregate methods and the simulation modelling, as illustrated in Fig. 3.2.

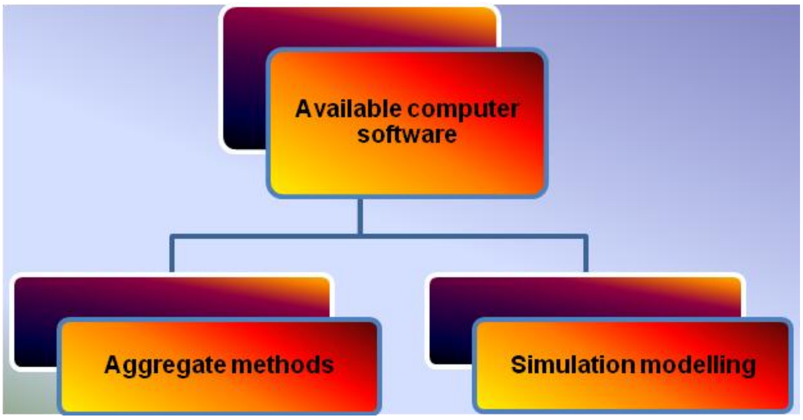


Figure 3.2: Main categories of computer software according to modelling approach

3.3.2 Existing computer software and gaps

Aggregate methods are mainly used for traffic network modelling. They use formulae with average values of the parameters such as capacity and vehicle flow over a period of time (Willoughby and Emmerson, 1999). The aggregate models are indeed very effective during the evaluation and assessment of traffic management schemes. These models, however, are not developed for purposes of modelling public transport in detail, though they contain some facilities for this kind of transport. AVL systems, which are basic elements in the application of DBFM, and other significant parameters, can not be modelled by aggregate methods. CONTRAM, TRIPS and SATURN are all examples of this type of model. To sum up, and with the detailed modelling of bus operations' features being primary aim of this research work, the review of these models led to the conclusion that they would be insufficient to carry out this work with its unique requirements and expectations.

With simulation modelling, on the other hand, the aim is to model the behaviour of each vehicle/driver as it moves through the road network, based on the characteristics of the vehicle/driver behaviour (Willoughby and Emmerson, 1999). Microscopic simulation provides an excellent experimental base for the design of control strategies, enabling the visual and statistical evaluation of the results (Czogalla and Hoyer, 1997).

The SMARTEST Project (Algers *et al.,* 1997) found more than 60 developers producing modelling programs around the world (Bagot, 1999). The simulation technique is regarded by many researchers as the ideal tool for the study of bus operation in a transport corridor (Agarwal *et al*., 1994). Some of the commercially available packages for simulation modelling are PARAMICS, VISSIM, TRAF-NETSIM, FLEXSYT II and HUTSIM (PRISCILLA, 2001).

A review of the most commonly used commercially available simulation models was carried out by PRISCILLA (2001). A comparison of the main features of the models has been made and given in Table 3.1 (PRISCILLA, 2001), where '√' is used for features that were reported as being modelled and 'x' for features that were not available as built-in facilities:

Table 3.1: Modelling components of commercially available simulation models

Available simulation models					
	FLEXSYT II	HUTSIM	PARAMICS	TRAF-NETSIM	VISSIM
Bus	√	√	√	√	√
Bus stop	x	x	x	√	√
Traffic signals	√	√	√	√	√
AVL systems	x	x	x	x	X

VISSIM and PARAMICS appear as the simulation models with the most similarities to the needs and requirements of this specific research; thus, their potentiality was investigated in detail. VISSIM is a general purpose computer-based traffic simulation system, which contains a psycho-physical car following model for longitudinal vehicle movement and a rule-based algorithm for lane changing (Algers *et al*., 1997). VISSIM is a microscopic and stochastic simulation with fixed time-slices, while PARAMICS, on the other hand, includes dynamic and intelligent routing and inclusion of ITS (Druitt, 1998). It takes into account public transport involvement and its interaction with other modes at bus-stops. However, there is no sufficient literature suggesting the potentiality of DBFM using ITS, such as AVL systems, for either of these two simulation models. Furthermore, even though TRAF-NETSIM is capable of modelling the components of bus stops with detail, it does not incorporate real time location of the buses which is a key element of dynamic bus fleet management. Other microscopic simulation models, including STEER (Signal Traffic Emulation with Event-based responsiveness), DRACULA (Dynamic Route Assignment modelling where, when and how to travel), TRANSYT (using an algorithm to calculate network signal timings) and AIMSUN (used to improve road infrastructure, reduce emissions but not for bus fleet management), were also reviewed. None of the above models, though, was found capable of achieving the research objectives of this work with particular focus on the detailed bus stop activities and bus fleet management options implemented.

The literature review depicts the lack of a commercially available simulation model capable of modelling bus and traffic incidents and their impact on bus operations for bus fleet management purposes using ITS. This gap was partially filled with the development of the simulation model called SIMBOL (Shrestha, 2003). The objectives of SIMBOL meet some of the initial targets of the author’s research, although the direction is dissimilar, since SIMBOL was developed in order to investigate bus priority options and evaluate the comparison output for each of these priority methods. SIMBOL is a very useful tool in the field of public transport simulation modelling using ITS, thus, the review of its components in detail was considered of high importance.

3.3.3 The simulation model SIMBOL

3.3.3.1 Description and components

There is a considerable research worldwide focused on the field of bus priority strategies. Therefore, a variety of simulation models for bus priority has been developed. Shrestha (2003) developed a simulation model called SIMBOL

(SImulation Model for Bus priority at traffic signals); it is a microscopic simulation model with fixed time scanning interval of one second, and simulates advanced bus priority strategies at traffic signals. The programming language used in developing the model is C++ (Borland Builder version5). A simple diagrammatic model of SIMBOL is shown in Figure 3.3 (Shrestha, 2003).

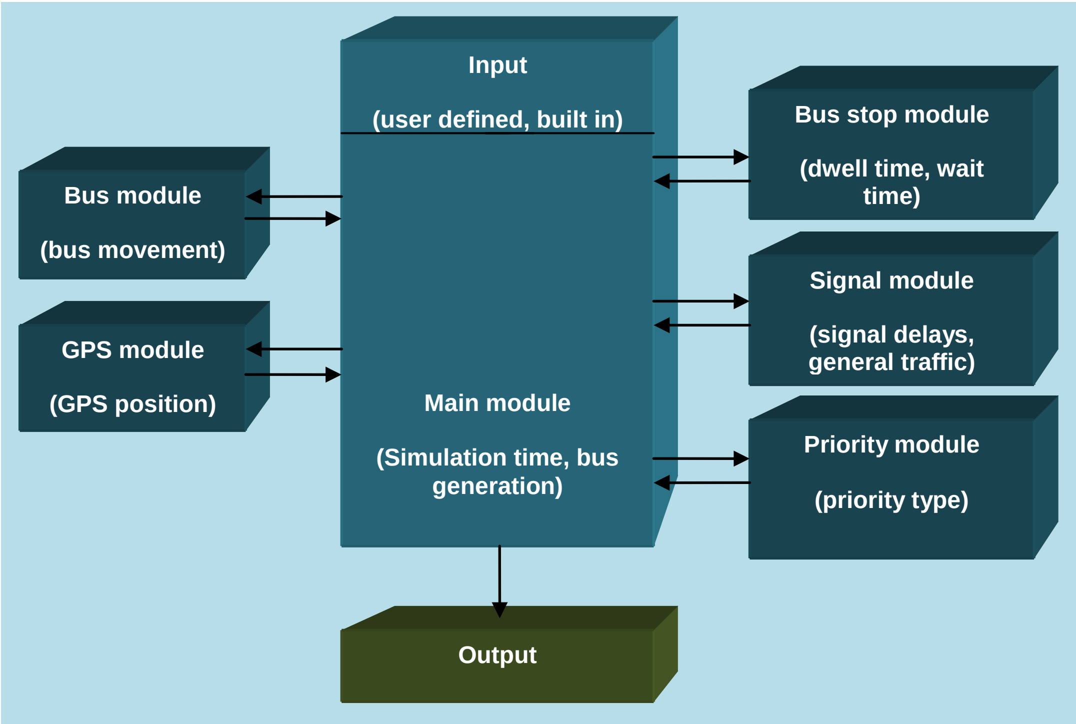


Figure 3.3: Diagrammatic model of simulation model SIMBOL

C++ is a highly flexible and adaptable programming language and has been used for a wide range of programs including operating systems, applications and graphics programming. C++ owes its success to the fact that it allows the programmer to organise and process information more effectively than most other languages. One of the major goals of the C++ language is to organize instructions into reusable components. A key innovation of the language is the idea of combining data and instructions together in a construct called a class or object (Oualline C++ Programming, 1995). The characteristics of the programming language C++, some of which are briefly mentioned above, set the choice of this language for developing simulation models for bus operations, such as bus priority and bus fleet management, as a quite reasonable, practical and advantageous choice.

SIMBOL consists of 6 different modules, which interact with each other harmonically and whenever they are activated via the Main module. The modules are (Shrestha, 2003):

1. the Main module
2. the Bus module,
3. the Bus Stop module,
4. the Signal module,
5. the Bus Priority module, and
6. the GPS module.

Looking into the core of the model development and process, is should be mentioned that the main module is responsible for plotting the overall simulation by taking input, coordinating the interaction between all other modules and generating output. The main function of this module is to manage the simulation time and the generation of buses. The main output parameters of the main module are the parameters used to evaluate the performance of different bus priority strategies. The bus module models the movement of buses by updating its position every second. The bus stop module models the interaction between buses and passengers at bus stops. It estimates the lateness of a bus at departure by estimating the dwell time of buses and the waiting time of passengers at bus stops. The interaction of buses with traffic signals is modelled by the signal module. This module calculates the signal period and the delays at traffic signals every second. This module calculates the signal period and the delays at traffic signals every second. The priority module is only activated when the model runs in priority mode. It is responsible for providing priority to the buses at traffic signals. The priority strategies are tested and evaluated through this module. Hence, this module is a key issue for SIMBOL, but will not be part of the new simulation model that will be developed to model bus fleet management options. Finally, the GPS module provides the GPS location of buses to detect a bus approaching traffic signals (Shrestha, 2003).

3.3.3.2 Conclusions on review of SIMBOL

SIMBOL is capable of simulating a bus route taking into account buses, bus stops, traffic signals, AVL systems and advanced bus priority strategies. The model has opened up possible areas for further work including further application of the existing model and development of the model to incorporate other issues which it does not yet cater for. These include the continuous circulation of the route, the modelling of bus and traffic incidents affecting the bus operation and the issue of bus fleet management. While the model provides solutions to the Origin-Destination route, there is a practice of running buses continuously around a route (i.e. Origin-Destination-Origin). In such cases, any deviation at the starting time at the origin of the first leg may affect the starting time at the second (returning) leg of the journey. Modelling this continuous circulation of buses on route is part of this research's objectives and is a significant element to the bus fleet management domain vital to explore possible bus fleet management and optimisation.

The high adaptability and flexibility of the computer language C++ offers the possibility of using part of the code from the model SIMBOL as a base case during the development of a new simulation model capable of achieving the targets of this research. However, issues like the continuous circulation, the parameter of the bus fleet size, the incorporation of bus and traffic incidents and their impact and the suggestion of bus fleet management options define a very innovative and challenging frame or research towards the development of a new model. The latter requires a perfect understanding of each characteristic and parameter of the model to be developed and a clear definition of its objectives and expected output. The lack of available computer software capable of modelling bus and traffic incidents to support bus fleet management direct the author to produce a methodology to develop a new microscopic simulation model for this purpose.

3.4 Chapter Summary and Research Gap

Between 2000 and 2008, public transport patronage increased by 19% in England, a very significant rise if the DfT's Public Service Agreement to increase the use of public transport by more than 12% by 2010 compared with 2000 levels (DfT, 2008). The average rating for overall satisfaction for bus services in England was 82 points in 2009, unchanged from 2008, with the rating in London being lower at 80 points (DfT, 2009).

The critical review and comprehension of the above statistics on public transport, as published by DfT in 2009 Public Transport Statistics Bulletins, highlights the need for continuous support of bus operations with the use of ITS and the implementation of new simulation models capable of modelling incidents for bus fleet management purposes. Fleet management is the most significant function for any bus based public transport organization. The fact that public transport organizations pay much of their attention to this sector reveals the importance of the research in this field and the need for a development of a complete methodology which will involve all the key issues of bus and traffic incidents modelling for bus fleet management guidance. Dynamic bus fleet management, a scenario developed recently, should be encouraged and supported with the use of ITS applications and especially AVL systems, which offers remarkable potential benefits.

To determine the suitability of the existing models, both aggregate methods and simulation modelling programs available for modelling public transport were reviewed. Among all the existing and described simulation models, SIMBOL is the most suitable for the purpose of this research and offers the possibility of using part of its code as a base during the development of a new simulation model capable to achieve the targets of this research. The new simulation model, called SIBUFEM (Simulating Incidents for BUs FlEet Management) (Polyviou *et al.,* 2011), will incorporate the simulation of continuous circulation, the modelling of various bus and traffic incidents according to their characteristics, the constraint of the bus fleet size parameter and the ability to provide useful visual and text outputs to recommend bus fleet management options to address the impact of each incident. The specifications, requirements and expectations adopted to develop the model are described in detail in Chapter 3 which analyzes the development of the new microscopic simulation model proposed, called SIBUFEM.

-Chapter 4-

Model Methodology and Data

4.1 Introduction

The literature review highlighted the need for the use of simulation modelling to develop a new microscopic simulation model capable of exploring bus and traffic incidents in a continuously circulated bus operation of high frequency. This model will also suggest strategies to control and eliminate the impact of these events on normal bus operations. The new simulation model developed, which aims to cover this research gap is called SIBUFEM (Simulating Incidents for BUs FlEet Management). The methodology followed to develop this model, the data used and the model expectations set are presented in detail in this Chapter. Developing such a model for high frequency bus operation systems requires an expert knowledge of the elements involved and a careful assessment of all the transport related parameters.

Chapter 2 highlighted how part of the existing simulation model SIMBOL was used during the development of the new model. The significant changes made, the additions involved and the innovative characteristics and aims targeted through this PhD research create the new and unique specifications of the model SIBUFEM. Each element and parameter of the model needs to be presented thoroughly beginning with the identification of the model requirements.

4.2 Model Requirements

The frame of the specifications identified for this work was set by the initial objectives of this research. Simulation modelling enables the developer to apply a wide variety of functions and parameters and to alter them in order to investigate different scenarios to be modelled. The model developed incorporates all these functions required to model bus and traffic incidents for bus fleet management strategies with the use of ITS and particularly AVL systems. In addition, SIBUFEM was constructed to incorporate a visual display of the results, including an adequate route, buses, bus-stops and AVL systems. The key model requirements, which set the development basis, were to:

- ✓ represent an adequate route with a significant number of bus stops, using existing data,
- ✓ develop a method of generating buses for a high frequency bus service,
- ✓ model the continuous circulation of buses and their movement around the route, i.e. Origin-Destination-Origin,
- ✓ model bus stops including estimation of elements such as boarding and alighting times, waiting times and dwell times,
- ✓ model the use of AVL systems and in particular GPS systems,
- ✓ model various bus and traffic incidents and their impact on the bus operation,
- ✓ identify, calculate and compare the impacts of the incidents on the key performance parameters of the model and produce the relevant output as text and visual display.

The framework of the modelling methodology is defined by these specifications. Each of the requirements is fulfilled during the development of SIBUFEM and the methodology is described analytically in the following sub-section. Understanding the issues that arose during the modelling methodology is vital for the accurate determination of possible challenges to be anticipated and the complete comprehension of the capabilities and expectations of this research.

4.3 Modelling Methodology

4.3.1 Route

Several attempts have been made to classify bus routes fairly or equitably, thus categories such as linear, radial, diametrical, ring-road, cross-town, mainline, local, express, peak express or even special bus routes have been established. For the purpose of this research linear routes are investigated only, an assumption which is justified considering that linear routes are widely used in bus operations. The route, during the model development, is presented linearly, rather than in its actual geographical shape, providing the opportunity of using an existing corridor of the city of Southampton, UK and relevant available route data. It needs to be highlighted, though, that one of the key targets of this work is the continuous running of buses around the route, i.e. Origin-Destination-Origin.

4.3.2 Bus generation

The generation, movement and interactions of the buses with other components of the model are the main focus of the methodology. Buses are generated according to their time headways or schedule. In this research, the focus is on the high frequency bus services, which operate on a time headway basis and usually refers to services with five or more buses per hour (TfL, 2010) and is generally regarded as one where passengers turn up randomly at the bus stops. A normal distribution has been used for the generation of the buses in the simulated network, incorporating a deviation time to accommodate the randomness of the generation at the starting point. The bus generation is constantly checked with the simulation time and is one of the main functions of the main module. Until the number of buses generated becomes equal to the bus fleet size of the service modelled, a bus is generated whenever the simulation time reaches the predefined bus generation times. For the rest of the buses, the generation is based on the headway predefined for the service. Overall the bus headway plays a key role for any bus operation and is vital to be given the adequate attention.

Time headway

This research targets headway-based high frequency bus operations, where buses are generated at a defined time interval and frequency of five or more buses per hour (TfL, 2010). In a high frequency route waiting passengers tend not to look at the timetable (TfL, 2010) and the operation is subject to influences that can greatly impact the level of service provided to passengers (Milkovits, 2008). High frequency bus routes have short headways (i.e. the time between successive buses) and high passenger demand which leads to interactions between buses (Milkovits, 2008). In reality, however, generation of buses is affected by several factors, such as availability of buses, weather conditions and condition of buses. The direct consequence is that buses often are not generated exactly at the defined headway, which is an issue to be included and reflected in the model by using a distribution for generating buses. Using this distribution, whenever a bus is generated the headway of the next bus is calculated. The next bus is then generated according to this headway. The distribution used for the generation of the buses is an appropriate 8-minutes headway distribution with +/-1 minute error. The bus headway used during this research is the difference between the departure times of two consecutive buses; an 8-minutes interval is used as bus headway, which is a typical value for high frequency bus routes in London as shown by reports on route results for London buses services (TfL, 2007).

Fleet size

Another crucial parameter of the modelling methodology is the bus fleet size. The fleet size, taking into consideration the round trip bus journey time, is the factor that determines the bus headway of a service and it is very important during the scheduling phase of the operation as well as the reestablishment of the bus reliability in case of delays. This was further justified by interviews held with bus operators:

> *'This line operates every 10 minutes during the day time, but in the afternoon in rush hour it dropps down to every 12 minutes because the journey time is longer due to the congestion issue. But this is still unreliable so now for this timetable, which started in April we dropped the time down to every 15 minutes, which is again another paradox in the fact that in the busiest time of the day we are running fewer buses, which is in fact the same number of buses but they take longer to complete their journey. It didn't justify another bus in the fleet because to keep this every 10 minutes in the peak you need another bus, but what do you do with the bus the rest of the time?'*

The bus fleet size determines the bus headway and, hence, the schedule of a bus service. The decision around the exact number of buses used in a line is crucial for the bus operations and might have an impact on the bus reliability and passengers' satisfaction degree.

For the needs of the specific route, the distribution is used for the generation of a number of buses equal to the fleet size. Once this number is reached, which means all the available buses have been generated, the next bus is generated according to the 8-minutes headway with 1 minute error. The bus availability is a factor which determines the bus generation at the starting point of the route and, after all available buses being generated, a further bus is only generated once the first bus completes its return journey.

4.3.3 Bus movement

The key element which sets the framework of the bus movement is the need of a circulation of the movement, since there is a practice of continuously running buses around the route. The primary aim is to investigate the implications of this issue, as any deviation at the starting time at origin of the first leg may affect the starting time of the returning leg of the bus journey. Modelling the continuous circulation of buses is essential to explore the impact of bus and traffic incidents and suggest possible bus fleet management options to address them.

Once the buses are generated at the origin, they start moving along the route through the route links. A link, in this case, is described as a section of a bus route between two consecutive bus stops. The bus movement is based on various parameters such as type of the road, bus incidents, traffic conditions, road conditions, bus conditions, weather conditions, traffic accidents and various special events. Modelling all these aspects of random variability is very complex but categorizing them and modelling their key parameters and impact on the bus movement is a key focus of this research. For the movement of the buses, existing link journey times which incorporate the effect of normal traffic conditions in each link are used. The extra impact caused by bus and traffic incidents, which may create significant delays in link journey times, is incorporated during the incidents modelling.

4.3.4 Passenger generation

Passenger generation takes place at the bus stops along the route. Bus stops are the locations where passengers arrive, wait for a bus and where boarding and alighting occur. The calculation of the passengers boarding and alighting is an essential component of the model, since passenger demand is the main parameter determining the route, frequency and more widely the bus operation in total. The above numbers are dependent on the passenger arrival at the bus stops, a component for which there are various methods of modelling it. This may be achieved by using a distribution with a mean representing the average passenger flow to the bus stop and a variance fixed for all bus stops (Liu *et al.*, 1999), or, as decided for the purposes of this research, it may be accomplished using simple relationships between the rate of boarding and alighting passengers and the headway (Hounsell and McLeod, 1999). These relationships are simplified, assuming the random arrival of passengers, into:

Boarding passengers = boarding rate × bus headway (1)

Alighting passengers = alighting rate × bus headway (2)

The assumption of a random passenger arrival at bus stops is taken for a high frequency service where the passenger has no expectation concerning the bus arrival time before reaching the bus stop (Seddon and Day, 1974). Even though the number of boarders depends on the bus headway of the operation, this may not be the case for the number of passengers alighting, which is dependent on the passengers already on board. Therefore, the number of alighting passengers is calculated as a percentage of the total passengers inside the bus. This percentage is uniquely assigned for each bus stop and the relationship is:

Alighting passengers = passengers inside bus × (% of alighting passengers) (3)

The numbers of passengers alighting and boarding at each bus stop were obtained from existing data collected from previous studies on the route during the model development of SIMBOL (Shrestha, 2003). These numbers were then used to calculate a boarding passenger rate at every bus stop assuming the random arrival of passengers. This assumption was justified by previous studies (Rajbhandari, 2002), which means that a uniform rate of passengers' arrival can be used to estimate an average passengers' arrival rate for each bus stop.

4.3.5 Passengers waiting time

Various definitions may be given for the waiting time of a passenger in the field of bus operations, of which the most common ones classify the waiting time as the time between the arrival of the passenger and the arrival of the bus or the boarding of the passenger or even the departure of the bus. For the development of SIBUFEM, waiting time is the time between the arrival of the passenger at a bus stop and the arrival of a bus in which the passenger boards into. The waiting time may be approximately calculated by half the time between the departure of the first bus and the arrival of the second, which during this work is called 'time gap'. This assumption is reasonably justified by the uniform arrival of passengers at a bus stop and the methodology is followed for the development of simulation models such as SIMBOL (Shrestha, 2003).

However, in SIBUFEM the focus is on high frequency headway-based bus operation. In high frequency services, buses are operated on the basis of the scheduled time headway between buses rather than the scheduled timetable. As there is no published timetable at bus stops, passengers arrive at bus stops without knowing when a bus will arrive (i.e. passengers do not arrive for a specific bus). Hence, the bus headway influences the number of passengers arriving for a bus and the waiting time of these passengers. Assuming a uniform rate of passenger arrival, the average waiting time of the passengers for all buses on a route can be calculated as (McLeod, 1998):

$$\textit{Average waiting time of passenger} = \left(\sum H_i^2\right) / \left(2 \times \sum H_i\right) \qquad (4)$$

where,
H_i is the time headway for the bus i

The average waiting time of a passenger is the average actual wait (AWT) of the passengers (TfL, 2010). The average time passengers would wait if the service ran exactly as scheduled is the average scheduled waiting time (SWT). The difference between the average scheduled waiting time (SWT) and the average actual wait time (AWT) is the average excess waiting time (EWT), which shows the additional time that a passenger has to wait above the expected waiting time (TfL, 2010). Thus, the excess waiting time, which is a key measure of bus reliability on high-frequency routes (Gardner *et al.*, 2006) is calculated through the expression:

$$\textit{Excess waiting time} = \textit{Average waiting time} - \textit{Scheduled waiting time} \qquad (5)$$

4.3.6 Dwell time

A further key modelling issue of a bus stop and extremely significant part of the overall journey time is the dwell time, the amount of time spent by buses at the bus stop during the boarding and alighting of passengers (York, 1993). The dwell time may be expressed in relation to the alighting and boarding times and the dead time, which is a fixed time for each bus relating to the time taken to open and close the bus doors and check the traffic. The dwell time is hence expressed by (York, 1993):

Dwell time for buses with single door: $T = D + A\alpha + \sum B_i b_i$ *(6)*

Dwell time for buses with two doors: $T = D_b + \sum B_i b_i$ *(7a)*

or $T = D_\alpha + A\alpha$ *(7b),*

whichever is greater.

Where T is the dwell time, A the alighting time per passenger, B_i the boarding time per passenger, D and D_α, D_b the 'dead times' for one door buses and two door buses respectively, α the number of passengers alighting, b_i the number of passengers boarding using the *ith* process of boarding, where there are m different processes of boarding.

The values of the variables in these equations are all dependent on the type of the bus and the ticket paying and they have been obtained from previous field survey (Shrestha, 2003). Other formulas, such as the one relating the bus dwell time to the number of passengers waiting at the bus stop (Liu et al, 1999), may not incorporate the effect of the number of passengers alighting or the variation in size or type of buses. Therefore, York's expressions are used for the calculation of the bus dwell time for this research.

4.3.7 Incidents parameters

4.3.7.1 Introduction

The role of incidents affecting bus operations is key to this research. The literature review, carried out in Chapter 2 states that in the field of public transportation, 'incident' is the term used for any non-recurring event that causes a roadway capacity reduction or abnormal demand increase. As a wide range of events can be classified as an 'incident', it is necessary therefore to categorise them into groups according to their characteristics in order to incorporate them into the modelling methodology.

In the field of incident management, incidents are grouped according to their frequency, size, severity, impacts, duration and location (Farradyne, 2000). For the purposes of this work, incidents are identified by 5 main characteristics, as shown in Figure 4.1:

Figure 4.1: Identifying characteristics of incidents.

There is a wide range of incidents which affect bus operations and a number of bus performance parameters used to quantify these effects, which are often experienced by bus operators:

> *'We report lost mileage, so if a bus is unable to complete its journey, then the reason for that would be recorded. Now we'd split that into controllable and non-controllable. So controllable lost mileage if you like, is mainly due to not having a driver or a driver is sick or if the bus breaks down. But then there are a lot of reasons outside our control, in which we include the smallest problems such as roadworks, bad parking, general traffic congestion etc.'*

The vital importance of the parameters presented in Figure 4.1 for the model development, as well as the fact that incorporating these incidents characteristics into the model is a key element of this research, requires an in-depth analysis of their contribution to the modelling methodology and all the aspects around it. These parameters are integrated and modelled during the model development via an input-dialog box which describes the options for each of the incidents' characteristics. The dialog box is shown and described in Chapter 5.

4.3.7.2 Starting time

The time that an incident initializes is one of the unique characteristics of the incident and, thus, should be integrated during the model development. Simulation modelling enables the user to set the time parameter according to his preferences, which offers the possibility to simulate for as long as is required. The methodology of SIFUBEM is also based on this concept; the time is the most vital parameter in modelling and the model provides the user with the option to define any required starting time for the incident.

A significant element of the model is the input-dialog box, which includes, among other incident related information, a box in which the user may specify the incident starting time. The starting time of the incident sets the initialization of the incident. At this point the effect of the event starts taking place and the normal operation suffers the impacts of it. Once the starting time has been decided, the next parameters have to be defined: the type, the location, the severity and the duration of the incident.

4.3.7.3 Type

According to information collected through talks between the author and operators of bus companies such as 'London United', in London, 'First Bus', in Southampton and 'Brighton and Hove Buses' in Hove, UK, there are numerous events reported and classified as bus and traffic incidents, such as bus breakdowns, road works, burst water main, traffic incidents and diversion (Polyviou, 2007). A technical report of a bus company usually has the form of a data table, as the one used by London United, which is presented in Appendix A. The focus of this particular research is on bus breakdowns and events that reduce the roadway capacity, such as congestion, road works and illegal parking vehicles. The assumption made is that the route is kept unchanged, which means there is no option of diversion.

A bus breakdown enhances a multidimensional scenario to explore, since the fleet size, which is a parameter under constraint during the model development, could be decreased and simultaneously the impact of the incident is likely to affect the rest of the traffic and therefore the efficiency and regularity of the bus operation.

The average age of the bus fleet in Great Britain in 2008 was 8.3 years indicating the high likelihood of an incident such as a breakdown occurring and affecting the normal bus operations. Attempts have been made recently to replace old buses with new ones and increase the reliability of the fleet but bus breakdowns remain a common incident in bus operations:

> *'On this particular route we've just replaced some buses which were built in 1999, so they were 12 years old, with brand new buses; so we had an upgrade in the fleet with buses that are 2-3 months old.'*

On the other hand, any of the incidents causing a decrease in the route capacity, even though they do not affect the fleet size parameter, may have minor or major impact on the bus operation efficiency. Consequently, two types of traffic incidents are identified and modelled for the purposes of this research, bus breakdown incidents and 'other type' events. Both of these categories and their impacts on overall bus performance are modelled in SIBUFEM, so that useful outcome related to the key bus performance measures are provided via the model output.

4.3.7.4 Location

A characteristic which has particular significance in cases of events occurring at specific points of the bus route is the location of occurrence. For the purposes of the model development, two points have been selected as possible locations: (a) the middle of the route and (b) the end of the route. This assumption is supported by the fact that the middle and the end of the route are the areas that would cause the most severe impact to the bus operation, as opposed to an incident occurring at the start of the route. A further justification is that an incident taking place at the origin of the route is much more controllable than any other location, since the origin is the point where buses are generated and actions are more direct and prompt. The parameter of the incident location is decisive for the links to be affected, since any such event may cause disruption of the normal operation of the upstream links.

4.3.7.5 Severity

Once the type, the starting time and the location of the incident have been included into the model, the parameter that expresses the size of the event is to be modelled.

During the model development, the severity of the incident indicates the degree of the impact that the incident has on the route capacity. For the purposes of this research slight and moderate incidents are examined, since a major incident may cause route diversion or total collapse of the operation, which is not within the scope of this research. The slight and moderate incidents are classified in this research work according to the percentage of decrease of the capacity that they cause. The road capacity is defined as the maximum number of vehicles that can pass over a given section of a lane or roadway in one or both directions during a given time period under prevailing environmental roadway and traffic conditions (TCRP, 2003b).

Bus operators use various terms to classify incidents according to their impact on the operation; they are categorised into 'slight' or 'severe', 'minor' or 'major', 'small' or 'big', 'controllable' or 'non-controllable' categories, yet the focus remains on the frequency of their occurrence and their effect on the bus performance. Bus operators, however, believe that the combination of minor incidents may cause very negative impacts on the overall bus operation.

> *'A big incident does not occur very often, once a month maybe. But most days there are lower level ones. You get the big problems, which could be roadworks, a lorry breaking down or protests. You've got those big things, which severely cut down the road width or they do so at a critical place. On every day we deal with lower level issues which are mainly parking. Each vehicle might be there for a couple of minutes but it's the combination of the badly parked vehicles that cause the delays...'*

The severity of an incident, along with its duration, is the main determinant factor for the length of possible car queues that may form as a result of the event. The incorporation of the severity parameter involves a change of the speeds of the upstream links of the route. This deviation from the normal link speeds requires the understanding and implementation of simple deterministic queuing theory to calculate the length of the queues created (Appendix B).

For the calculation of the length of the traffic queues, initially, the capacity and demand values of the modelled bus route are defined. The decrease of the roadway capacity, which is caused by the incident, beyond the demand values causes a cumulative delay to the system, which is represented by the queue length created.

The respective delays caused and queue lengths created by the incident are calculated for each of the four links upstream (from the location of the incident's occurrence), reaching a maximum of 1 kilometre queue length in SIBUFEM. The additional times needed and the discharge times to accommodate the delay caused are then calculated, which enables the calculation of the new link speeds which dynamically change every second in SIBUFEM. Speed factors are inserted in the model to incorporate the change in bus speed caused by the incident delays provided using the queuing theory described and presented in Appendix B.

Once these queues are calculated and the delay caused by them estimated, the decreased link speeds may be defined and the severity of the incident specified. The procedure of using queuing theory and cumulative delays to calculate the value of severity in terms of link speed decrease is analyzed in steps as depicted in Figure 4.2.

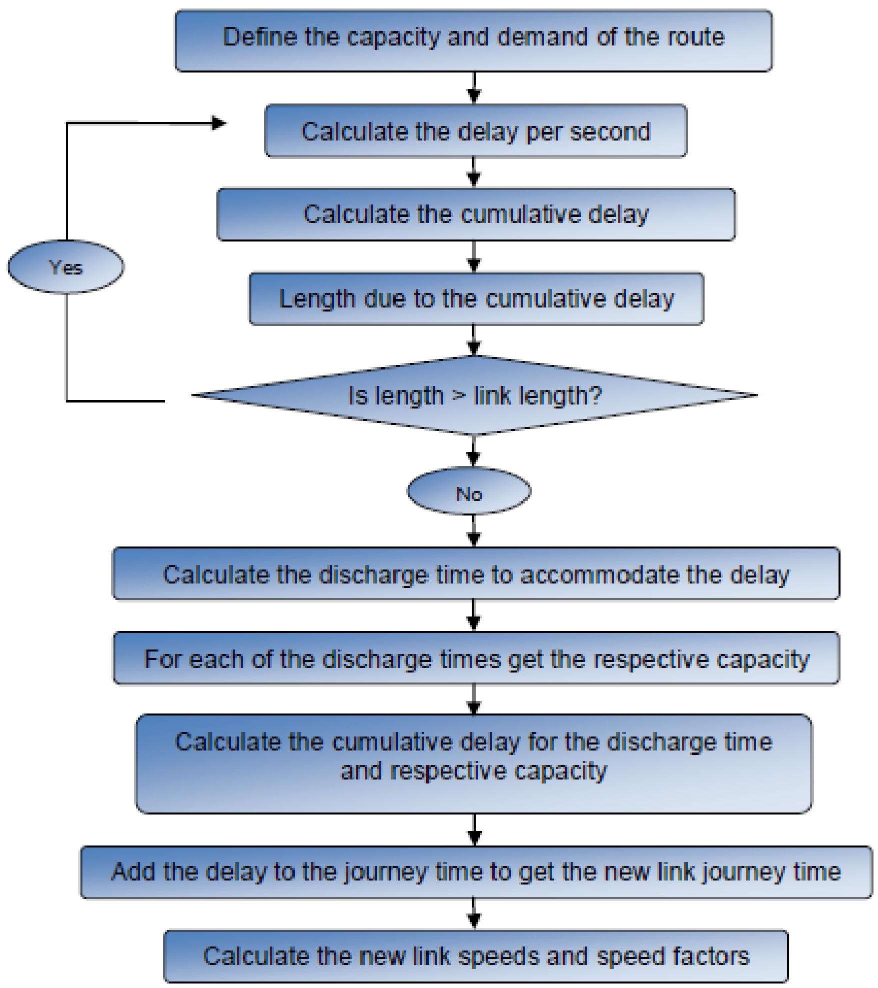


Figure 4.2: Calculation of delay and bus speed decrease due to incident

4.3.7.6 Duration

The impact of a bus or traffic incident is also dependent upon the duration that the incident lasts for. The longer the disruption due to the unexpected event is, the more severe the impact on the normal bus operation will be. For the specific model development, three options of duration time are applied and examined: (a) short, (b) medium and (c) long duration. For each of these options a specific time is assigned. Attention should be drawn to the fact that, although the duration of an incident is the time that the incident takes effect, the time required for the route to re-establish its normal operation characteristics may be considerably longer than that. The recovery time is calculated, as mentioned in section 4.3.7.5, via the use of queuing theory and the cumulative delays calculation.

4.3.8 Slack time

A very significant component of the bus operation is the accommodated slack time. Naturally, transit planners add slack times when making a schedule. The slack time, in the field of bus operations, is the difference between the scheduled and the actual expected travel times (Zhao *et al.,* 2006). The amount of slack can greatly affect the service quality. If the slack time is insufficient, buses are unlikely to run to schedule when they fall behind, thus deteriorating the service reliability (Zhao *et al.,* 2006). On the other hand, an excessive slack time reduces the service frequency, which may also cause inconvenience to passengers. Thus, determining the optimal slack time that minimizes the expected waiting time involves a trade-off between the service reliability and the service frequency (Zhao *et al.,* 2006). Defining the optimum slack time for a bus operation involves queue simulation modelling and comparison to approximation approaches for this purpose. However, this is beyond the scopes of this research, thus, a slack time equal to the 10% of the journey time was chosen to be added at the end of each leg of the round journey. This extra time is sufficient to tackle the regulation of 'late' buses. As for the 'early' buses, the holding option, in which the bus generation is delayed, is responsible for accommodating this issue.

4.3.9 Bus holding option

A further aim of this particular modelling methodology is to incorporate the holding option for 'early' buses. When a bus is available for generation, the model verifies whether it is the appropriate time to generate a bus by checking the last bus generation at this point and comparing it with the predefined 8-minutes headway. In case a bus arrives earlier, the generation is delayed in order to meet this headway requirement. The holding option is responsible, along with the slack time provided, for the regulation of the bus operation and ensuring that the desired headway of the service is met.

4.3.10 Bus bunching

Although transit agencies build slack time into their schedules to alleviate the problem of irregularity in bus arrivals at bus stops, their attempts often fail because practical amounts of slack can not prevent large localized disruptions from spreading system-wide (Daganzo, 2009). As the headways between buses change from the designed value external disturbances of the bus operation are magnified over time until buses travel in pairs instead of evenly spaced; this effect is referred to as bus bunching (Pilachowski, 2009). When a bus is lagging behind the schedule time, it picks up more passengers than average and hence the dwell times are higher, which makes the bus lag even more. The result is that it may cause an accumulation of other buses behind it causing bus bunching. The main cause for bunching comes from the fact that the time a bus spends at a bus stop increases with the number of users that need to board and alight the bus (Pilachowski, 2009). There is no clear evidence that a bus will follow a particular pattern in case of bunching. However, the pattern followed for the model is specific. If the arrival time of a bus approaching the bus stop is later than the departure time of the previous bus, then this bus stops to alight and board passengers. However, if the arrival time is earlier than the departure time of the previous bus, then the bus only stops to alight passengers. Therefore, the model allows the possibility of overtaking buses at bus stops.

4.3.11 Model assumptions

A number of assumptions surround the modelling methodology that was followed for the designing and development of the simulation model SIBUFEM. Most of these assumptions have already been mentioned in previous Sections, but a complete list, along with the reasoning behind them is given below:

- For the purpose of this research linear routes are investigated only, as these are widely used in bus operations.
- A further assumption surrounds the lack of bus lanes throughout the route. The effect of this as well as any other bus priority measures is not a area of investigation and modelling in SIBUFEM; the main focus of this research is modelling traffic incidents to suggest bus fleet management strategies.

- In order to model the running of buses on a circulated basis along the route, the return journey uses the exact same route with 16 identical bus stops on the other side of the corridor and at the same location to the first existing 16 bus stops.
- Assuming that passengers arrive randomly at the bus stops along the examined route, a uniform rate of arrival of passengers is used for the model development of SIBUFEM. The assumption of a random passenger arrival at bus stops is taken for a high frequency service (which is the focus of this research) where the passengers have no expectations concerning the bus arrival time before reaching the bus stop (Seddon and Day, 1974). This assumption is justified by studies carried out on the same route confirming the random passengers' arrival (Rajbhandari, 2002).
- The effect of the traffic signals is taken into account using journey time profiles. However the modelling of the traffic signals is not the focus of this research, therefore the direct interaction with the traffic signals along the route has been replaced by average journey time delays. Furthermore, average link journey time profiles and speed factors reducing the bus speed have been inserted into SIBUFEM, using time dependent deterministic queuing theory, to incorporate the effect of the incidents on the rest of traffic and the consecutive effect of the traffic conditions on the bus performance.
- A 'holding' option has been inserted for the buses in SIBUFEM, suggesting a delay may be allowed to early buses; the assumption here is that this option is only available at the first bus stop of each leg; consequently bus stop 0 and bus stop 16 are the only 'holding' option action points of the route. In SIBUFEM these are the control actions points, as this usually occurs in real conditions when decisions are made by operators at each of the route ends.
- In regards to the location of the occurrence of the bus related incidents it is assumed that the location is that of the referred bus stop. For the purposes of the model development, two points have been selected as possible locations: (a) the middle of the route and (b) the end of the route. This assumption is supported by the fact that the middle and the end of the route are the areas that would cause the most severe impacts to the bus operation, as opposed to an incident occurring at the start of the route. A further justification is that an incident taking place at the origin of the route is more controllable than any other location, since the origin is the point where

buses are generated and actions are more direct and prompt (see Methodology).

- No major incidents were modelled in SIBUFEM as the focus of this research is on slight and moderate events. According to conversations with bus operators, major incidents that may severely affect the bus operation only occur about once a month, while less severe events are dealt with on a daily basis.
- A slack time equivalent to 10% of the round journey time has been inserted at each end of the bus route to accommodate incident delays. This is justified by the fact that bus operators may include up to twice that time as a recovery time to ensure that the agreed level of service is met.
- In SIBUFEM the benefits of the strategy of reinforcing the bus line with an additional bus has been explored. This fleet management action pre-supposes that an extra bus and a driver are available at the bus depot. This assumption is justified by the fact that most of the bus operations constantly keep a number of ‘spare’ buses available for this case. The percentage of the extra buses may be up to 20% of the total bus fleet size, according to evidence provided by bus operators.

4.4 Model Data

A key element to building a model is the data collection phase. This data is required (i) to utilize the modelling methodology, (ii) to show the special characteristics of the chosen route and (iii) to explore the potential of applying the model to various route types. A significant amount of model building data, for the purposes of this work, is available, as the site used for the model development of SIBUFEM has already attracted significant research attention (Shrestha, 2003). The earlier study of the chosen route provided invaluable information and data concerning the route and its parameters. This data was collected from the field and was categorised according to related components. Although, there was a need for just minor alterations to the new model development, the complete description of the data used is required.

4.4.1 Route related data

The parameters of the route used for the model development are defined by the route data, which include the number of bus stops, the amount of links and their characteristics, the number and width of lanes and the capacity and demand parameters of the route. A key issue during the procedure of site and bus service selection is the requirement to meet the model expectations and the research objectives of a linear route, a service with buses running on a circulation basis and a high frequency bus operation.

The site selected for the data use is the Portswood corridor bus route in Southampton. The location of the route in the map of Southampton is depicted in Figure 4.3.

Figure 4.3: Selection of Portswood corridor in Southampton as the model site.

Southampton has a substantial population and a key role as a regional centre. Historically it has strong levels of bus use, although this has changed little overall for many years (Richardson, 2010).The specific bus route extends from the area of Swaythling to the city centre of Southampton and return, covering 8.64 kilometres. The first leg of the route consists of 16 bus stops, keeping buses moving in the Portswood corridor towards the city centre. Since the circulation of the route is vital for this research, an assumption made at this stage is that the return journey uses the exact same route with another 16 bus stops on the other side corridor and at the same location to the first 16 bus stops.

In order to keep the route as realistic as possible and justify this assumption, the information used for the return trip bus stops meet the real data used for the first 16 bus stops. The effect of the traffic signals is taken into account using journey time profiles. However the modelling of the traffic signals is not the focus of this research, thus, the direct interaction with the traffic signals of the route has been replaced by average journey time delays. A further assumption surrounds the lack of bus lanes in the route. The effect of this or any other bus priority measures is not a matter of investigation and modelling during the model development of SIBUFEM; the main focus of this research is modelling traffic incidents to support bus fleet management. In this respect, SIBUFEM has significantly built on the previously developed simulation model SIMBOL (Shrestha, 2003), exploring a different function of bus operations, the bus fleet management.

The route is part of the bus route covered by the service 7/7A (which is operated by 'FIRST' bus company), which was prior to 12th of March 2006 named 11/11A. The line route has been recently altered, with the route modelled in SIBUFEM being covered by a combination of lines of the specific bus operator. Although the alighting patterns have consequently changed since then, the data being used for the purposes of the model development throughout this research are based on real conditions and data collected in previous studies (Shrestha, 2003). Validating the model with updated route data, after carrying out a series of data collection surveys and discussions with bus operators of 'FIRST', and comparing the model outcome against the route data used in SIBUFEM would be an issue for further consideration. However, for the purposes of the model development of SIBUFEM existing real route data of a busy Southampton route was used, setting this route as a study case and enabling the future use of the model in other bus routes.

The data including the details of the route, including but not limited to the length of the links and the position of the bus stops in the route, is shown in Table 4.1. The return leg of the route consists of bus stops set at the exact same location to the respective bus stops of the first leg, situated at the opposite side of the corridor; therefore, the return leg bus stops use the same identifying names to the first leg as depicted in Table 4.1.

Table 4.1: Details of the bus route used for the model

Bus stop ID	Bus stop name	Distance (m)	Cum. Distance (m)	Link Number
0	Stoneham road	0.0	0.0	0
1	McDonald restaurant	150.0	150.0	1
2	Woodmill road	290.0	440.0	2
3	Mayfield road	280.0	720.0	3
4	Sirdar road	180.0	900.0	4
5	Somerset road	350.0	1250.0	5
6	Bus depot	360.0	1610.0	6
7	Somerfield	190.0	1800.0	7
8	Safeway	260.0	2060.0	8
9	Spring crescent	350.0	2410.0	9
10	Cedar road	210.0	2620.0	10
11	Stage gate	370.0	2990.0	11
12	Middle street	220.0	3210.0	12
13	Law court	280.0	3490.0	13
14	Cenotaph	520.0	4010.0	14
15	Marland centre	310.0	4320.0	15
16	Marland centre	0.0	4320.0	16
17	Cenotaph	310.0	4630.0	17
18	Law court	520.0	5150.0	18
19	Middle street	280.0	5430.0	19
20	Stage gate	220.0	5650.0	20
21	Cedar road	370.0	6020.0	21
22	Spring crescent	210.0	6230.0	22
23	Safeway	350.0	6580.0	23
24	Somerfield	260.0	6840.0	24
25	Bus depot	190.0	7030.0	25
26	Somerset road	360.0	7390.0	26
27	Sirdar road	350.0	7740.0	27
28	Mayfield road	180.0	7920.0	28
29	Woodmill road	280.0	8200.0	29
30	McDonald restaurant	290.0	8490.0	30
31	Stoneham road	150.0	8640.0	31

4.4.2 Bus related data

The bus related data describes the characteristics of the buses in the system examined and includes the time of bus generation, the origin and destination of the buses, their type and capacity and the average link speed for each of the links of the route. The main source of the model building bus related data is the bus generation and the average link speed, hence, they are described in detail.

4.4.2.1 Bus fleet size and bus generation

The parameter of bus fleet size is an important modelling element of the specific research, since this is the factor to determine the amount of buses available for the system. The amount of buses required for the normal circulated operation of the bus service has been calculated with the use of the service frequency and the average bus round trip journey time. The service frequency used is the 8-minutes headway of bus generation and the average round trip journey time is provided by existing data on the route (Shrestha, 2003). The average bus journey time is calculated to be approximately 3300 seconds, which includes the slack time added at each end to accommodate the late buses and regulate the system. For the purposes of this research, 7 buses are required for the normal operation of the service.

According to Section 4.3.2, the buses are generated following an 8-minutes headway service, which means that the time between two consecutive bus generations is approximately 8 minutes, since a 1 minute error is also included to take into account the variation due to various factors.

The bus generation times used for the amount of buses generated equal to the fleet size of the service are based on the normal distribution used. Once the fleet size (i.e. 7 buses) of the system is exceeded, the next bus is generated at the base of 8-minutes headway only if the bus availability criterion at the generation point is met. For the purposes of the model development, the generation of the buses takes place at bus stop 0.

4.4.2.2 Average link speed

The average speed of a link, which describes the space mean speed of each section of the route between two consecutive bus stops, is based on available data on the modelled route (Shrestha, 2003), and excludes any stoppage time at a bus stop or traffic signal. Since the link characteristics used for the return leg of the route were identical to the opposite links which are part of the first leg, the average link speeds used for the return leg are exactly the same as described on Table 4.2.

Table 4.2: Average link speed of buses along the route.

Bus stop ID		Name	Distance (m)	Speed (km/hr)
1st leg:	0	Stoneham road	0	-
	1	McDonald restaurant	150	5.90
	2	Woodmill road	290	16.47
	3	Mayfield road	280	27.15
	4	Sirdar road	180	25.35
	5	Somerset road	350	18.30
	6	Bus depot	360	28.34
	7	Somerfield	190	23.81
	8	Safeway	260	11.96
	9	Spring crescent	350	35.00
	10	Cedar road	210	9.97
	11	Stage gate	370	17.19
	12	Middle street	220	29.77
	13	Law court	280	22.86
	14	Cenotaph	520	12.99
	15	Marland centre	310	11.60
2nd leg:	16	Marland centre	-	-
	17	Cenotaph	310	11.60
	18	Law court	520	12.99
	19	Middle street	280	22.86
	20	Stage gate	220	29.77
	21	Cedar road	370	17.19
	22	Spring crescent	210	9.97
	23	Safeway	350	35.00
	24	Somerfield	260	11.96
	25	Bus depot	190	23.81
	26	Somerset road	360	28.34
	27	Sirdar road	350	18.30
	28	Mayfield road	180	25.35
	29	Woodmill road	280	27.15
	30	McDonald restaurant	290	16.47
	31	Stoneham road	150	5.9

4.4.3 Bus stop related data

Apart from the identifying characteristics of the route and the buses running along it, the parameters of each bus stop, a crucial modelling component of this research, need to be specified. The bus stop related data consists of the position of the bus stops, the starting times of the passenger generation, the boarding and alighting rates and the dwell times. Key procedures of the model take place at the bus stops, including but not limited to the bus generation, the passengers' interaction, the incidents occurrence and the suggested control options, setting bus stop as the most vital component of this research work.

4.4.3.1 Boarding and alighting passenger rate

Assuming that the passengers arrive randomly at the bus stops of the examined route, a uniform rate of arrival of passengers is used for the model development of SIBUFEM. The assumption is based on a study carried out in the same route concluding to the random passenger arrival (Rajbhandari, 2002). The rate may be higher in peak period than in off-peak. However, since the data used in SIBUFEM refers to an off-peak period, the variation of the rate is not significant and a uniform rate is used to estimate the boarding passengers in the model.

For the purposes of the model development the alighting and boarding passenger rates used are listed in Table 4.3, with the rates used for the return leg being identical to the respective rates of the opposite bus stops of the first leg of the route. Even though a uniform passenger arrival rate has been assigned for each bus stop, this rate is unique for each bus stop and reflects to real field data. Furthermore, in order to avoid unrealistically high number of passengers, the starting time of the passengers' generation was unique for each bus stop as described in following Section.

Table 4.3: Alighting and boarding passenger rates used in model (Shrestha, 2003).

Bus stop ID		Name	Alighting ratio	Boarding rate (s/pass)
1st leg:	0	Stoneham road	0.070	90.0
	1	McDonald restaurant	0.026	72.0
	2	Woodmill road	0.010	257.3
	3	Mayfield road	0.010	129.1
	4	Sirdar road	0.008	240.0
	5	Somerset road	0.008	276.9
	6	Bus depot	0.008	720.0
	7	Somerfield	0.305	69.9
	8	Safeway	0.048	141.2
	9	Spring crescent	0.014	171.9
	10	Cedar road	0.030	225.0
	11	Stage gate	0.038	180.0
	12	Middle street	0.056	1800.0
	13	Law court	0.226	1200.0
	14	Cenotaph	0.470	3600.0
	15	Marland centre	1.000	9999.0
2nd leg:	16	Marland centre	0.070	90.0
	17	Cenotaph	0.026	72.0
	18	Law court	0.010	257.3
	19	Middle street	0.010	129.1
	20	Stage gate	0.008	240.0
	21	Cedar road	0.008	276.9
	22	Spring crescent	0.008	720.0
	23	Safeway	0.305	69.9
	24	Somerfield	0.048	141.2
	25	Bus depot	0.014	171.9
	26	Somerset road	0.030	225.0
	27	Sirdar road	0.038	180.0
	28	Mayfield road	0.056	1800.0
	29	Woodmill road	0.226	1200.0
	30	McDonald restaurant	0.470	3600.0
	31	Stoneham road	1.000	9999.0

The boarding and alighting rates were obtained using calculations on existing survey data collected for a previous study on the same route (Shrestha, 2003) and adjusting the passenger numbers for the bus headway of 8 minutes. The alighting ratio expresses the percentage of passengers alighting at a bus stop out of the total passengers already inside the bus, while the boarding rate is the average time headway between passengers boarding.

The arrival rate used to estimate the boarding passengers, which is expressed by the ‘boarding rate’ heading in Table 4.3, was presented in the form of time gaps in order to generate a passenger for the model input. For the model development of SIMBOL a 2 hours (i.e. 7200 seconds) period of data collection was used, thus, the boarding rate, calculated in seconds per passenger, is expressed by the equation:

$$Boarding\ rate = 7200 / total\ passenger\ number \qquad (8)$$

In the case of alighting passengers, though, the rate was expressed in terms of a percentage of passengers already inside the bus. Therefore, the equation describing the alighting ratio used in SIBUFEM is:

$$Alighting\ ratio = alighting\ passengers / passengers\ inside \qquad (9)$$

4.4.3.2 Dwell time

A suitable dwell time equation needs to be used to calculate the dwell time for the estimated number of passengers alighting and boarding. Introducing the alighting and boarding rates into the model enables the estimation of the dwell time according to the dwell time parameters described in Section 4.3.6 and in particular by expression 7a for buses with two doors. Most of the buses (approximately 85%) serving the modelled route were double deck buses. According to previous research on the route (Shrestha, 2003), the dwell time expression for the double deck buses of the bus service 7/7A examined is:

$$T = 6.85 + 1.69 \times \alpha + 9.00 \times b \qquad (10)$$

The calculation of the coefficients of the above dwell time equation, which was also discussed in Section 4.3.6, was carried out using multiple linear regression (Shrestha, 2003). The above relationship, which is included in the model, provides the total dwell time ‘*T*’ in seconds in terms of numbers of alighting and boarding passengers, represented by ‘α’ and ‘b’ respectively.

The field parameters used for the above expression have been tested in comparison to the dwell time parameters calculated via a similar method in a London study (York, 1993) and their compatibility was confirmed by statistically testing the dwell time parameters using t-test for paired data (Shrestha, 2003). The latter enables this research work to use the above dwell time equation as the reasonable and realistic calculation of the dwell time.

4.4.3.3 Starting time of passenger generation

The parameter to decide upon the passenger numbers for the first bus arriving at each bus stop is the starting time of passenger generation. Hence, an accurate and realistic estimation of the passenger generation starting time is required for the model development. The time of passenger generation initialization affects directly the dwell time of the buses, at least the first few.

A further key issue is the fact that passengers should not be generated simultaneously at all bus stops or else the first bus reaching the bus stops will have to carry an unrealistically high number of passengers. In order to take into account these significant issues, the starting time of the first bus stop is based on the bus headway interval and the average deviation of the buses for the service. Then the start time for the rest of the bus stops of the route is calculated using the average travel time taken by the buses to arrive at each bus stop. The details of the starting times of passenger generation for each of the bus stops of the route are summarized in Table 4.4.

Table 4.4: Starting times of passenger generation for each bus stop.

Bus stop ID		Name	Passenger generation starting time (s)
1st leg:	0	Stoneham road	-433
	1	McDonald restaurant	-286
	2	Woodmill road	-154
	3	Mayfield road	-92
	4	Sirdar road	-25
	5	Somerset road	69
	6	Bus depot	138
	7	Somerfield	171
	8	Safeway	343
	9	Spring crescent	420
	10	Cedar road	529
	11	Stage gate	635
	12	Middle street	695
	13	Law court	752
	14	Cenotaph	919
	15	Marland centre	1045
2nd leg:	16	Marland centre	1144
	17	Cenotaph	1271
	18	Law court	1437
	19	Middle street	1494
	20	Stage gate	1555
	21	Cedar road	1660
	22	Spring crescent	1770
	23	Safeway	1847
	24	Somerfield	2009
	25	Bus depot	2052
	26	Somerset road	2120
	27	Sirdar road	2214
	28	Mayfield road	2282
	29	Woodmill road	2343
	30	McDonald restaurant	2476
	31	Stoneham road	2624

4.4.4 Incident related data

There are 5 main parameters characterizing the nature of the incident for the needs of this research as described in Section 4.3.7. The starting time is the parameter which defines the time of the occurrence and decides upon which of the buses of the system are affected. Using various starting times as input so as to estimate the impact of this parameter is one of the potential capabilities of the model.

The types of the incident to be modelled are: (a) bus breakdown and (b) other type incidents. The nature of a bus breakdown and the consequent decrease of the fleet size, which is a crucial modelling parameter of this work, require the partition of the incidents into these two groups. There are two possible options for the next incident parameter, the location of the event. For the purposes of this research, two locations are examined, as described in Section 4.3.7.3: (a) the middle of the route, which refers to a random bus stop at the middle of the route length and, thus, bus stop 6 is chosen and (b) the end of the route, which refers to the last bus stop of the first leg, the bus stop 15. It should be emphasized that the assumption concerning the location of the occurrence of an incident is that it meets the location of the referred bus stops.

The impact of an incident is dependent on its size; hence, the severity factor of the incident is crucial for the effects on the bus operation. The aim of the model is to investigate slight and moderate bus and traffic incidents and their impact; under this category fall all the incidents that cause capacity reduction of up to 40%. For the purposes of the model, a slight bus or traffic incident causes 25% capacity decrease, while a moderate incident reflects a 40% reduction. These percentages are depicted on the journey time profiles through the bus speed decrease. The link speed factor used to describe the bus speed reduction is expressed by the relationship:

Link speed factor = reduced speed due to incident / normal link speed (11)

The link speed factor is a number which varies on the range between 0 and 1, where a factor equal to 1 refers to the case that the incident causes no impact to the link speed. In other words, the smaller the speed factor is, the higher the effect of the incident, since the decrease of the speed is more considerable. The speed factors are calculated in excel file format for all the possible combinations of location, duration and severity and then used via text file format into the model. They are representative of the incident effect on each link upstream to the location where the incident occurs. The file responsible for the calculation of the link speed factors is based on the methodology of Figure 4.2 and it is given in Appendix B; due to the size of the file (based on time intervals of one second for 12 hours simulation period) only a part of the file (30 seconds simulation time) is shown. Figure 4.4 demonstrates an example of how the speed factors used for each of the 4 upstream links vary according to the simulation time for the scenario of a medium severity incident taking place for 1 hour at the middle of the bus route:

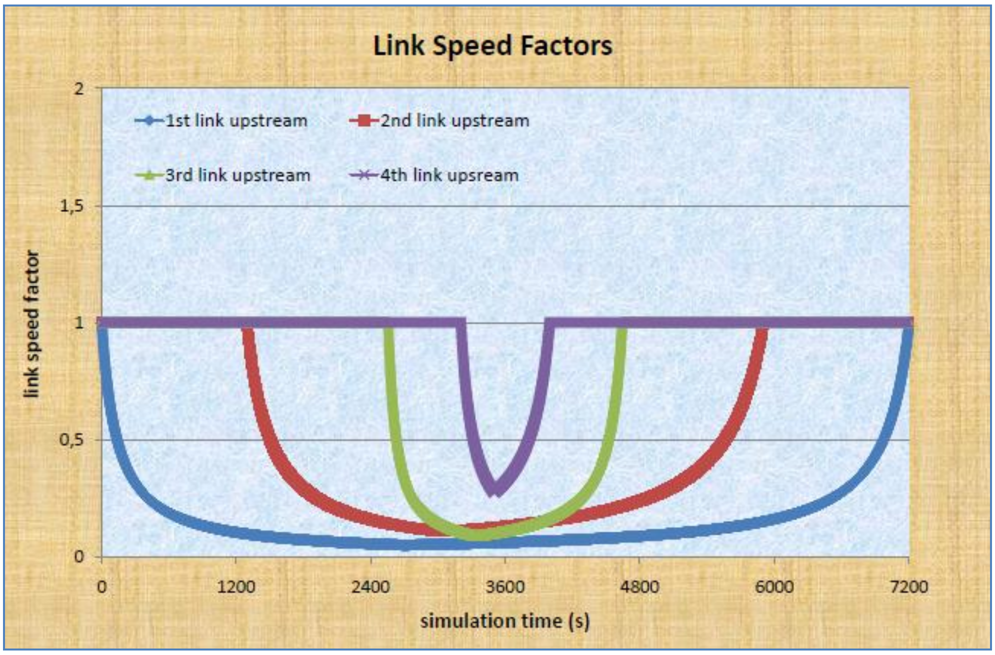


Figure 4.4: Link speed factors incorporating incidents impact[1].

Section 4.3.7.5 described the process of calculation of the link speed decrease using deterministic queuing theory and cumulative delays. Using the expression 11 above, the link speed factors for each of the upstream links are calculated. In Figure 4.4 the behaviour of these factors in relation to the simulation time is shown. The second link upstream is affected only once the length of the traffic queue exceeds the length of the first link upstream; then again a queue length which is longer than the length of the first two links upstream will initialize the effect on the speed factor of the third link upstream and so forth. The curves in Figure 4.4 describe the change of the link speed, and consequently the effect on the bus speed for the specific links, during the 12 hours simulation time.

[1] The link speed factors are used as built-in input data in SIBUFEM and equation 11 is used for their calculation.

4.5 Model Expectations

A number of performance parameters are required to produce efficient and complete conclusions on the various impacts of bus and traffic incidents on the overall performance of a high frequency bus service. The part of the model responsible for the output generation is of vital significance and should satisfy each of the model expectations and more widely the research objectives set. Hence, the comprehensive and accurate definition of the model expectations is essential.

4.5.1 Expectations

A number of expectations are identified in order to meet the research objectives, as these described in section 1.2. The expected outcome includes a wide range of parameters and components, offering a multidimensional view of the model results and they achieve to:

- Show visually the progression of each bus along the route;
- Check the deviation from the bus headway of the service;
- Calculate the total journey time;
- Provide the number of passengers boarding and alighting at bus stops;
- Calculate the waiting time;
- Calculate the total dwell time at bus stops;
- Provide the total passenger journey time;
- Calculate the bus delays due to various incidents;
- Incorporate the impacts of various incidents on the bus service;
- Calculate the effect of incidents according to key performance parameters in order to suggest fleet management strategies to address each problem;

4.5.2 Performance parameters and model output

Investigating the impact of bus and traffic incidents on the service level of bus operations involves defining the key performance parameters of the simulation model. The decision over these parameters is of vital importance for the accurate evaluation of the results provided by the model. For the purposes of this research, the performance parameters chosen are the average bus round-trip journey time, the average excess waiting time and the average bus speed. The choice of these measures was based on the fact that transit managers, schedulers, decision makers, metropolitan planning organisations and the public are all interested in these performance parameters, as these significantly influence decision making and public transport use.

Any change on one of these three parameters indicates a change on the service standards of the bus operation. The fact that the simulation period used for the accurate and realistic estimation of model results is 12 hours, which fulfils the requirements of providing realistic journey time profiles and offering the system the opportunity to exceed the overall effects caused by incidents and recover its normal service standards, should be underlined.

An idea for future enhancement of the model development would be to include performance indicators to reflect the loss of bus patronage related to the specific incident induced delay. This could be achieved by inserting the parameter of elasticity of travel volume with respect to travel time. The travel time is directly linked to the time delays caused by bus related incidents. At the current stage of model development, though, bus patronage loss was not incorporated in SIBUFEM.

The layover time is another parameter considered to be important and is regularly used by bus operators to make instant decisions regarding the performance of each bus individually:

> *‘We have the information about the delay and the layover which is 10 seconds, in other words if he stays no later than that for the rest of his journey he’ll be able to leave on time to come back to the other end.’*

Once the performance parameters are identified, the model output is produced in form of texts, tables, graphs and visualizations. The text formatted files provide useful results related parameters such as journey time, passengers’ numbers, dwell time, waiting time and bus speed. Tables and graphs are used to illustrate the change of the key performance parameters in respect to simulation time and provide a comparison between the different scenarios modelled and visual displays aim to illustrate the route, the bus progression and the incidents occurring and enhance the complete understanding of the model core.

4.6 Interviewing Bus Operators

As part of the modelling methodology, meetings were held with bus operators to obtain an insight into operators' needs and feedback at different stages of this research. The method of interviewing operators of three different large bus companies was adopted, which offered invaluable qualitative data supporting the development stage of SIBUFEM as well as the applications phase. Meetings with operators of 'FIRST' bus company in Southampton, UK, 'London United' in London, UK, and 'Brighton and Hove Buses' in Hove, UK were organised and carried out ensuring a research methodology approach as representative of real conditions as possible was adopted.

In order to gain a better understanding of the bus fleet management domain, relevant field procedures and to determine the frame of the modelling methodology adopted in this research, meetings with operators within 'FIRST' and 'London United' were held in Southampton, UK and London, UK respectively. Invaluable information and data, such as bus route data, necessary for the model development of SIBUFEM, was obtained through these meetings with operators. Furthermore, interviews with experts in bus operations provided knowledge and information regarding the nature and the characteristics of the incidents that affect overall bus performance.

Through direct liaising with bus operators in Hove, this offered an in-depth understanding of issues related to bus related incidents and bus fleet management procedures. A number of questions regarding the pre-scheduling process of the bus lines, the bus headways determination, the fleet size and the availability of 'spare buses', the incidents' detection, the monitoring of buses along the route, the bus fleet management actions adopted and the impacts of them on the performance of the bus operation, were addressed offering valuable input to this research. Various quotes have been included throughout this PhD Thesis to support the model development of SIBUFEM and highlight the model's functionality and applicability of this research.

4.7 Chapter Summary

Chapter 4 set the outline of the modelling methodology of the research, highlighting the significance of the main modelling components. The route and its characteristics are incorporated into the model, the bus movement is based on average link journey time, the bus stops are responsible for the generation of the passengers, the traffic signals are not modelled as this is not within the scope of this PhD research, the incidents are modelled according to their 5 main parameters and issues like bus holding or bus bunching are taken into account. Model data, including route, bus, bus stop and incident related information, has been presented and the expectations and requirements have been set. The key performance parameters, which are included in the model output, are vital for the evaluation of the outcome of SIBUFEM. The development of the model is based on the methodology described and the data presented.

-Chapter 5-

Model Development

5.1 Introduction

With the literature review completed and the consequent modelling methodology defined, the next step is the model development of the new microscopic simulation model called SIBUFEM (Simulating Incidents for BUs FlEet Management) (Polyviou *et al.* 2010). This Chapter describes the development of SIBUFEM and is based on the methodology and related issues described in previous Sections. Based on this methodology and key parameters, the model is developed with the use of the programming language C++ (Borland Builder Version 6). The complete computer source code of the newly developed simulation model SIBUFEM is given in Appendix C.

C++ is a highly flexible and adaptable programming language and has been used for a wide range of programs including operating systems, applications and graphics programming (Oualline, 2003). C++Builder is an object-oriented, visual programming environment for rapid application development (RAD) (Borland Software Corporation, 2002), (Parsons, 1997). Using C++Builder, enables the programmer to create highly efficient applications for Microsoft Windows XP, Microsoft Windows 2000 and Microsoft Windows 98 with a minimum of manual coding. The characteristics of the programming language C++, some of which are mentioned above, set the choice of this language for developing simulation models for bus operations as a reasonable and practical one. Furthermore, the flexibility of the computer language and the possibility of using categorised objects called 'classes' offer a great advantage to the user and presents C++ as a powerful tool altogether.

The model consists of four main different parts, which in programming language are called modules: the main, the bus, the bus stop and the incidents module. The main module interacts with the rest of the modules and is responsible for steering the overall simulation. It is the most significant part of the model and should be regarded as its 'brain'. A diagrammatic description of SIBUFEM (full source code in Appendix C), which was developed for modelling traffic incidents to support bus fleet management, is shown in Figure 5.1.

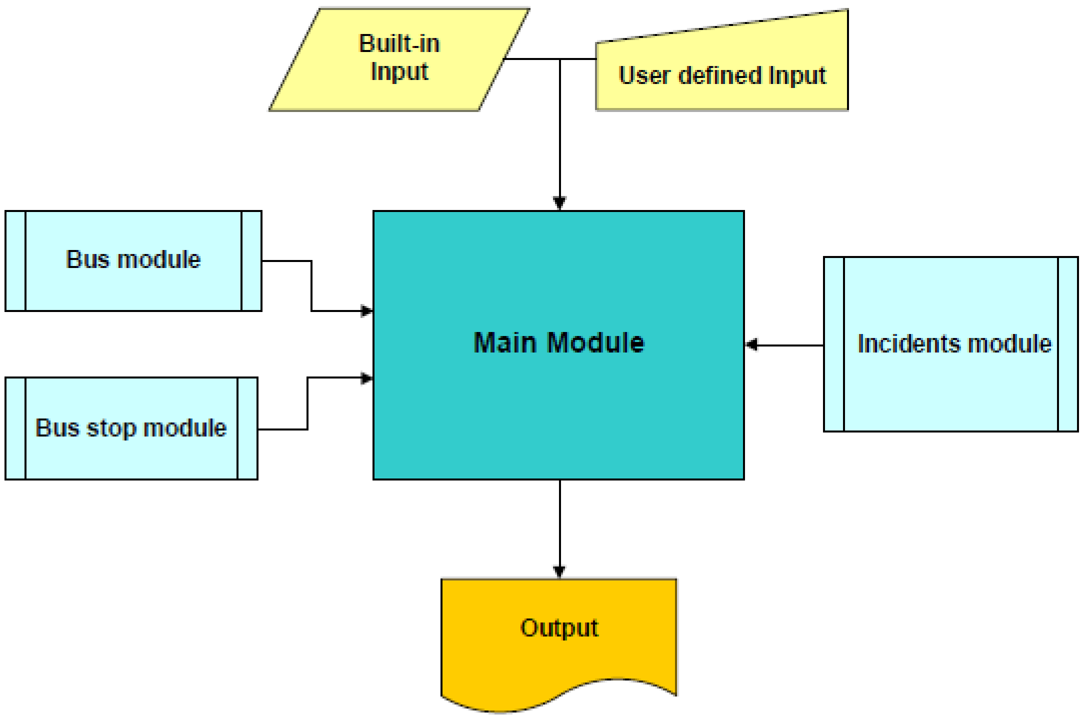


Figure 5.1: Diagrammatic description of the simulation model SIBUFEM

The main module receives the input parameters, defined via built-in and user defined forms, for all the modules designed to perform certain specific tasks as their name suggests. At every change in simulation time, the interaction between the main module and the rest of the modules activates or deactivates these tasks and a useful output based on the outcome of the tasks is produced both visually and in text form (Polyviou *et al.,* 2011).

5.2 Main Module

The main module is responsible for the smooth operation of the model. Taking input, coordinating the interaction between all modules and generating output are the main tasks undertaken by the main module. The main module receives input parameters for all the modules and coordinates their process. The main function of the main module, besides the adjustment of the interaction between modules, is to manage the simulation time and generate the buses of the service. The input passed to the main module, the working of the module and the output provided are described analytically in the following Sections.

5.2.1 Model input

The input of the model is defined by the main module and is separated into built-in input and user defined input. Parameters related to the bus service, the route and their components belong to the first category, thus, they are modelled inside the main module as built-in input. These parameters are defined during the model development and can not be altered for the various scenarios modelled. On the other hand, parameters which can be modified by the user during the start of the simulation are modelled as user defined input for the model. The simulation period, the simulation speed and the identifying characteristics of the bus and traffic incidents occurring are the user defined input of SIBUFEM; hence these parameters can be modified by the user and they are passed to the main module while starting a simulation run.

While built-in input is modelled inside the main module and is not defined by the user, the second category of input is dependent on the preferences of the user and therefore an input-dialog box is required for this purpose. The input-dialog box, as illustrated in Figure 5.2, receives the choices of the user related to the simulation period and speed and the characteristics of the incident. Once the above parameters are defined, the user is asked to choose between button 'READY!' which proceeds to the simulation run and button 'cancel' which exits the simulation.

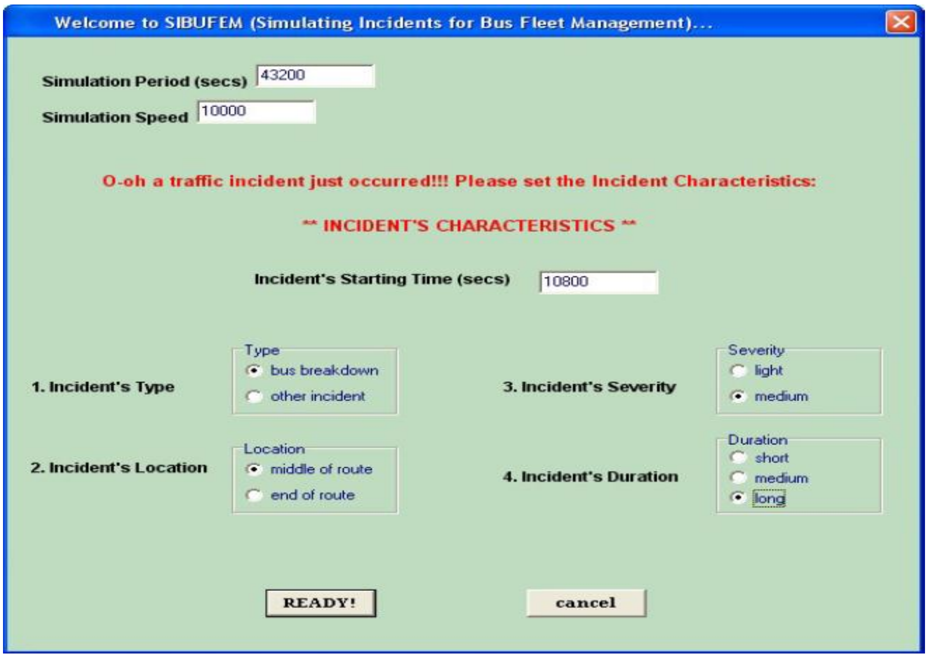


Figure 5.2: Dialog-box for the user defined incidents-related input in SIBUFEM (Polyviou et al., 2011)

In Figure 5.2, simulation period is set on 12 hours time and the starting time of the occurring incident is 3 hours after the simulation initialization; the input-dialog box illustrated in this case represents the modelling of a bus breakdown scenario of a medium severity level which occurs at the middle of the route and lasts for long time duration. The definitions and values for these individual incidents' characteristics are provided in Chapter 6, where the model application is described.

5.2.2 Working of the module

At the start of the simulation, the module receives the input described in previous Section and passes all the information for the different modules to their respective module. The model has a fixed time scanning interval of one second, represents a route with defined number of bus stops and provides the generation of buses in controlled intervals. Every second time is checked inside the main module so as to generate a bus if the predefined generation time is reached. Every generated bus' actual position and GPS position is updated in order to be checked with the bus stop and the incident modules to track its presence there. A visual representation of these activities is displayed as an output, at the same time that the rest output in text form is produced. The final output of the main module should give parameters such as total passenger waiting time, passenger journey time, bus journey time, bus occupancy, bus delays, which is used in further analysis to evaluate the impact of various bus and traffic incidents modelled and be capable of suggest control strategies to address the effect.

The total simulation period, since it is part of the user defined input, is dependent on the user preferences. However, two parameters need to be highlighted and taken into account at this point; (a) the model is developed under the scope of integrating a continuous circulation of the buses around the route and (b) one of the key components of the model development is the use of journey time profiles to incorporate the interaction of the buses with the rest traffic. Both of these issues require a simulation time long enough to provide realistic results, thus a minimum time of 12 hours is necessary as simulation period. The simulation time is updated every second and the model keeps 'running' until the simulation period is completed.

The bus generation follows the modelling methodology described in Section 4.3.2, is constantly checked with the simulation time and is one of the main functions of the main module. Until the number of buses generated becomes equal to the bus fleet size of the service modelled, a bus is generated whenever the simulation time reaches the predefined bus generation times. For the rest of the buses, the generation is based on the headway predefined for the service. The research focuses on high frequency bus services, which usually refers to services with five or more buses per hour (TfL, 2010) and is generally regarded as one where passengers turn up randomly at the bus stops. For this reason bus operators are more interested in how reliable and evenly spaced the service is, rather than the time a bus was due to arrive at a bus stop (TfL, 2010). For the purposes of this research, buses arrive at bus stops every 8 minutes, as defined in section 4.3.2, and the generation takes place at the origin of the route, which is bus stop 0. A range of one minute is provided for headway in order to take several factors, such as weather conditions or bus condition, into account. An important detail of the bus generation procedure is the bus availability constraint, which means that a bus is only generated if a bus is available at the generation point. If a bus is not available, then the bus generation time is delayed; this constraint is applied both at the beginning of the first leg of the journey (i.e. bus stop 0) and at the start of the return leg of the bus trip (i.e. bus stop 16). In each of the cases time headways of the buses are checked and compared to the scheduled time headway of 8 minutes, taking also account of the one minute variation and, thus, delayed if required. Each bus generated is assigned a unique identification number, which relates to its functional parameters such as size, type and passenger capacity, and offers the model user the opportunity of keeping a track of the movement of each bus.

These two functions, the simulation time managing and the bus generating, are two of the main tasks of the main module and of vital importance for the model development. The key responsibility of the module, however, is ensuring the interaction between modules and the smooth operation of the model. The flowchart describing the working of the main module, as well as the interaction with the rest modules, is shown in Figure 5.3.

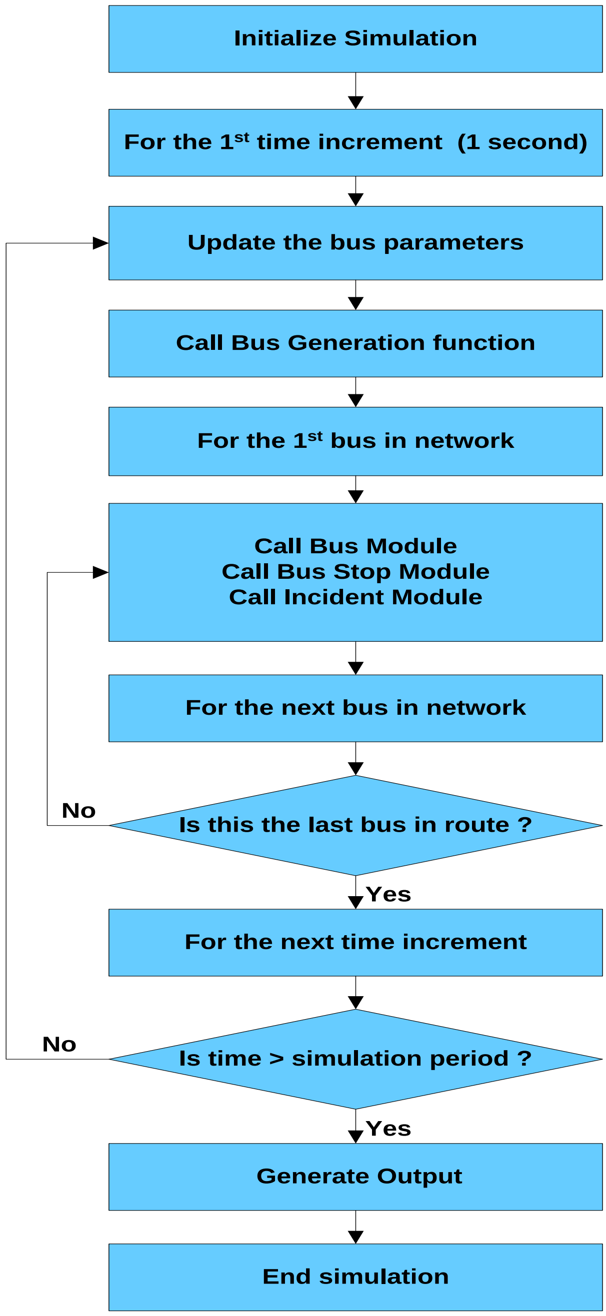


Figure 5.3: Flowchart of working of the main module of SIBUFEM

5.3 Bus Module

According to the flowchart illustrated in Figure 5.3, the first module to be called by the main module is the bus module. The movement of all buses is modelled by this module. The module keeps a record of all of the buses in the system. At the time of the generation of buses the input parameters such as the identification number of the bus, its origin, destination and capacity are defined by the main module. The bus module then checks (as is depicted diagrammatically in Figure 5.4) whether the bus is moving or not and calculates the distance travelled during the specified time interval of one second.

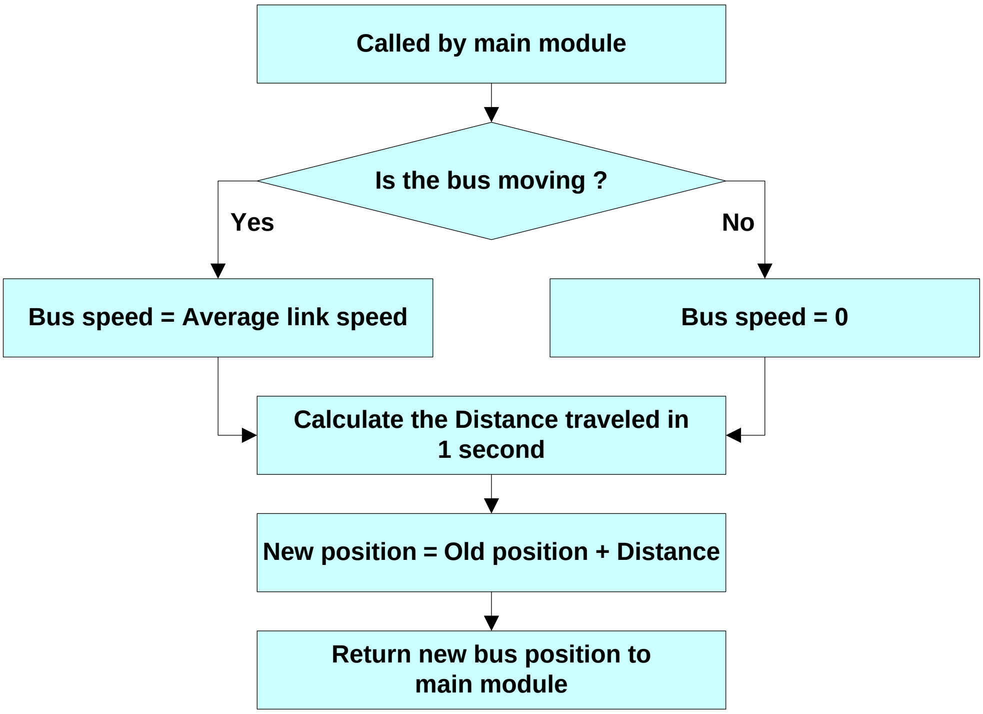


Figure 5.4: Flowchart of the bus module of SIBUFEM

When the bus is moving, the present speed of the bus is assigned as the average link speed according to its location in the route. The position of all buses in the route and their speed are updated every second, offering a detailed observation of the model process and outcome. If the bus reaches a bus stop the interaction between the bus and the bus stop component is modelled by the respective module: the bus stop module.

5.4 Bus Stop Module

The passenger numbers is a vital component of any bus operation and a significant part of any model simulating it. The module responsible for the modelling of the interaction between buses and passengers at the bus stops is the bus stop module. The input parameters required for this module are the bus stop identification number, the position of the bus stop, the percentage of alighting passengers and the rate of boarding.

The main function of the module is to estimate the dwell time of a bus using the number of passengers at a bus stop and calculating the waiting time of the passengers and the lateness of the bus at it's time of departure. When a bus departs a detailed output, including but not limited to the arrival and departure time of the bus, the time gap, the numbers of alighting and boarding passengers, the waiting time of passengers and the dwell time of the bus, are produced.

5.4.1 Working of the module

The bus stop module is called by the main module in order to check whether a bus has reached a bus stop. Once a bus arrives at a bus stop, parameters such as the bus identification number, its arrival time, the service number and the number of passengers inside the bus are recorded. This information is used for the calculation of the number of alighting and boarding passengers, the total dwell time, total passenger waiting time and the journey time. The speed of the bus is at this point changed into zero until the departure time is reached when the speed is changed into the average link speed according to its location in the route. The number of total passengers inside, the passengers alighting and boarding and the dwell time are then calculated. In case of 'holding' option, a delay might be provided to early buses; the assumption here is that this option is only available at the first bus stop of each leg; thus bus stop 0 and bus stop 16 are the only 'holding' option action points of the route. During the delay period all the passengers generated board the bus. The methodology of the bus stop module is illustrated in Figure 5.5 showing each step of the module process.

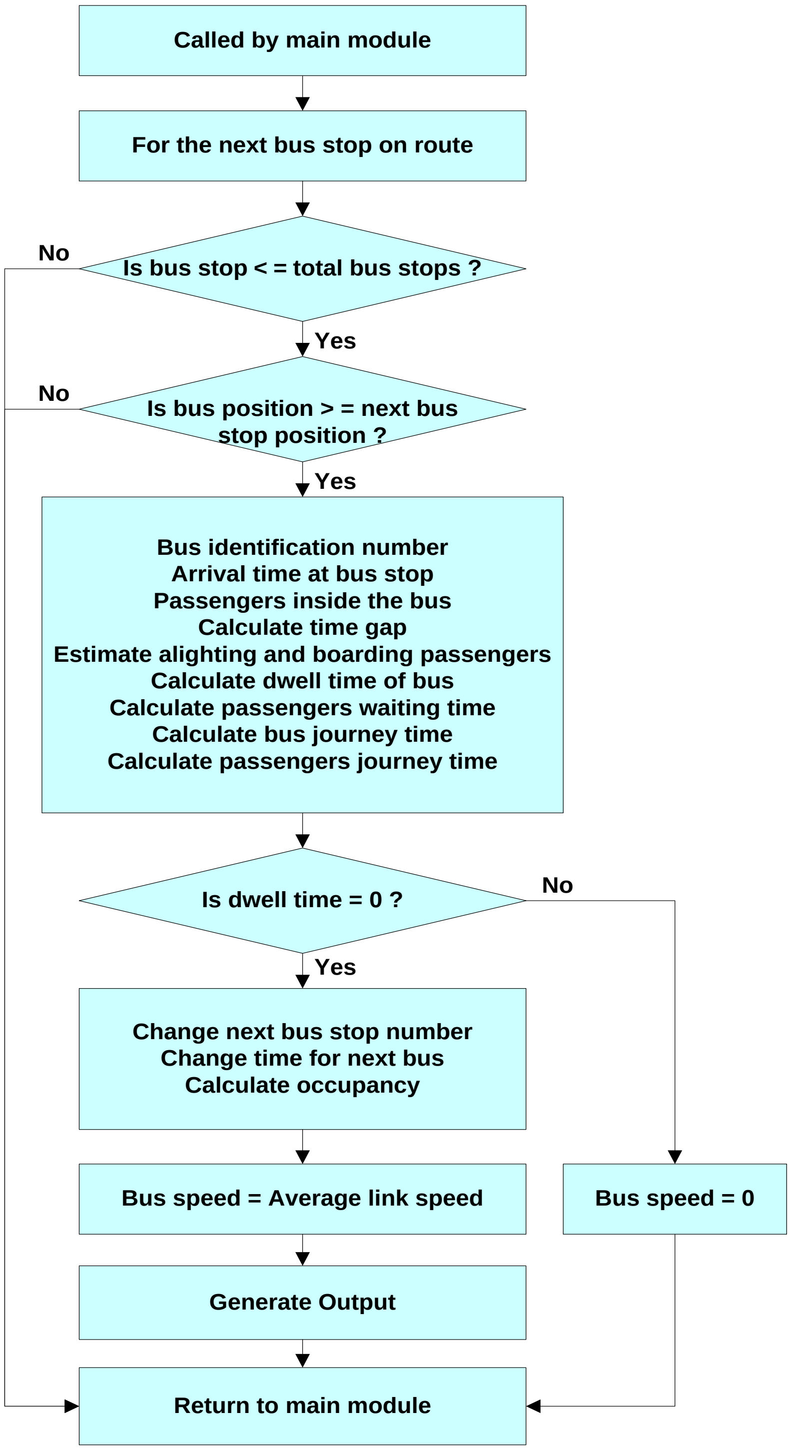


Figure 5.5: Flowchart of the bus stop module of SIBUFEM

Such is the significance of the role of the bus stop as a key model component of SIBUFEM that the description of the functions carried out by the module is necessary. Hence, the next Sections are dedicated to this purpose and the functions involved in bus stop module and incorporate the time gap calculation, the estimation of passenger numbers, and the calculation of dwell time and waiting time, are investigated in detail.

5.4.2 Passengers numbers estimation

The number of alighting passengers is estimated in the model using the alighting and boarding rate as described in Section 4.4.3.1. The alighting ratio is the percentage of passengers alighting at a bus stop out of the total passengers inside the bus. It depends on the bus stop and varies according to its location, while the number of total passengers inside the bus is recorded by the bus stop module once a bus arrives at a bus stop.

On the other hand, the number of boarding passengers is estimated using the time gap calculation. In order to estimate the number of passengers boarding and their waiting time, the time between the arrival time of a bus and the departure time of the earlier bus is required. This time is called time gap and, for the purpose of calculating it, the arrival and departure time of each bus in the system are recorded at each bus stop. Using these recorded times, the time gap of every arriving bus is calculated. As for the first bus, since there is no earlier bus to use its departure time, this is calculated using the start time of the passenger generation which is calculated according to Section 4.4.3.3. Time gap calculation is the first step of the bus stop module process and is followed by the estimation of the boarding passengers' number. An issue related to the use of time gap is the fact that the calculation of the departure time is calculated recursively. The reason is that the departure time can not be calculated using the parameter of dwell time, since the dwell time is not known yet. Hence, the boarding passengers are estimated for the time gap between departure and arrival according to this relationship:

$$\textit{Boarding passengers} = \textit{Time gap} / \textit{Rate of passenger arrival} \qquad (10)$$

Once the number of boarding passengers is calculated, the dwell time is estimated using this number and the number of alighting passengers. During this period of the dwell time more passengers are calculated and these need to be taken into account, thus, this is the step following. Then the dwell time for these extra passengers is calculated and the total dwell time is the sum of the dwell times. The total number of passengers is used to calculate the total number of boarding passengers.

5.4.3 Dwell time calculation

Dwell time is an extremely significant element of any bus operation. The correct calculation of the dwell time is based on field data collection as described in Section 4.4.3.2. The equation used by the model to calculate the dwell time is given by relationship 8 of the same Section. The equation, which is obtained from field data (Shrestha, 2003), incorporates the number of alighting and boarding passengers, expressed by α and b respectively, to calculate the dwell time T.

5.4.4 Passengers waiting time calculation

According to the flowchart of the bus stop module depicted in Figure 5.5, the process step following the calculation of the dwell time is the passengers waiting time calculation. Waiting time is the time between the arrival of the passenger at a bus stop and the arrival of a bus in which the passenger boards into. In this research, the focus is on high frequency headway-based bus operation; thus in SIBUFEM the average passengers waiting time is calculated through the expression of waiting time in relation to the bus headway, as described by relationship 4 in Section 4.3.5.

5.5 Incident Module

If the main module of SIBUFEM is to be characterised as the 'brain' of the model, being responsible for the overall operation and the decision making and interaction of the modules, then the incident module would be the 'heart' of the model. The incident module is the part of the model which involves modelling bus incidents occurring during the bus operation and, thus, it is the main focus of this research. The main target of this research work is to investigate these incidents and their impact on the key performance parameters of the model. The task of the incident module is to model the incident occurrence incorporating its 5 main parameters, as described in Section 4.3.7, and model their impacts on the bus operation.

5.5.1 Working of the module

The module is activated by the main module and it is responsible to receive the incident characteristics and model the incident and its impacts. The 5 incident parameters are passed to the incident module through the input-dialog box depicted in Figure 5.2 of Section 5.2.1. The main module, which is responsible for the time simulation tracking, checks whether the incident starting time is reached and if so the incident module is called. The model assumption for the occurrence of the incident is that the event takes place only once a bus reaches the incident location. The buses on the route are continuously checked and once their location meets the location of the incident, as this defined by the user via the input-dialog box, the incident is activated. The next step for the module process is to enquire on the type of the incident. Figure 5.6 describes the process of the incident module.

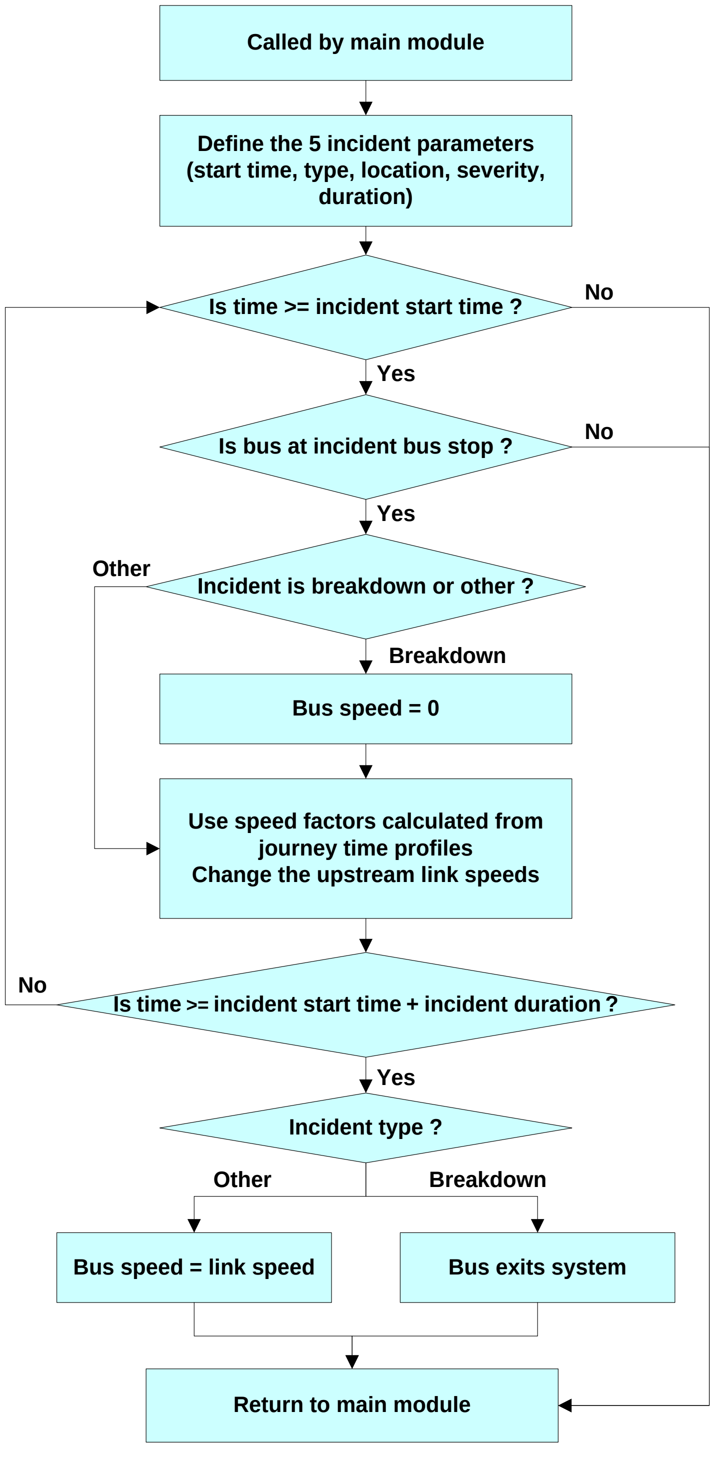


Figure 5.6: Flowchart of the incident module of SIBUFEM

5.5.2 Bus breakdown or other type

The focus of this research is on bus breakdown and other events which may reduce the roadway capacity, such as congestion, road works and illegal parking. In case of a bus breakdown, the speed of the bus is changed to zero and link speeds of the link upstream to the incident location are decreased to integrate the incident effect. Although the rest of the traffic and the effect is not modelled in detail during this work, the impact of such a bus or traffic incident on the traffic is included in the model using deterministic queuing theory and calculating cumulative delays, as presented in Section 4.3.7.5. Similar effect takes place for the case of other incidents, with the size of the impact being dependent on the location duration and severity of the event. The key difference between the two categories of incident types modelled is the fact that the behaviour of the parameter fleet size. Fleet size is a vital component for any bus based operation and a significant model constraint of SIBUFEM. A bus breakdown causes a decrease of the fleet size, since it is assumed that the bus is removed from the system once a breakdown occurs.

5.5.3 Duration and severity

The impact of the incident is directly related to its severity and duration parameters through the link speed factors. Figure 4.2 illustrates the methodology of the calculation of the bus speed decrease for each of the upstream links via a file of excel form. The longer the duration and the higher the severity of an incident, the longer the queue to be formed and consequently the smaller the values for the link speed factor and thus the bus speed. The duration of the incident is the parameter to determine when the incident terminates. However, the impact of the incident lasts longer, since more time is needed to accommodate the car queue created. This extra time is taken into account by the calculation of the link speed factors via the excel file. The severity options modelled for the purposes of the model vary from 25% roadway capacity decrease, representing a slight incident, to 40% capacity decrease of the route, referring to a moderate incident.

The effect that incidents such as roadworks, bus breakdowns or illegally parked vehicles cause to bus operations is modelled taking into account traffic queue lengths created in SIBUFEM. This effect is a matter of high concern in any bus operation, as confirmed by bus operators in Hove, UK.

> *'General traffic congestion is an issue but it's often down to a cause; you've only got to get one badly parked vehicle or one set of roadworks not managed very well. There was a set of roadworks down here in the Miles Road where is was cut down from two lanes to one lane and I contacted local authority to see whether it mattered to them as it happened between 9.30am and 4.30pm; it wasn't thought to be a very high priority but it caused congestion up to the busy roundabout close to the University and solid non moving traffic as far as we could see up the road.'*

5.6 Model Output

A simulation model cannot be complete unless the outcome is produced and presented efficiently and sufficiently. The model output is part of the main module and could be presented in Section 5.2; however, in order to emphasize the significance of the outcome for the evaluation of all the modules and the critical review of the key performance parameters, this Section is devoted to the output of SIBUFEM. The model continuously updates the status of various components in visual output and produces text files.

5.6.1 Visual output

The model is providing the continous progress of each component in the system. Figure 5.7 illustrates an example of the visual output of a simulation run of SIBUFEM (Polyviou and Hounsell, 2011).

Figure 5.7: An example of visual output produced by SIBUFEM

The output is shown with a reference to a linearly represented bus route. The route, which covers 8.64 kilometers, is depicted in grey colour. The positions of all buses and bus stops are arranged on this linear route.

The bus stops are represented by blue squares and buses by red and purple rectangles. Buses that are on their return trip (from busstop16 to busstop31) are depicted in purple, as a comparison to the first leg (from busstop0 to busstop15) red coloured buses. Furthermore, the simulation time and the simulation period are displayed on the top right corner of the visual output box, offering the user the opportunity to observe the progression of the buses and incidents occurrence in relation to time.

5.6.2 Text output

Apart from producing useful visual display, the model generates text files at the end of each simulation run. The main module produces a concise text file including but not limited to the key performance parameters described in Section 4.5.2. The average bus journey time, the average excess waiting time and the average bus speed are the performance parameters this research focuses on. Other parameters such as bus departure and arrival times, number of passengers boarding and alighting, dwell time, occupancy and waiting time are also contained in the output text files.

The significance of the key performance parameters for the comprehension of the model outcome and the efficient evaluation of the bus and traffic incidents modelled and their impacts on bus operation should be underlined. The details of the model applications and the results produced for each case set the focus outline of the Chapter 6.

5.7 Chapter Summary

A highly flexible and powerful computer language was used for the development of the simulation model SIBUFEM. The model consists of four modules, which interact with each other while the general operation and the simulation time are managed by the main module of the model. This Chapter has described the development process of the model with the contribution of flowcharts illustrating diagrammatically the function of each of the modules developed. Furthermore, the model input of SIBUFEM was presented with a particular focus on the incident characteristics and the dialog-box incorporating these parameters. Finally, the model is capable of generating output, both in visual display and text form, which set the model application and results evaluation procedure as the next research step.

-Chapter 6-

Model Application and Results (Phase A)

6.1 Introduction

One of the greatest advantages of using the powerful and highly flexible programming language C++ is the potential of adjusting the model according to the unique needs of the applications required, without altering the attributes of the model thereby increasing its adaptability. Consequently, various scenarios were investigated and became the subject of simulation using SIBUFEM. The aim of the application was to explore the effect of various bus and traffic incidents on the key performance parameters of the bus operation. Although it may not be possible to test these impacts in the field due to lack of field data about the incidents, which challenges the task of the complete model validation with real data, the simulation procedure offers invaluable results, the comparison of which is vital in current research.

This Chapter, therefore, describes the model verification and the validation of the base case scenario, then focuses on the model application procedure which consists of the investigation of the base case and the incidents case scenarios followed by the simulation results produced for each of the scenarios modelled. A discussion of the results for each of the model cases is provided followed by the comparison of results, and finally, the Chapter summarizes the key points.

6.2 Model Verification and Validation

Simulation models are increasingly being used to solve problems and aid in decision making processes. The model verification and validation are essential parts of the model development process for any simulation model to be accepted and used to support decision-making (Rakha *et al.*, 1996). The developers and users of such a model, the decision makers and the individuals affected by decisions based on the model are all rightfully concerned with whether the model and its results are correct (Rossetti *et al.*, 2009). This concern is addressed through model verification and validation.

6.2.1 Model verification

Model verification is defined to be the process of determining whether the logic that describes the underlying mechanics of the model, as specified by the designer, is faithfully captured by the computer code (Rakha *et al.*, 1996). The tasks associated with checking the model and corresponding programs to ascertain that they perform as intended set the outline of the model verification process (Davies and O'Keefe, 1989). The verification of a simulation model may be performed independently of field data (Rakha *et al.*, 1996), thus, hypothetical data was used to check the output of SIBUFEM from series of test runs. The objective was to ensure that, for a given input, the program code provides the output that is consistent with the logic on which the code is based. For this purpose, visual output and text output files were used.

Adjusting the simulation speed input, so as the speed of the simulation time was equal to real time, offered a practical and efficient method for the visual verification process. Using the visual output during simulation runs, the simulation time and period were checked that they were activated, the input-dialog box reflected each time the chosen options and the generation of buses was checked to ensure it occurred at the specified generation point (i.e. bus stop 0) according to the generation times. Furthermore, the progression of buses was checked via the journey time parameter and it was ensured that each of the buses was provided with an identification number which was clearly indicated on top of the bus. Buses stopped at the bus stops to board and alight passengers and the option of overtaking buses was confirmed. The visual output also verified the right operation of the model and the initialization and termination of SIBUFEM according to the predefined simulation time and period.

More detailed verification of the model was carried out using the text output files provided once a simulation run was completed. The performances of each of the modelling parameters are included in output files generated by different modules. The main detailed output provided information including but not limited to bus position, travel time and stoppage at bus stops. The latter information was counterchecked with data from the bus stop module and the same concept was followed for the rest of the verification process. For example, arrival and departure time obtained from the bus stop module were checked against the bus module to verify that the bus stopped and moved from the bus stop respectively.

6.2.2 Model validation

Once the simulation model was verified, the research focus moved on to the model validation process. Model validation is the process of determining to what extent the model's underlying fundamental rules and relationships are able to adequately capture the targeted emergent behaviour, as specified within the relevant theory and as demonstrated by field data (Rakha *et al.*, 1996). Validating every aspect of SIBUFEM, including the modelling of bus and traffic incidents, may not be achieved due to lack of sufficient field data for the incident scenarios modelled. However validation of the base case model was carried out to make sure this reflected the system in the field.

For the purposes of model validation, the method of bus journey time profiles incorporating passenger-dependent bus stop dwell times and deterministic time-dependent queuing theory was used. Section 4.3.7 described the incident characteristics and the steps followed during the queuing theory analysis as shown in Figure 4.2. Car queue lengths were calculated according to the roadway capacity, the demand and the incident impact on capacity. All incident characteristics were taken into account, resulting in different cumulative delays and queue lengths describing each incident scenario. An example of demonstrating the impact of incident and the traffic queue lengths formed for each of the links upstream to the incident location is provided in Figure 6.1. The graph was used for the validation of the scenario of a medium severity incident which takes place at the middle of the route and lasts one hour. The traffic at the first link upstream to the location of the incident is affected resulting to the building up of a queue, which moves to the next upstream link once the maximum length of the first link is reached and so forth.

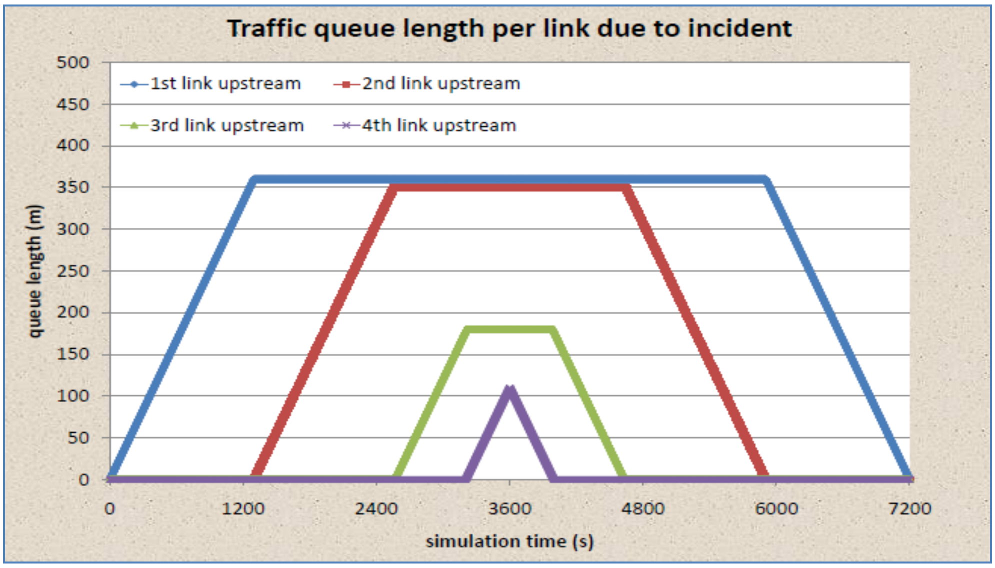


Figure 6.1: Model validation using deterministic time-dependent queuing theory.

6.3 Model Application

The key target and essence of a simulation model is to be used and applied to various scenarios in order to address the tasks it is developed for. The aim of the model application, for the purposes of SIBUFEM, was to explore the impact of bus and traffic incidents on the bus operation key performance parameters. In order to achieve this target, the base case was the first scenario formulated and modelled, which was the comparison tool for the evaluation of the incidents scenarios effects.

6.3.1 Incident scenarios modelled

A total number of 25 different scenarios were identified and investigated, taking into account the 5 key incident parameters (i.e. starting time, type, location, severity and duration of the incident) described in section 4.3.7. The number of scenarios was dependent on the possible options for each of the parameters of the incidents. There may be 2 options for the type of the incident, 2 for the location, 2 for the severity and 3 for the duration parameter, thus, 24 scenarios were identified and investigated in total. The base case scenario, which describes the normal operation of the service without any disruption, is the first case to be modelled and, therefore, the total number of scenarios becomes 25. In order to produce realistic journey time profiles and offer the system the opportunity to overcome any incident impacts, the simulation period used for each of the scenarios during the model application was 12 hours.

The scenarios formulated and tested for the purposes of model development are described in Table 6.1. The first column shows the identification number of the incident, the next column provides the code of the incident, which is used to link the incident to its identifying parameters and the scenario description is under the third column presenting the characteristics of the incident. The scenario code is a 4 digit number, where each of the digits refers to the option used for each of the parameters. For example, a scenario coded 1223 refers to the scenario of a bus breakdown (i.e. the first digit is 1), occurring at the end of the route (i.e. the second digit is 2), of a medium severity (i.e. the third digit is 2) and long duration (i.e. the last digit is 3).

Table 6.1: Incident scenarios formulated and modelled.

Number	Scenario Code	Scenario Description
1	0000	No Incidents (Base Case)
2	1111	Bus Breakdown, middle route, slight severity, short duration
3	1112	Bus Breakdown, middle route, slight severity, medium duration
4	1113	Bus Breakdown, middle route, slight severity, long duration
5	1211	Bus Breakdown, end route, slight severity, short duration
6	1212	Bus Breakdown, end route, slight severity, short duration
7	1213	Bus Breakdown, end route, slight severity, short duration
8	1121	Bus Breakdown, middle route, medium severity, short duration
9	1122	Bus Breakdown, middle route, medium severity, medium duration
10	1123	Bus Breakdown, middle route, medium severity, long duration
11	1221	Bus Breakdown, end route, medium severity, short duration
12	1222	Bus Breakdown, end route, medium severity, medium duration
13	1223	Bus Breakdown, end route, medium severity, long duration
14	2111	Other Incident, middle route, slight severity, short duration
15	2112	Other Incident, middle route, slight severity, medium duration
16	2113	Other Incident, middle route, slight severity, long duration
17	2211	Other Incident, end route, slight severity, short duration
18	2212	Other Incident, end route, slight severity, short duration
19	2213	Other Incident, end route, slight severity, short duration
20	2121	Other Incident, middle route, medium severity, short duration
21	2122	Other Incident, middle route, medium severity, medium duration
22	2123	Other Incident, middle route, medium severity, long duration
23	2221	Other Incident, end route, medium severity, short duration
24	2222	Other Incident, end route, medium severity, medium duration
25	2223	Other Incident, end route, medium severity, long duration

6.3.2 Base case scenario

If a simulation model is to be applied to various scenarios, defining the base case scenario as a first step is essential. The base case is the comparison tool which, besides the useful output related to the current state of the bus route, offers the developer and the user the opportunity to analyze, compare and evaluate the scenarios applied. Application of the model with a simulation, where the current state of the bus operation without the occurrence of any incidents was modelled, sets the outline of the base case scenario. The bus operation, a high frequency headway based service, provides buses along the continuous route according to its 8-minutes-headway. The simulation for this first scenario, which may also be referred as the 'no-incident' case, aims to provide the model output to be used for the evaluation method and follows all the guidelines and assumptions defined during the modelling development process.

6.3.3 Bus and traffic incidents scenarios

The role of incidents' characteristics and their impacts are predominant throughout this research; significant consideration was taken to identify the possible options for each of these characteristics and, therefore, formulate the scenarios to be investigated using SIBUFEM. The selection of the scenarios was based on the 5 key incident components presented in section 4.3.7 and key performance parameters, as described in section 4.5.2, were used to support the method of evaluation of the impact of each incident.

6.3.3.1 Selection of incident scenarios

The incident scenarios were constructed according to the 5 main incident parameters: (i) starting time, (ii) type, (iii) location, (iv) severity and (v) duration. For the needs of the model, 2 options were provided for the type of incident, 2 for the location parameter, 2 for the severity and 3 for the duration, leading to a number of 24 scenarios, as illustrated in Figure 6.2.

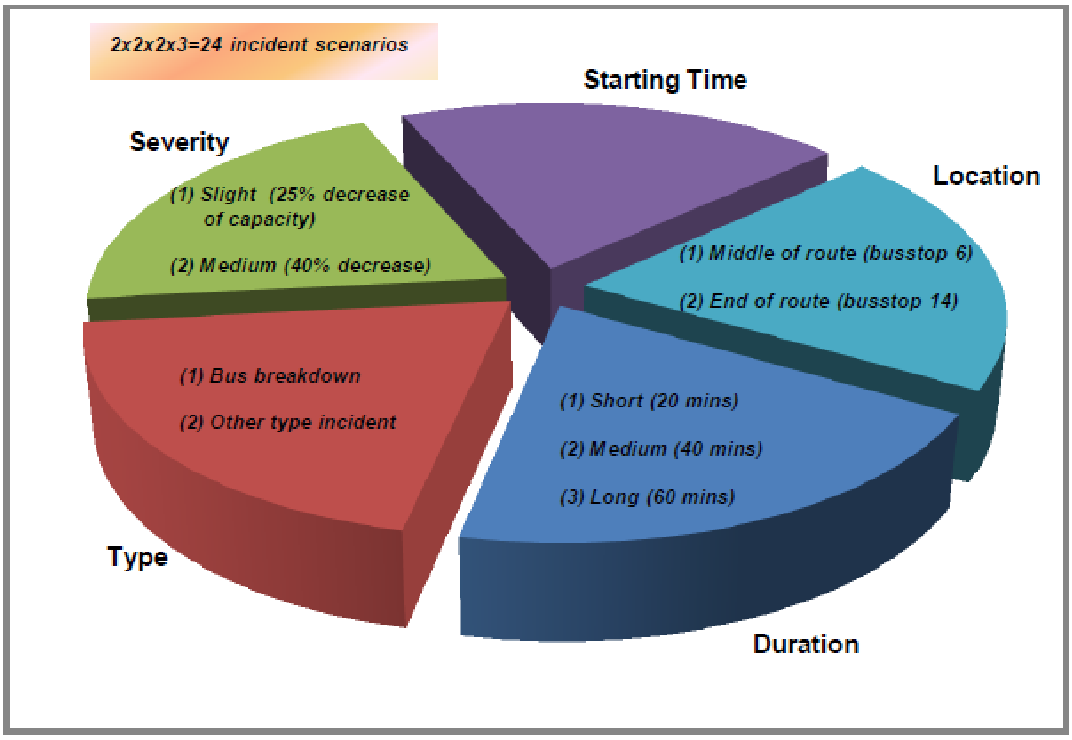


Figure 6.2: 24 incident scenarios selected based on incident parameters.

It should be highlighted that the first incident parameter, the starting time, was given a fixed value equal to 3 hours, which means that each incident occurs 3 hours after the simulation initialization. Although this parameter was steady during the analysis of these 24 scenarios, the effect of the change in this component was investigated subsequently and it was decided not to be analyzed at this stage of the model application. Table 4.1 describes the complete list of scenarios modelled during the application process.

6.3.3.2 Evaluation method

The significance of the performance parameters of a model is even more evident during the stage of model application and more specifically the evaluation process. The evaluation of the modelled scenarios was based on the comparison of their performances, as these are indicated by the output file generated at the end of each simulation run. The 3 key performance parameters of SIBUFEM, described in Section 4.5.2 during the model methodology definition are identified as the performance criteria for this comparison.

Text output files generated after the end of each simulation run provided, among other parameters, the average bus speed and the average bus round-trip journey time. The calculation of the excess waiting time involves the contribution of an excel file which illustrates the calculation of the excess waiting time per bus stop for each of the buses on route (Appendix B). The excess waiting time, which is the difference between the scheduled waiting time and the actual waiting time (TfL, 2010), represents the additional time that passengers have to wait for a bus and is a key performance indicator for any bus operation, thus, special attention has been drawn on this parameter during this research.

6.4 Model Results

In order to evaluate each of the cases investigated for the purposes of this research, 25 simulation runs were carried out to cover all scenarios. The simulation output was then stored and expanded to provide values for all 3 key performance parameters. The comparison between the simulation results for each of the incident cases and the respective output for the no incident case (i.e. base case) was carried out. The simulation results of SIBUFEM and the output comparison carried out as an evaluation method is described in this section.

6.4.1 Base case scenario

Before simulating the various incidents components and their impacts on the bus operation, a simulation of ‘no incident’ case scenario was carried out. The base case or scenario ‘0000’, as called for model development purposes, describes the circulation of the buses along the route, the progression of the buses and their interactions with the key model component of bus stop. A 12 hours simulation period was chosen in order to incorporate performance parameters’ changes in relation to the time of the day and text output files generated at the end of the run, such as the ‘BusstopOut.txt’ and ‘SibufemOutput.txt’ files, provided information related to average bus journey time, average bus speed and average excess waiting time.

Bus journey time was the first performance parameter to be investigated, with simulation results depicted in Figure 6.3 representing the bus journey time changes in relation to the simulation time and the average bus journey time. Investigating the circulation of the buses around the route and assuming that the turn around of the buses at each end does not affect the journey time, the simulation run for the base case scenario showed that the average bus journey time for the round bus trip is 3024 seconds. The journey time for each of the buses is obtained and shown in Figure 6.3 according to the generation time of the bus during the 12 hours simulation run.

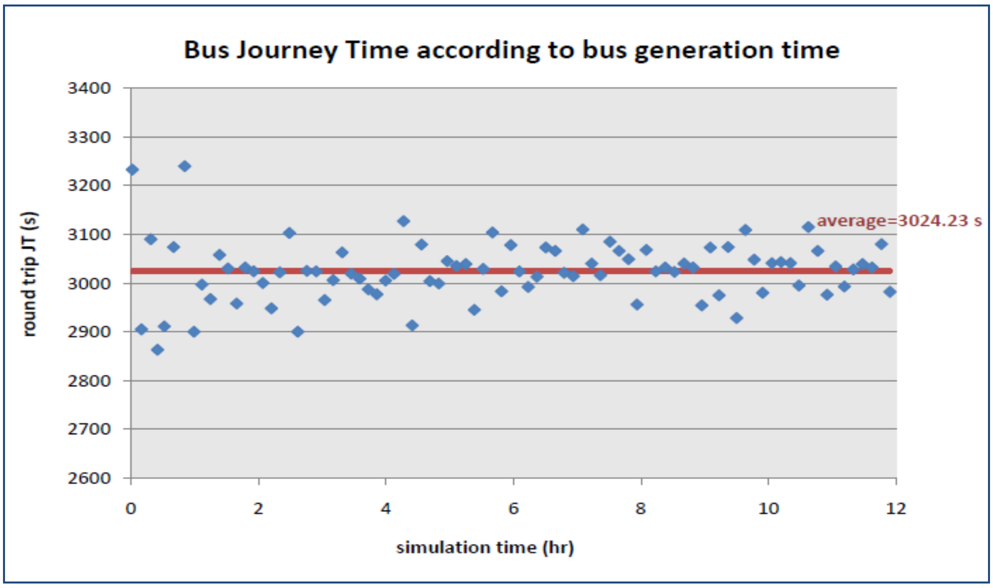


Figure 6.3: Bus journey time for the base case.

Similarly, the bus speed changes in relation to bus generation time and the average bus speed in terms of kilometres per hour (km/h) are illustrated in Figure 6.4. The average overall bus speed was found to be 10.34 km/h for the base case scenario. This value was the average overall bus speed taking into account all delays at generation or bus stops. The speed is considerably less than the field average bus speed of existing bus service which covers the same route in Southampton (approximately 14km/h); this is due to the fact that the data used for the model development of SIBUFEM represents the situation of a bus service which was in operation in 2001. With the values of two out of the three model performance parameters defined, the focus shifts to excess waiting time of the no incident case.

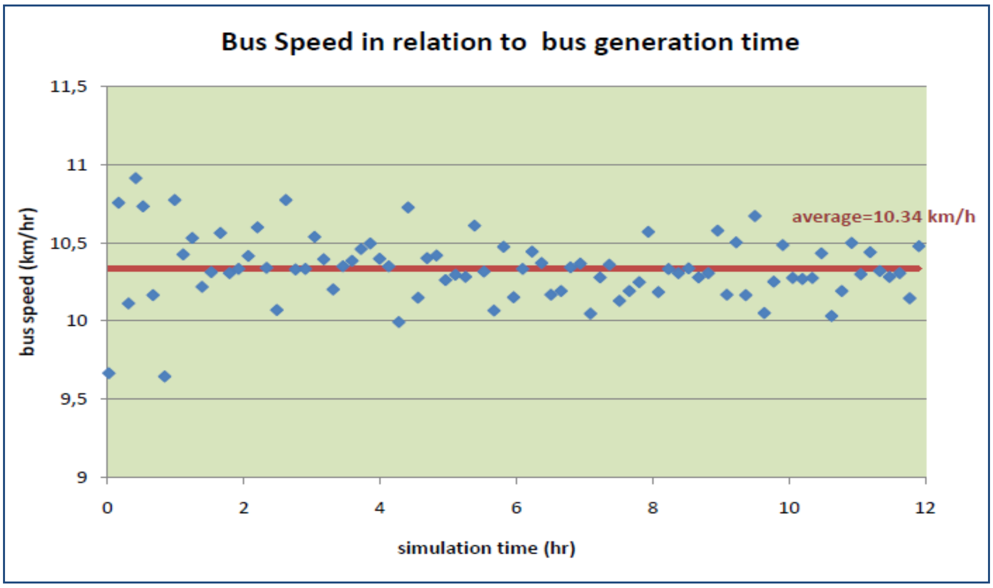


Figure 6.4: Bus speed for the base case.

For the calculation of the third model performance parameter, an excel file receiving model output was used to compute the excess waiting time per bus stop for each bus on the system. Figure 6.5 illustrates the way the average excess waiting time, as this was calculated for the total number of buses operating during the 12 hours simulation run (i.e. 86 buses in total), changes along the route according to the bus stop ID. Due to the lack of incidents occurring or any other abnormal conditions, the average excess waiting time is starting at 12.75 and reaching a peak of 19.23 seconds. The average value is 15.80 seconds or 0.26 minutes, which is relatively low given the fact that average excess waiting time for high frequency London bus services is 0.96 minutes (TfL, 2010). This is mainly explained by the fact that the modelled route is based on Southampton data, where congestion is less than in London and because SIBUFEM does not capture all the bus operational variability which occurs in practice (see Chapter 7 for further discussion of this). For the base case, SIBUFEM was applied to 'clear' conditions, whereas the London data for excess waiting time includes incident-affected data which may have occurred.

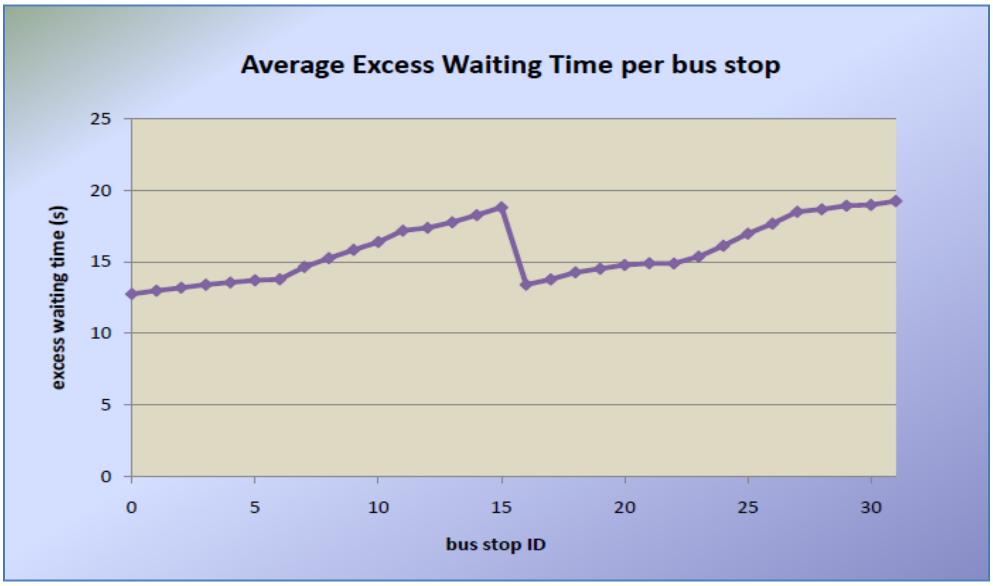


Figure 6.5: Average excess waiting time for the base case[2].

6.4.2 Bus breakdown scenarios

As described in section 6.3.2.1 and presented in Table 6.1, 12 scenarios constitute the complete application of the model for the case of a bus breakdown. A bus breakdown, as modelled in this research, involves the exit of the bus out of the system and the re-establishment of the bus fleet size with a reserve bus when the incident duration is exceeded. The way the components of location, severity and

[2] The bus stop-related data used in Figure 6.5 are provided in Table 4.1.

duration affect the 3 key model performance parameters is a matter of interest in this section. Simulation results describing the various scenarios of a bus breakdown are illustrated with the use of graphs and tables and a comparison against the base case results reflects the impacts of such an incident on the bus operation. For the purposes of a thorough analysis and representation of the simulation results, a detailed description of the first of the 12 simulation runs is given, supported by key summarizing points of the total number of runs with tables and graphs providing figures of the key performance indicators.

6.4.2.1 Detailed bus breakdown scenario example

Each of the 12 bus breakdown scenarios was simulated using SIBUFEM for a period of 12 hours and the simulation output was stored according to their code number. The simulation period of 12 hours was required in order to investigate the behaviour of the incidents and the impact on bus operations during whole day periods. Figure 6.6 illustrates the change of bus journey time along the simulation time and average journey times before and after the incident occurrence. The specific graph describes the simulation for the breakdown scenario with code '1111', which, according to Figure 6.2., represents a bus breakdown that occurs at bus stop 6 and causes 25% reduction of the roadway capacity for 20 minutes time. The bus breakdown takes place 3 hours after the initialization of the simulation run and the effect on the journey time of the next buses is evident. The gap at 3 hours simulation time reflects the bus which suffers the break down and the journey time of this bus is not incorporated in the calculation of the average bus journey time.

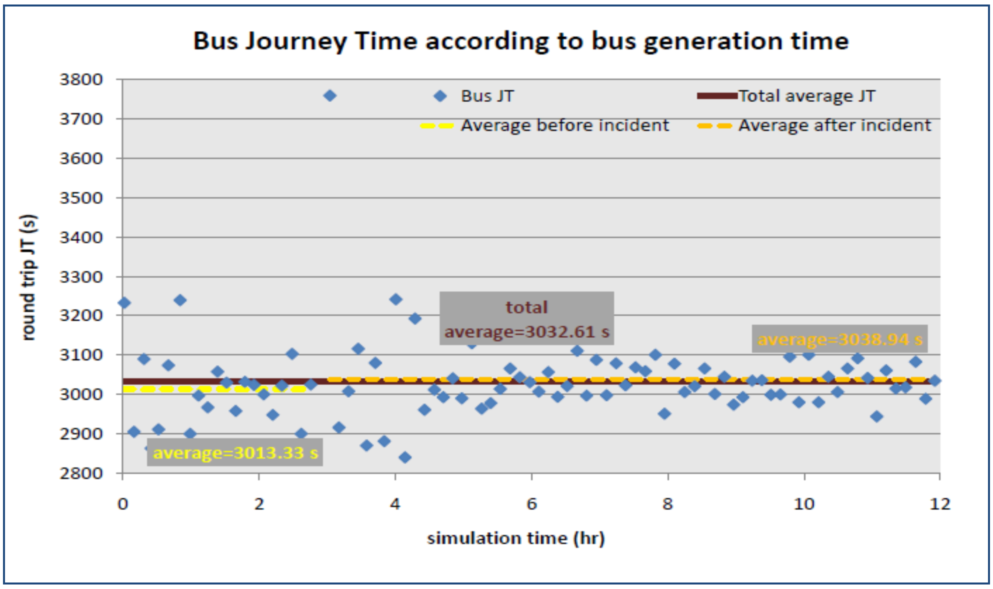


Figure 6.6: Bus journey time for the bus breakdown scenario '1111'.

Similarly to the calculation of bus speed in relation to bus generation time carried out for the base case scenario, the total average bus speed of the scenario '1111' was computed and Figure 6.7 depicts the simulation outcome. Yellow and orange dashed lines related to the average bus speed before and after the occurrence of the breakdown, giving 10.38 km/h and 10.29 km/h respectively. The decrease of the bus speed after the first 3 hours of the simulation is clear and was anticipated, as this is the time when the bus breaks down and disruption of the bus operation occurs.

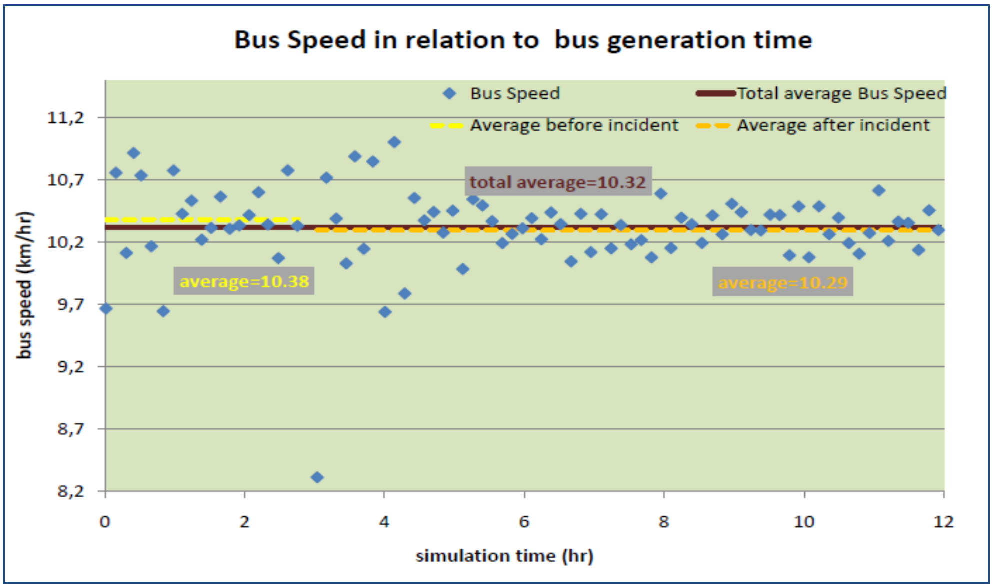


Figure 6.7: Bus speed for the bus breakdown scenario '1111'[3].

The effect of the incident on the key performance parameters of the model is clearly highlighted in Figures 6.6 and 6.7, however the average excess waiting time is a key quality indicator for this research and the calculation and comparison against the figures provided by Figure 6.5 was fundamental. The excess waiting time (EWT) comparison is depicted graphically in Figure 6.8, which shows that the parameter, even though it is approximately even to the values for the base case for the first 6 stops, it increases rapidly at bus stop 6, where the breakdown occurs, and follows the red line to reach its maximum value of 35.03 seconds.

[3] In Figure 6.7 the bus breakdown occurs 3 hours after the model simulation starts.

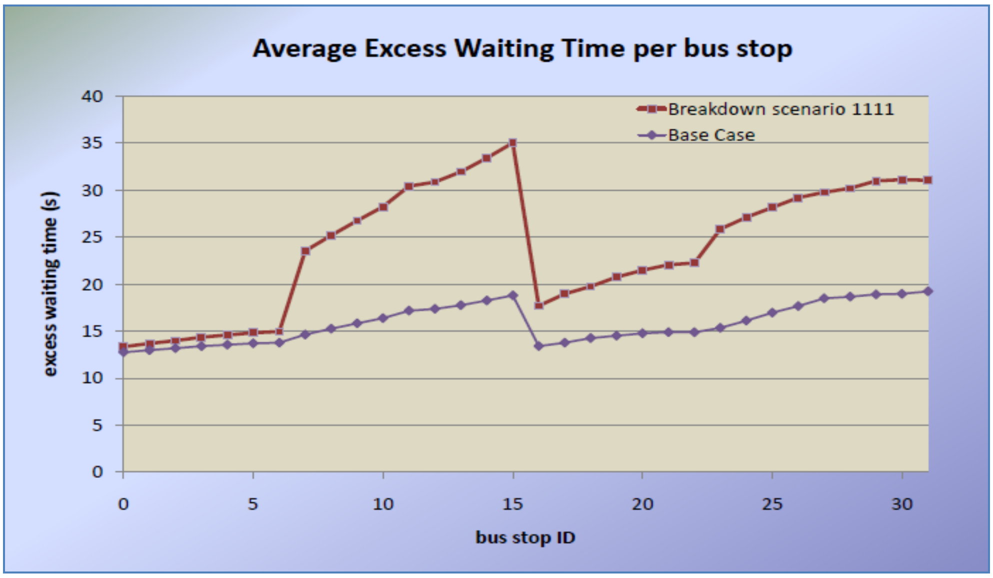


Figure 6.8: Average excess waiting time for the bus breakdown scenario '1111'.

Looking more thoroughly into the impact of the incident on the three key performance measures, as demonstrated through Figures 6.6-6.8, a detailed analysis on the time period that the event takes place for was required to support the complete and efficient study of the incident effect. For this purpose, the progression of individual buses immediately before and after the occurrence of the incident is examined. The journey time and speed per section (i.e. route section between two consecutive bus stops) of the route were calculated for five cases of buses:

(i) 1st bus before the incident occurrence
(ii) Incident bus (i.e. the bus which breaks down)
(iii) 1st bus after the incident occurrence
(iv) 2nd bus after the incident occurrence
(v) 3rd bus after the incident occurrence

The reason for choosing these buses was to explore the effect of the bus breakdown on individual buses on which the incident has more effect. This is of high interest for the development and analysis of potential DBFM strategies, particularly in cases where the severity and duration of the event are low resulting in a minor overall impact on the bus performance. Figures 6.9 and 6.10 illustrate the simulation results related to these five individual buses for the same bus breakdown scenario '1111'; they describe the behaviour of journey time and speed per link of the route for each of these buses.

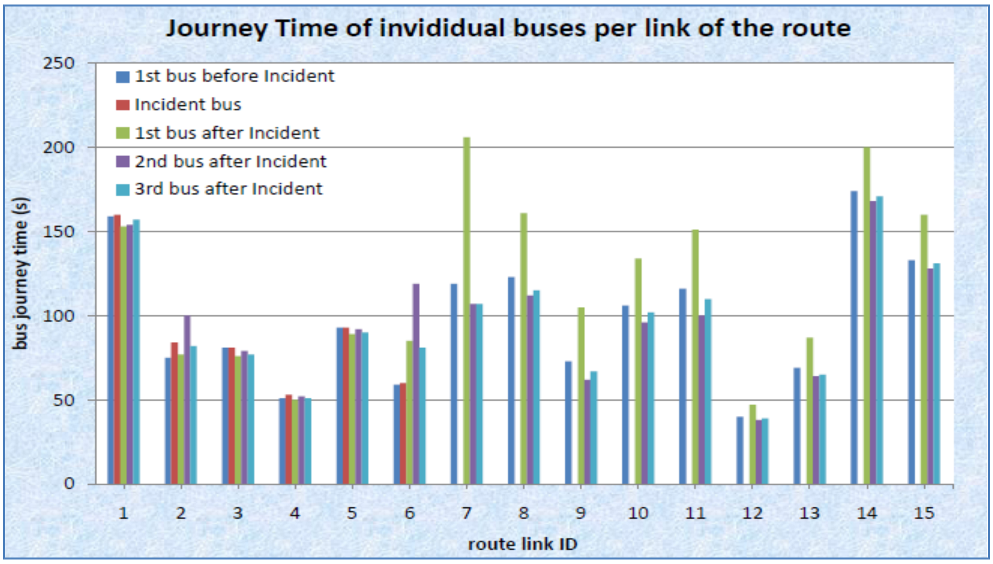


Figure 6.9: Journey time of individual buses per link of the route for the bus breakdown scenario '1111'.

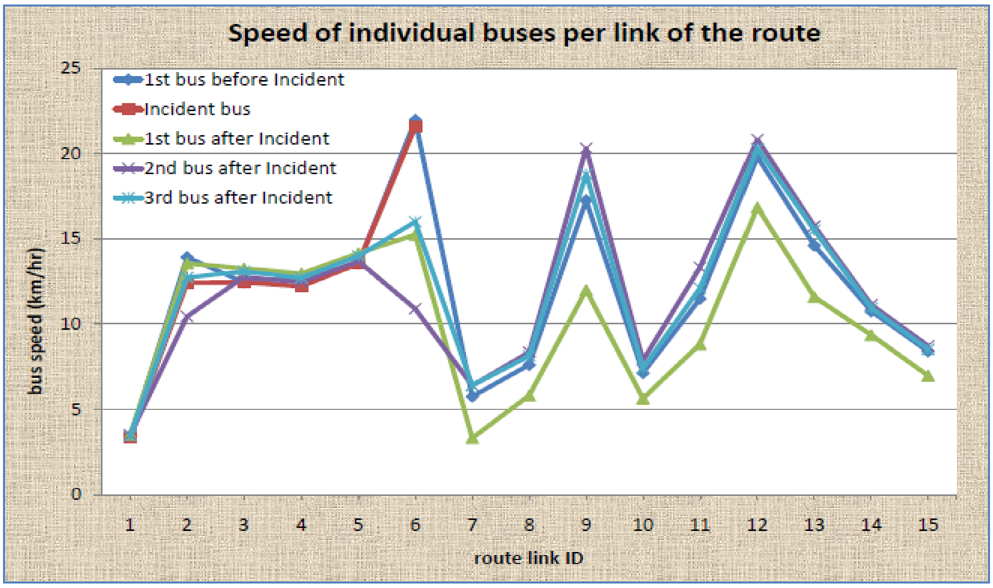


Figure 6.10: Speed of individual buses per link of the route for the bus breakdown scenario '1111'[4].

The location of the bus breakdown in this case is at bus stop 6 (i.e. start of link 7 of the route). However, the incident impact on the rest of the traffic resulting in disruption of the bus operation is evident at link 6 of the route as shown in Figures 6.9 and 6.10. If attention is aimed at link 6, results show that the journey time for the buses following the bus breakdown occurrence is increased compared to the

[4] In SIBUFEM, a route link is described as any section of the bus route between two consecutive bus stops.

journey time of the bus which suffers the breakdown, consequently causing a decrease in bus speed compared to 'normal' bus operation. The rise of the journey time for the first bus going through the incident location, once the breakdown takes place, is substantial at link 7 of the route with an increase from 119 seconds to 206 seconds, resulting in a 42% speed reduction at the specific link, according to Figures 6.9 and 6.10 (Polyviou and Hounsell, 2011). The model results suggest that for the bus breakdown scenario '1111' only the very first bus is substantially affected by the bus breakdown, while the following two buses are mainly influenced at the link of the route where the even occurs.

The application of simulating this first incident case, which may be characterized as a 'slight' bus breakdown scenario, caused changes in the key performance model parameters, suggesting specific attention be paid to the remaining simulation runs. The methodology used during the rest runs was identical with each simulation providing output related to journey time, bus speed and excess waiting time. The next sub section summarizes the results of these simulation runs with focus on the three model performance indicators.

6.4.2.2 Bus breakdown scenarios results

Using SIBUFEM and enabling the respective incident characteristics options via the Input-Dialog box, which is the first step once the model simulation has been initialized, generated output characterizing the conditions of the incident and its impacts on the system. Partial and total average bus journey times were calculated in order to cover every aspect of the incident effect. The second and third columns of Table 6.2 provide the partial average bus journey times before and after the incident occurrence in seconds. The results for the total bus journey time of 12 hours simulation time is listed in column 3. Different colours are used where the dashed line is inserted to separate the first 6 scenarios of incident taking place in the middle of the route from the next 6 where the incident location is the end of the route.

As expected, the bus journey time parameter is only affected once the incident occurs and total journey time varies from 3032.61 to 3159.61 seconds for the first six scenarios and from 3028.98 to 3146.95 seconds for the following six. The calculation of the total average journey time represents the total number of buses that are generated in SIBUFEM for the 12 hours simulation period, whereas the partial journey times are based on a smaller number of buses, which represents the number of buses generated at the respective time gap.

The term 'partial' is used, here, to characterise the average journey times related to a partial number of buses running along the route; they refer to a part of the total number of the generated buses. For example, the partial average journey time values listed in the second column in Table 6.2 refer to the buses generated between the beginning of the simulation and the time of the incident occurrence. Similarly, the values of the third column in Table 6.2 are the average journey times for the buses that were generated between the time when the incident occurred and the end of the simulation run. The scenario codes at Table 6.2 represent the scenarios modelled which are described in detail in Table 6.1.

Table 6.2: Average bus journey time for all bus breakdown scenarios.

	Average Bus Journey Time (seconds)		
Scenario Code	**JT before Incident**	**JT after incident**	**total average JT**
1111	3013.33	3038.94	3032.61
1112	3013.33	3053.98	3043.82
1113	3013.33	3066.63	3053.31
1121	3013.33	3066.08	3052.89
1122	3013.33	3109.25	3085.27
1123	3013.33	3210.83	3159.63
1211	3012.75	3033.97	3028.98
1212	3012.75	3045.05	3037.45
1213	3012.75	3068.25	3054.88
1221	3012.75	3048.43	3040.04
1222	3012.75	3092.66	3073.63
1223	3012.75	3190.95	3146.95

According to the simulation results, SIBUFEM suggests that the expected average bus journey time for a bus breakdown scenario is approximately 3013 seconds for the buses going through the location of the incident before the occurrence of the event. If this value is checked with the model results for the base case scenario, where the average journey time was found 3024 seconds in Figure 6.4, there is a time difference of approximately 11 seconds; this difference is due to the fact that only 21 buses (21 buses go through the incident location before the event occurs) are taken into account for the calculation of the first column of Table 6.2, while 86 buses (all of the buses generated during the 12 hours simulation run) contribute to the average value of base case in Figure 6.3. The simulation results, however, for the average of the first 21 buses of the base case scenario agree with the results on the same number of buses of the bus breakdown scenario as it was expected.

A similar concept but different performance indicator is the focus of the next list of simulation results; a list which describes the model outcome in terms of average bus speed. Again bus speed was calculated before and after the starting time of the incident and in total as demonstrated in Table 6.3. The scenarios referring to each scenario code are described in Table 6.1.

Table 6.3: Average overall bus speed for all bus breakdown scenarios.

	Average Bus Speed (km/hr)		
Scenario Code	speed before Incident	speed after incident	total average speed
1111	10.38	10.29	10.32
1112	10.38	10.25	10.28
1113	10.38	10.21	10.25
1121	10.38	10.22	10.26
1122	10.38	10.10	10.17
1123	10.38	9.87	10.00
1211	10.38	10.31	10.33
1212	10.38	10.27	10.30
1213	10.38	10.21	10.25
1221	10.38	10.26	10.29
1222	10.38	10.14	10.20
1223	10.38	9.91	10.02

The average bus speed for each bus breakdown scenario is provided by the last column of Table 6.3. For example, during scenario '1111' buses operate with an overall speed of 10.32 km/h, as also shown in detail in Figure 6.7. A first observation is that average bus speed decreases when the duration of the incident increases. Furthermore, simulation results show that buses run slower when the incident becomes more severe. This decrease is evident if for example scenario '1121' (i.e. short duration scenario), '1122' (i.e. medium duration scenario) and '1123' (i.e. long duration scenario) are compared; according to calculations the change in speed caused by the incident is 1.54%, 2.70% and 4.91% respectively.

Some of the reductions in speed suggested by the model results seem to be small. However, it needs to be taken into account that the above bus speeds refer to the average bus speed results for the total or a partial number of the buses. In many cases this represents a number of more than 80 buses running along the route over the 12 hours simulation period. The impact of the incidents affects the bus operation overall but more severely a certain amount of buses as described in Figures 6.9 and 6.10. Even though the average bus speed may be decreased by 1-5% overall, the speed reduction for some buses, as shown in Figure 6.10, may reach the 42% compared to the normal conditions operation. Further comparisons and overall evaluation of the simulation results are presented in the following section.

Finally, the calculation of excess waiting time for each of the modelled scenarios was carried out and the average value per bus stop was computed. This is a model performance parameter of significant importance for the evaluation process, since it reflects the additional time that a passenger needs to wait at a bus stop and considering that the value of passenger waiting time is estimated as double the passenger journey time value in bus operations (HEN 2, 1997).

Scenarios were divided again in two groups depending on the location of the incident for the purposes of clear illustration of the results and better understanding of the incident effect, as shown in Figures 6.11 and 6.12.

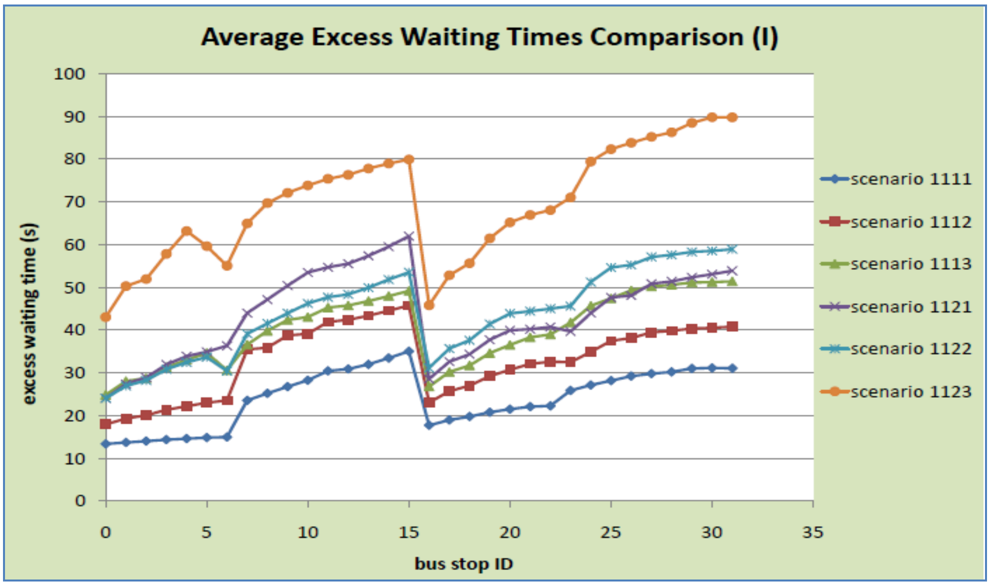


Figure 6.11: Average excess waiting times for bus breakdown scenarios when the incident occurs at the middle of the route[5].

[5] The list of the scenario codes used in this Figure is provided in Table 6.1

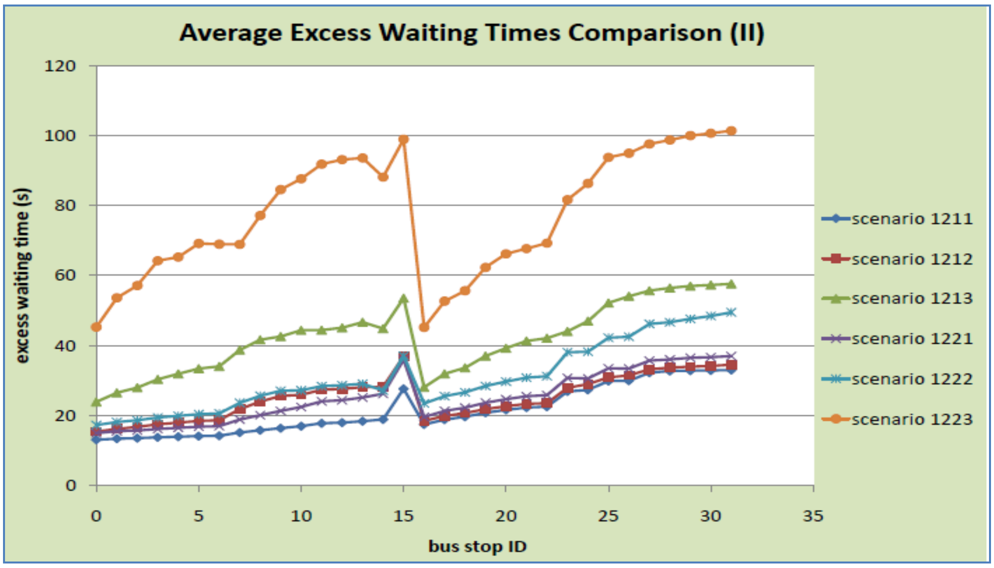


Figure 6.12: Average excess waiting times for bus breakdown scenarios when the incident location is at the end of the route[6]

According to Figures 6.11 and 6.12, simulation results demonstrate the similarity of the curves describing the behaviour of average excess waiting time in relation to the route location for each of the modelled scenarios. The average excess waiting time meets its minimum value at the start of the bus route (i.e. bus stop 0), then rises steadily until it receives the maximum value for the first leg of the route (i.e. bus stop 15). The return leg of the route initializes at bus stop 16; this is a decision point for the bus operation, since an added slack time is provided to accommodate the late buses and regulate the operation. Furthermore, early buses are provided with the 'holding' option, which delays their generation in order the designed 8-minutes headway is met. Therefore, at bus stop 16, excess waiting time is considerably less compared to the value of the parameter at bus stop 15. Then, the second leg (return trip) of the route contributes again to the gradually increase of the waiting time until the return trip is complete and bus generation point (i.e. bus stop 0) regulates the bus headway and, thus, the bus operation.

6.4.3 'Other type' incident scenarios

The identifying difference between the two categories of incidents modelled with the use of SIBUFEM (i.e. bus breakdown incident and other type of incident) is the behaviour of the fleet size. This key modelling parameter is a crucial component for any bus based operation and plays a crucial role as a constraint in the model. While

[6] The complete list of the scenario codes used in this Figure is presented in Table 6.1

in the case of the bus breakdown the fleet size is reduced by one until the incident duration is exceeded and the fleet size is then restored, in 'other type' incident scenarios the fleet size remains unchanged and the incident impact is due to the roadway capacity reduction caused and the effect on normal bus operation. It should be highlighted that, incidents of this category do not involve bus stopping (i.e. bus breakdown) and exiting the system, which is expected to produce low impact on the model performance parameters. However, the fact that the fleet size remains unchanged is an important difference between this type of incidents and bus breakdowns and SIBUFEM will model the effect on the performance parameters in each case. The simulation results and their comparison need to shed light on this interesting research issue.

This category of incidents involves events including but not limited to road works, congestion, traffic accidents and illegal parking. Congestion is a common example of such an incident which may have severe negative effect on the reliability of a bus schedule and overall performance, as confirmed by bus operators:

> *'This line operates every 10 minutes during the day time, but in the afternoon in rush hour it dropped down to every 12 minutes because the journey time is longer due to congestion. But that is still becoming unreliable, so now in this timetable, which started in April we now dropped the timetable down to every 15 minutes, which is again another paradox as for the busiest time of the day we are running fewer buses, which is in fact the same number of buses but they take longer to complete their journey.'*

The impact of these incidents on the bus operation is incorporated as a roadway capacity reduction of the modelled route and involves the build up of car queues and the effect of general traffic on the buses of the system. The option of Input-Dialog box which appears at the start of every simulation offers two options for the type of incident; the second option (i.e. 'other incident') is chosen for the new simulation runs, investigating the scenarios in Table 6.1 with a code starting with the digit 2. The calculation of the model outcome is based on these runs for each of the new 12 incident scenarios and results consist of values for all 3 key performance indicators, which are average bus journey time, bus speed and excess waiting time. Each of the simulation runs modelling the 12 new incident scenarios were carried out on a 12 hours period basis. An example of the simulation results produced for each of the scenarios is provided through Figures 6.13 and 6.14, where the variations of bus journey time and bus speed in relation to simulation time are shown respectively. The graphs represent results generated after simulating scenario '2111' (i.e. 'other type' incident occurs at bus stop 6 and causes 25% reduction of roadway capacity for 20 minutes time) with the use of SIBUFEM.

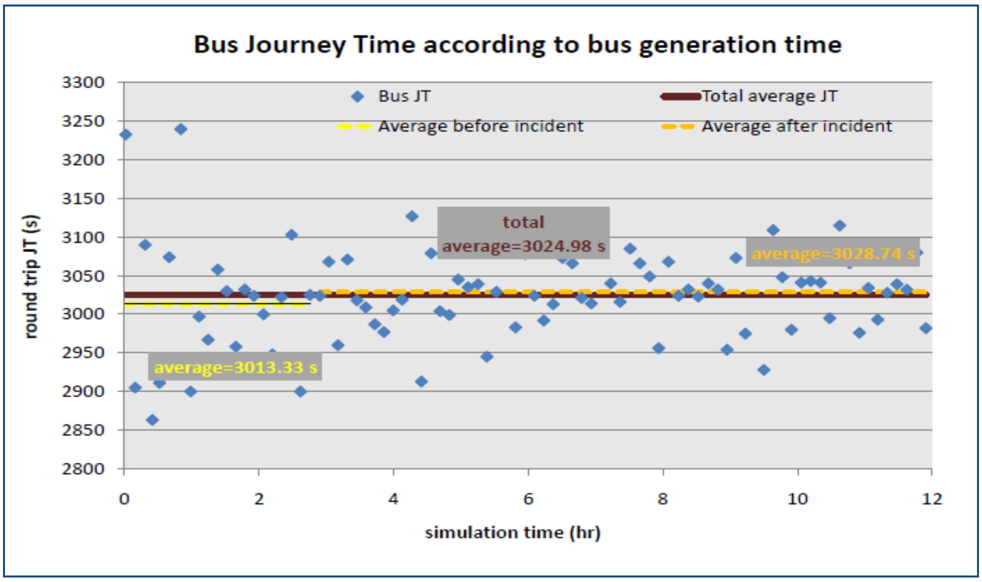


Figure 6.13: Bus journey time for 'other type' incident scenario '2111'.

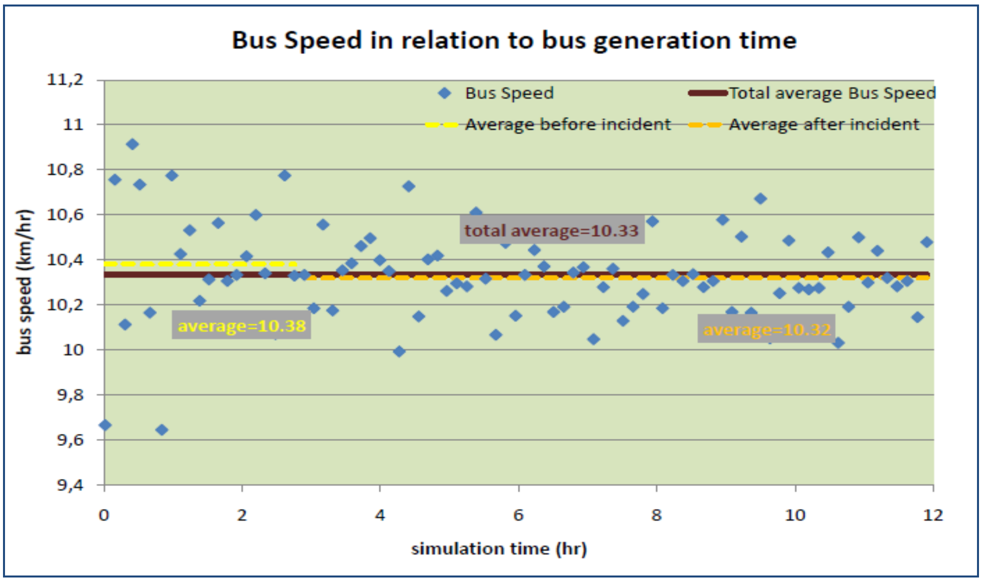


Figure 6.14: Bus speed for 'other type' incident scenario '2111'.

Besides collecting and presenting model results for the change of bus journey time along the simulation time, average values for the period before and after the incident as well as overall averages for both parameters (i.e. journey time and speed of buses) have been calculated and illustrated in Figures 6.13 and 6.14. According to the results, the total average bus journey time for scenario 2111 is 3024.98 seconds and the total average bus speed was found to be 10.33 km/hr. Similarly the methodology carried out during the remaining simulation runs and attention was drawn to all 3 key performance parameters. Enabling the respective options at the Input-Dialog box which comes up as soon as the simulation starts and, thus, applying the model to various incident characteristics led to vital outcome for each modelled case. Table 6.4 summarizes the simulation results related to average bus journey times. The effect on the journey time once the incident occurs is evident and the total average journey time was found to vary between 3024.98 and 3270.76 seconds when the incident location is at the middle of the bus route and between 3025.06 and 3112.58 seconds when the incident takes place at the end of the route. The same concept was followed for the calculation and presentation of bus speed, with results including partial and average values as shown in Table 6.5. Total average bus speed varies from 10.33 to 9.80 km/hr according to the location, severity and duration of the incident.

Table 6.4: Average bus journey time for all 'other type' incident scenarios.

	Average Bus Journey Time (seconds)		
Scenario Code	**JT before Incident**	**JT after incident**	**total average JT**
2111	3013.33	3028.74	3024.98
2112	3013.33	3029.68	3025.69
2113	3013.33	3033.63	3028.68
2121	3013.33	3036.69	3030.99
2122	3013.33	3187.49	3140.54
2123	3013.33	3376.76	3270.76
2211	3012.75	3028.79	3025.06
2212	3012.75	3030.29	3026.21
2213	3012.75	3031.98	3027.51
2221	3012.75	3034.12	3029.15
2222	3012.75	3070.16	3056.49
2223	3012.75	3145.31	3112.58

Table 6.5: Average overall bus speed for all 'other type' incident scenarios.

	Average Bus Speed (km/hr)		
Scenario Code	speed before Incident	speed after incident	total average speed
2111	10.38	10.32	10.33
2112	10.38	10.32	10.33
2113	10.38	10.30	10.32
2121	10.38	10.30	10.32
2122	10.38	9.94	10.06
2123	10.38	9.56	9.80
2211	10.38	10.32	10.33
2212	10.38	10.31	10.33
2213	10.38	10.31	10.33
2221	10.38	10.30	10.32
2222	10.38	10.21	10.25
2223	10.38	10.03	10.12

For the purposes of a comprehensive understanding of the simulation results and in order to set the grounds for a multidimensional comparison and evaluation of the scenarios modelled, the results were also analysed in terms of average excess waiting time. Scenarios were divided in two groups, as categorized during the simulation of breakdown scenarios in Section 6.4.2.2, according to the incident location, and the average excess waiting time per bus stop was then computed. Figures 6.15 and 6.16 demonstrate the values for this crucial performance indicator, with scenarios of the first group (see Figure 6.15) giving substantially higher excess waiting time values than the respective scenarios of the second group (see Figure 6.16).

However, in order to extract sufficient and complete conclusions about the model results, a thorough comparison of the output analysis is required. For the purposes of the model results evaluation, the next section focuses on investigating the effect of each of the main incident components on the key model performance parameters and evaluating the modelled scenarios.

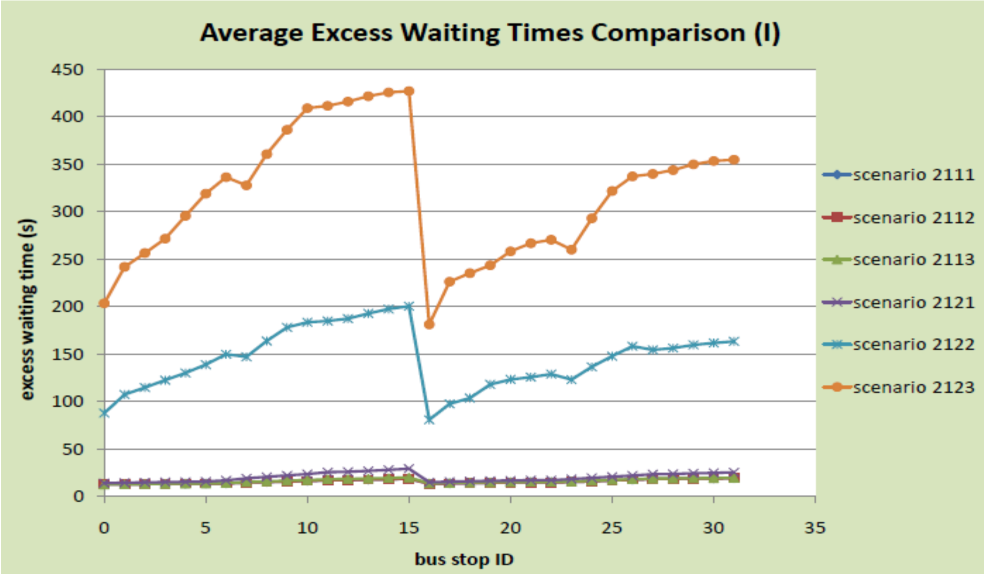


Figure 6.15: Average excess waiting times for 'other type' incident scenarios when the incident occurs at the middle of the route.

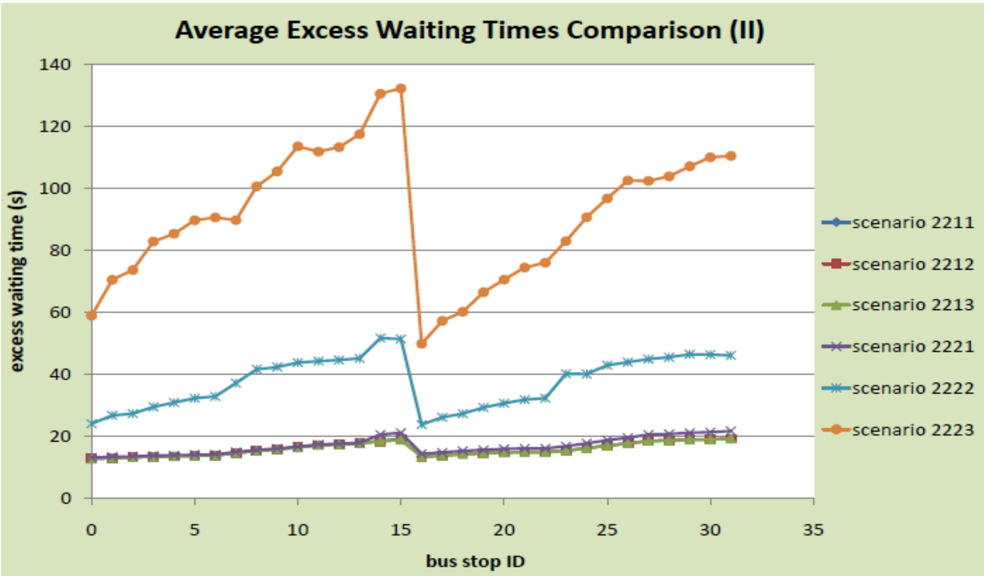


Figure 6.16: Average excess waiting times for 'other type' incident scenarios when the incident occurs at the end of the route[7].

[7] The complete description of the the scenario codes used in Figures 6.15-6.16 is provided in Table 6.1

6.5 Comparison and Evaluation of Results

The principal of simulation modelling is to apply the model to various scenarios, produce simulation results related to specified performance parameters and reach conclusions through the comparison of the results. SIBUFEM was applied to the base case or 'no incident' scenario, 12 breakdown scenarios and 12 'other type' incident scenarios. The simulation outcome for these 25 simulation runs was compared to evaluate the effect of each of the incident characteristics on the bus operation. Excess waiting time being a key model indicator of SIBUFEM was used, during the first stage of results comparison, to evaluate the effect of the incident parameters on the bus operation performance. The comparison was continued with focus on the remaining two performance parameters.

6.5.1 Effect of incident type on excess waiting time

In order to estimate the impact of incident type on the bus operation, both types of incident were compared against the base case scenario in terms of excess waiting time with Figure 6.17 demonstrating the outcome of this comparison.

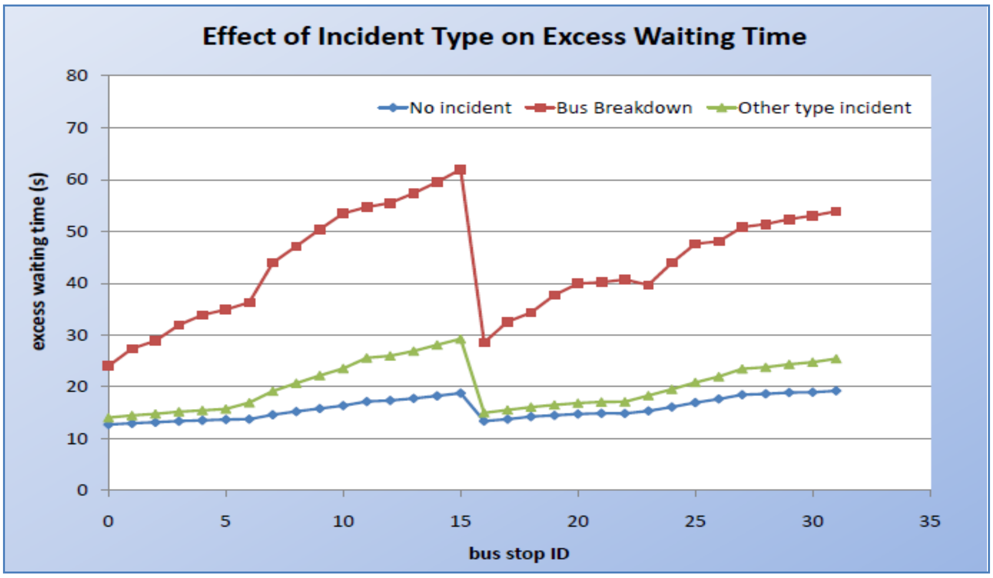


Figure 6.17: Effect of incident type on excess waiting time per bus stop for scenarios '0000', '1121' and '2121'.

For the purposes of this comparison, scenario '0000' (i.e. base case), scenario '1121', (i.e. bus breakdown of medium severity taking place at the middle of the route for short duration) and scenario '2121' (i.e. 'other type' incident of medium severity taking place at the middle of the route for short duration) were used. Figure 6.17 shows that excess waiting time is considerably higher for the case of bus breakdown incident than for the respective ‘other type’ event scenario. In other words, a bus breakdown causing 40% roadway capacity reduction at the middle of the bus route for 20 minutes time has greater effect on excess waiting time than a traffic accident, an illegal parking or a disabled vehicle causing the same capacity reduction, at the same place and for same duration. Simulation results show that a bus breakdown scenario is expected to cause 43.63 seconds average excess waiting time (average of all bus stops), while the same value for an ‘other type’ event case is just 20.15 seconds. If these results are compared against the base case, which causes an average value of 15.80 seconds for the same performance indicator, the impact of the bus breakdown and the ‘other type’ event scenarios in terms of excess waiting time increase are 176.1% and 27.5 % respectively.

However, in order to reach efficient conclusions concerning the incident types’ comparison, all modelled scenarios were compared following the same concept based on the type of the incident. For the needs of the comparison and the presentation of the outcome, scenarios were categorized into two groups according to their incident location. The aim is to compare scenarios that retain the same incident characteristics (i.e. starting time, location, severity, duration) but differ on the incident type parameter. Figure 6.18 summarizes the comparison results, with the graph on the left representing scenarios occurring at the middle of the route and the graph on the right those occurring at the end of the route.

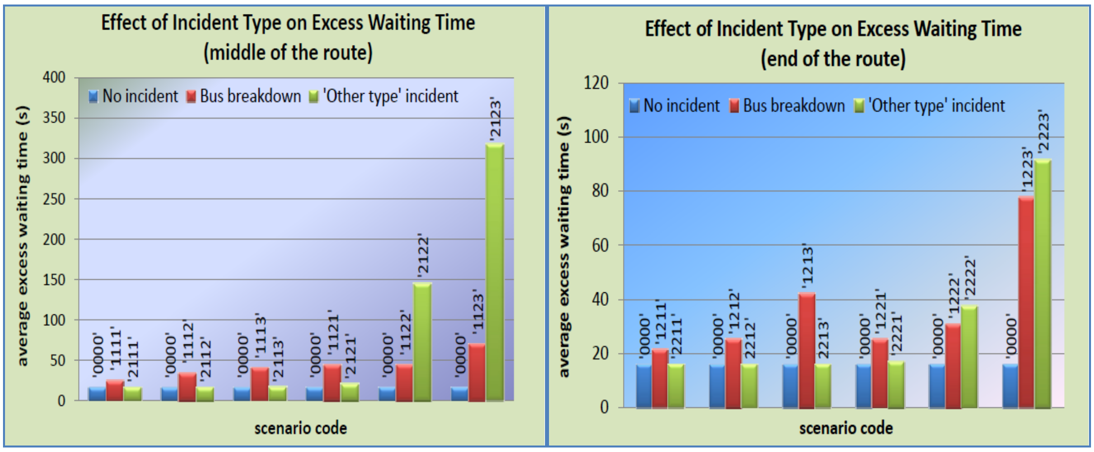


Figure 6.18: Effect of incident type on average excess waiting time[8].

For the needs of efficient presentation of the effect of incident type, scenarios are represented by the respective code in the above graphs. According to Figure 6.18,

[8] The model results for the incident case scenarios were compared to the results for the Base Case scenario. The list describing the scenario codes used in Figure 6.18 is presented in Table 6.1.

the comparison between the two incident categories modelled shows that the bus breakdown scenarios are responsible for higher average excess waiting times when the severity is slight. However, the same conclusion does not apply to the medium severity incidents. While scenarios ‘1121’ and ‘1221’ (i.e. for incident duration 20) give higher excess waiting time than cases ‘2121’ and ‘2221’, when incident duration is longer than 20 minutes the situation is reversed; for incidents lasting longer than 40 or 60 minutes ‘other type’ incident cases have a considerably greater impact on the equivalent bus breakdown scenarios as Figure 6.18 demonstrates. Investigating the effect of the location, severity and duration of the incident is vital to reach further model conclusions.

6.5.2 Effect of incident location on excess waiting time

Incident location is one of the incident components that affect the bus operation, depending on the specific characteristics of the route, such as the length of the links (i.e. length between two consecutive bus stops) upstream the location, the speeds on the links and the passenger numbers. These characteristics are responsible for the change of the model results according to the location. An example demonstrating the impact of the incident location on the model performance is shown in Figure 6.19. Simulation results for the scenarios ‘1111’ (i.e. bus breakdown of slight severity taking place at the middle of the route for short duration) and ‘1211’ (i.e. bus breakdown of slight severity taking place at the end of the route for short duration) are compared against the model case for this purpose.

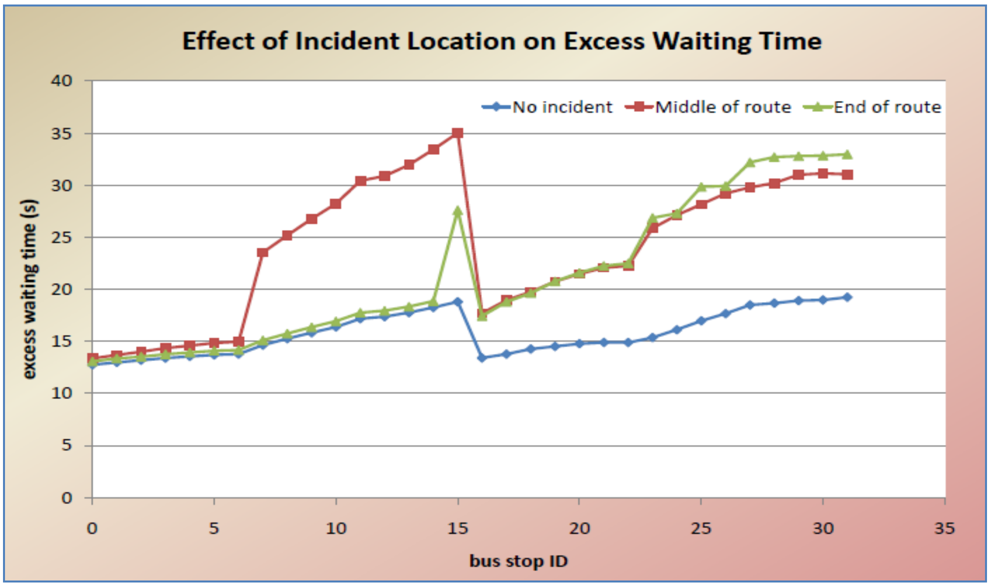


Figure 6.19: Effect of incident location on excess waiting time per bus stop for scenarios ‘0000’, ‘1111’ and ‘1211’.

The effect of the event is evident in Figure 6.19, with the increase in excess waiting time being substantial for the scenario ‘1111’ after bus stop 6 and for the scenario ‘1211’ after stop 14. In SIBUFEM, bus stop 6 and bus stop 14 are defined as the two options for location of the incident, thus, the occurrence of the event, which in this case is a bus breakdown, causes delay in the waiting time of passengers at the next stop (i.e. bus stop 7 or bus stop 15, depending on the scenario); consequently even more passengers are waiting at this stop causing even further delay and higher passenger waiting time. Therefore, the behaviour of the curve at this stop indicating the considerable rise in average excess waiting time was expected. Nevertheless, it is interesting that the excess waiting time follows a very similar pattern for the return leg of the bus trip.

A more detailed investigation of the complete range of scenarios followed this initial comparison, which aimed to reach conclusions on the incident location impacts. Figure 6.20 describes the impact of location on average excess waiting time, where the graph on the left summarizes the comparison results for all bus breakdown scenarios and the graph on the right presents the results for the respective ‘other type’ incident cases.

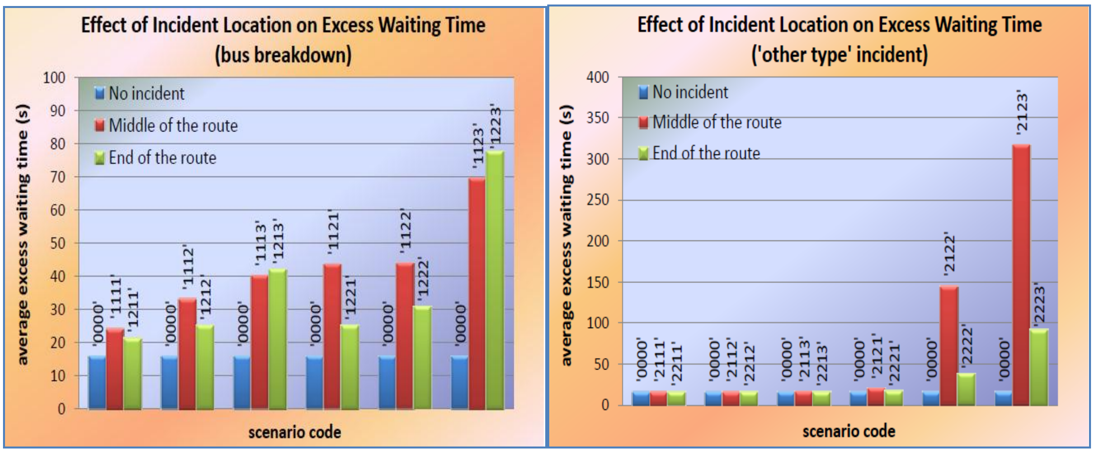


Figure 6.20: Effect of incident location on average excess waiting time[9].

The above graphs show that, in the case of a bus breakdown, the excess waiting time is higher when the event occurs at the middle of the route than when it takes place at the end of it for short or medium durations. However, for a long duration bus breakdown, the impact of the event taking place at the end of the bus route is greater than the scenario of the event occurring at the middle of the route. These conclusions may not be broadened to include the ‘other type’ scenarios, since the

[9] A complete description of the scenario codes used in Figure 6.20 is provided in Table 6.1

graph on the right in Figure 6.20 shows that the excess waiting time is always higher for scenarios of the incident occurring at the middle of the route; especially for incidents lasting 40 or 60 minutes the difference in excess waiting time values is considerably larger.

6.5.3 Effect of incident severity on excess waiting time

Having reached some conclusions regarding the impacts of the type and location of the incident, attention was drawn to the behaviour of another incident parameter, the severity component. A medium severity incident might be expected to cause a higher excess waiting time than the same event of a slight severity. In order to investigate this, three scenarios (i.e. ‘0000’, ‘1113’ and ‘1123’) were looked into in detail and the comparison produced the results shown in Figure 6.21. Scenario ‘1113’ represents a bus breakdown of short severity taking place at the middle of the route for long duration, while ‘1123’ refers to the equivalent scenario of a medium severity.

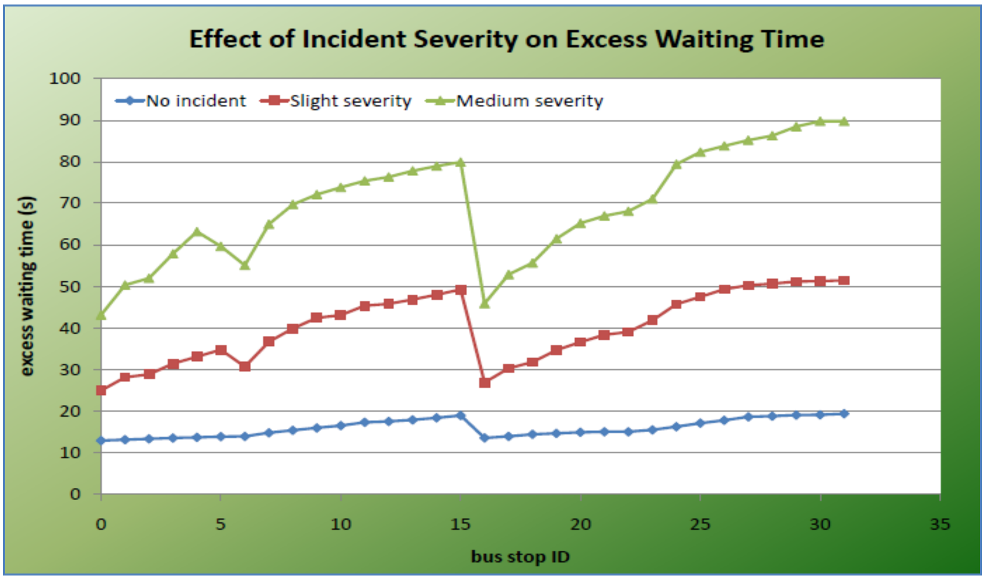


Figure 6.21: Effect of incident severity on excess waiting time per bus stop for scenarios ‘0000’, ‘1113’ and ‘1123’.

According to the simulation results, the lines describing the two scenarios of slight and medium severity follow the same pattern, with the line representing the medium severity case being shifted upwards in comparison to the slight severity scenario. An incident of medium severity was found to therefore have a greater impact on the excess waiting time than the equivalent slight severity event. A wider comparison of all the modelled scenarios with respect to the severity parameter is depicted in Figure 6.22. The graphs confirm that the medium severity incident scenarios have greater effect on the average excess waiting time than the slight severity incident ones and show that the impact of severity is more determinant in the case of an 'other type' incident scenario than in the case of a bus breakdown. The comparison against the base case shows that the maximum increase in average excess waiting time is 5.02 minutes for 'other type' incident scenarios, which is produced by scenario '2113', while maximum value for bus breakdown scenarios is just 1.03 minutes produced by case '1213'.

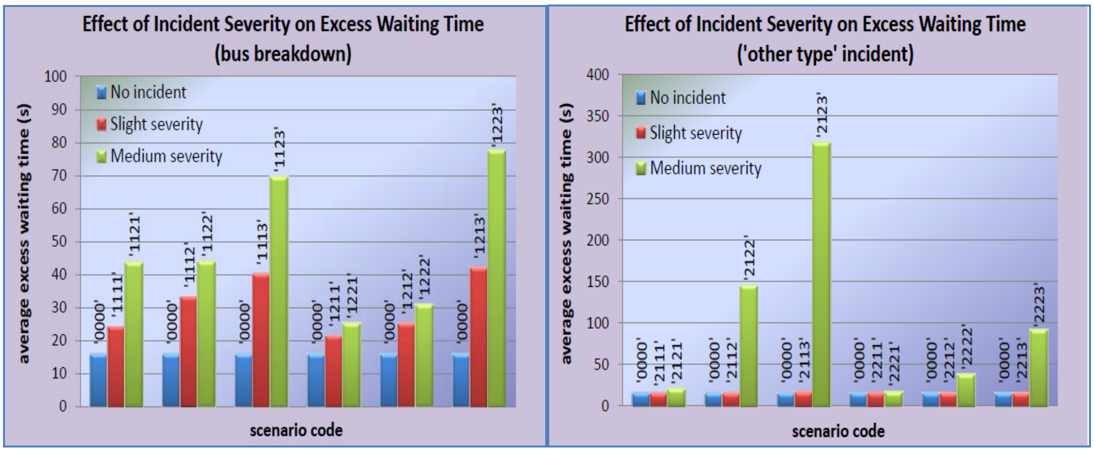


Figure 6.22: Effect of incident severity on average excess waiting time[10].

6.5.4 Effect of incident duration on excess waiting time

The behaviour of the incident duration parameter was the next step of the simulation results comparison. Similarly to the methodology carried out during the investigation of the impact of the rest of the incident characteristics, a detailed comparison of all scenarios was carried out. Figure 6.23 summarizes this comparison in two graphs, one for each category of incidents modelled.

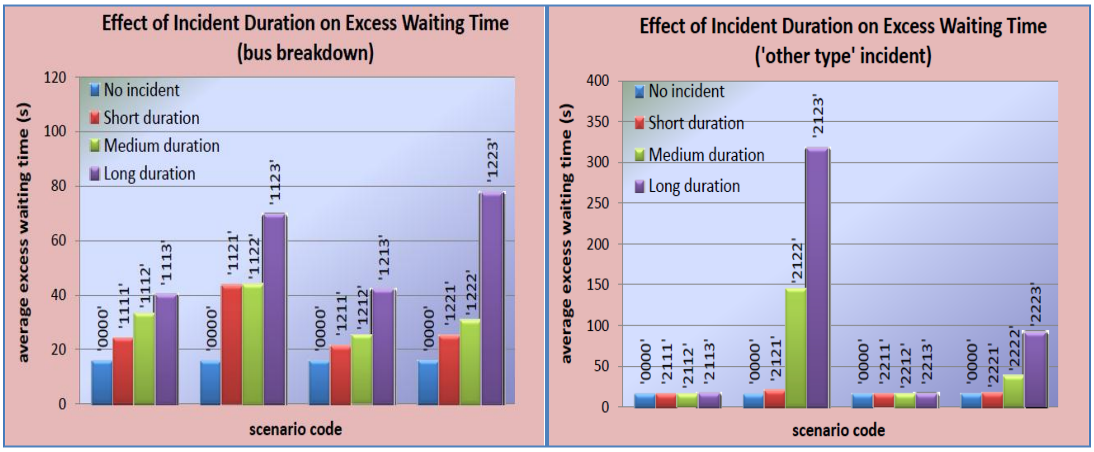


Figure 6.23: Effect of incident duration on average excess waiting time.

[10] A complete description of the scenario codes used in Figure 6.22 is provided in Table 6.1

The results demonstrate the impact of the incident duration element on the extent of normal bus operation disruption. According to Figure 6.23, bus breakdown scenarios of duration of 60 minutes (i.e. long duration) produce 1.9 to 6.6 times more average excess waiting time increase than the breakdown scenarios of 20 minutes duration. On the other hand, in the case of 'other type' incidents, the impact of duration is evident for medium severe incidents, while for slight severe events the excess waiting time is virtually unaffected.

An example of exploring the effect of incident duration is provided in Figure 6.24, which demonstrates the comparison of scenarios '1211', '1212' and '1213' against the base case scenario. All three bus breakdown scenarios occur at the middle of the route (i.e. bus stop 14). However they represent events of different duration of 20, 40 and 60 minutes respectively.

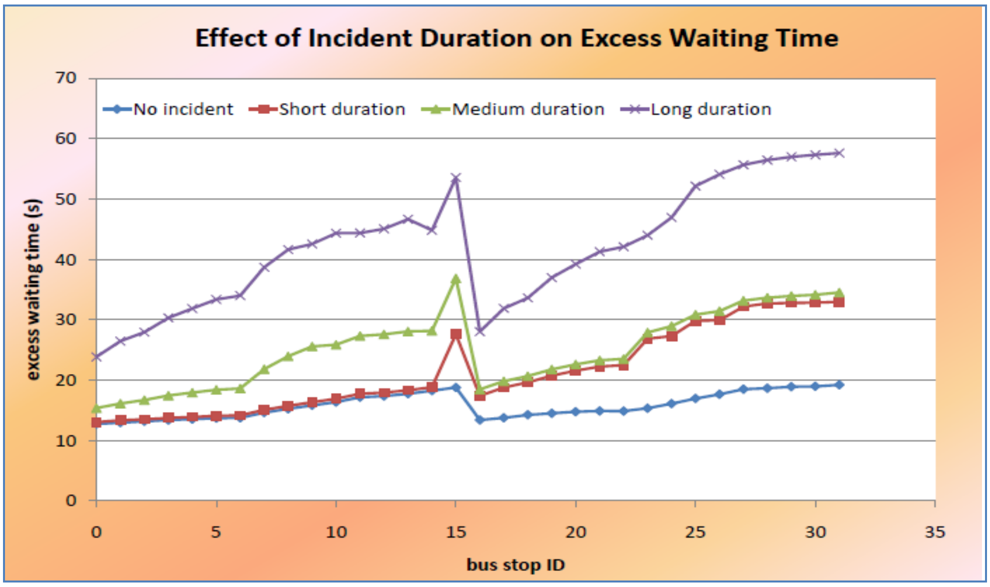


Figure 6.24: Effect of incident duration on excess waiting time per bus stop for scenarios '0000', '1211', '1212, and '1213'.

The graph illustrates the impact of the incident duration and, according to calculations, the average excess waiting time for scenario '1211' is 21.3 seconds, for scenario '1212' 25.1 seconds and for scenario '1213' 42.0 seconds. If these results are compared against the base case, it may be concluded that the 40 and 60 minutes scenarios produce 1.7 times and 4.8 times higher average excess waiting times than the 20 minutes case respectively. The effect that the incident duration has on the model performance parameters is clear and the longer the duration of the incident, the greater the impact caused.

6.5.5 Effect of incident on bus journey time and speed

Even though excess waiting time is a crucial model performance parameter and may offer significant contribution towards the investigation of the behaviour of the incident parameters and their effect on the bus operation, average bus journey time and average bus speed are equally important performance indicators. In order to lead to efficient conclusions concerning the simulation outcome, results comparisons were carried out based on these two parameters.

For the purposes of the model results comparison, scenarios were grouped into two categories according to the incident type that they involve. The effect of location, severity and duration of the incident on average bus journey time was studied through scenarios comparison and Figure 6.25 demonstrates an example of this comparison. The graphs provide the increase in bus journey time compared to the base case scenario. Bus breakdown scenarios ‘1111’ and ‘1211’ are used to explore the effect of location, scenarios ‘1113’ and ‘1123’ for the severity effect and scenarios ‘1121’, ‘1122’ and ‘1123’ for the impact of duration incident. The impact of the parameters on average bus journey time for the equivalent cases of ‘other type’ incident is also illustrated in Figure 6.25. More generalised outcome on the comparison and evaluation of the model results is provided in Chapter 7, where average performance measure changes caused by all modelled incident scenarios are used.

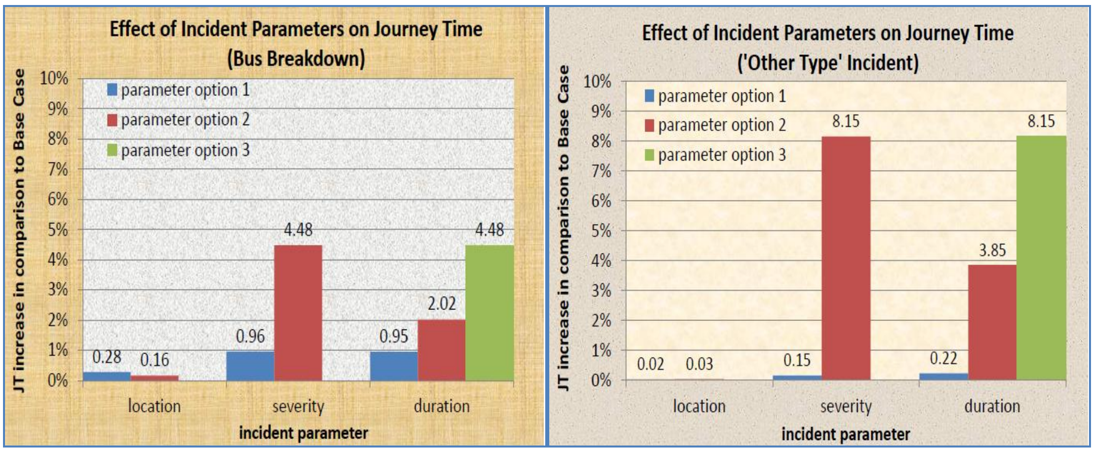
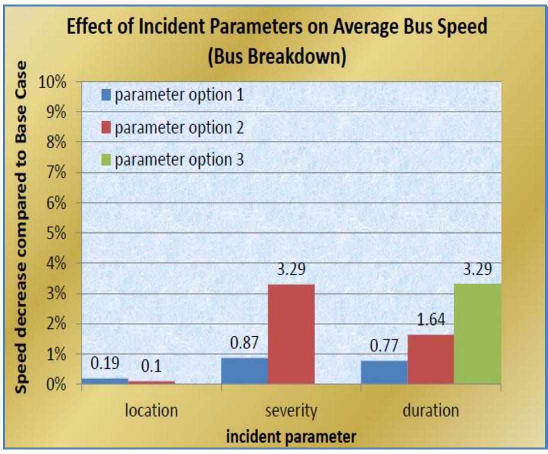


Figure 6.25: Impact of incident parameters on average bus journey time.

While the location parameter, according to Figure 6.25, has a minor effect on the bus journey time, severity and duration contribute to up to an 8.15% increase in comparison to the no incident case journey time. For example, a slightly severe bus breakdown causes 0.96% increase, which is 0.48 minutes, while a medium severe bus breakdown causes 4.48% rise, which is 2.26 minutes.

The behaviour of the average bus speed according to the incident characteristics was another interesting element of the model results analysis and evaluation. Comparisons between all scenarios were carried out to investigate the effect of the incident parameters on this performance parameter and an example is highlighted in Figure 6.26. According to the simulation results, the impact of incident severity and duration on average bus speed is crucial, while the effect of the location parameter is barely noticeable.



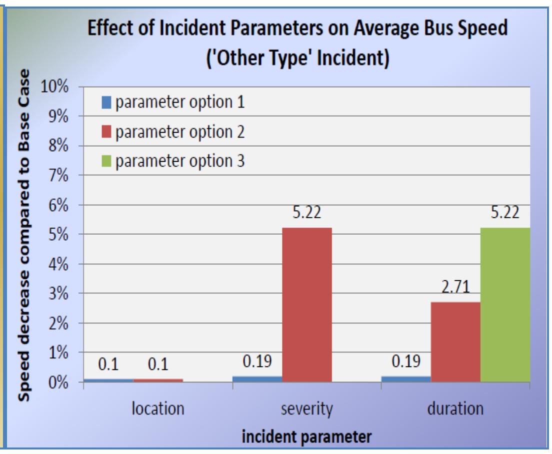


Figure 6.26: Impact of incident parameters on average bus speed.

6.6 Chapter Summary

The development of the innovative simulation model SIBUFEM, which investigates the bus and traffic incidents modelling into the continuous circulation of buses in a high frequency bus service, offers the opportunity to apply it to various scenarios incorporating the incident components and explore their effect on the key model performance indicators. Excess waiting time, bus journey time and bus speed set the grounds for the comparison of the simulation results, which demonstrated that incident characteristics play a vital role during the estimation of the impacts of an incident.

Looking into the effect of the incident type, a bus breakdown incident has a greater impact than an ‘other type’ incident unless the severity is medium and duration longer than 40 minutes. Concerning the incident location, the impact is usually higher when the event occurs at the middle of the bus route than when it takes place at it the end. The only exception to this is bus breakdown scenarios of 60 minutes in duration; in this case, the impact is higher at the end of the route. Severity and duration have a direct impact on the key model performance parameters of SIBUFEM and, according to the analysis and evaluation of the simulation results, the higher the values of these two incident components are, the greater the effect produced is.

-Chapter 7-

Model Applications and Results (Phase B)

7.1 Introduction

In order to fully address the research objectives and assess the impact of incidents on the bus performance and, hence, suggest potential fleet management strategies for improved efficiency, a second phase of model applications was carried out with particular focus on modelling two bus fleet management control actions. For the purposes of demonstrating the range of results and evaluation of simulation runs modelled, results were presented through graphs. This Chapter describes the individual characteristics of each of the two bus fleet management strategies incorporated in the model and presents the simulation model results, describing, comparing, and evaluating them which were essential in order to suggest potential control actions according to the characteristics of the occurred incident. Results from SIBUFEM were discussed in interviews with Bus Operators (see Section 4.6 for an overview of this approach) and relevant interview quotations have thereby been incorporated in this Chapter where appropriate, to demonstrate this. Finally, the summary highlights the key points of this Chapter indicating how SIBUFEM could be used by operators and users to produce a detailed strategy appraisal.

7.2 Bus Fleet Management Strategies

7.2.1 The decision making process

The purpose of bus fleet management is to improve the overall efficiency of vehicle fleet operations and thereby improve the quality of public transport services. The main aim is to enable operators to detect bus-related incidents by comparing a bus's schedule with real-time location data and take necessary actions to address the disruption caused by these incidents. Real time information systems offer the opportunity to monitor buses running along the route and deciding if action is required:

> *'What we'll do is the controllers will look at their screens, of which they have several, which shows where the buses are along the route. If the buses are light blue then they are between 1-2 minutes early, so slightly ahead of time; if they are between 1 minutes early to 5 minutes late, which is traffic conditions window, then they are green; between 5 and 10 minutes late they are yellow; and more than 10 minutes late they are red. For example here, they are reasonably well spaced apart from one which is slightly late. If you've got a late one, an early one is next to it, or vice versa. If you've got an early one it's probably because the other one in front is running late, because of course he is picking up extra passengers and the other one has fewer to pick up.'*

In case of an incident causing delays on a specific bus of the line, bus controllers contact the bus driver to enquire about the nature of the delay and get an update of the situation. In addition, information provided by their network overview regarding the time delay of the specific bus offers them enough evidence to decide whether an action is required or not.

> *'The controllers have a network overview which shows for each of the routes which bus is the worst, so they can have a look at some of these routes with red ones. The idea is we would contact the driver. He'll get to the end and ask what should I do? And then we might tell him to run empty to the other end or to a specific bus stop of the route. Or one of the strategies is asking a bus driver which runs late to 'run empty' up to a*

> *certain bus to get back on time, while the next one is asked to continue to run late and come out and run empty on the other end of the line.'*

A variety of bus fleet management strategies is applied according to the nature and characteristics of the incident, as described in Table 2.3 in Section 2.5.1. Through direct liaising with bus operators, two of these options were found to attract substantial interest as they are widely used in bus operations in the UK. These two strategies, which refer to reinforcing the service and increasing the speed of individual buses, were the main focus of this second phase of simulation runs. Each of these control actions has been modelled in SIBUFEM and a series of model results including the comparison of the model outcome to the No-Strategy option scenario has been carried out.

7.2.2 Reinforce bus service

Whether a bus related incident causes an individual delay, where one of the buses in a line is delayed, or a generalised delay, where several buses are delayed, bus operators may choose to add a reserve bus to the line. Especially in cases of saturation, where passengers can not take the bus because it is full, or a bus breakdown, the strategy of reinforcing the bus service might be very beneficial towards restoring the agreed bus service level or reinforcing the frequency of the line:

> *'This route for example is the busiest route of all. We were finding here that passenger loadings were very high, evening peak time in particular, so I took the opportunity here to put 2 extra buses just during peak time, so here whereas it only runs every 7 minutes in the peak time, its just every 5 minutes between 4.45pm-5.15pm.'*

This particular bus fleet management option, which suggests that both an extra bus and a driver are available, means that the operators decide to include this additional bus into the line to accommodate delays caused by an incident affecting the overall bus performance. In some cases, when there are no available reserve buses (as is the case in Malaga, Spain, for example), an alternative of this strategy is used including a bus from another line (Belmonte and Fernandez, 2005). For the purposes of model application, it has been assumed that a spare bus is available in the system and may be used as a control strategy to accommodate the incidents' effects. This assumption is justified by bus operators:

> *'We have a certain number of spare buses, because every day is different. We've got about 18% spare buses but from them the engineers have their requirements which vary and then buses that they don't need are for all the operational requirements which again vary.'*

In SIBUFEM, the generation point of the extra bus is the same as the generation point of the rest of the buses and is located at the beginning of the route, which is the bus service's depot.

7.2.3 Increase speed of individual buses

Another bus fleet management strategy, which does not require the availability of extra buses, is increasing or decreasing the speed of one or more individual buses. Bus operators do take this action after communicating directly with bus drivers of buses that are behind the schedule. Action is taken immediately in this case but bus speeds are relevant to the traffic conditions and bus speed limitations. An indirect approach to increase the average bus speed of certain buses is reducing the waiting time of buses at each of the ends of the route in order to tackle the incident delays caused. Every bus schedule has to build in enough time, called slack time, to accommodate for any delays, as confirmed by operators:

> *'This one was late earlier on, but the schedule has to build in enough time to make up for any delays. Any time at the end of the route of course is unproductive. We tend to allow if possible about 10 minutes in the hour but of course it depends on the characteristics and the length of the route.'*

The increase of bus speed of certain buses as a bus fleet management option has been explored in SIBUFEM. The modelling methodology followed was to increase the bus speed of the buses upstream the incident's location by 10% for the time period that the event is taking place and provide model results showing the impact of this strategy on the overall bus performance.

7.3 Model Results

An extensive number of simulation runs were carried out to provide sufficient evidence on the behaviour of each bus fleet management strategy during this second phase of model application. The simulation output of the No-Strategy scenarios for each of the various bus breakdown and ‘other type’ cases was stored and expanded to provide values for the main bus performance parameters. Simulation runs modelling the incorporation of each of the bus fleet management strategies outlined in Section 7.2 to address each incident scenario were carried out, leading to 48 new simulation runs, 24 for each fleet management option. The simulation results of SIBUFEM and the subsequent output comparison carried out for the various bus breakdown scenarios as an evaluation method is described in this section.

7.3.1 Bus breakdown scenarios

Each of the 12 scenarios described in Section 6.3.2.1 and presented in Table 6.1 were modelled initially without applying any bus fleet management strategy (i.e. Do-Nothing option) and then using each of the two control actions investigated, the option of reinforcing the line with an extra bus and the strategy of increasing the speed of the buses by 10%, offering a total of 36 bus breakdown simulation runs overall throughout both application phases. An example of model results showing the impact of the incident on average excess waiting time and the effect of the bus fleet management strategies applied to address the delays caused by the incident in the case of the ‘1121’ incident scenario (i.e. a bus breakdown of medium severity which occurs in the middle of the route for 20 minutes time) is presented in Figure 7.1.

According to calculations on the change of this key bus performance parameter caused by the bus fleet management actions, an average of 61% decrease of the average excess waiting time occurred compared to the incident case scenario by adding a spare bus into the route (i.e. BFM1), whereas the impact of increasing bus speeds (i.e. BFM2) on the same performance parameter was found to cause a 44.5% decrease on average.

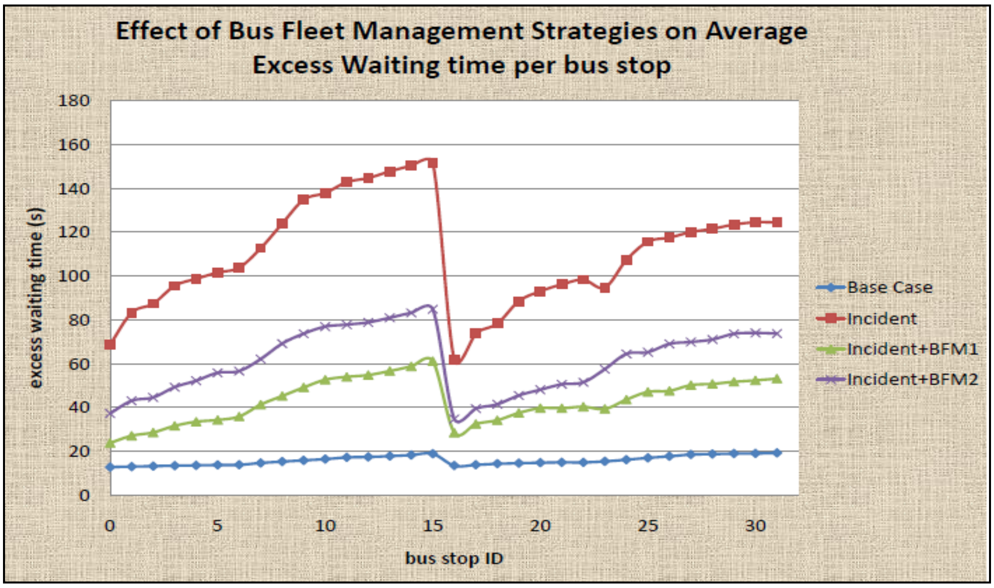


Figure 7.1: Impact of bus fleet management strategies on average excess waiting times per bus stop for the bus breakdown scenario '1121'.

Each of the incident scenarios was simulated using SIBUFEM for a period of 12 hours and the simulation output was stored according to their code number. The simulation period of 12 hours was required in order to investigate the behaviour of the bus fleet management strategies and their effect on the overall bus performance during whole day periods. Figure 7.2 illustrates the model outcome of SIBUFEM for bus breakdown scenarios showing the effect of the two bus fleet management strategies on the average excess waiting time, which, according to direct liaising with bus operators, is considered to be the most crucial bus performance parameter.

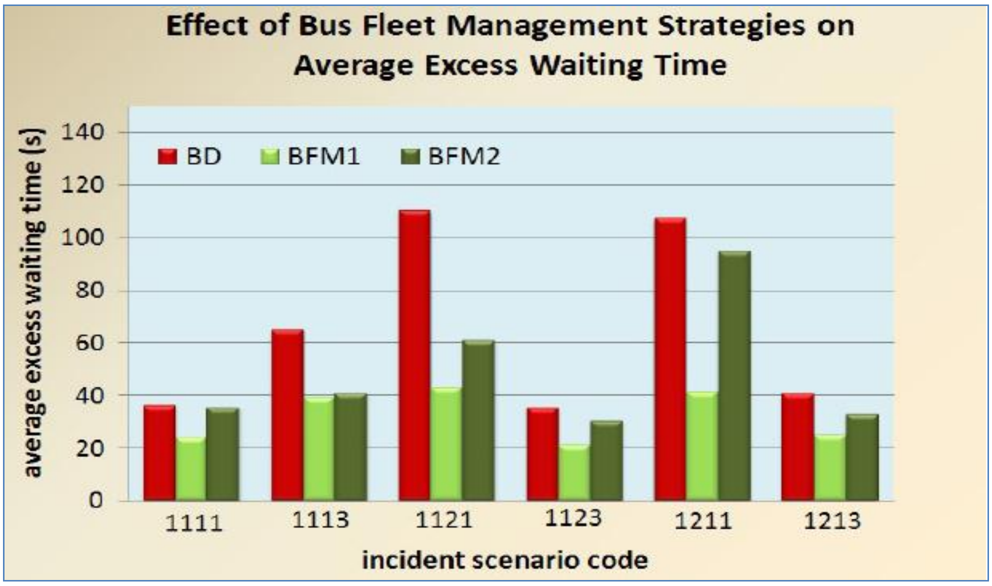


Figure 7.2: Model outcome of SIBUFEM showing the effect of bus fleet management strategies on the overall bus performance

The graph presents the average waiting times for 8 simulation scenarios, each describing an incident with different levels of location, severity and duration, which are indicated by their unique scenario code. The red columns refer to the breakdown scenario ('BD') when no fleet management strategy is applied and describe the Do-nothing strategy. Scenarios including the first bus fleet management option ('BFM1', i.e. reinforce the line with an extra bus) are illustrated in light green, whereas the dark green coloured columns refer to adopting the second bus fleet management strategy ('BFM2', i.e. increase the bus speeds by 10%).

Each of the two modelled bus fleet management actions has a direct impact on the average excess waiting time, which is slight or severe according to the characteristics of the incident occurring. That is the first key finding regarding the model results of this second phase of application using SIBUFEM, which can be further enriched by the general conclusion that the option of reinforcing the line with an additional bus seems to be more beneficial for the bus performance compared with increasing bus speed. This is complimented by the operators' views on the benefits that the action of inserting an additional bus into the line offers.

> *'If a route is valuable to us, and one of our best routes, then its important that the buses run to time which may require an extra bus, which can serve two purposes: if it works well you can put an extra bus in a route like No 7 during peak time, and you can provide an extra journey as well as slackening it out a bit.'*

7.3.2 Comparison and evaluation of results

A further comparison and evaluation of the model results was required to reach more accurate and detailed concluding comments regarding the behaviour of each strategy in each incident scenario. Therefore, the relationship between the effect of each modelled bus fleet management option on the bus performance and each incident's parameter individually was further investigated in order to reach conclusions appraising the merits of each strategy and suggesting the more beneficial option to address the delays caused according to the incidents' characteristics.

The calculation of excess waiting time for each of the modelled scenarios was carried out and results of various incident model scenarios were compared against each other to explore the behaviour of each fleet management strategy under different incident conditions. The average excess waiting time is a model performance parameter of high importance for the evaluation process, as it reflects the additional time that a passenger needs to wait at a bus stop and, considering that the value of passenger waiting time is estimated as double the passenger journey time value in bus operations (HEN 2, 1997), it was chosen as a key model indicator in the evaluation of results. Both types of fleet management actions were compared against the Do-Nothing strategy with Figure 7.3-7.5 demonstrating the outcome of this comparison according to each of the incident parameters of location, severity and duration.

Comparisons of incident scenarios differing in location, such as the pairs of '1113' and '1213' or '1121' and '1221', revealed that the line reinforcement with an extra bus offers substantially greater benefits in terms of a decreased average excess waiting time. Comparing the above incident pairs, simulation results using SIBUFEM showed that the two bus fleet management strategies caused comparable decrease of the excess waiting time when the incident took place in the middle of the route, especially for incidents of slight severity, with the decrease of the excess waiting time due to the first and second strategy found to be 39.4% and 37% respectively. However, implementing the line reinforcement strategy outperformed the strategy focussed on increasing bus speed when the severity of the incident was medium with the effects changing to 61% and 44.5% for each strategy respectively.

This situation differed when the incident occurred at the end of the route. The strategy to reinforce the service line provides a substantial advantage in comparison to the action to increase the bus speed as shown in Figure 7.3.

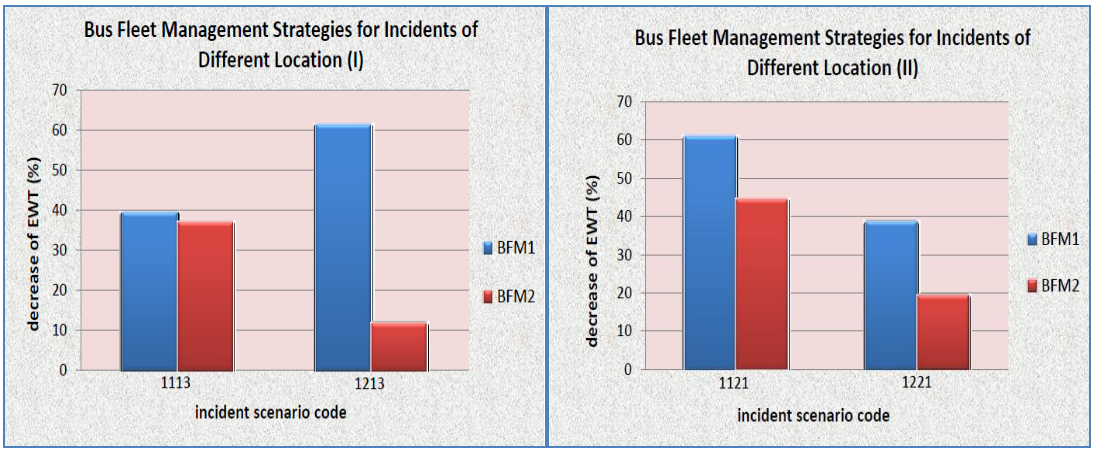


Figure 7.3: Effect of bus fleet management strategies on bus performance according to the incident's location[11].

[11] BFM1 and BFM2 represent the actions to reinforce the line by adding an extra bus and increase the bus speed respectively.

The same methodology was followed for the evaluation of the model results regarding the parameter of the incident's severity. Figure 7.4 illustrates a characteristic example of the comparison of incident scenarios describing the behaviour of each of the two control actions relevant to the level of severity of the incident. For slight severity incidents (i.e. incidents causing 25% decrease of the roadway capacity), reinforcing the line offered substantially greater benefits than increasing the bus speed, for short duration incidents. When the incidents of a slight severity are of a long duration the positive impact of both strategies was found to be roughly equivalent. For medium severity incidents (i.e. incidents causing 40% decrease of the roadway capacity), reinforcing the line by adding an extra bus caused a substantially larger decrease of the excess waiting time as opposed to increasing the bus speed.

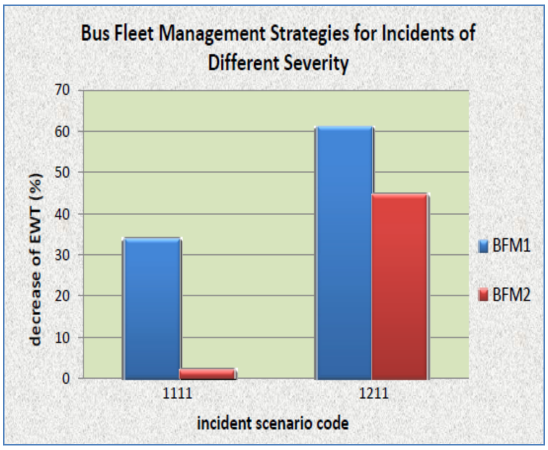


Figure 7.4: Effect of bus fleet management strategies on bus performance according to the incident's severity.

The effect of the two bus fleet management actions modelled in SIBUFEM according to different levels of incident duration was also investigated using model comparisons and analysing graphs, such as the ones presented in Figure 7.5. Overall, independently of the incident's duration, the first strategy to reinforce the line with a reserve bus was found to significantly more beneficial than the second option of increasing the bus speed. More specifically, for incidents occurring in the middle of the bus route, the former strategy offered better results, causing a 33.7% decrease in the average excess waiting time, than the latter strategy which only caused a 2% decrease of the same parameter, for short duration incidents. However, the decrease of the excess waiting time was very similar for long duration incidents. On the other hand, when the incident took place at the end of the route, the line reinforcement strategy outperformed the second fleet management action independently of the duration parameter.

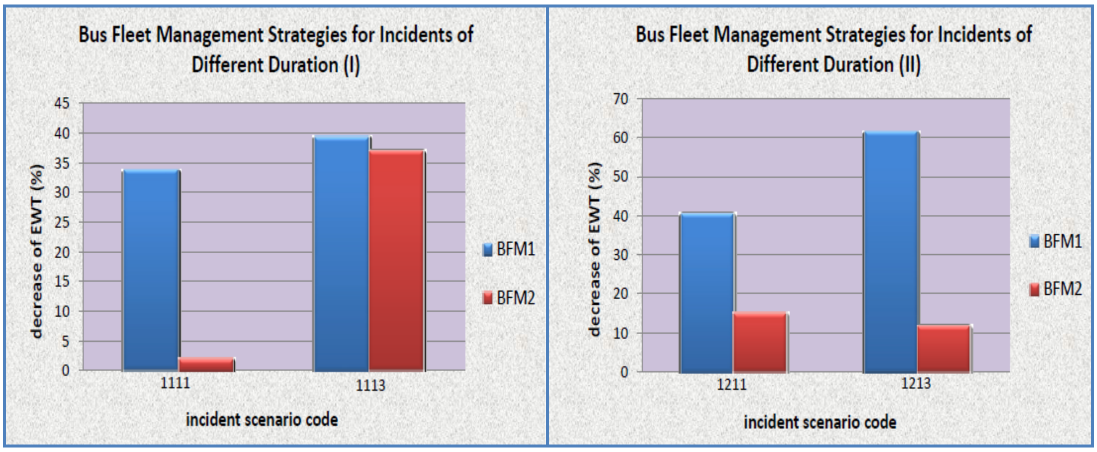


Figure 7.5: Effect of bus fleet management strategies on bus performance according to the incident's duration.

Further comparisons were carried out between the rest of the modelled incident scenarios to reach accurate and reliable conclusions about the impact of these two bus fleet management strategies on the overall bus performance and, thus, suggest potential dynamic bus fleet management actions to accommodate the delays caused by incidents of different characteristics. Overall it can be concluded that the option to reinforce the line with an extra bus was found to be more advantageous in terms of an average excess waiting time decrease. In some cases a substantial reduction of up to 62% compared to the Do-Nothing incident case was caused by applying this action to the bus service. On the other hand, an increase of speed of some buses by 10% had a positive impact on the bus performance in some cases, a neutral effect (for example a 2% change of the excess waiting time for the '1111' scenario) in other cases and, finally, a negative impact in rare cases, which indicates that the decision making process of bus operators is extremely dependent on each particular scenario, taking into account the unique characteristics of the occurring incident.

7.4 Chapter Summary

In order to expand the model application of SIBUFEM and support Dynamic Bus Fleet Management, a second phase of application was carried out to investigate the impact of each incident parameter and suggest bus fleet management options to address the delays caused by each incident. SIBUFEM investigates the bus and traffic incidents modelling into the continuous circulation of buses in a high frequency bus service and offers the opportunity to apply the model to various incident scenarios. It explores their effect on key model performance indicators and investigates the incorporation of various bus fleet management options and their impact on the bus service.

Two fleet management strategies were identified, modelled and investigated in SIBUFEM, with the research focus being on the impact of each of these control actions on the key performance parameter of average excess waiting time. Overall, the strategy of reinforcing the line with an extra bus outperformed the control action of increasing the speed of buses to address incident delays. In some cases, though, the two options were found to have a very similar level of impact on the performance of the bus operation, which then leads to the dilemma of choosing between them by using further criteria. In terms of the average excess waiting time decrease reinforcing the line with an additional bus seemed advantageous. However, this strategy suggests the availability of an extra bus and an additional driver, which one can argue demands higher operational costs for the bus operators.

> *'It didn't justify another bus in the fleet because to keep this every 10 minutes in the peak you need another bus, but what do you do with the bus the rest of the time? What do you do with it on Sundays, what do you do in the day time? It'll all have to be paid for, the whole route, the whole bus and its drivers that'll have to be paid for by this route on one extra journey.'*

Even though the operational costs of each of these strategies and their economic appraisal are beyond the scope of this research work which focuses on the benefits in terms of bus performance parameters, in terms of service reliability, further use of the model exploring economic implications of these strategies would be extremely interesting.

-Chapter 8-

Discussion and Future Work

8.1 Introduction

The model development of SIBUFEM was based on the establishment and analysis of a base case scenario of 'normal' operations for a high frequency bus service. The comparison of the base case to the results from a number of scenarios based on bus and traffic incidents, using the key model performance parameters of average bus journey time, bus speed and excess waiting time, and the incorporation of the comparison evaluation in the bus fleet management process are the key areas of focus of this research. Incidents were specified in terms of their type, location, duration and severity. Modelling various incident scenarios using the simulation model SIBUFEM offers, among other benefits, the opportunity to identify the extent to which passenger waiting times, a key criterion influencing the use of public transport, are affected by the variation of the incident characteristics. This research outlines the functionality and necessity of such a model within the context of bus functionality and sustainability and contributes towards reinforcing Dynamic Bus Fleet Management with a tool enabling bus operators to estimate the impact of incidents on the level of service of bus operations and suggest fleet management strategies. This Chapter focuses on the discussion of the model outcome, highlights the main findings and contributions of this research and investigates possible areas for further work.

8.2 Discussion on Significant Findings

The simulation model developed is capable of producing both visual and text files output enabling the visualisation and computation of the effect of various bus and traffic incidents on the model performance. The target of SIBUFEM was to establish conclusions on the behaviour of each of the incident characteristics individually and to conclude on their impact on key parameters of the bus operation level of service, as well explore the potential benefits of bus fleet management strategies.

The first stage of model application, presented in Chapter 6, provides a comparison and evaluation of the output generated from the simulation of each scenario modelled using SIBUFEM. Chapter 7 expanded on the model application, incorporating the modelling of two widely used bus fleet management options and model results evaluating the benefits provided by each strategy were presented.

The comparison of simulation results focused on the effect of incident parameters on three key performance measures: (i) average bus journey time, (ii) average bus speed and (iii) average excess waiting time. The choice of these measures was based on the fact that transit managers, schedulers, decision makers, metropolitan planning organisations and the public are all interested in these performance parameters, as these significantly influence decision making and public transport use. According to the results presented in Sections 6.4 and 6.5, each of these measures is affected to a small or large extent by the change of one of the incident characteristics. The user of SIBUFEM is provided with an option of defining these characteristics in terms of their location, severity and duration, offering a great adaptability to the model.

Section 6.5 described the evaluation of the simulation results through comparison between the incident scenarios and the base case scenario. In order to demonstrate the importance of this comparison and highlight the impacts of incident parameters on the key bus performance measures, average values representing the change that each parameter causes to each of the performance measures were calculated. For this purpose, the impact on average bus journey times and bus speeds is given in percentage of change and the effect on excess waiting times in minutes. Table 8.1 summarizes the model outcome regarding the effect of incident parameters on the bus operation for the bus breakdown scenarios modelled and Table 8.2

demonstrates the respective outcome for the 'other type' incident scenarios, including events such as traffic accidents, illegal parking, disabled vehicle and roadworks. Values in Tables 8.1 and 8.2 represent the mean value of average performance measure changes caused by each incident parameter when all possible combinations of the remaining incident parameters are taken into account. The above values provide the impact of each of the two options for the incident's location (i.e. in the middle of the route or at the end of the route), the two options for the incident's severity (i.e. slight or moderate) and the three options for the duration of the incident (i.e. short, medium or long) on three key bus performance parameters. Values in Table 8.1 and 8.2 represent the average performance measure changes caused by each incident parameter when all possible combinations of the remaining incident parameters are taken into account (Polyviou and Hounsell, 2011).

Table 8.1: Model outcome of SIBUFEM for bus breakdown scenarios

	Incident Parameter						
	Location		Severity		Duration		
	middle	end	slight	moderate	short	medium	long
Average Bus Journey Time Change (%)	1.55	1.30	0.58	2.28	0.34	1.01	2.25
Average Bus Speed Change (%)	-1.23	-1.05	-0.5	-1.77	-0.39	-0.99	-2.03
Average Excess Waiting Time Change (minutes)	0.44	0.35	0.25	0.54	0.21	0.29	0.69

Table 8.2: Model outcome of SIBUFEM for 'other type' incident scenarios

	Incident Parameter						
	Location		Severity		Duration		
	middle	end	slight	moderate	short	medium	long
Average Bus Journey Time Change (%)	2.07	0.73	0.07	2.73	0.11	1.26	2.83
Average Bus Speed Change (%)	-1.42	-0.58	-0.11	-1.89	-0.15	-0.94	-1.91
Average Excess Waiting Time Change (minutes)	1.21	0.27	0.01	1.48	0.02	0.63	1.57

Results show, for example, that a bus breakdown occurring at the middle of the route causes a 1.55% increase in average bus journey time compared to the base case scenario. This value refers to the average of the changes for each of the six possible scenarios incorporating the remaining incident parameters; two possible options for the severity and three options for the duration. These six scenarios, for which the effect of incident location characteristic was investigated, are used to produce the mean value of average bus journey time.

According to simulation results, all incident parameters affect the key performance measures of the bus operation. Regarding the location component, results show that incidents occurring in the middle of the route caused higher impact on the performance parameters than incidents occurring at the end of the route. Once an incident takes place passenger numbers waiting at bus stops downstream the incident location increase causing delays to the bus operation. If the event takes place at the end of the route, smaller number of bus stops of the system is affected; therefore, delays caused are slighter than when the incident location is in the middle of the route. Furthermore, slack time included at the end of the route regulates the operation, a further reason explaining the difference in the location parameter effect on the model performance measures and, more widely, on the bus operation level of service. According to Tables 8.1 and 8.2, 'other type' incident scenarios are more severely affected by the incident location than bus breakdown scenarios. The latter underlines the dominant role that bus route characteristics play in bus-based public transport sector and especially the effect that they retain in the case of traffic incidents such as roadworks, traffic accidents and illegal parking.

Investigation of the incident severity parameter showed that incidents of a medium severity (i.e. causing a 40% roadway capacity reduction) produced greater changes in the key performance indicators than incidents of a slight severity (i.e. causing 25% capacity loss). Changes were calculated in reference and comparison to the base case scenario model results. Although this result relating to incident severity was anticipated, the overall performance of 'other type' incident scenarios was clearly more severely affected by the change in incident severity than the performance of bus breakdown scenarios; this provides a significant insight into the impact of different types of incidents on the bus operations. A slight severity bus breakdown incident is expected to cause an increase of 0.25 minutes on the average excess waiting time, while a medium severity bus breakdown is expected to cause a rise of 0.54 minutes, which is an additional 0.29 minutes if the two scenarios are compared. On the other hand, a slight severity 'other type' incident was responsible for an increase of just 0.01 minutes on the excess waiting time, while a medium severity incident of this type caused a rise of 1.48 minutes; the difference between these two cases is 1.47 minutes.

Further comparison and evaluation of simulation results relating to the duration incident parameter were investigated. Tables 8.1 and 8.2 provide the average values of change in performance measures for incidents lasting 20 minutes, 40 minutes and 60 minutes. Results highlighted a similarity of the behaviour of this incident characteristic to the severity component: the longer the duration of the incident, the more severe the impacts on the key model performance measure were found. Nevertheless, similarly to the severity characteristic, the extent of the effect of the incident duration was greater in the case of 'other type' events than in the case of bus breakdown incidents.

Apart from reaching significant conclusions on the effect of specific characteristics of bus related incidents on the overall performance of the bus operation, interesting research findings were also found surrounding the decision making process over the bus fleet management actions operators need to take when required. In order to explore the potential benefits that bus fleet management strategies entail to minimize the delays caused by incidents and support the service's reliability and efficiency, two specific actions were modelled in SIBUFEM. This formed the second stage of the model application, leading to model results related to two strategies: reinforcing the line with an extra bus and increasing the bus speed by 10%. The effect of each of the bus fleet management strategies on excess waiting time was calculated. Initially, several comparisons between the model results of the Base Case, the Incident case and the Bus Fleet Management scenarios were carried out, as the example illustrated in Figure 7.1. Then, model results summarizing the impact of the two fleet management actions on the excess waiting time were provided, as shown in Figure 7.2 and, finally, the behaviour of the modelled strategies according to different levels of incidents' location, severity and duration was explored (Figures 7.3-7.5).

Concerning the second stage of model application, it may be concluded that both of the modelled fleet management strategies caused a direct effect on the bus performance measures. However, the main key finding of the model results' evaluation is that reinforcing the line with an additional bus outperformed overall the control action of increasing the speed of the buses. However, a complete evaluation of the benefits offered by each of these two options should take into account the economic requirements and appraisal of each strategy, which is crucial in the decision making procedure for any bus-based transport organisation.

These research findings are significant in establishing the role of bus and traffic incidents modelling and evaluating the merits of each bus fleet management strategy, which in turn enables the support of dynamic bus fleet management as an innovative application of ITS within the bus-based public transport sector.

8.3 Research Contribution

Buses are the most dominant of public transport, representing 64% of total passenger journeys on public transport in England (DfT, 2009). Provisional figures, however, suggest that bus passenger journeys in England decreased by 1.8% between 2008-09 and 2009-10 (DfT, 2010). Innovative applications are being integrated increasingly into bus operations throughout the world, in order to oppose to the recent trend and support the use of bus-based public transport. The need for a continuous support with research on key bus operations issues is required in order to improve the level of bus service and, thus, the degree of bus passengers' satisfaction. Average rating for overall satisfaction for bus services in England was 82 points out of 100 in 2009, unchanged from 2008 (DfT, 2009), indicating that passengers' expectations of bus operations are stagnating and may not be fully met. The role of ITS applications towards increasing the use of public transport and passengers' satisfaction is crucial as verified by bus operators:

> *'With the real time information, when we've got the times counting down at bus stops, the customers, the people who matter, they go to the bus stop and they don't look at the timetable book they look at the real time information. So if the book says that the bus should be there now but in fact it is 3 minutes late, passengers look at the real time information which says the bus will be at the bus stop in 3 minutes but they won't know that this 3 minutes is 3 minutes later than the book, they will be happy that the bus will be there in 3 minutes, and that becomes their timetable. So when the bus turns up, the bus is actually to them on time because it comes exactly when the display says.'*

One of the key components for any bus operation is the bus and traffic incidents management and how this contributes towards supporting DBFM. This thesis has provided a review of the current state of art of ITS and DBFM as one of its most significant applications. In addition, a review of existing computer software used for modelling bus and traffic incidents for bus fleet management purposes underlined the need for a new simulation model capable of providing the impact of various incidents on bus operations. This thesis presented a new model, SIBUFEM, which investigated the effect of each of the incidents characteristics on key performance measures of a high frequency bus service. In addition, the model incorporates the continuous circulation of buses around the route. These characteristics entail unique attributes for this model and significantly contribute to the research field of supporting bus operations on a path parallel to the model SIMBOL (Shrestha, 2003). Findings related to performance measures such as bus journey time, bus speed and passenger waiting time is of significance for transport planners, transit managers, decision makers and the general public, demonstrating the significance of the research findings.

Valuable feedback on findings from this research was been obtained from bus operators, which supported the appraisal and evaluation of various bus incident management strategies. The strategy of adding an extra bus into the route, which has been modelled in SIBUFEM, is a popular action which is widely implemented in real conditions.

> *'What happens is over time congestion increases, and we've got a history of track record of having to increase the number of buses on a route just to maintain the same frequency or widening the headways in order to accommodate the extra running time and we tend to do both as necessary. If a route is valuable to us, and one of our best routes, then its important that the buses run to time so may require an extra bus, which can serve two purposes: if it works well you can put an extra bus in a route like No 7 at the peak time and you can provide an extra journey as well as slackening it out a bit.'*

SIBUFEM is capable of modelling the bus and traffic incidents and their impact on high frequency bus operations. The bus performance measures are affected by the occurrence of the incident, which requires decision making from the bus operators. Results produced by SIBUFEM may be significant in influencing these decisions

> *'If we can provide a bus that has lots of people on it, in simple terms, then its worth doing it; and if we have the right evidence to justify this action.'*

According to statistics provided by Transport for London, the average excess waiting time for high frequency London bus services is 0.96 minutes (TfL, 2010). The simulation model developed in this thesis showed that the time that passengers have to wait for a bus above the average scheduled waiting time varies between 0.28 and 1.83 minutes depending on the incident occurring. The high flexibility and adaptability of the model offers great potential for use in a wide range of applications and scenarios to assess the impact of incidents on bus performance and hence suggest potential DBFM strategies for improved bus service efficiency.

SIBUFEM may be used as a tool for implementing bus fleet management control options and to evaluate the benefits of each strategy. Reliable and accurate research on incident modelling is essential for bus operators to respond promptly and accurately to traffic incidents; the usefulness of this research and the convenience that a highly flexible and adaptable tool like SIBUFEM could offer to bus operators during the decision making for DBFM have been highlighted in this thesis. Bus operators confirm the great potential of DBMF and the huge benefits that accurate and prompt fleet management actions entail in terms of reliability.

> *'So with the extra 2 buses that we put on we found that they were worth our investment and, although that didn't increase the journey times, it did offer the opportunity to increase recovery times, because as well as increasing the frequency at the busiest times, because there is an extra bus there I could alter some of the layover time as we call it in the end which increased the reliability. It would also, had our need to increase the running time, provided an opportunity, so there is the opportunity to do both, because 2 buses is more than enough to increase the frequency, and also when they get to the end you can slacken them out slightly after the evening peak.'*

The model could be used during the development of a Decision Support System (DSS) which would be capable of modelling traffic incidents and suggest the optimum strategy to address the impacts and re-establish the overall quality of bus service, without disregarding the fact that the final decision and responsibility for the decisions remain to the operators.

> *'It tends to be just one person talking to another person in bus services. So a little bit more structure to it to shows the impacts of what is out there and the impacts of your decisions would be possible is something that bus operator groups might be interested to use as a training guide but also to speak to local authorities, because if you can model effectively what happens when a road's died upon the bus operations, then it could be useful to make the decision of whether or not to dig the road up before then, because the model could be used to show what the problems would*

> *be other than digging the road up and then you've got the problems. So in that sense it could be a very useful tool for local authorities.'*

This research itself contributes towards the direction of increasing the sustainability of road transport and increasing public transport use. The UK has set itself the world's first legally binding target of a 34% reduction of Greenhouse Gas emissions below 1990 levels by 2020 through its Low Carbon Transition Plan (DfT, 2009) and with road transport contributing to a quarter of the UK's emissions a shift in modal use is required. However, public transport use needs to be supported by the overall sustainable town planning, as highlighted by transport operators.

> *'The American Express building, which is one of the biggest employers in this city, was built in 60s deliberately without any car parking spaces to encourage public transport use in the city centre.'*

Emissions from road transport are growing yet provisional figures suggest that bus passenger journeys in England decreased by 1.8% between 2008-09 and 2009-10 (DfT, 2010). This may be due to the lack of focus on addressing issues raised by public view on the position of public transport to satisfy travel needs and desires confirmed by DfT's bus passenger satisfaction survey for the first quarter of 2010, which highlighted that overall satisfaction of bus operations in England was unchanged at 82 points out of 100 and that passengers were the least satisfied with reliability and value for money (DfT, 2010). SIBUFEM and, more widely, this research highlight key areas to address that would significantly contribute to increasing public acceptability of buses whilst government need to target key issues surrounding reliability, accessibility and attractiveness of buses (Stradling *et al.,* 2008). If traffic incidents affecting bus operation are addressed promptly and accurately and control actions are taken in a timely and efficient manner, bus passenger satisfaction should rise, consequently driving an increase in bus use thereby contributing to the sustainability of UK's public transport system.

8.4 Areas for Further Work

8.4.1 Further model validation and development

One of the major contributions of the current research is that besides leading to significant findings in the field of traffic incidents modelling to support bus fleet management it has opened up possible areas for further work in the field of traffic incidents and bus fleet management. This may include both further application of the existing model and additional development of the model itself to model other issues. Currently, the validation of the model is based on the fact that the base case model is based on field data. Even though the resultant traffic queues from the model were found similar to the queues predicted by the deterministic queuing theory used, the next step of the validation process of SIBUFEM would require real incidents data. For the complete model validation a series of surveys related to bus related incidents' data would be required and information on the strategies that operators follow to address their impacts on each case.

However, the feedback received from bus operators regarding the model validation and potential of future use were very positive and offered invaluable information for further development and application of the model in real incidents' conditions.

> *'And if we could then demonstrate through something like this (showing the model results of SIBUFEM) that this is the effect, then we could see what the use is. I suppose really it's a combination of the road capacity and its use, isn't it? You've got this amount of capacity which is used to 90% and you've got X number of vehicles flowing through it, then if you cut it in half its going to exceed its capacity.'*

Furthermore, through direct liaising with operators, a further application of the model was recommended, comparing the impact of roadworks on the bus operations in day and night time.

> *'It could be a very useful tool for local authorities. If you can come up with some kind of illustration and researched modelling to show that if you do roadworks in the night there will be no impact but if you it in the day time this will be the impact, with impacts on the number of passengers and*

causing that much inconvenience and affecting the profits so fares might have to go up by this much etc.'

8.4.2 Investigating the impact of headway changes

Investigating the impact that the time headway of the service has on the overall performance of the bus operation is an example of application where no further development of the model is required. The bus headway used for the purposes of the model development of SIBUFEM is the typical value of 8 minutes. However, bus headways for non-timetabled services vary according to the passenger demands and the unique requirements of each case. A change in bus headway would cause different passengers waiting time, as described in Section 4.3.5, thus it would be interesting to explore the impact of bus headway on average excess waiting time and, more widely, on the bus performance. This is also verified through direct liaising with bus operators, who consider the option of changing the bus headway in certain cases as very beneficial.

'What happens is over time congestion increases, and we've got a history of track records having to increase the number of buses on a route just to maintain the same frequency or widening the headways in order to accommodate the extra running time and we tend to do both as necessary.'

8.4.3 Applying SIBUFEM to different route types

The current research focuses on a specific route, without addressing the issue of overlapping routes. The model can be applied to situations where there is a choice of bus services, which will depend on the field data collection of passenger O-D matrix and characteristics of bus services. Furthermore, an additional direction of reinforcing the model to include more applications would be to investigate other types of bus routes, such as radial, diametrical, ring-road and cross-town. Currently, SIBUFEM is developed on the basis of linear routes and network level of buses is not considered. For example, in SIBUFEM, passengers located inside a bus that breaks down, alight the bus and wait at the bus stop to board the next bus. In real conditions, they would have the option of using other bus services, walking or using other means of transport. For network level analysis to be incorporated, further model development is required including the passenger route choice and aiming to a more representative model of the field conditions.

8.4.4 Linking the incident delays to the bus patronage levels

Transportation activities tend to follow a predictable pattern, in line with the 'law of demand' (Victoria Transport Policy Institute, 2011). When the travel time declines, the amount of mobility measured in terms of trips tends to increase. On the other hand, when incidents affecting bus operations cause delays which have a negative impact on the travel time, then mobility tends to decline and bus patronage levels will fall. For the analysis of the relationship between travel time and service use, transportation elasticities are used defined as the percentage change in consumption of a good caused by a one-percent change in its price or other characteristics (such as traffic speed or road capacity) (Victoria Transport Policy Institute, 2011). In transportation, a transit service elasticity is defined as the percentage change in transit ridership resulting from each 1% change in transit service, such as bus-miles or frequency.

Bus related incidents' delays and their impacts on short-term and long-term bus patronage levels are related to travel time elasticity, which is the focus of attention by transportation economists. According to a study by leading UK transportation economists, the elasticity of travel volume with respect to travel time is -0.5 in the short term and -1.0 over the long term (SACTRA, 1994). This means that increasing the travel time by 10% typically decreases the traffic volumes by 5% in the short term and 10% in the long term. Another study found the elasticity values for vehicle travel with respect to travel time in urban roads equates to -0.27 in the short term and -0.57 in the long term (Goodwin, 1996). Among the several factors affecting transit ridership, the service in lost mileage is a parameter which can be calculated in SIBUFEM. According to studies, a 1% decrease of the service is likely to decrease transit ridership by 0.71% (Kain and Liu, 1999).

The delays caused by bus related incidents and the impacts on short-term and long-term bus patronage levels may be linked to travel time elasticity. SIBUFEM provides simulation results related to the delays on the bus journey times. According to Tables 6.2 and 6.4, in some of the incident scenarios the travel time is increased by up to 8.5% due to the incident's occurrence. The travel time results suggested in SIBUFEM may be used for further evaluation of the impact of the incidents on bus patronage levels. However, in order to address this issue adequately, further quantitative data regarding traffic and bus incidents are required.

An idea for future enhancement of the model development would be to include a performance indicator to reflect the loss of bus patronage related to the specific incident induced delay. Therefore, valuable conclusions can be reached regarding the effect of the incidents on the bus patronage levels, if the elasticity of transit use with respect to the service or travel time is inserted into the model.

8.4.5 Investigating the case of low frequency services

Investigating traffic incidents to support bus fleet management for high frequency bus services is one of the central aims of this research. High frequency operations tend to attract more interest by bus operators. However, passengers using low frequency services are considered to be more inconvenienced by the occurrence of bus related incidents.

> *'People are more inconvenienced by the 20 minute buses since they'll have an extra 20 minutes to wait. But it's still in the back of our mind that you get more money from the buses that are on the high frequency route but you have to weigh that up against the overall good of the network.'*

The model development was carried out using a system with random passenger generation which is currently the strategy for non-timetabled services. An innovative application of the model, involving partial alteration of the model, would be looking into low frequency services and how SIBUFEM could be used to model traffic incidents in timetabled services. Passenger arrival rate is closer to the bus arrival time in this case, which can be modelled by defining a passenger generation distribution expressing it.

8.4.6 Using SIBUFEM for weather related incidents

For the purposes of this research, traffic incidents are categorized and modelled according to five main characteristics: (i) type, (ii) starting time, (iii) location, (iii) severity and (iv) duration. Bus breakdown scenarios and 'other type' incident cases are the two categories investigated in this thesis. While the second group of scenarios includes events such as roadworks, illegal parking and traffic accidents, a key area to investigate in further research would be the impacts of each event individually including more traffic related incidents. Investigating the behaviour of traffic incidents in congestion cases would be a challenging idea for further work. Nevertheless, the impact of adverse weather conditions on bus performance measures, for example, would be another direction towards incorporating traffic related incidents modelling to support bus fleet management. Sever weather conditions often impact on the road network by significantly increasing congestion issues, this highlights new challenges in the development of SIBUFEM as such meteorological conditions are difficult to predict, which makes operators task of decision making even harder thereby calling for the use of SIBUFEM.

8.4.7 Modelling and evaluating alternative DBFM strategies

As well as categorizing traffic incidents according to their main characteristics and modelling the impact of their components on the bus performance, SIBUFEM could be extended to model potential fleet management strategies for bus operations to ensure that the overall level of service is met. In the case of an incident, operators have to devise a management plan in order to minimise the impacts of the incident on the overall quality of the bus service; thus several control actions are identified (Belmonte and Fernandez, 2005). DBFM control strategies may involve actions such as increase or decrease of the speed of an individual bus, jump stops, help between services and line head retention as demonstrated in Table 2.3.

SIBUFEM already incorporates one of these actions, the option of service reinforcement. During the model applications process, the strategy of adding an extra bus into the line was implemented in the case of a bus breakdown, assuming that an extra bus was available. Implications may be caused when the fleet size is limited to the operating buses, without the option of reinforcing the service being available. In this case, the strategies of changing the regulation frequency or increasing/decreasing the speed of individual buses or helping the line with bus from another service may be adopted.

Modelling simulation offers the opportunity of following and analyzing the continuous progression of all buses of the system; SIBUFEM updates continuously all the information at intervals of one second. User can identify the buses that run behind the designed headway and, thus, control actions may be applied for specific buses of the service. In the case of bus bunching for example, the model results of SIBUFEM help to identify the individual bunching buses and suggest a fleet management strategy, such as the line head retention where one of the buses is forced to await on a bus stop. The decision on which bus stop to apply this strategy is again dependent on the SIBUFEM results which indicate the bus delays and passenger waiting times on each bus stop.

A further development of the model would involve an additional module responsible for the bus fleet management control strategies, part of which is already investigated through the service reinforcement action modelled for bus breakdown scenarios. Depending on the prioritization of the bus operation parameters, SIBUFEM can provide useful bus performance results and indicate potential DBFM strategies. The main modelling requirements for the modelling and evaluation process of alternative DBFM strategies would be:

i. Validate the model with field data on incident scenarios
ii. Identify the operational agreements on interaction between the specific service and other services
iii. Define the operational speed of the service
iv. Decide on the BDFM locations of the route (control action points)
v. Identify whether there are extra available buses that can be used by the service if required
vi. Quantify the effect of incidents in terms of cost effectiveness
vii. Define the main performance indicators for the benefits from DBFM strategies

8.4.8 Using SIBUFEM as a guidance tool for operators

Once data is collected through bus operators' responses on surveys related to traffic incidents and the strategies followed in each case, they could be used for further development of SIBUFEM and its incorporation into a DSS as a guidance tool for bus operators. It should be underlined though that a DSS is not a substitute for operators, but a useful tool that helps them explore the potential consequences of their control actions. Bus operators are the ultimate and indispensable decision makers in any bus-based public transport issues but, currently, improvements in technology and sufficient guidance regarding the tools they are supplied with are required to keep bus operations at the forefront of improvements of bus services.

> *'Could model results from a future form of SIBUFEM be used as a guidance for bus operators in case of bus related incidents to suggest strategies of bus fleet management? Absolutely, this is going to happen, when you do this, these are the impacts. So yes.'*

8.5 Chapter Summary

The average age of the bus fleet in Great Britain in March 2008 was 8.3 years, slightly lower than the average age of 8.4 years in 2005 (DfT, 2008), it is evident that bus related incidents may affect the normal bus operation level of service and, thus, customer satisfaction. Traffic incidents such as roadworks, traffic accidents, disabled vehicles, burst water mains, sport related events as well as other special events, may have slight or severe impacts on overall bus performance. SIBUFEM offers bus operators a highly flexible, open and adaptable tool capable of modelling bus and traffic incidents efficiently and cost effectively to support DBFM. The overall comparison of the simulation results showed that incident severity and duration have a direct proportional impact on the overall bus performance, while the impact of incident location is greater when the incident takes place in the middle of the bus route than when it occurs at the end of the route. The impacts of incidents on bus operations when these incidents take place in the middle of the route are more complex to address than if they were to occur at the end of the route as passenger numbers and travel time are more likely to be affected in the former case.

Results from this research have opened up new paths for further work, with a wide range of opportunities including further application of SIBUFEM or even further development of the model. A critical review of the public transport statistics for 2009 underlines the need for continuous support of bus operations with the use of ITS (DfT, 2009); the implementation of new simulation models, such as SIBUFEM, supports one of the main ITS applications, DBFM. However, bus operators should be provided with better guidance on how to use ITS and the general public should be offered sufficient public transport information in order to maximise bus users' degree of satisfaction and, thus, reinforce the use of public transport which is a necessity for sustainable transport.

-Chapter 9-

Research Conclusions

9.1 Background

The continuous implementation of highly technological functions such as ITS in public transport has increased the need for a highly efficient, accurate and reliable bus-based public transport system. Addressing traffic incidents in order to maintain high bus performance levels plays a crucial role in any bus operation. Attempts to evaluate the impact of these incidents on bus performance measures have highlighted the strong need for research able to categorize, analyze and calculate the impact of specific incidents on bus operations according to the incidents' characteristics. In order to address this issue and support Dynamic Bus Fleet Management, a key application of ITS, this PhD research has investigated the effect of various bus and traffic incidents according to their main parameters (type, location, severity, duration) and the benefits of two bus fleet management actions by introducing a new microscopic simulation model developed specifically for this purpose.

9.2 Presenting the New Simulation model SIBUFEM

The model developed in this research, called SIBUFEM (Simulating Incidents for BUs FlEet Management), simulates a high frequency bus service using existing field data and incorporates the continuous circulation of buses along the bus route. It uses journey time profiles, passenger-dependent bus stop dwell times and deterministic time-dependent queuing theory to model traffic incidents and the impact of their characteristics on the bus performance parameters. SIBUFEM was developed and applied to a variety of incident scenarios, which were categorised according to their type, location, severity and duration. A base case scenario of 'normal' operational conditions was established for comparison with model results from the various incident scenarios. The comparison and evaluation of the results was based on the key performance parameters of average bus journey time, bus speed and excess waiting time.

9.3 The Main Research Findings

The simulation model developed provides significant findings related to the behaviour of the individual elements of bus and traffic incidents. It offers a fundamental outline on how to model incidents of considerable significance and should be incorporated into bus fleet management strategies. Simulation results showed that each of the incident parameters influences to a slight or severe extent the key performance measures of the bus operation. The main conclusions summarized relating to the overall comparison of SIBUFEM results (Polyviou *et al.,* 2011) were based on the performance parameters of bus journey time, bus speed and excess waiting time:

1. The higher the severity of a traffic incident, the higher the expected impact of the event on the overall bus performance is. For example, a bus breakdown causing a 25% roadway capacity decrease is expected to cause an increase of 0.25 minutes on the average excess waiting time; while a bus breakdown causing a 40% capacity loss is expected to cause a rise of 0.54 minutes on the same performance parameter.

2. The longer the duration of a traffic incident, the more severe the expected effect of the incident on the overall bus performance is found to be. For example, an 'other type' traffic incident, such as a traffic accident lasting 20 minutes is expected to cause an increase of 0.11% on the average bus journey time, while if the same incident lasted 40 or 60 minutes the increase would be 1.26% or 2.83% respectively.

3. In terms of the incident location parameter, the effect is greater when the incident occurs in the middle of the bus route than when it occurs at the end of the route. The effect of incident location is especially evident in the case of ‘other type’ traffic incidents.

4. Findings from this research also highlighted that ‘other type’ traffic incidents, such as roadworks, illegal parking, traffic accidents and disabled vehicles, are usually more severely affected by a change in an incident parameter than bus breakdown incidents; this applies to all three incident parameters of location, severity and duration.

5. Results from the second stage of the model application showed that both fleet management strategies modelled have a substantial direct effect on the bus performance. Overall, the action of reinforcing the line with an extra bus outperformed the strategy of increasing the speed of buses by 10%.

More widely, this research brings forth significant findings related to performance measures, such as passenger waiting time, which are particularly useful for transit managers, transport planners, schedulers and the general public, to support DBFM as an innovative ITS application. The research findings from this PhD may be used by bus operators to assess the impact of various incidents on bus performance and thereby supply crucial data for potential fleet management control actions.

9.4 Using SIBUFEM to Support Bus Operations

In addition, key areas to address that would significantly contribute to increasing bus-based public transport efficiency and reliability are highlighted throughout this PhD Thesis. Possible areas for further development of SIBUFEM involve investigating the effect of bus headway, the issue of overlapping route, various types of routes, the issue of passenger modal choice, the behaviour of incidents in congested areas and the impact of adverse weather conditions on bus operations.

Buses account for 64% of total passenger journeys on public transport in England (DfT, 2009), highlighting their dominant role in the field of public transport. However, according to provisional figures based on quarterly surveys of the largest bus operators, bus passenger journeys have decreased since the first quarter of 2008, with a reduction of 2.4% in non-metropolitan areas and 3.3% in metropolitan areas between 2008-09 and 2009-10 (DfT, 2010). There was, according to the same survey, a slight annual increase of 0.5% in London, where most of the ITS applications are being implemented, demonstrating the emerging potential of ITS applications in increasing bus patronage.

In spite of this, the stagnating bus passengers' satisfaction score of 82 points out of 100 (DfT, 2010) highlights the need for continuous research to support and improve bus operations. Even though steps have been made to improve efficiency and reliability of bus operations, bus passengers are increasingly raising their expectations and requirements for the level of bus service. The convenience and comfort that private car use offers is constantly reinforcing this rise in expectations.

9.5 Addressing the Sustainability of Public Transport

Not only does the public require high standards of overall bus performance but governmental policies throughout the world set targets in order to support public transport in order to reduce greenhouse gas emissions from transport. The UK in particular, has the world's first legally binding target to reduce Greenhouse Gas emissions by 34% below 1990 levels by 2020 through its Low Carbon Transition Plan (DfT, 2009); for this to be achieved, emissions from road transport need to be decreased substantially by supporting a shift in modal use. The environmental impact of road transport, which contributes to approximately a quarter of the UK's emissions (UK SDC, 2010) is a crucial issue which needs to be addressed if a sustainable transport system is to be implemented. SIBUFEM can play a significant role in increasing public transport use thereby reducing transport's share of the UK's Greenhouse Gas emissions.

Although, efforts contribute towards enhancing public transport, the lack of reliability and attractiveness in bus operations is evident according to surveys on the degree of satisfaction. Developing highly flexible and adaptable tools like SIBUFEM, capable of modelling traffic incidents to support Dynamic Bus Fleet Management and, thus encourage the use of ITS applications in bus operations offers great potential in the field of bus-based public transport. If efficient and timely actions to address the impact of traffic incidents on bus operations are taken, the bus level of service should rise leading to an increase in customer satisfaction and thus the development of a sustainable transport system.

9.6 Increasing the Public Transport Service Overall

A number of additional conditions need to be met for the sustainability of public transport system to be achieved. Bus operators need to be provided with sufficient and complete guidance regarding the tools they are supplied with, especially related to ITS applications (Hounsell, 2004). Efforts should be made to bridge the existing gap between research and implementation, which is evident in some cases of public transport. Furthermore, a package of accurate and sufficient information provision options is required with real-time information being the most significant method of information. According to surveys of passenger opinions on current passenger information provided in Dublin, 39% of the respondents believe that the information provided on transport options is insufficient (Caulfield *et al.,* 2005). Unless public transport information becomes sufficient and easy to use for the public, the usage of bus is not likely to reach its full potential. In addition, enhancing the quality of bus corridors and Park-And-Ride schemes seem to represent good value and great potential and should therefore be supported (CfIT, 2002), (Hine and Preston, 2003). Public transport is also required to increase its attractiveness to older people, especially if the 'Help The Aged' report which forecasts that by 2021 one in three people in the UK will be aged over 60 is taken into account (DETR, 2001). Nevertheless, the gap between present and efficient transport prices in public transport should be filled. Estimates of the current inefficiencies in the pricing policies of transport in Europe suggest that public transport prices need to be revised in many cases and important increases in the money prices of peak private transport using spatial and time differentiated road pricing are required (Proost *et al.*, 2002).

From the public's perspective, more attractive bus prices and higher reliability is required for them to 'sacrifice' the comfort, convenience and time saving that travelling in private car offers. This conclusion is in agreement with the recent DfT's survey on individual aspects of bus passenger satisfaction, which confirms that passengers are the least satisfied with reliability and value for money (DfT, 2010). The levels of passengers' satisfaction with reliability and value for money in England for the first quarter (i.e. January to March) of 2010 are both at 73 points out of 100, unchanged in comparison to the immediate previous quarter (DfT, 2010). SIBUFEM and more widely this research provide a significant contribution to address this issue and contribute to increasing public acceptability of buses. More action from all sectors surrounding transport need to be incorporated to maximise bus passenger satisfaction and increase public transport use thereby contributing to the sustainability of the transport and the environment.

References

Adamski A., Turnau A., 1998, Simulation Support Tool for Real-Time Dispatching Control in Public Transport, *Transportation Research A*, vol. 32, No 2, pp. 73-87

Agarwal P. K., Jain S. S., Khanna S. K., 1994, Development of a Simulation Model for Performance Evaluation of Bus Operation on Urban Transportation Corridor, *Proceedings of the 36th Annual Conference of Transportation Research Forum*, Florida, 1994, pp. 759-776

Aisu H., Kentaro T., Toshiaki T., Yoshikazu O., Yoshiro S., 2000, A Dynamical Operation Scheduling for Dual-mode Transport System, *IEEE Conference on ITS*, Proceedings, ITSC, 2000 pp. 440-445

Algers S., Bernauer E., Boero M., Breheret L., Di Taranto C., Dougherty M., Fox K., Gabard J. F., 1997, *SMARTEST Project: Review of Microsimulation Models*, Institute for Transport Studies, University of Leeds, United Kingdom

Bagot N., 1999, Try Before You Buy: Trends in Traffic Simulation, Traffic Technology International, 1999, pp. 68-74

Belmonte M. V., Fernandez A., 2005, Agent Coordination for Bus Fleet Management, *ACM Symposium on Applied Computing*, Santa Fe, New Mexico, USA, 2005, pp.462-466

Bodin L. D., 1990, Twenty Years of Routing and Scheduling, *Operations Research*, Vol. 38, No 4, August 1990, pp. 571-579

Borland Software Corporation, 2002, *Developers Guide Borland C++ Builder 6 for Windows,* California, Scotts Valley

Caulfield B., O' Mahony M., Farrell S., 2005, An Examination of the Public Transport Information Requirements of Users, *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems,* Vienna, Austria, 2005

Clowes D.J., 2000, National ITS including Transportation Information Highways (TIH), National Control Centres and RTA. In: 7th World Congress on Intelligent Transport Systems, Paper number 5001, Turin, 2000

Commission for Integrated Transport (CfIT), 2002, Obtaining Best Value for Public Support for the Bus Industry, LEK Research Report, CfIT, London

Cuena J., Hernandez J., 1997, An Exercise of Knowledge Oriented Design: Architecture for Real Time Decision Support Systems. In *Knowledge Based Systems: Advanced Concepts, Techniques and Applications*, S. G. Tzafestas (ed.), World Scientific, pp. 497-524

Czogalla O., Hoyer R., 1997, Simulation Based Design of Control Strategies for Urban Traffic Management and Control, *Proceedings of the 4th World Congress on Intelligent Transport Systems*, 21-24 October 1997, Berlin (Paper No. 2265)

D'Acierno L., Armando C., Montella B., 2009, Estimation of Urban Traffic Conditions Using an Automatic Vehicle Location (AVL) System, *European Journal of Operational Research, v. 196, pp 719-736*

D'Souza C., Gardner K., Hounsell N.B., Shrestha B.P., 2010, New Developments for Bus Priority at Traffic Signals in London Using iBus, In: *Road Transport Information and Control Conference and the ITS United Kingdom Members' Conference (RTIC 2010),* London

Daganzo C.F., 2009, A Headay-Based Approach to Eliminate Bus Bunching: Systematic Analysis and Comparisons. *Transportation Research Part B,* 43 (10), 2009, pp 913-921

Davies R.M. and O'Keefe R., 1988, *Simulation Modelling with Pascal*, Prentice-Hall, Englewood Cliffs, New Jersey

Department for Transport, 2005, *Intelligent Transport Systems (ITS) – The policy framework for the roads sector,* DfT-UK, Product Code PPU3617/ES

Department for Transport, 2008, *Public Transport Statistics Bulletin GB: 2008 edition.* Available at: <http://www.dft.gov.uk/pgr/statistics/datatablespublications>

Department for Transport, 2009, *Public Transport Statistics Bulletin GB: 2009 edition.* Available at: <http://www.dft.gov.uk/pgr/statistics>

Department for Transport, 2010, *Quarterly Bus Statistics, Great Britain Q1 2010,* Available at: <http://www.dft.gov.uk/pgr/statistics>

Department of the Environment, Transport and the Regions (DETR), 2001, Older People: Their Transport Needs and Requirements, DETR, London. Available at: <http://www.mobility-unit.dft.gov.uk/older/needs/summary/02.html>

Der-Hong L., Shin-Ting J., Chandrasekar P., 2002, Investigating an Incident Situation Using Data Mining, In Proceedings 81st *Annual Meeting of TRB* (CD-ROM), Washington, D.C. 2002

Diogenes, L., 250-300 A.D., *Lives and Opinions of Eminent Philosophers*

Druitt S., 1998, An Introduction to Microsimulation, *Traffic Engineering and Control*, Vol. 39 (9), pp. 480-483

European Commission, 2001, White Paper: European Transport Policy for 2010: Time to Decide, COM (2001), publication No 370, White Papers. Available at: <http://europa.eu/documentation/official-docs/white-papers/index_en.htm>

Fang C.F., 2006, Information Technology Systems for Use in Incident Management and Work Zones, FHWA-CT-RD-222-39-06-01, 2006

Farradyne P.B., 2000, *Traffic Incident Management Handbook,* Washington, DC, Federal Highway Administration Office of Travel Management

Fernandez A., Alonso E., Ossowski S., 2004, A Multiagent Architecture for Bus Fleet Management, *Integrated Computer-Aided Engineering*, Vol. 11, No. 2, pp. 101-115

Firmin, P. E., 2006, Satellite Navigation Technology Applications for Intelligent Transport Systems: A European Perspective. *European Navigation Conference*, Manchester, May 2006.

Gardner K., Melhuish T., McKenna D., Rice T., 2006, The Benefits of Bus Priority Within the Central London Congestion Charging Zone In: Association for European Transport and contributors, 2006 *European Transport Conference Proceedings*

Garmin Inc, 2000, *GPS Guide for Beginners*, Kansas, U.S., Garmin International Inc

Gendreau M., Hertz A., Laporte G., 1994, A Tabu Search Heuristic for the Vehicle Routing Problem, *Management Science*, Vol. 40, No 10, October 1994 pp. 1276-1290

Goodwin P., 1996, *Empirical Evidence on Induced Traffic,* Transportation, Vol. 23, No 1, pp. 35-54

Greenfeld J., 2000, Automatic Vehicle Location (AVL) for Transit Operation, *10th Mediterranean Electrotechnical Conference*, MEleCon 2000, Vol. 2, pp. 656-659

Haase K., Desaulniers G., Desroisers J., 2001, Simultaneous Vehicle and Crew Scheduling in Urban Mass Transit Systems, *Transportation Science*, Vol. 35, No 3, August 2001, pp. 286-303

HEN 2, 1997, Highway Economics Note No. 2, Section 2, Values of Time and Vehicle Operating Costs, Chapter 2, Design Manual for Roads and Bridges, Volume 13, 1997, Highways Agency, HMSO

Hine J., Preston J., 2003, Integrated Futures and Transport Choices: UK Transport Policy Beyond the 1998 White Paper and Transport Acts, Aldershot, Ashgate, pp. 235-236.

Hounsell N. B., 2004, Keeping Buses Moving: Role of Intelligent Transport Systems, *Proceedings of the Institution of Civil Engineers*, Municipal Engineer, Vol. 157, No. 1, March 2004, pp. 55-60

Hounsell N. B., Graham W., 2001, Examples of New ITS Applications in Europe to Improve Bus Services, *Transportation Research Record*, No. 1971, 02-3451, 2002, pp. 85-91

Hounsell N. B., McLeod F. N., Landles, J. R., Gardner K., 1996, Bus Priority in London: Building on Prompt, *Proceedings of the Third World Congress on Intelligent Transport Systems*. Orlando, USA, October 1996

Hounsell N. B., McLeod F., 1999, Automatic Vehicle and Bus Priority: The London System, *World Transport Research, Selected Proceedings of the 8th World Conference on Transport Research*, Vol. 2, Planning Operation Management and Control, pp. 279-292

Hounsell N.B., Shrestha B.P., 2005, AVL Based Bus Priority at Traffic Signals: A Review of Architectures and Case Study, *European Journal of Transport and Infrastructure Research, (5)1, pp 13-29*

Huisman D. and Wagelmans A.P.M., 2006, A Solution Approach for Dynamic Vehicle and Crew Scheduling, European Journal of Operational Research, Elsevier, vol. 172(2), pp. 453-471.

Huisman D., Freling R., Wagelmans A. P. M., 2004, A Robust Solution Approach to the Dynamic Vehicle Scheduling Problem, *Transportation Science*, Vol. 38, No 4, November 2004, pp 447-458

Hwang M., Kemp J., Lerner-Lam E., Neuerburg N., Okunieff P., *Advanced Public Transportation Systems: The State of the Art, Update 2006*, Washington DC, U.S. Department of Transportation, Federal Transit Administration

ITS UK, 2010, Intelligent Transport Systems Review Spring/Summer 2010, Edgware, Middlesex, Regent Publications Ltd

Kain J. F. and Liu Z., 1999, *Secrets of Success,* Transportation Research A, Vol. 33, No. 7/8, Sept/Nov 1999, pp. 601-624

Kirannoudis C., 2004, *An introduction to Metaheuristics*, Operational Research Lecture Notes, MsC in Computational Fluid Mechanics, National Technical University of Athens, Greece

Koehne J., Mannering F. L, Hallenbeck M. E., Nee J., 1995, *Framework for Developing Incident Management Systems-Revised*, Washington State Transportation Center (TRAC), 1995 Report for U.S. Department of Transportation

Kontaratos M., Balis V., Liarakis C., 1996, GPS-Based AVL System: A Tool for Supporting Public Transport Operation in the Urban Area, *Proceedings of the Third World Congress on Intelligent Transport Systems*. Location: Orlando, Florida, 1996. pp. 102

Lam W. H. K., 2001, Development of Intelligent Transport Systems in Hong Kong, *IEEE Transport Systems Conference Proceedings*, 2001, Oakland (CA) USA, pp. 1000-1005

Laporte G., Gendreau M., Potvin J.-Y., Semet F., 2000, Classical and modern heuristics for the vehicle routing problem, *International Transaction in Operational Research*, v. 7, 2000, pp. 285-300

LAWLER E. L., 2001, *Combinatorial Optimization: Networks and Matroids,* New York, Dover Publications

Lee Y-J., Chon K. S., Hill D. L., Desai N., 2001, Impact of AVL System on Schedule Adherence for MTA Bus System, *Transport Research Board*, 2001, Washington, D.C., No. 01-1984

Lin W.H., and Bertini R.L., 2004, Modelling Schedule Recovery Processes in Transit Operations for Bus Arrival Time Prediction, *Journal of Advanced Transportation*, Vol. 38, No. 3, pp 347-365

Liu R., Clark S., Montgomery F., Watling D., 1999, Microscopic Modelling of Traffic Management Measures for Guided Bus Operation, *Proceedings of 8th World Congress on Transport Research,* vol. 2, Planning, Operation, Management and Control, Pergamon, 1999, pp. 367-380

Lobo A. X., 1998, A Review of Automatic Vehicle Location Technology and Its Real Time Applications, *Transport Reviews*, 1998, Vol. 18, No. 2, pp. 165-191

Luke S., 2006, Public Transport Mode Selection: A Review of International Practice, *Proceedings of the European Transport Conference,* Strasbourg, 2006

Marques A., Sebastian V., Darby-Dowman K., Brand T., 1999, *ATT in Urban Sites with Integration and Standardisation, AUSIAS TR1006, Final Report*, ETRA S.A., Telematics Applications Programme (Transport), Version number V1.0

MathWorks, 2010, *Simulink 7: Simulation and Model-Based Design.* Available at: <http://www.Mathworks.com/products/simulink/>

McDonald et al, 2006, *Intelligent Transport Systems in Europe: opportunities for future research'*, Singapore, World Scientific Publishing

McLeod F. N., 1998, Headway-based selective priority to buses, *Proceedings of the Third Institute of Mathematics and its Applications International Conference on Mathematics in Transport Planning and Control* (ED JD Griffiths), April 1-3, 1998, Cardiff UK: 69-78 ISBN 008 043430 4, Pergamon, pp. 69-78

Milkovits M.N., 2008, *Simulating Service Reliability of a High Frequency Bus Route Using Automatically Collected Data*, MsC Dissertation, MIT University, 2008

Nagatani T., Yoshimura J., 2002, Dynamical Transition in a Coupled-Map Lattice Model of a Recurrent Bus, *Physica A: Statistical Mechanics and its Applications*, Vol. 316, No. 1-4,Dec 15, 2002, pp. 625-636

National Incident Management Coalition, 1997, *Incident Management: Challenges, Strategies and Solutions for Advancing Safety and Roadway Efficiency* – Final Technical Report, Chicago, ATA Foundation in association with Cambridge Systematics

OR/MS today, 2004, *Vehicle Routing and Scheduling Survey,* volume 31, no.3 issue, Available at: <http://www.lionhrtpub.com/orms/ORMS-archive.html >

OR/MS today, 2010, *Vehicle Routing and Scheduling Survey,* volume 37, no.1 issue Available at: <http://www.lionhrtpub.com/orms/ORMS-archive.html >

Oualline S., 2002, *Practical C++ Programming*, 2nd edition, O'Reilly Media

Parsons D., 1997, *Object Oriented Programming with C++*, 2nd Edition, Aldine Place, London, Letts Educational

Peripimeno A., Anquita D., Chiappini P., 2004, Smart Data Re-Sampling for Bus Fleet Management, *IEEE Intelligent Vehicles Symposium*, University of Parma, Parma, Italy, 2004, pp. 178-181

Pilachowski J.M., 2009, *An Approach to Reducing Bus Bunching*, PhD Thesis, University of California, 2009

Polk A.E., 2000, What ITS has to offer Municipal Governments. In: 7th World Congress on Intelligent Transport Systems, Paper number 1051, Turin, 2000

Polyviou P. and Hounsell N.B., 2011, *How can Modelling of Traffic Incidents support Intelligent Transport Systems and Reinforce Bus-Based Public Transport?,* 8th Intelligent Transport Systems (ITS) European Congress, Lyon, France. 6th-10th January 2011.

Polyviou P., Hounsell N.B., Shrestha B.P., 2011, *Modelling Bus and Traffic Incidents for Bus Fleet Management Purposes using Intelligent Transport Systems,* 43rd Annual Universities' Transport Study Group (UTSG) Conference, Milton Keynes, London, UK. 5rd-7th January 2011.

Polyviou P., 2008, *Evaluation of New Bus Fleet Management Options using Intelligent Transport Systems*, Universities' Transport Study Group (UTSG), Marriot Hotel, Portsmouth, UK. 3rd-5th January 2008.

Polyviou P., Hounsell N.B., Shrestha B.P., 2010, *Intelligent Transport Systems and Simulation Modelling of Dynamic Bus Fleet Management*, The Young European Arena of Research 2010, Brussels, Belgium. 7th-11th June 2010.

Polyviou P., 2006, *Discussion on Fleetroute,* Personal Communication with Civix Software (conversation via email), May 2006

Polyviou P., 2007-2011, *Discussion on Traffic Incidents Affecting Bus Operations*, Personal Communication with Bus Operators of 'London United' in London, 'First Bus' in Southampton and 'Brighton and Hove Buses' in Hove, UK (in person conversation), 2007-2011

Powell W. B., Jaillet P., Odoni A., 1995, Stochastic and dynamic networks and routing. Ball M. O., Magnanti T. L., Monma C. L., Nemhauser G. L., eds. *Handbooks in Operations Research and Management Science*, vol. 8. Network Routing, North-Holland, Amsterdam, The Netherlands, pp. 141-295

Pradeep S. K., Gopalkrishna B., Prakash D.C., 2000, Fleet Management Using GPS and GIS, *The Indian Journal of Transport Management*, 2000

PRISCILLA, 2001, *Bus Priority Strategies and Impact Scenarios Development on a Large Urban Area*, Deliverable 2, Public Transport Priority: State of the Art Review, March 20021, pp. 66-67, 108

Proost S., Van Dender K., Courcelle C., De Borger B., Peirson J., Sharp D., Vickerman R., Gibbons E., O'Mahony M., Heaney Q., Van de Bergh J., Verhoef E., 2002, How Large is the Gap Between Present and Efficient Transport Prices in Europe?, *Transport Policy,* vol. 9, pp. 41-57

Rajbhandari B., 2002, *Modelling Intelligent Transport System Applications for Public Transport,* PhD Thesis, Department of Civil Engineering, University of Southampton, pp. 113-114

Rakha H., Hellinga B., Van Aerde M., and Perez W., 1996, Systematic Verification, Validation and Calibration of Traffic Simulation Models, 1996 *Transportation Research Board Annual Meeting*

Reeves C.R., 1995, *Modern Heuristic Techniques for Combinatorial Problems*, London, McGraw-Hill

Richardson N., 2010, Southampton's Vision: Can we double bus use?, The Journal of The Chartered Institute of Logistics and Transport (UK), vol. 12, no. 9, September 2010.

ROSETTA, 2004, *Real Opportunities for Exploitation of Transport Telematics Applications, Project Document,* Final Report, 2004a

Rossetti M.D, Hill R.R., Johansson B., Dunkin A., Ingalls R. G., 2009, Verification and Validation of Simulation Models, *Proceedings of the 2009 Winter Simulation Conference*, 2009, pp. 162-176

SACTRA, 1994, *Trunk Roads and the Generation of Traffic,* UKDoT, HMSO, London, UK, 1994

Sadeh N. M., Kott A., 1996, *Models and Techniques for Dynamic Demand-Responsive Transportation Planning*, Technical Report, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 1996

Seddon P. A., Day M.P., 1974, Bus Passenger Waiting Times in Greater Manchester, *Traffic Engineering and Control,* January 1974, pp. 442-445

Shrestha B. P., 2003, *Simulating Advanced Bus Priority Strategies at Traffic Signals,* Ph.D. Thesis, Department of Civil Engineering and the Environment, University of Southampton, 2003

Stradling S., Anable J., Anderson T., Cronberg A., 2008, Car use and Climate Change: Do we practise what we preach?. In Park A., Curtice J., Thomson K., Philips M., Johnson M., Clery E. (eds.) British Social Attitudes: the 24th Report, London: Sage, Chapter 7.

TCRP, 2003a, *Bus Rapid Transit, Vol. 2: Implementation Guidelines,* TCRP Report 90, Washington DC, Transportation Research Board

TCRP, 2003b, Transit Capacity and Quality of Service, TCRP Report 100, 2nd edition, Washington DC, Transportation Research Board

TCRP, 2008, *AVL systems for Bus Transit: Update, A synthesis of Transit Practice,* Synthesis 73, Washington DC, Transportation Research Board

Transport for London, 2007, London Buses Quality of Service Indicators, Route results for London Buses Services, First Quarter 2007/08, TfL London Buses, pp. 3

Transport for London, 2010, *QSI results for all bus routes*. Available at: <http://www.tfl.gov.uk/tfl/businessand partners/buses/boroughreports/>

U.S. Department of Transportation, 2010, *Traffic Incident Management Handbook*, FHWA-HOP-10-013, Washington, DC, Federal Highway Administration Office of Travel Management

UK Sustainable Development Commission (SDC), 2010, Smarter Moves: How Information and Communication Technologies can promote Sustainable Mobility, SDC Report, Jan 2010.

Union Internationale des Transports Publics (UITP), 2003, Vehicle for Mobility Advancing Public Passenger Transport in Europe, State-of-the-art Report, GTC2-2000-33029, Nov. 2003

UTCA, 2003, *Deploying Advanced Public Transportation Systems in Birmingham*, UTCA Report FHWA/CA/OR, 2003, University of Alabama at Birmingham,

Victoria Transport Policy Institute, 2011, *Transportation Elasticities-How Prices and Other Factors Affect Travel Behavior,* Report, VTPI, 2011

Willoughby P., Emmerson P., 1999, Network Interaction – A Review of Existing Modelling Techniques, *Traffic Engineering and Control*, Vol. 39(2), pp. 81-82

Wren A., Kwan R. S. K., 1999, Installing an Urban Transport Scheduling System, *Journal of Scheduling*, vol. 2, pp. 3-17

York I.O., 1993, *Factors Affecting Bus-Stop Times,* TRL Project Report 2, T1/25, Transport Research Laboratory, Crowthorne, UK

Zhao F, Shen L. D., 1997, *Planning and Implementation of Automatic Vehicle Location Systems for Public Transit*, Technical Report, NUTI-FIU-3-2, prepared for the U.S. Department of Transportation, Washington, DC. June 1997

Zhao J., Dessouky M., Bukkapatnam S., 2006, Optimal Slack Time for Schedule-Based Transit Operations, *Transportation Science,* 40(4), pp. 529-539

Zhicai J., Jianping W., McDonald M., 2006, Socio-Economic Impact Assessment of Intelligent Transport Systems, Tsinghua Science and Technology, vol. 11, no. 3, June 2006, pp. 339-350

Zhu T., Wang J., Lv W., 2009, Outlier Mining Based Automatic Incident Detection on Urban Arterial Road, *Proceedings of the 6th International Conference on Mobile* Technology, Application and Systems, ACM, New York, 2009

APPENDIX A

Incident-Related Technical Report of London United Buses

CentreComm Real Time Bus Delay Information - Garage Report Form

This form should be faxed to the CentreComm Information Collator on 020-7730-1278 or the details telephoned on 020-7730-6188

As soon as practically possible after the details of the delay have been established.

Garage: | Controller:

	Date	Start Time	Cause	End Time	Road	Junction	Postal Area	Direction	Max Delay
1									
2									
3									
4									
5									
6									
7									
8									

Bus Delay Cause Codes

1. - Roadworks 2. - Diversion 3. - Burst Water Main 4. - Traffic Signals 5. - Road Traffic Collision 6. - Police Incident 7. - Security Alert.
8. - Parking 9 - March/Demo 10 - Sport/Special Event 11 - Other (must be specified)

CentreComm Collators 020 7730 6188, fax 020-7730-1278. or email clc@tfl-buses.co.uk. CentreComm Management Desk 020 7730 5287 or email CentreCommCC@tfl-buses.co.uk - Bus delay template Ver.2

LONDON UNITED

LONDON SOVEREIGN

TRANSDEV Group

THIS SIDE FOR INTERNAL COMPANY USE ONLY – NOT REQUIRED BY CENTRECOMM

To assist with accurate recording within our company, please use this side of the sheet to record the route number(s) that have been affected by the delays shown on the other side of this sheet.

1)____________________

2)____________________

3)____________________

4)____________________

5)____________________

6)____________________

7)____________________

8)____________________

Database Ref:____________________

Entered By:____________________

Date:____________________

APPENDIX B

Calculation of Link Speed Factors used in SIBUFEM

duration (s)	simulation time (s)	capacity (veh)	demand (veh)	delay (veh)	CumuDelay (veh)	Total queue length (m)	1st link queue (m)	2nd link (m)	3rd link (m)	4th link (m)
		1000						360,000	710,000	890,000
3600		600	800		0					
	1	0,167	0,222	0,056	0,056	0,278	0,278	0,000	0,000	0,000
	2	0,167	0,222	0,056	0,111	0,556	0,556	0,000	0,000	0,000
	3	0,167	0,222	0,056	0,167	0,833	0,833	0,000	0,000	0,000
	4	0,167	0,222	0,056	0,222	1,111	1,111	0,000	0,000	0,000
	5	0,167	0,222	0,056	0,278	1,389	1,389	0,000	0,000	0,000
	6	0,167	0,222	0,056	0,333	1,667	1,667	0,000	0,000	0,000
	7	0,167	0,222	0,056	0,389	1,944	1,944	0,000	0,000	0,000
	8	0,167	0,222	0,056	0,444	2,222	2,222	0,000	0,000	0,000
	9	0,167	0,222	0,000	0,444	2,222	2,222	0,000	0,000	0,000
	10	0,167	0,222	0,056	0,500	2,500	2,500	0,000	0,000	0,000
	11	0,167	0,222	0,056	0,556	2,778	2,778	0,000	0,000	0,000
	12	0,167	0,222	0,056	0,611	3,056	3,056	0,000	0,000	0,000
	13	0,167	0,222	0,056	0,667	3,333	3,333	0,000	0,000	0,000
	14	0,167	0,222	0,056	0,722	3,611	3,611	0,000	0,000	0,000
	15	0,167	0,222	0,056	0,778	3,889	3,889	0,000	0,000	0,000
	16	0,167	0,222	0,056	0,833	4,167	4,167	0,000	0,000	0,000
	17	0,167	0,222	0,056	0,889	4,444	4,444	0,000	0,000	0,000
	18	0,167	0,222	0,056	0,944	4,722	4,722	0,000	0,000	0,000
	19	0,167	0,222	0,056	1,000	5,000	5,000	0,000	0,000	0,000
	20	0,167	0,222	0,056	1,056	5,278	5,278	0,000	0,000	0,000
	21	0,167	0,222	0,056	1,111	5,556	5,556	0,000	0,000	0,000
	22	0,167	0,222	0,056	1,167	5,833	5,833	0,000	0,000	0,000
	23	0,167	0,222	0,056	1,222	6,111	6,111	0,000	0,000	0,000
	24	0,167	0,222	0,056	1,278	6,389	6,389	0,000	0,000	0,000
	25	0,167	0,222	0,056	1,333	6,667	6,667	0,000	0,000	0,000
	26	0,167	0,222	0,056	1,389	6,944	6,944	0,000	0,000	0,000
	27	0,167	0,222	0,056	1,444	7,222	7,222	0,000	0,000	0,000
	28	0,167	0,222	0,056	1,500	7,500	7,500	0,000	0,000	0,000
	29	0,167	0,222	0,056	1,556	7,778	7,778	0,000	0,000	0,000
	30	0,167	0,222	0,056	1,611	8,056	8,056	0,000	0,000	0,000

2nd link(time)	3rd link(time)	4th link(time)	2nd link(Capacity)	3rd link(Capacity)	4th link(Capacity)	2nd link(Delay)	3rd link(delay)	4th link(Delay)
			1000	1000	1000			
			600	600	600			
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000
0	0	0	0,000	0,000	0,000	0,000	0,000	0,000

2nd link(CumuDel)	3rd link(Cumudel)	4th link(CumuDel)	1st link queue length	2nd link queue length (m)	3rd link queue length(m)	4th link queue length(m)
				350,000	180,000	280,000
0,000	0,000	0,000	0,278	0,000	0,000	0,000
0,000	0,000	0,000	0,556	0,000	0,000	0,000
0,000	0,000	0,000	0,833	0,000	0,000	0,000
0,000	0,000	0,000	1,111	0,000	0,000	0,000
0,000	0,000	0,000	1,389	0,000	0,000	0,000
0,000	0,000	0,000	1,667	0,000	0,000	0,000
0,000	0,000	0,000	1,944	0,000	0,000	0,000
0,000	0,000	0,000	2,222	0,000	0,000	0,000
0,000	0,000	0,000	2,222	0,000	0,000	0,000
0,000	0,000	0,000	2,500	0,000	0,000	0,000
0,000	0,000	0,000	2,778	0,000	0,000	0,000
0,000	0,000	0,000	3,056	0,000	0,000	0,000
0,000	0,000	0,000	3,333	0,000	0,000	0,000
0,000	0,000	0,000	3,611	0,000	0,000	0,000
0,000	0,000	0,000	3,889	0,000	0,000	0,000
0,000	0,000	0,000	4,167	0,000	0,000	0,000
0,000	0,000	0,000	4,444	0,000	0,000	0,000
0,000	0,000	0,000	4,722	0,000	0,000	0,000
0,000	0,000	0,000	5,000	0,000	0,000	0,000
0,000	0,000	0,000	5,278	0,000	0,000	0,000
0,000	0,000	0,000	5,556	0,000	0,000	0,000
0,000	0,000	0,000	5,833	0,000	0,000	0,000
0,000	0,000	0,000	6,111	0,000	0,000	0,000
0,000	0,000	0,000	6,389	0,000	0,000	0,000
0,000	0,000	0,000	6,667	0,000	0,000	0,000
0,000	0,000	0,000	6,944	0,000	0,000	0,000
0,000	0,000	0,000	7,222	0,000	0,000	0,000
0,000	0,000	0,000	7,500	0,000	0,000	0,000
0,000	0,000	0,000	7,778	0,000	0,000	0,000
0,000	0,000	0,000	8,056	0,000	0,000	0,000

1st link (additional)	2nd link(addit.)	3rd link(addit.)	4th link(addit.)	1st link (discharge time)	2nd link(disch. time)	3rd link(disch. time)	4th link(disch. time)
0,333	0,000	0,000	0,000	1,333	0,000	0,000	0,000
0,667	0,000	0,000	0,000	2,667	0,000	0,000	0,000
1,000	0,000	0,000	0,000	4,000	0,000	0,000	0,000
1,333	0,000	0,000	0,000	5,333	0,000	0,000	0,000
1,667	0,000	0,000	0,000	6,667	0,000	0,000	0,000
2,000	0,000	0,000	0,000	8,000	0,000	0,000	0,000
2,333	0,000	0,000	0,000	9,333	0,000	0,000	0,000
2,667	0,000	0,000	0,000	10,667	0,000	0,000	0,000
2,667	0,000	0,000	0,000	11,667	0,000	0,000	0,000
3,000	0,000	0,000	0,000	13,000	0,000	0,000	0,000
3,333	0,000	0,000	0,000	14,333	0,000	0,000	0,000
3,667	0,000	0,000	0,000	15,667	0,000	0,000	0,000
4,000	0,000	0,000	0,000	17,000	0,000	0,000	0,000
4,333	0,000	0,000	0,000	18,333	0,000	0,000	0,000
4,667	0,000	0,000	0,000	19,667	0,000	0,000	0,000
5,000	0,000	0,000	0,000	21,000	0,000	0,000	0,000
5,333	0,000	0,000	0,000	22,333	0,000	0,000	0,000
5,667	0,000	0,000	0,000	23,667	0,000	0,000	0,000
6,000	0,000	0,000	0,000	25,000	0,000	0,000	0,000
6,333	0,000	0,000	0,000	26,333	0,000	0,000	0,000
6,667	0,000	0,000	0,000	27,667	0,000	0,000	0,000
7,000	0,000	0,000	0,000	29,000	0,000	0,000	0,000
7,333	0,000	0,000	0,000	30,333	0,000	0,000	0,000
7,667	0,000	0,000	0,000	31,667	0,000	0,000	0,000
8,000	0,000	0,000	0,000	33,000	0,000	0,000	0,000
8,333	0,000	0,000	0,000	34,333	0,000	0,000	0,000
8,667	0,000	0,000	0,000	35,667	0,000	0,000	0,000
9,000	0,000	0,000	0,000	37,000	0,000	0,000	0,000
9,333	0,000	0,000	0,000	38,333	0,000	0,000	0,000
9,667	0,000	0,000	0,000	39,667	0,000	0,000	0,000

	1st link	2nd link	3rd link	4th link	1st link (Delay)	2nd (Delay)	3rd (Delay)	4th (Delay)	1st link (CumuDelay)	2nd link (C.D.)	3rd link (C.D.)	4th link (C.D.)
0,074	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	0,333	0,000	0,000	0,000
0,148	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	0,667	0,000	0,000	0,000
0,222	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	1,000	0,000	0,000	0,000
0,296	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	1,333	0,000	0,000	0,000
0,370	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	1,667	0,000	0,000	0,000
0,444	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	2,000	0,000	0,000	0,000
0,519	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	2,333	0,000	0,000	0,000
0,593	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	2,667	0,000	0,000	0,000
0,593	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	3,000	0,000	0,000	0,000
0,667	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	3,333	0,000	0,000	0,000
0,741	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	3,667	0,000	0,000	0,000
0,815	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	4,000	0,000	0,000	0,000
0,889	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	4,333	0,000	0,000	0,000
0,963	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	4,667	0,000	0,000	0,000
1,037	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	5,000	0,000	0,000	0,000
1,111	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	5,333	0,000	0,000	0,000
1,185	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	5,667	0,000	0,000	0,000
1,259	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	6,000	0,000	0,000	0,000
1,333	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	6,333	0,000	0,000	0,000
1,407	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	6,667	0,000	0,000	0,000
1,481	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	7,000	0,000	0,000	0,000
1,556	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	7,333	0,000	0,000	0,000
1,630	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	7,667	0,000	0,000	0,000
1,704	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	8,000	0,000	0,000	0,000
1,778	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	8,333	0,000	0,000	0,000
1,852	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	8,667	0,000	0,000	0,000
1,926	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	9,000	0,000	0,000	0,000
2,000	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	9,333	0,000	0,000	0,000
2,074	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	9,667	0,000	0,000	0,000
2,148	0,167	0,000	0,000	0,000	0,333	0,000	0,000	0,000	10,000	0,000	0,000	0,000

Delay (s)	1st link (JT)	2nd (JT)	3rd (JT)	4th (JT)	1st link (SPEED)	2nd (SP.)	3rd (SP.)	4th (SP.)	1st link (Sp Factor)	2nd (Sp Factor)	3rd (Sp Factor)	4th (Sp Factor)
	360,000	350,000	180,000	280,000	360,000	350,000	180,000	280,000	*28,340*	*18,300*	*25,350*	*27,150*
	45,730	68,852	25,562	37,127	7,872	5,083	7,042	7,542	*1,000*	*1,000*	*1,000*	*1,000*
0,074	46,064	68,852	25,562	37,127	7,815	5,083	7,042	7,542	*0,993*	*1,000*	*1,000*	*1,000*
0,148	46,397	68,852	25,562	37,127	7,759	5,083	7,042	7,542	*0,986*	*1,000*	*1,000*	*1,000*
0,222	46,730	68,852	25,562	37,127	7,704	5,083	7,042	7,542	*0,979*	*1,000*	*1,000*	*1,000*
0,296	47,064	68,852	25,562	37,127	7,649	5,083	7,042	7,542	*0,972*	*1,000*	*1,000*	*1,000*
0,370	47,397	68,852	25,562	37,127	7,595	5,083	7,042	7,542	*0,965*	*1,000*	*1,000*	*1,000*
0,444	47,730	68,852	25,562	37,127	7,542	5,083	7,042	7,542	*0,958*	*1,000*	*1,000*	*1,000*
0,519	48,064	68,852	25,562	37,127	7,490	5,083	7,042	7,542	*0,951*	*1,000*	*1,000*	*1,000*
0,593	48,397	68,852	25,562	37,127	7,438	5,083	7,042	7,542	*0,945*	*1,000*	*1,000*	*1,000*
0,667	48,730	68,852	25,562	37,127	7,388	5,083	7,042	7,542	*0,938*	*1,000*	*1,000*	*1,000*
0,741	49,064	68,852	25,562	37,127	7,337	5,083	7,042	7,542	*0,932*	*1,000*	*1,000*	*1,000*
0,815	49,397	68,852	25,562	37,127	7,288	5,083	7,042	7,542	*0,926*	*1,000*	*1,000*	*1,000*
0,889	49,730	68,852	25,562	37,127	7,239	5,083	7,042	7,542	*0,920*	*1,000*	*1,000*	*1,000*
0,963	50,064	68,852	25,562	37,127	7,191	5,083	7,042	7,542	*0,913*	*1,000*	*1,000*	*1,000*
1,037	50,397	68,852	25,562	37,127	7,143	5,083	7,042	7,542	*0,907*	*1,000*	*1,000*	*1,000*
1,111	50,730	68,852	25,562	37,127	7,096	5,083	7,042	7,542	*0,901*	*1,000*	*1,000*	*1,000*
1,185	51,064	68,852	25,562	37,127	7,050	5,083	7,042	7,542	*0,896*	*1,000*	*1,000*	*1,000*
1,259	51,397	68,852	25,562	37,127	7,004	5,083	7,042	7,542	*0,890*	*1,000*	*1,000*	*1,000*
1,333	51,730	68,852	25,562	37,127	6,959	5,083	7,042	7,542	*0,884*	*1,000*	*1,000*	*1,000*
1,407	52,064	68,852	25,562	37,127	6,915	5,083	7,042	7,542	*0,878*	*1,000*	*1,000*	*1,000*
1,481	52,397	68,852	25,562	37,127	6,871	5,083	7,042	7,542	*0,873*	*1,000*	*1,000*	*1,000*
1,556	52,730	68,852	25,562	37,127	6,827	5,083	7,042	7,542	*0,867*	*1,000*	*1,000*	*1,000*
1,630	53,064	68,852	25,562	37,127	6,784	5,083	7,042	7,542	*0,862*	*1,000*	*1,000*	*1,000*
1,704	53,397	68,852	25,562	37,127	6,742	5,083	7,042	7,542	*0,856*	*1,000*	*1,000*	*1,000*
1,778	53,730	68,852	25,562	37,127	6,700	5,083	7,042	7,542	*0,851*	*1,000*	*1,000*	*1,000*
1,852	54,064	68,852	25,562	37,127	6,659	5,083	7,042	7,542	*0,846*	*1,000*	*1,000*	*1,000*
1,926	54,397	68,852	25,562	37,127	6,618	5,083	7,042	7,542	*0,841*	*1,000*	*1,000*	*1,000*
2,000	54,730	68,852	25,562	37,127	6,578	5,083	7,042	7,542	*0,836*	*1,000*	*1,000*	*1,000*
2,074	55,064	68,852	25,562	37,127	6,538	5,083	7,042	7,542	*0,830*	*1,000*	*1,000*	*1,000*
2,148	55,397	68,852	25,562	37,127	6,499	5,083	7,042	7,542	*0,826*	*1,000*	*1,000*	*1,000*
2,222	55,730	68,852	25,562	37,127	6,460	5,083	7,042	7,542	*0,821*	*1,000*	*1,000*	*1,000*

APPENDIX C

Computer Source Code of the Simulation Model SIBUFEM

```
//---------------------------------------------------------------------------

#ifndef InputDialogFH
#define InputDialogFH
//---------------------------------------------------------------------------
#include <vcl\System.hpp>
#include <vcl\Windows.hpp>
#include <vcl\SysUtils.hpp>
#include <vcl\Classes.hpp>
#include <vcl\Graphics.hpp>
#include <vcl\StdCtrls.hpp>
#include <vcl\Forms.hpp>
#include <vcl\Controls.hpp>
#include <vcl\Buttons.hpp>
#include <vcl\ExtCtrls.hpp>
//---------------------------------------------------------------------------
//enum TPriorityType {eptNo, eptSVD, eptAVL};

class TInputDialog : public TForm
{
__published:
		TButton *OKBtn;
		TButton *CancelBtn;
	TEdit *editSimPeriod;
	//TEdit *editInciStartTime;
	TLabel *Label2;
	TLabel *Label3;
	TEdit *editSimSpeed;
	TLabel *lblGpsAccuracy;
	TCheckBox *cbGpsError;
	TLabel *Label11;
	TEdit *editGpsAccuracy;
   TLabel *Label1;
   TLabel *Label4;
   TLabel *Label5;
   TLabel *Label6;
	TLabel *Label7;
	TRadioGroup *Type;
	TRadioGroup *Location;
	TRadioGroup *Severity;
	TRadioGroup *Duration;
	TLabel *Label8;
	TEdit *editInciStartTime;
	void __fastcall FormCreate(TObject *Sender);
	void __fastcall editSimPeriodKeyPress(TObject *Sender, char &Key);
 //	void __fastcall rgAvlClick(TObject *Sender);
	void __fastcall cbGpsErrorClick(TObject *Sender);
   void __fastcall LocationClick(TObject *Sender);
   void __fastcall SeverityClick(TObject *Sender);
   void __fastcall DurationClick(TObject *Sender);
	void __fastcall TypeClick(TObject *Sender);
	void __fastcall RadioButton1Click(TObject *Sender);
	void __fastcall RadioButton3Click(TObject *Sender);
private:

public:
		virtual __fastcall TInputDialog(TComponent* AOwner);
	int newPriorityOption;
	int newSimPeriod;
	int newSimSpeed;
	int newInciStartTime;
	float newGpsAccuracy;
	int newAvlSystem;
	int newInciType;
	int newInciLocation;
	int newInciSeverity;
	int newInciDuration;
	int gpsError;
	int newPassGenerateMode;
```

```
        int signalPhasesVisible;
        int carGeneratedVisible;
        bool __fastcall  Execute (void);
};
//---------------------------------------------------------------------------
extern PACKAGE TInputDialog *InputDialog;
//---------------------------------------------------------------------------
#endif
```

```
//---------------------------------------------------------------------
#include <vcl.h>
#pragma hdrstop
#include "InputDialogF.h"
//---------------------------------------------------------------------
#pragma resource "*.dfm"
TInputDialog *InputDialog;
//---------------------------------------------------------------------
__fastcall TInputDialog::TInputDialog(TComponent* AOwner)
        : TForm(AOwner)
{ }
//---------------------------------------------------------------------
void __fastcall TInputDialog::FormCreate(TObject *Sender)
{
 newSimPeriod=43200;//400000;    // 43200=12 hours simulation
 newSimSpeed=10000; //Def.=1000                  //starting values
 newGpsAccuracy=10.0;//0.0;//Def.=10
 newInciStartTime=10800;                //Incident starting in 3hrs from simulation start.

}
//---------------------------------------------------------------------------
bool __fastcall TInputDialog::Execute(void)
{
 editSimPeriod->Text=IntToStr(newSimPeriod);
 editSimSpeed->Text=IntToStr(newSimSpeed);
 editInciStartTime->Text=IntToStr(newInciStartTime);
 editGpsAccuracy->Text=FloatToStrF(newGpsAccuracy,ffNumber,4,2);

 ActiveControl=editSimPeriod;
 if(cbGpsError->Checked) gpsError=1;
 else editGpsAccuracy->Text=0.0;
  }
  else if(Location->ItemIndex==1) newInciLocation=2;

  if(Severity->ItemIndex==0)
  {
  newInciSeverity=1;
  }
  else if(Severity->ItemIndex==1) newInciSeverity=2;

  if(Duration->ItemIndex==0) newInciDuration=1;
  else if(Duration->ItemIndex==1) newInciDuration=2;
  else if(Duration->ItemIndex==2) newInciDuration=3;

  else {}

  if(cbGpsError->Checked) gpsError=1;
  else   gpsError=0;

  carGeneratedVisible=1;
  newAvlSystem=1;
  newSimPeriod=StrToInt(editSimPeriod->Text);

  return true;
 }
 else return false;
}

//---------------------------------------------------------------------------
void __fastcall TInputDialog::editSimPeriodKeyPress(TObject *Sender,char &Key)
{
 if((Key<'0')||(Key>'9'))
 {
  MessageBeep(0);
  Key='\0';
 }
}
```

```
//---------------------------------------------------------------------------
void __fastcall TInputDialog::cbGpsErrorClick(TObject *Sender)
{
 if(cbGpsError->Checked)
 {
  lblGpsAccuracy->Visible=true;
  editGpsAccuracy->Visible=true;
  editGpsAccuracy->Text=3.3;
 }
 else
 {
  lblGpsAccuracy->Visible=false;
  editGpsAccuracy->Visible=false;
  editGpsAccuracy->Text=0.0;
 }
}
```

```
//---------------------------------------------------------------------------

#ifndef MainSimFH
#define MainSimFH
//---------------------------------------------------------------------------
#include <Classes.hpp>
#include <Controls.hpp>
#include <StdCtrls.hpp>
#include <Forms.hpp>
#include <math.h>

#include "BusList.h"
#include "BusStop.h"
#include "Link.h"
#include "Bus.h"
#include "Signal.h"
#include "InputDialogF.h"

#include <ExtCtrls.hpp>
//---------------------------------------------------------------------------
class TMainSim : public TForm
{
__published:
	TButton *StartBtn;
	TButton *CloseBtn;
	void __fastcall StartBtnClick(TObject *Sender);
	void __fastcall CloseBtnClick(TObject *Sender);
private:
	int gpsCounter;
	int enterSimTime;
	int seedNumber;
	int transmitCounter;

	int simulationPeriod;
	int simulationSpeed;
	int InciStartTime;
	int InciStartTime2;
	int lag1;

	int routeLength;
	int windowWidth;
	int directionChangeLength;

	float updateInterval;
	int timeFactor;
	int busGenerationCounter;
	int busGenerationCounter2;
	int busGenerationCounterD;
	int busAvailableAtEnd;
	int busAvailableAtStart;
	int FleetSize;
	int busAvailableTime[5];
	int busAvailableTime2[5];

	int finalDist;
	int finalDistD;

	int avlSystem;

	int IncibusstopNo;
	int InciType;
	int InciLocation;
	int InciSeverity;
	int InciDuration;

	int Inumber;
	int Inumber2;
```

```
int Inumber3;
int Inumber4;


int PreviousGenTimeEnd;
int PrevGenTimeStart;
int PreviousGenTimeStart;
int TimeAtBusExitEnd;
int TimeAtBusExitStart;
int temp1111;
int InitialSlackTime;
int slackTime;

float InciPassInside;
int inciDistance;
int inciDistancePass;
int IncidentLinkNo;

float incilinksp;

int incidentBusId;
int incidentBusId2;
int incidentStartTime;
int incidentStartTime2;
int busJourneyTime;
float avBusJourneyTime;
int incidentDuration;
int incidentDuration2;

float gpsAccuracy;
int gpsErrorModelled;
float passengerRateFactor;

int passGenOption;
int signalVisible;
int carVisible;

int totalBusstopNo;
int totalSignalNo;
int totalLinkNo;

int busAtBusstopNos;
void RouteParameters(int startLag,float passRateFactor);
void DrawRoute();
void DrawSignalPhase();
void ChangeSignalPhase(int times);
void DisplaySimTime(int SimTime);
int CountDigit(int number);

void Simulate(int time,int busNumber,float busPosi2,int holdBusStop);
void MoveBus(int oldpos, int newpos, int nosBuses, int busNumber);
void MoveBusD(int oldpos, int newpos, int nosBuses);
void GenerateCar(int times,int timeFactors);
void DrawCar();
void DrawPass(int times);
void ChangeLinkPara(int busNumber,float busPosi2);
void CheckBusstops(int times,int busNumber,float busPosi2,int holdBusStop);

void InciCheck (int time, int IncibusstopNo, float busPosi2);
float linkSpeed;
float normlinkSpeed;

float AvlSystem(int time,int busNumber,float busPosi2);
int RandomBusGenerateO();
int RandomBusGenerateD();

void OutputHeaders();
void OutputFile(int busNumber,float passRateFactor);
void BusGenerateO(BusList*bList1, int option, int times);
```

```
        void BusGenerateD(BusList*bList2, int option, int times);
        void BusAvailabilityEnd(int time);
        void BusAvailabilityStart(int time);
        void CheckBusAvailabilityEnd(int time);
        void CheckBusAvailabilityStart(int time);
        char* SwitchOn(int switchNumber);

        BusList* list;
        BusList* list2;
        BusList* listD;
        BusStop*busstop[32];
        Link*link[32];
        Bus*currentBus;
        BusStop* currentBusstop;
        Link* currentLink;

public:
        __fastcall TMainSim(TComponent* Owner);
};
//---------------------------------------------------------------------------
extern PACKAGE TMainSim *MainSim;
//---------------------------------------------------------------------------
#endif
```

```
//---------------------------------------------------------------------------
#include <vcl.h>
#pragma hdrstop
#include "MainSimF_NEW.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource "*.dfm"
TMainSim *MainSim;
FILE
*stream201,*stream311,*stream212,*stream2121,*stream213,*stream214,*stream221,*stream
231,*stream232,
*stream241,*stream242,*stream111,*stream131,*stream119,*stream1111,*stream3111,*strea
m621,*stream622,*stream623,*stream624,
*stream512,*stream3221,*stream2111,*stream6565, *stream6566;

//stream 1** for input, 2** output, *1* bus related, *3* busstop related &
//---------stream 6 incident related-------------------------------------

//*************************************************************************************************
//-----------------1--------##Route Information##-----------------------------------------------
//-------------------------------------------------------------------------------------------
void TMainSim::RouteParameters(int startLag,float passRateFactor)
{
  routeLength = 4340;
  windowWidth = 1000;
  directionChangeLength =30000;


  busstop[0] = new BusStop(0,1,5,-434,0.07,90.0/passRateFactor,9999.0,0.0);
  busstop[1] = new BusStop(1,2,150,-286,0.026,72.0/passRateFactor,9999.0,0.0);
  busstop[2] = new BusStop(2,3,440,-154,0.01,257.3/passRateFactor,9999.0,0.0);
  busstop[3] = new BusStop(3,4,720,-92,0.01,129.1/passRateFactor,9999.0,0.0);
  busstop[4] = new BusStop(4,5,900,-25,0.008,240.0/passRateFactor,9999.0,0.0);
  busstop[5] = new BusStop(5,6,1250,69,0.008,276.9/passRateFactor,9999.0,0.0);
  busstop[6] = new BusStop(6,7,1610,138,0.008,720.0/passRateFactor,9999.0,0.0);
  busstop[7] = new BusStop(7,8,1800,171,0.305,69.9/passRateFactor,9999.0,0.0);
  busstop[8] = new BusStop(8,9,2060,343,0.048,141.2/passRateFactor,9999.0,0.0);
  busstop[9] = new BusStop(9,10,2410,420,0.014,171.9/passRateFactor,9999.0,0.0);
  busstop[10] = new BusStop(10,11,2620,529,0.03,225.0/passRateFactor,9999.0,0.0);
  busstop[11] = new BusStop(11,12,2990,635,0.038,180.0/passRateFactor,9999.0,0.0);
  busstop[12] = new BusStop(12,13,3210,695,0.056,1800.0/passRateFactor,9999.0,0.0);
  busstop[13] = new BusStop(13,14,3490,752,0.226,1200.0/passRateFactor,9999.0,0.0);
  busstop[14] = new BusStop(14,15,4010,919,0.47,3600.0/passRateFactor,9999.0,0.0);
  busstop[15] = new BusStop(15,15,4335,1045,1,99999.0/passRateFactor,9999.0,0.0);


  busstop[16] = new BusStop(16,16,30000+2*routeLength-4335,1345-
201,0.07,90.0/passRateFactor,9999.0,0.0);
  busstop[17] = new BusStop(17,17,30000+2*routeLength-4010,1472-
201,0.026,72.0/passRateFactor,9999.0,0.0);
  busstop[18] = new BusStop(18,18,30000+2*routeLength-3490,1638-
201,0.01,257.3/passRateFactor,9999.0,0.0);
  busstop[19] = new BusStop(19,19,30000+2*routeLength-3210,1695-
201,0.01,129.1/passRateFactor,9999.0,0.0);
  busstop[20] = new BusStop(20,20,30000+2*routeLength-2990,1756-
201,0.008,240.0/passRateFactor,9999.0,0.0);
  busstop[21] = new BusStop(21,21,30000+2*routeLength-2620,1861-
201,0.008,276.9/passRateFactor,9999.0,0.0);
  busstop[22] = new BusStop(22,22,30000+2*routeLength-2410,1971-
201,0.008,720.0/passRateFactor,9999.0,0.0);
  busstop[23] = new BusStop(23,23,30000+2*routeLength-2060,2048-
201,0.305,69.9/passRateFactor,9999.0,0.0);
  busstop[24] = new BusStop(24,24,30000+2*routeLength-1800,2210-
201,0.048,141.2/passRateFactor,9999.0,0.0);
  busstop[25] = new BusStop(25,25,30000+2*routeLength-1610,2253-
201,0.014,171.9/passRateFactor,9999.0,0.0);
```

```
  busstop[26] = new BusStop(26,26,30000+2*routeLength-1250,2321-
201,0.03,225.0/passRateFactor,9999.0,0.0);
  busstop[27] = new BusStop(27,27,30000+2*routeLength-900,2415-
201,0.038,180.0/passRateFactor,9999.0,0.0);
  busstop[28] = new BusStop(28,28,30000+2*routeLength-720,2483-
201,0.056,1800.0/passRateFactor,9999.0,0.0);
  busstop[29] = new BusStop(29,31,30000+2*routeLength-440,2544-
201,0.226,1200.0/passRateFactor,9999.0,0.0);
  busstop[30] = new BusStop(30,30,30000+2*routeLength-150,2677-
201,0.47,3600.0/passRateFactor,9999.0,0.0);
  busstop[31] = new BusStop(31,30,30000+2*routeLength-5,2825-
201,1,99999.0/passRateFactor,9999.0,0.0);
  totalBusstopNo =32;


//-------------DEFINING THE LINKS OF THE ROUTE(link: bus-stop to bus-stop)
//------------------------------------------------------------------------------
  //Link(int lID, int lStart,int lDist,float lTime,float lSpeed)
  link[0] = new Link (0,-10,-10,0.0,3.6);  //non-existing link for better output
  link[1] = new Link (1,0,150,0.0,5.9);
  link[2] = new Link (2,150,290,0.0,16.47);
  link[3] = new Link (3,440,280,0.0,27.15);
  link[4] = new Link (4,720,180,0.0,25.35);
  link[5] = new Link (5,900,350,0.0,18.30);
  link[6] = new Link (6,1250,360,0.0,28.34);
  link[7] = new Link (7,1610,190,0.0,23.81);
  link[8] = new Link (8,1800,260,0.0,11.96);
  link[9] = new Link (9,2060,350,0.0,35.00);
  link[10] = new Link (10,2410,210,0.0,9.97);
  link[11] = new Link (11,2620,370,0.0,17.19);
  link[12] = new Link (12,2990,220,0.0,29.77);
  link[13] = new Link (13,3210,280,0.0,22.86);
  link[14] = new Link (14,3490,520,0.0,12.99);
  link[15] = new Link (15,4010,330,0.0,11.60);

  link[16] = new Link (16,34340-10,-10,0.0,3.6); //non-existing link for better output
  link[17] = new Link (16,34340,330,0.0,11.60);
  link[18] = new Link (17,34670,520,0.0,12.99);
  link[19] = new Link (18,35190,280,0.0,22.86);
  link[20] = new Link (19,35470,220,0.0,29.77);
  link[21] = new Link (20,35690,370,0.0,17.19);
  link[22] = new Link (21,36060,210,0.0,9.97);
  link[23] = new Link (22,36270,350,0.0,35.00);
  link[24] = new Link (23,36620,260,0.0,11.96);
  link[25] = new Link (24,36880,190,0.0,23.81);
  link[26] = new Link (25,37070,360,0.0,28.34);
  link[27] = new Link (26,37430,350,0.0,18.30);
  link[28] = new Link (27,37780,180,0.0,25.35);
  link[29] = new Link (28,37960,280,0.0,27.15);
  link[30] = new Link (29,38240,290,0.0,16.47);
  link[31] = new Link (30,38530,150,0.0,5.9);

     totalLinkNo = 32;//17;//18;

}




//------------------------------------------------------------------------------------
void TMainSim::BusAvailabilityEnd(int time)
{
 int slackTime;
 i

   stream6565 = fopen("C:Slacktime.txt", "a+");
   fprintf(stream6565, " %7d", slackTime);
   fclose(stream6565);
```

```
 if (busAvailableTime[0]==999999) busAvailableTime[0]=time+slackTime;
 else if (busAvailableTime[1]==999999) busAvailableTime[1]=time+slackTime;
 else if (busAvailableTime[2]==999999) busAvailableTime[2]=time+slackTime;
 else if (busAvailableTime[3]==999999) busAvailableTime[3]=time+slackTime;
 else if (busAvailableTime[4]==999999) busAvailableTime[4]=time+slackTime;
}
//---------------------------------------------------------------------------
void TMainSim::CheckBusAvailabilityEnd(int time)
{
 if (time>=busAvailableTime[0])
 {

  }
  busAvailableTime[4]=999999;
 }
}
//---------------------------------------------------------------------------
///*
void TMainSim::CheckBusAvailabilityStart(int time)
{
 if (time>=busAvailableTime2[0])
 {
  }
  busAvailableTime2[4]=999999;
 }
}
//*/
//---------------------------------------------------------------------------
///*
void TMainSim::BusAvailabilityStart(int time)
{
 int slackTime2;
 int TimeAtBusExitStart=time;


);
   fprintf(stream6566, "\n %7d", slackTime2);                    /
   fclose(stream6566);
 if (busAvailableTime2[0]==999999)
 {
  busAvailableTime2[0]=time+slackTime2;
 }
 else if (busAvailableTime2[1]==999999) busAvailableTime2[1]=time+slackTime2;
 else if (busAvailableTime2[2]==999999) busAvailableTime2[2]=time+slackTime2;
 else if (busAvailableTime2[3]==999999) busAvailableTime2[3]=time+slackTime2;
 else if (busAvailableTime2[4]==999999) busAvailableTime2[4]=time+slackTime2;
}

//---------------------------------------------------------------------------
//*****************************************************************************

}
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
void TMainSim::ChangeLinkPara(int busNumber,float busPosi2)
{
 int nextLinkNos=currentBus->nextLink();
 if (nextLinkNos<totalLinkNo)
 {
  currentLink = link[nextLinkNos];
  if (busPosi2>=(currentLink->linkPosition()))
  {
   if(nextLinkNos==2)
   {
    currentBus->getSpeed(currentLink->ActualLinkSpeed(currentLink-
>InputLinkSpeed(),busNumber));
   }
   else currentBus->getSpeed(currentLink->InputLinkSpeed());
```

```
  }
 }
}
//-----------------------------------------------------------------------------

//-----------------------------------------------------------------------------------------------------------
//------------------------------------******* Checking for Incidents *******------------------------------------------------
//---------------------------------------InciCheck-------------------------------------------------------------
void TMainSim::InciCheck (int time, int IncibusstopNo, float busPosi2)
{
 if (InciLocation==1)
  {
   IncibusstopNo=6;
  }
 else if (InciLocation==2)
  {
   IncibusstopNo=14;
  }
 int nextBusstopNos=currentBus->nextBusstop();
 if (time>=InciStartTime&& currentBus->nextBusstop()==IncibusstopNo)
  {
  currentBusstop = busstop[currentBus->nextBusstop()];
  int

=1200;
     if (InciDuration==2)      incidentDuration=2400;
     if (InciDuration==3)      incidentDuration=3600;

     currentBus->changeSpeed(1);
     float bspeed=currentBus->BusSpeed();

     incidentBusId=currentBus->BusIdNumber();
     incidentStartTime=time;
     InciStartTime2=incidentStartTime+480;

     if (InciSeverity==1)
     {
            currentLink->IncidentLinkSpeed(1.0, time);
     }
      currentLink->IncidentLinkSpeed(1.0, time);
     }
    {

     int arrivalTime=currentBus->arriveBusstopTime(time,IncibusstopNo);


     if (InciDuration==1)      incidentDuration=1200;
     if (InciDuration==2)      incidentDuration=2400;
     if (InciDuration==3)      incidentDuration=3600;


     float bspeed=currentBus->BusSpeed();


     if (InciSeverity==1)
     {
      currentLink=link[currentBus->nextLink()-2];
      currentLink->IncidentLinkSpeed(1.0, time);
     }
     if (InciSeverity==2)
     {
      currentLink=link[currentBus->nextLink()-2];
      currentLink->IncidentLinkSpeed(1.0, time);
     }
    }
    InciStartTime=999999;
   }
  }
```

```
 if (time>=InciStartTime2).
  {
   currentBusstop = busstop[currentBus->nextBusstop()];
   int inciDistance=currentBusstop->BusstopPosition();

    if (InciType==1)
    {
     int arrivalTime=currentBus->arriveBusstopTime(time,IncibusstopNo);

     if (InciDuration==1)
     {
      incidentDuration2=1200;
     }
     if (InciDuration==2)
     {
      incidentDuration2=2400;
     }
     if (InciDuration==3)
     {
      incidentDuration2=3600;
     }

     }
    }
    InciStartTime2=999999;
  }
}

//-----------------------------------------------------------------------------------------------------------------



//------------------------------------------------------------------------------------------
//----------------------------------------------------------------------------------------------
//----4---Non changing functions-Drawing Routes, time display,etc.
//--------------------------------------------------------------------------------------------
//-------------------------------------------------------------------------------------------------

void TMainSim::DrawRoute()
{
 int routeSegmentNos=routeLength/windowWidth;      /
 Canvas->Pen->Color = clLime;O
 Canvas->Brush->Color = clLime;
 Canvas->Rectangle (0,0,windowWidth,routeSegmentNos*103);
 Canvas->Brush->Color = clLime;//clYellow;
 Canvas->Rectangle (0,routeSegmentNos*103, windowWidth,routeSegmentNos*500);
 for (int i=0; i<=routeSegmentNos; i++)
 {
  int segmentLength=windowWidth;
  if (i==routeSegmentNos) segmentLength=routeLength%windowWidth;
  int y=100+i*60;
  Canvas->Pen->Color = clBlack;
  Canvas->Pen->Style = psSolid;
  Canvas->MoveTo (0,y);
  Canvas->LineTo (segmentLength,y);
  Canvas->MoveTo (0,y+30);
  Canvas->LineTo (segmentLength,y+30);


  Canvas->Pen->Color = clSilver;
  Canvas->Brush->Color = clSilver;
  Canvas->Rectangle (0,y+1,segmentLength,y+30);
  Canvas->Pen->Color = clBlack;
  Canvas->Pen->Style = psDash;
  Canvas->MoveTo (0,y+15);
  Canvas->LineTo (segmentLength,y+15);
```

```
 }
//------------------------------------------------------------------------------------------------------------

 for (int j=0; j<totalBusstopNo; j++)    //BUS-STOPS DRAWING
 {
  if (j<(totalBusstopNo))
  {
   currentBusstop = busstop[j];
   int verticalShift=100;
   int busstopPosi = currentBusstop->BusstopPosition();
   if (busstopPosi>30000)
   {
    busstopPosi=30000+2*routeLength-busstopPosi;
    verticalShift=100+46;
   }
   int busstopPosition=busstopPosi%windowWidth;
   int y=verticalShift+(int (busstopPosi/windowWidth))*60;
   Canvas->Pen->Color = clBlue;
   Canvas->Brush->Color = clBlue;
   if(verticalShift<=100)Canvas->Rectangle (busstopPosition,y-6,busstopPosition+6,y);
   if(verticalShift>100)Canvas->Rectangle (busstopPosition,y-9,busstopPosition+6,y-15);

   Canvas->Font->Size=8; Canvas->Font->Color = clBlack; Canvas->Brush->Color = clLime;
   if(verticalShift<=100)
   {
   Canvas->TextOut(busstopPosition+7,y-14,"bus-stop");
   Canvas->TextOut(busstopPosition+50,y-14,j);
   }
   if(verticalShift>100)
   {
    Canvas->TextOut(busstopPosition+7,y-14,"bus-stop");
    Canvas->TextOut(busstopPosition+50,y-14,j);
   }
  }
 }

 //----------------Displaying Junction name, TRG name and time etc.------------------------------------------

 Canvas->Font->Color=clNavy; Canvas->Font->Size=12; Canvas->Brush->Color = clLime;
 Canvas->Font->Style=TFontStyles()<<fsBold; Canvas->Font->Style=TFontStyles()<<fsBold;
 Canvas->TextOut(10,10,"TRG, University of Southampton");
 Canvas->Font->Color=clNavy; Canvas->Font->Size=8; Canvas->Brush->Color = clLime; Canvas-
>Font->Style=TFontStyles()<<fsBold;
 Canvas->TextOut((windowWidth-210),20,"Simulation Time");

 Canvas->Font->Color=clBlack; Canvas->Font->Size=8; Canvas->Font->Style=TFontStyles();
 Canvas->TextOut((windowWidth-210),40,"Simulation period = ");
 Canvas->TextOut((windowWidth-210)+148,40,simulationPeriod);
 Canvas->TextOut((windowWidth-210),60,"GPS error SD = ");
 Canvas->TextOut((windowWidth-210)+148,60,FloatToStrF(gpsAccuracy,ffNumber,4,2));
}


//-------------------------------------------------------------------------------------------
void TMainSim::DisplaySimTime(int SimTime)   //Displaying Simulation Time
{
 char *str;
 double num;
 int dec, sign, ndig = 0;//5;
 num = SimTime;///60.0;//2.0;
 str = fcvt(num, ndig, &dec, &sign);
 Canvas->Brush->Color = clLime; Canvas->Font->Color=clNavy; Canvas->Font->Size=8;//12;
 Canvas->Font->Style=TFontStyles()<<fsBold;
 Canvas->TextOut((windowWidth-62),20,str);
 Canvas->Font->Style = TFontStyles(); //clears
}
int TMainSim::CountDigit(int number)
{
 int digit;
```

```
 if(number<=9) digit =1;
 else digit =2;
 return digit;
}

//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
void TMainSim::DrawPass(int times)
{
 for (int j=0; j<totalBusstopNo; j++)
 {
  currentBusstop = busstop[j];
  int busstopPosi = currentBusstop->BusstopPosition();
  int busstopPosition = busstopPosi%windowWidth;
  int y1=85+(int(busstopPosi/windowWidth))*80+5;
  int newPassNos = currentBusstop->boardPassGenerate(times);

  int passPosi=busstopPosition;
  Canvas->Pen->Color = clSilver;
  Canvas->Brush->Color = clSilver;
  Canvas->Rectangle (passPosi,y1,(passPosi-(newPassNos+1)*3),y1+4);

  if(newPassNos>0)
  {
   for(int i=0;i<newPassNos;i++)
   {
    int passPosi=busstopPosition-(i*3);
    Canvas->Pen->Color = clPurple;
    Canvas->Brush->Color = clPurple;
    Canvas->Rectangle (passPosi,y1,(passPosi-2),y1+4);  //adding passengers
   }
  }
 }
}
//-----------5----------------##FORMATTED OUTPUT (TXT FILES)##--------------------------------------------------
//------------------------------------------------------------------------------------------------------------

void TMainSim::OutputHeaders()
{

 stream201 = fopen("SibufemOutputs.txt", "w+");
 fclose(stream201);

 stream311 = fopen("zBusGenerateFile.txt", "w+");
 fprintf(stream311, "\n %s %s %s %s %s", "BusNum
","Time","Busstopdeviate","finalp","busAvailableAtStart");
 stream3111 = fopen("zBusGenerateFile_D.txt", "w+");
 fprintf(stream3111, "\n %s %s %s %s", "BusNumD  ","Time","busAvailableAtEnd","finalpD");
 stream212 = fopen("SimOut.txt", "w+");
 fprintf(stream212, "\n %c %c %s %c %s %c", ' ',' ',"Time", ' ', "Distance", ' ');
 fclose(stream212);
 stream2121 = fopen("SimOut_D.txt", "w+");
 fprintf(stream2121, "\n %c %c %s %c %s %c", ' ',' ',"Time", ' ', "Distance", ' ');
 fclose(stream2121);
 stream213 = fopen("BusAVLPosition.txt", "w+");
 fprintf(stream213, "\n %c %c %s %c %s %c", ' ',' ',"Time", ' ', "Distance", ' ');
 fclose(stream213);
 stream214 = fopen("AVLCentrePosition.txt", "w+");
 fprintf(stream214, "\n %c %c %s %c %s %c", ' ',' ',"Time", ' ', "Distance", ' ');
 fclose(stream214);
 stream221 = fopen("LinkOut.txt", "w+");
 fprintf(stream221, "\n %s %s %s", "LinkNo","BusNo","Time");
 stream231 = fopen("BusstopOut.txt", "w+");
 fprintf(stream231, "\n %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s",
"Stop","Bus","Arrive","Depart","Headway","PassA","PassB","PassIn","Occupy","Dwell","Wait","Jo
urT","SchT","Late","devTime","busAtBusstopNos","busPosi2", "BusJourneyTime");
 stream2111 = fopen("BusJourneyTimeOutput.txt", "w+");
 fprintf(stream2111, "\n %s %s %s", "busNumber","busJourneyTime","LinkID");
```

```
 stream232 = fopen("BusArrivals.txt", "w+");
 stream241 = fopen("SignalStages.txt", "w+");
 fprintf(stream241, "\n %s %s %s %s %s %s %s %s %s", " 1","    2","    3","    4","    5","    6","    7","
8","    9");
 fprintf(stream241, "\n %s %s %s %s %s %s %s %s %s",
"Time","Stage","Car","Stage","Car","Stage","Car","Stage","Car");
 stream242 = fopen("SignalOut.txt", "w+");
 fprintf(stream242, "\n %s %s %s %s %s %s %s %s %s",
"Signal","BusNo","Arrive","Depart","QCar","CrLag","SgLag","TotLag","DelayToCars");
}





//--------------------------Output file with main Simulation results--------------------------------------------------
//----------------------------------------------------------------------------------------------------------------

void TMainSim::OutputFile(int busNumber,float passRateFactor)
{
 float totalCarDelays=0.0; float grandTotalCarDelays=0.0;

 stream201 = fopen("SibufemOutputs.txt", "w+");
 fprintf(stream201, "\n %s\n %s","           S I B U F E M , Simulation Output","
___________________________________");
 fprintf(stream201, "\n \n \n %s","Bus Fleet Management Simulation of hypothetical
corridor");
 fprintf(stream201, "\n %s","AVL System =");
 fprintf(stream201, "\n %s %2.1f","Passenger generation factor =",passengerRateFactor);
 fprintf(stream201, "\n %s %5d %s","Simulation period =",enterSimTime,"secs");
 fprintf(stream201, "\n %s %5d %s","Actual simulation period =",simulationPeriod,"secs");
//--------------------------------------------------------------------------------
 fprintf(stream201, "\n\n
%s","==============================================================");
 fprintf(stream201, "\n %s\n %s","                ***INPUT
PARAMETERS***","==============================================================");

 fprintf(stream201, "\n \n %s %5.2f %s","GPS error SD =",gpsAccuracy,"m");
 fprintf(stream201, "\n %s","----------------------------------------------------------");

 fprintf(stream201, "\n %s%s","Passenger generation option - ",SwitchOn(passGenOption));
 fprintf(stream201, "\n %s%s","Visible signal period option - ",SwitchOn(signalVisible));
 fprintf(stream201, "\n %s%s","Visible general traffic option - ",SwitchOn(carVisible));

 fprintf(stream201, "\n\n %s\n %s","Bus stop Parameters","---------------------------------------------------------
");
 fprintf(stream201, "\n %s %s %s %s %s
%s","Stop","StpPos","DwTime","DwSd","Alight%","BoardRate");
 for (int j=0; j<totalBusstopNo; j++)
 {
  if (j>0)fprintf(stream201, "\n%3d %5d %8.1f %5.1f %6.3f %10.2f",j,busstop[j]-
>BusstopPosition(),
  busstop[j]->AvDwellTime(),busstop[j]->SdDwellTime(),busstop[j]-
>AlightPercentage(),busstop[j]->BoardPassRate());
 }

 fprintf(stream201, "\n \n
%s","==============================================================");
 fprintf(stream201, "\n %s\n %s","                    ***SIMULATION
RESULTS***","==============================================================");
 fprintf(stream201, "\n %s %s %s
%s","busstopID","avBusJourneyTime","avExcessWaitingTime","avBusSpeed");
 for (int j=0; j<totalBusstopNo; j++)
 {
  currentBusstop = busstop[j];
  float avBusJourneyTime=currentBusstop->AvBusJourneyTime();
j,avBusJourneyTime,avExcessWaitingTime,avBusSpeed);
```

```
  if (j==1)//(j>0)
  {
   Canvas->Brush->Color=clLime;
   Canvas->Font->Color=clNavy; Canvas->Font->Size=10; Canvas->Font->Style=TFontStyles();
   Canvas->TextOut(150,430,"Summary of SIBUFEM results: ");
   Canvas->Font->Color=clMaroon; Canvas->Font->Size=10;
   Canvas->TextOut(150,400+j*60,"The Average Bus Journey Time is ");
   Canvas->TextOut(354,400+j*60,avBusJourneyTime);
   Canvas->TextOut(150,420+j*60,"the Average Excess Waiting Time is ");
   Canvas->TextOut(369,420+j*60,avExcessWaitingTime);
   Canvas->TextOut(150,440+j*60,"and the Average Bus Speed is ");
   Canvas->TextOut(336,440+j*60,avBusSpeed);
  }
 }

 fclose(stream201);  // 18/11/09

 //Bus arrival at different bus stops output
 //---------------------------------------------------------------------------
 stream232 = fopen("BusArrivals.txt", "w+");
 fprintf(stream232, "\n \n %s","BusStp");
 for (int i = 0; i<totalBusstopNo; i++) fprintf(stream232, "%6d",i);
 fprintf(stream232, "\n %s","Distce");
 for (int i = 0; i<totalBusstopNo; i++) {fprintf(stream232, "%6d",busstop[i]->BusstopPosition());}
 fprintf(stream232, "\n %s","BusNum");
 for (int j=0;j<3; j++)
 {
  fprintf(stream232, "\n %4d %c",j+1, ' ');
  for (int i = 0; i<totalBusstopNo; i++) {fprintf(stream232, "%6d",stopOut[i][j]);}
 }
 fclose(stream232);
}
//---------------------------------------------------------------------------
char* TMainSim::SwitchOn(int switchNumber)
{char* switching;
 if(switchNumber==1) switching="On";
 else switching="Off";
 return switching;
}

//------------6---##On-time drawing cars stops buses etc in place------------------------------------------------
//------------------------##Normal Distribution of Bus Generation##----------------------------------------------
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------

void TMainSim::MoveBus(int oldpos, int newpos, int nosBuses, int busNumber)
{
 if(newpos<0+directionChangeLength)
 {
  int oldpos1=oldpos-0;
  int oldPosition = oldpos1%windowWidth;
  int y1=102+(int (oldpos1/windowWidth))*60;
  Canvas->Pen->Width=1;
  Canvas->Pen->Color = clSilver;
  Canvas->Brush->Color = clSilver;
  Canvas->Rectangle ((oldPosition),y1,((oldPosition)-14),y1+6);

   //for display of busNumbers...
   Canvas->Font->Size=4; Canvas->Pen->Width=3; Canvas->Font->Color = clSilver; Canvas-
>Brush->Color = clSilver;
   //Canvas->TextOut(oldPosition+4,y1-1,"");
   Canvas->TextOut(oldPosition-23,y1-1,busNumber);
   //

  if(newpos<0+routeLength)
  {
   int newpos1=newpos-0;
   int newPosition = newpos1%windowWidth;
```

```
    int y2=102+(int ((newpos1-2)/windowWidth))*60;
    Canvas->Pen->Width=1;
    Canvas->Pen->Color = clMaroon;
    Canvas->Brush->Color = clRed;
    Canvas->Rectangle (newPosition,y2,((newPosition)-14),y2+6);

     //for display of busNumbers...
     Canvas->Font->Size=4; Canvas->Pen->Width=3; Canvas->Font->Color = clRed; Canvas-
>Brush->Color = clSilver;
     //Canvas->TextOut(newPosition+4,y2-1,"");
     Canvas->TextOut(newPosition-23,y2-1,busNumber);
     //
   }
  }

  if(newpos>directionChangeLength)    //moving buses in the Leftbound direction
  {
   int oldpos2=30000+2*routeLength-oldpos;
   int oldPosition2 = oldpos2%windowWidth;
   int y12=124+(int (oldpos2/windowWidth))*60;      /
   Canvas->Pen->Width=1;
   Canvas->Pen->Color = clSilver;
   Canvas->Brush->Color = clSilver;
   Canvas->Rectangle((oldPosition2),y12,(oldPosition2-14),y12+6);

    //for display of busNumbers...
    Canvas->Font->Size=4; Canvas->Pen->Width=1; Canvas->Font->Color = clSilver; Canvas-
>Brush->Color = clSilver;
    //Canvas->TextOut(oldPosition2-38,y12-8,"");
    Canvas->TextOut(oldPosition2-40,y12-8,busNumber);
    //

   if(newpos<30000+2*routeLength)
   {
    int newpos2=30000+2*routeLength-newpos;
    int newPosition2 = newpos2%windowWidth;
    int y22=124+(int ((newpos2-2)/windowWidth))*60;
    Canvas->Pen->Width=1;
    Canvas->Pen->Color = clNavy;
    Canvas->Brush->Color = clPurple;
    Canvas->Rectangle (newPosition2,y22,(newPosition2-14),y22+6);

     //for display of busNumbers...
     Canvas->Font->Size=4; Canvas->Pen->Width=1; Canvas->Font->Color = clMaroon; Canvas-
>Brush->Color = clSilver;
     //Canvas->TextOut(newPosition2-38,y22-8,"");
     Canvas->TextOut(newPosition2-40,y22-8,busNumber);
     //
   }
  }
  for (int j=1; j<500000000/(simulationSpeed*nosBuses); j++)  {}
}
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
int TMainSim::RandomBusGenerateO()
{
 int valueX1=0;
 while(valueX1==0)
 {
  float valuePx;
  int valueX = rand()%1500;
  if (valueX>250)  valuePx = ((exp(-1*((valueX-250)/50)))/50);
  else valuePx=-1.0;

  float valueY = (rand()%1000)/250000.0;
  if(valueY<=valuePx)    valueX1= valueX;
  else valueX1=0;
 }
```

```
 return valueX1;
}


//------------7-------------##Starts set the model initialization##----------------------------------------------
//-----------------------------------------------------------------------------

__fastcall TMainSim::TMainSim(TComponent* Owner): TForm(Owner){ busAtBusstopNos=1;}
//-----------------------------------------------------------------------------
void __fastcall TMainSim::StartBtnClick(TObject *Sender)
{
 if(InputDialog->Execute())
 {

  InciType=InputDialog->newInciType;
  InciLocation=InputDialog->newInciLocation;
  InciSeverity=InputDialog->newInciSeverity;
  InciDuration=InputDialog->newInciDuration;

  gpsErrorModelled=InputDialog->gpsError;
  passengerRateFactor=1;
  passGenOption=InputDialog->newPassGenerateMode;
 }
 else Close();

//-----------8------------------## Initial values ##;------------------------------------------------------------
//-----------------------------------------------------------------------------------------------------------


 Inumber=0;
 Inumber2=0;
 Inumber3=0;
 Inumber4=0;
  inciDistance=99999;

  if (InciLocation==1)
  {
   IncibusstopNo=6;
  }
  else if (InciLocation==2)
  {
   IncibusstopNo=14;
  }


=0
 busAvailableAtStart=FleetSize;
 PreviousGenTimeEnd=0;
 PrevGenTimeStart=0;
 TimeAtBusExitEnd=0;
 TimeAtBusExitStart=0;



 for (int k=0;k<=4;k++)
  {
   busAvailableTime[k]=999999;
   busAvailableTime2[k]=999999;
  }
 seedNumber=9;
 updateInterval=1.0;
 timeFactor = 1.0;
 //-----------------------------------------------------------------------------
 BusList* list = new BusList();
 BusList* list2 = new BusList();



 RouteParameters(0,passengerRateFactor);
```

```
DrawRoute();
OutputHeaders();






//-------------------------------------------------------------------------------------------
//------------9--------##Simulation Activating##---------------------
//---------------------------------------------------------------------------------------------
//-----------------Start of Simulation - Generating Buses at Origin according to time-----------
//--------------------------------------------------------------------------------------------------------------
randomize();
int simuperiod=simulationPeriod;
enterSimTime=simuperiod;
gpsCounter=1;
for (int time = 0; time <simulationPeriod; time++)
{
 DisplaySimTime(time/timeFactor); //Displays time
 float realTime = (time/timeFactor);
 stream212 = fopen("SimOut.txt", "a+");
 fprintf(stream212, "\n %c %6.2f %c", ' ', realTime, ' ');
 fclose(stream212);
 stream2121 = fopen("SimOut_D.txt", "a+");
 fprintf(stream2121, "\n %c %6.2f %c", ' ', realTime, ' ');
 fclose(stream2121);
 fprintf(stream213, "\n %c %6.1f %c", ' ', realTime, ' ');
 fclose(stream213);  // 18/11/09

    stream3221 = fopen("C:zLinkBusSpeedProfileOut22222.txt", "a+");
  fprintf(stream3221, "\n %c %6.2f %c", ' ', realTime, ' ');
  fclose(stream3221);



 if (realTime<simulationPeriod) fprintf(stream241, "\n %5.1f",realTime);

 if(passGenOption==1)DrawPass(time/timeFactor);
 //-----------------------------------------------------------------------------
 //-----------------------------------------------------------------------------
 BusGenerateD(list2,0,time/timeFactor);

// CIRCULATION!
//  if(time<simuperiod)
 //-------------------------------------------------------------------------------------------
 //------------10---------------------##Changing attributes of buses in a loop##-----------
 //-----------------------------------------------------------------------------------------
 int exitNumber=0;
 int exitNumber2=0;
 for(int number = 1; number <=list->getLength(); number++)
 {

  currentBus = list->getBus(number-1);
  int busNumber=currentBus->BusIdNumber();
  int numberOfBuses = list->getLength();
  float busPosi1 = currentBus->oldPosition();
  float busPosi2 = currentBus->newPosition(timeFactor);


//---------------------------------------------------Bus Breakdown Incident---------------------------------------------------


   }

                  // SLIGHT SEVERITY HERE...  4 links uspream effect.

    if (InciSeverity==1)
    {
```

```
if (InciDuration==1)
{
 if (InciLocation==1)
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link4.txt", "r");
 }
 else
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link4.txt", "r");
 }
}
if (InciDuration==2)
{
 if (InciLocation==1)
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link4.txt", "r");
 }
 else
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link4.txt", "r");
 }
}
if (InciDuration==3)
{
 if (InciLocation==1)
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link4.txt", "r");
 }
 else
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOC2Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOc2Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOc2Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOc2Link4.txt", "r");
 }
}

float temp;                                   // 1st link upstream
long curpos = Inumber*7;
fseek(stream621, curpos, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream621);
 temp = atof(msg);
}
fclose(stream621);

float temp2;                                  // 2nd link upstream
long curpos2 = Inumber2*7;
fseek(stream622, curpos2, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream622);
 temp2 = atof(msg);
```

```
}
fclose(stream622);

float temp3;                          // 3rd link upstream
long curpos3 = lnumber3*7;
fseek(stream623, curpos3, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream623);
 temp3 = atof(msg);
}
fclose(stream623);

float temp4;                          // 4th link upstream
long curpos4 = lnumber4*7;
fseek(stream624, curpos4, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream624);
 temp4 = atof(msg);
}
fclose(stream624);

if (temp==0.0)                        // 1st link upstream
{
 currentLink=link[currentBus->nextLink()-2];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-2];
 currentLink->IncidentLinkSpeed(temp, time);
}

if (temp2==0.0)                       // 2nd link upstream
{
 currentLink=link[currentBus->nextLink()-3];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-3];
 currentLink->IncidentLinkSpeed(temp2, time);
}

if (temp3==0.0)                       // 3rd link upstream
{
 currentLink=link[currentBus->nextLink()-4];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-4];
 currentLink->IncidentLinkSpeed(temp3, time);
}

if (temp4==0.0)                       // 4th link upstream
{
 currentLink=link[currentBus->nextLink()-5];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-5];
 currentLink->IncidentLinkSpeed(temp4, time);
}

lnumber++;
lnumber2++;
```

```
 Inumber3++;
 Inumber4++;

 if (temp==0 && temp2==0 && temp3==0 && temp4==0)
 {
  busPosi2=99999;
 }
}

                     // MEDIUM SEVERITY HERE....

if (InciSeverity==2)
{
 if (InciDuration==1)
 {
  if (InciLocation==1)
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link4.txt", "r");
  }
  else
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link4.txt", "r");
  }
 }
 if (InciDuration==2)
 {
  if (InciLocation==1)
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link4.txt", "r");
  }
  else
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link4.txt", "r");
  }
 }
 if (InciDuration==3)
 {
  if (InciLocation==1)
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link4.txt", "r");
  }
  else
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link4.txt", "r");
  }
 }
 float temp;                              // 1st link upstream
 long curpos = Inumber*7;
 fseek(stream621, curpos, SEEK_CUR);
 {
  char msg[20];
```

```
 fgets(msg,20,stream621);
 temp = atof(msg);
}
fclose(stream621);

float temp2;                              // 2nd link upstream
long curpos2 = lnumber2*7;
fseek(stream622, curpos2, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream622);
 temp2 = atof(msg);
}
fclose(stream622);

float temp3;                              // 3rd link upstream
long curpos3 = lnumber3*7;
fseek(stream623, curpos3, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream623);
 temp3 = atof(msg);
}
fclose(stream623);

float temp4;                              // 4th link upstream
long curpos4 = lnumber4*7;
fseek(stream624, curpos4, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream624);
 temp4 = atof(msg);
}
fclose(stream624);

if (temp==0.0)                            // 1st link upstream
{
 currentLink=link[currentBus->nextLink()-2];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-2];
 currentLink->IncidentLinkSpeed(temp, time);
}

if (temp2==0.0)                           // 2nd link upstream
{
 currentLink=link[currentBus->nextLink()-3];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-3];
 currentLink->IncidentLinkSpeed(temp2, time);
}

if (temp3==0.0)                           // 3rd link upstream
{
 currentLink=link[currentBus->nextLink()-4];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[currentBus->nextLink()-4];
 currentLink->IncidentLinkSpeed(temp3, time);
}

if (temp4==0.0)                           // 4th link upstream
```

```
      {
       currentLink=link[currentBus->nextLink()-5];
       currentLink->IncidentLinkSpeed(1.0, time);
      }
      else
      {
       currentLink=link[currentBus->nextLink()-5];
       currentLink->IncidentLinkSpeed(temp4, time);
      }

      Inumber++;
      Inumber2++;
      Inumber3++;
      Inumber4++;

      if (temp==0 && temp2==0 && temp3==0 && temp4==0)
      {
       busPosi2=99999;
      }
     }
    }
   }


//---------------------------------------------Other type Incidents------------------------------------------------------------


                  // SLIGHT SEVERITY HERE...  4 links uspream effect.

     if (InciSeverity==1)
     {
      if (InciDuration==1)
      {
       if (InciLocation==1)
       {
        stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link1.txt", "r");
        stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link2.txt", "r");
        stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link3.txt", "r");
        stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1Loc1Link4.txt", "r");
       }
       else
       {
        stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link1.txt", "r");
        stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link2.txt", "r");
        stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link3.txt", "r");
        stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur1LOC2Link4.txt", "r");
       }
      }
      if (InciDuration==2)
      {
       if (InciLocation==1)
       {
        stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link1.txt", "r");
        stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link2.txt", "r");
        stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link3.txt", "r");
        stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2Loc1Link4.txt", "r");
       }
       else
       {
        stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link1.txt", "r");
        stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link2.txt", "r");
        stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link3.txt", "r");
        stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur2LOC2Link4.txt", "r");
       }
      }
      if (InciDuration==3)
      {
       if (InciLocation==1)
       {
```

```
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3Loc1Link4.txt", "r");
 }
 else
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOC2Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOc2Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOc2Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev1Dur3LOc2Link4.txt", "r");
 }
}

float temp;                                   // 1st link upstream
long curpos = Inumber*7;
fseek(stream621, curpos, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream621);
 temp = atof(msg);
}
fclose(stream621);

float temp2;                                  // 2nd link upstream
long curpos2 = Inumber2*7;
fseek(stream622, curpos2, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream622);
 temp2 = atof(msg);
}
fclose(stream622);

float temp3;                                  // 3rd link upstream
long curpos3 = Inumber3*7;
fseek(stream623, curpos3, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream623);
 temp3 = atof(msg);
}
fclose(stream623);

float temp4;                                  // 4th link upstream
long curpos4 = Inumber4*7;
fseek(stream624, curpos4, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream624);
 temp4 = atof(msg);
}
fclose(stream624);


if (temp==0.0)                                // 1st link upstream
{
 currentLink=link[IncibusstopNo];
 currentLink->IncidentLinkSpeed(1.0, time);

}
else
{
 currentLink=link[IncibusstopNo];
 currentLink->IncidentLinkSpeed(temp, time);
}

if (temp2==0.0)                               // 2nd link upstream
{
```

```
 currentLink=link[IncibusstopNo-1];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[IncibusstopNo-1];
 currentLink->IncidentLinkSpeed(temp, time);
}

if (temp3==0.0)                                    // 3rd link upstream
{
 currentLink=link[IncibusstopNo-2];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[IncibusstopNo-2];
 currentLink->IncidentLinkSpeed(temp, time);
}

if (temp4==0.0)                                   // 4th link upstream
{
 currentLink=link[IncibusstopNo-3];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
{
 currentLink=link[IncibusstopNo-3];
 currentLink->IncidentLinkSpeed(temp, time);
}

Inumber++;
Inumber2++;
Inumber3++;
Inumber4++;

}
                              // MEDIUM SEVERITY HERE....

if (InciSeverity==2)
{
 if (InciDuration==1)
 {
  if (InciLocation==1)
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1Loc1Link4.txt", "r");
  }
  else
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur1LOC2Link4.txt", "r");
  }
 }
 if (InciDuration==2)
 {
  if (InciLocation==1)
  {
   stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link1.txt", "r");
   stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link2.txt", "r");
   stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link3.txt", "r");
   stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2Loc1Link4.txt", "r");
  }
  else
  {
```

```
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur2LOC2Link4.txt", "r");
 }
}
if (InciDuration==3)
{
 if (InciLocation==1)
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3Loc1Link4.txt", "r");
 }
 else
 {
  stream621 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link1.txt", "r");
  stream622 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link2.txt", "r");
  stream623 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link3.txt", "r");
  stream624 = fopen("C:\\SIBUFEM\\SimulationData\\Sev2Dur3LOC2Link4.txt", "r");
 }
}
float temp;                         // 1st link upstream
long curpos = Inumber*7;
fseek(stream621, curpos, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream621);
 temp = atof(msg);
}
fclose(stream621);

float temp2;                        // 2nd link upstream
long curpos2 = Inumber2*7;
fseek(stream622, curpos2, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream622);
 temp2 = atof(msg);
}
fclose(stream622);

float temp3;                         // 3rd link upstream
long curpos3 = Inumber3*7;
fseek(stream623, curpos3, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream623);
 temp3 = atof(msg);
}
fclose(stream623);

float temp4;                        // 4th link upstream
long curpos4 = Inumber4*7;
fseek(stream624, curpos4, SEEK_CUR);
{
 char msg[20];
 fgets(msg,20,stream624);
 temp4 = atof(msg);
}
fclose(stream624);

if (temp==0.0)                         // 1st link upstream
{
 currentLink=link[IncibusstopNo];
 currentLink->IncidentLinkSpeed(1.0, time);
}
else
```

```
      {
       currentLink=link[IncibusstopNo];
       currentLink->IncidentLinkSpeed(temp, time);
      }

      if (temp2==0.0)                    // 2nd link upstream
      {
       currentLink=link[IncibusstopNo-1];
       currentLink->IncidentLinkSpeed(1.0, time);
      }
      else
      {
       currentLink=link[IncibusstopNo-1];
       currentLink->IncidentLinkSpeed(temp, time);
      }

      if (temp3==0.0)                    // 3rd link upstream
      {
       currentLink=link[IncibusstopNo-2];
       currentLink->IncidentLinkSpeed(1.0, time);
      }
      else
      {
       currentLink=link[IncibusstopNo-2];
       currentLink->IncidentLinkSpeed(temp, time);
      }

      if (temp4==0.0)                    // 4th link upstream
      {
       currentLink=link[IncibusstopNo-3];
       currentLink->IncidentLinkSpeed(1.0, time);
      }
      else
      {
       currentLink=link[IncibusstopNo-3];
       currentLink->IncidentLinkSpeed(temp, time);
      }

      Inumber++;
      Inumber2++;
      Inumber3++;
      Inumber4++;

     }
    }
   }

//-------------------------------------------------------------------------------------------------------------------------------

     else simulationPeriod=simulationPeriod;
    }
   if (busPosi2==99999)
   {
    exitNumber=number;
   }
  }

  if(exitNumber>0)
  {
  }
  CheckBusAvailabilityEnd(time);


//CIRCULATION!
  for(int number2 = 1; number2 <=list2->getLength(); number2++)
  {


   }
```

```
   if(busPosi22==99999) exitNumber2=number2;// delete buses only after current loop is
completed
  }
  if(exitNumber2>0)
  {
   list2->exitBus(exitNumber2-1);// delete buses after completing the loop
   exitNumber2=0;
   BusAvailabilityStart(time);
  }
  CheckBusAvailabilityStart(time);
 }
 OutputFile(list->getLength(),passengerRateFactor);
}


//---------------------------------------------------------------------------
//---------------------------------------------------------------------------

void __fastcall TMainSim::CloseBtnClick(TObject *Sender)
{
 Close();
}
//===========================================================
```

```
//---------------------------------------------------------------------------

#ifndef LinkH
#define LinkH

#include <stdlib.h>
#include <iostream.h>


class Link
{
  public:
    Link(int lID, int lStart,int lDist,float lTime,float lSpeed);
    ~Link();
    int linkIdentity();
    int linkPosition();
    int linkDistance();
    int thisLinkTime(int times);
    float InputLinkSpeed();
    float ActualLinkSpeed(float presentSpeed, int busNumber);
    void IncidentLinkSpeed(float sevdegree, int time);


  private:
    int linkID;
    int linkStart;
    int linkDist;
    float linkTime;
    float linkSpeed;
    float inputSpeed;
    float normlinkSpeed;

};
//---------------------------------------------------------------------------
#endif
```

```
#include "link.h"

FILE *stream121, *stream321, *stream3221, *stream361;

Link::Link(int lID,int lStart,int lDist,float lTime,float lSpeed)
{
  linkID = lID;
  linkStart = lStart;
  linkDist = lDist;
  linkTime= lTime;
  linkSpeed = lSpeed;
  inputSpeed = lSpeed;

   stream3221 = fopen("C:zLinkBusSpeedProfileOut22222.txt", "w+");
   fprintf(stream3221, "\n %s %s %s %s", "  Time  ","     linkID","
linkSpeed","busJourneyTime");
   fclose(stream3221);

}
//------------------------------------------------------------------------
Link::~Link()                    {}
int Link::linkIdentity()         {return linkID;}
int Link::linkPosition()         {return linkStart;}
int Link::linkDistance()         {return linkDist;}
int Link::thisLinkTime(int times) {return linkTime;}
float Link::InputLinkSpeed()
{
   stream3221 = fopen("C:zLinkBusSpeedProfileOut22222.txt", "a+");
   fprintf(stream3221, "%c %c %5d %c %c %5.2f %c",' ',' ',linkID,' ',' ', linkSpeed, ' ');
   fclose(stream3221);

 return linkSpeed;
}

//----------------------------------------------------------------------------



float Link::ActualLinkSpeed(float presentSpeed, int busNumber)
{

   stream121 = fopen("C:\\SIBUFEM\\SimulationData\\abusVaryTimeWidth8.txt", "r");
   float temp;
   long curpos = (busNumber-1)*10; // shift cursor pos by 8 places
   fseek(stream121, curpos, SEEK_CUR);
   {
    char msg[20];
    fgets(msg,20,stream121);
    temp = atof(msg);
   }
   fclose(stream121);



   float actualLinkSpeed=linkDist*3.6/deviatedTime;
   stream321 = fopen("C:zLinkBusSpeedProfileOut.txt", "a+");
   fprintf(stream321, "\n %3d %5d %9.6f %5.2f",linkID,busNumber,temp,actualLinkSpeed);
   fclose(stream321);  // 18/11/09

   return actualLinkSpeed;
}
//----------------------------------------------------------------------------


void Link::IncidentLinkSpeed(float sevdegree, int time)
{
```

```
 if (sevdegree==2.0)
 {
  linkSpeed = 0.4*linkSpeed;
 }

 if (sevdegree==1.0)
 {
  linkSpeed = inputSpeed;
 }

 if (sevdegree==9.9)
 {
  linkSpeed = inputSpeed;
 }

 if (sevdegree==100.0)
 {
  linkSpeed = inputSpeed;
 }

 if (sevdegree==3.0)
 {
  linkSpeed = 0.1*linkSpeed;
 }

 if  (sevdegree==10.0)
 {
  linkSpeed = inputSpeed;
 }


 else
 {
  linkSpeed = sevdegree*inputSpeed;
 }

}

//---------------------------------------------------------------------------------------------
```

```
//---------------------------------------------------------------------------
#include <iostream.h>
#include "bus.h"
#include "BusListItem.h"

class BusList
{
  public:
    BusList();
    virtual ~BusList();
    void addBus(Bus* abus);
    int getLength();
    Bus* getBus(int index);
    Bus* exitBus(int index);

  protected:
    BusListItem* head;
    BusListItem* stop;
};

#endif
```

```
//---------------------------------------------------------------------------
#pragma hdrstop

#include "BusList.h"

BusList::BusList()
{
  head = NULL;
}

BusList::~BusList()
{
  if(head != NULL) delete head;
}

void BusList::addBus(Bus* abus)
{
  if(head == NULL) head = new BusListItem(abus);
  else head->addBus(abus);
}

int BusList::getLength()
{
  if(head == NULL)
  return 0;
  else
  {
    int count = 0;
    BusListItem* current = head;
    while(current != NULL)
    {
      count++;
      current = current->next;
    }
    return count;
  }
}

Bus* BusList::getBus(int index)
{
  if(index > getLength())
  return NULL;
  else
  {
    BusListItem* current = head;
    while(current != NULL && index > 0)
    {
      current = current->next;
      index--;
    }
    return current->abus;
  }
}

Bus* BusList::exitBus(int index)
{
  if(index >= getLength()) return 0;
  else
  {
    BusListItem* current = head;

    {
      head = current->next;
      stop = current;
    }
    else
    {
```

```
      previous->next = current->next;
      stop = current;
    }
    return 0;
  }
}
#pragma package(smart_init)
```

```
///*-----------------------------------------------------------------------
//-----------------------Incident Module-----------------------------------
//----------------------------------------------------------------------

#ifndef IncidentH
#define IncidentH

#include <iostream.h>

class Incident
{
  public:
    Incident(int InciId,int InciType,int InciLocation,int InciSeverity,int InciStartTime,int
InciDuration);
    ~Incident();


    int IncidentIdNumber();
    int IncidentType();
    int IncidentLocation();
    int IncidentSeverity();
    int IncidentStartTime();
    int IncidentDuration();


   private:
    int InciIds;
    int InciTypes;
    int InciLocations;
    int InciSeveritys;
    int InciStartTimes;


};
//-----------------------------------------------------------------------
#endif



///*-----------------------------------------------------------------------
//------------------------------ ------------------------------
//-----------------------------------------------------------------------

#include "Incident.h"

FILE *stream611;


Incident::Incident(int InciId,int InciType,int InciLocation,int InciSeverity,int InciStartTime,int
InciDuration)
{

 InciIds = InciId;
 InciTypes = InciType;
 InciLocations = InciLocation;
 InciSeveritys = InciSeverity;
 InciStartTimes = InciStartTime;



stream611 = fopen("D:\\SIBUFEM\\polyvios made\\SIBUFEM model\\IncidentOut.txt", "w+");

fprintf(stream611, "\n %s %s %s", "L","B","T");

}
```

```
#ifndef BusH
#define BusH

#include <iostream.h>
#include <string.h>
#include <math.h>
class Bus
{
  public:
    Bus(int busId,int serveNo,float sp,float oldp,int capa,int firstStop,int firstSig,
    int lastStop,int lastSig,int passStart,float finalp,float busstopPosiDeviate);
    ~Bus();
    int BusIdNumber();
    int ServiceNumber();
    int nextLink();
    int FirstBusstop();
    int LastBusstop();
    int LastSignal();
    float BusstopPosiDeviation();

    void GetGpsErrorAtDoorClose(float gpsErrorDC);
    float GpsErrorAtDoorClose();

    void getSpeed(float lSpeed);
    void getSpeed2(float lSpeed2);
    float BusSpeed ();
    int arriveBusstopTime(int times,int busstopNumber2);
    void StopBusstopTime(int times,int busstopNumber2,int bHeadway,float pAlight,float
pBoard,float pWait,int reserveNos,float updateInterval);

    void GetBusIsAtBusstop(int busBusstop);
    int BusIsAtBusstop();
    int BusstopDwellTime();
    int TotalBusstopDwellTime();
    int BusHeadway();
    float PassAlighted();
    float PassBoarded();
    float PassWaited();

    void getNextBusstop(int busstopNums);
    int nextBusstop();
    int arriveSignalTime(int times,int busstopNumber2);
    int GreenTimeStart(int times,int signalNumber2);
    void FrontCarRemain(int signalNumber2,float frontCars,float delCars);
    int FrontCarRemainNumber();
    int TotalFrontCarNumber();
    void getNextSignal(int signalNums);
    int nextSignal();
    int totalSignalTime();

    void changeSpeed(int times);
    void ShiftOldPosition (float shiftedPosition);
    float oldPosition ();
    float newPosition(float timeFactor);
    float totalPassInside(float passAlight,float passBoard);
    float occupancy(float totalPassg);

    int PassJourneyTime ();
    int BusJourneyTime();
    int TotalBusJourneyTime();

    void GetBusstopDeviation(int deviTimes);
    int LastBusstopDeviation();
    void GetBusstopSchedule(int schTimes);
    int LastBusstopSchedule();

    float GpsErrorGenerateNorm(int maxError);
```

```
    float linkSpeed;

  private:
    int busID;
    int endNo;
    int serviceNos;
    float speed;
    float normalSpeed;
    //char* name;
    float oldBusPosition;
    int nextBusstopNos;
    int nextSignalNos;
    int firstBusstopNos;
    int lastBusstopNos;
    int lastSignalNos;
    float passAlready;
    float finalDist;
    float finalDistD;

    float busstopPosiDeviation;

    int busstopDelayTime;
    int busstopNumber1;
    int signalDelayTime;
    int signalNumber1;
    int signalNumber111;
    int greenTimeStart;
    int signalNumber11;
    int frontCarNumber;
    int totalFrontCarNumber;
    int totalSignalDelayTime;
    int passCapacity;
 //
    float pSpeed;
    int nextBeaconNos;

    int linkNumber1;
    int nextLinkNos;
    int busstopArriveTime;
    int signalArriveTime;
    int busstopNumber11;
    float busstopStopTime;
    int totalBusstopStopTime;
    int busHeadway;
    float passAlight;
    float passBoard;
    float passWait;
    int alreadyReserved;
    int busIsAtBusstop;

    int previousTimes;
    int passengerJourneyTime;

    int busJourneyTime;
    int totalBusJourneyTime;

    int lastBusstopDeviation;
    int lastBusstopSchedule;
    int timeDuration;
    int timeDuration1;

    float gpsErrorDoorClose;
};

#endif
```

```
#include "bus.h"

FILE *stream112;

Bus::Bus(int busId,int serveNo,float sp,float oldp,int capa,int firstStop,int firstLink,
int lastStop,int lastSig,int passStart,float finalp,float busstopPosiDeviate)
{
  busID=busId;
  serviceNos=serveNo;
  speed = sp;
  normalSpeed = sp;
  linkSpeed=0.0;
  oldBusPosition = (oldp - (speed/3.6));
  passCapacity = capa;
  nextBusstopNos =firstStop;//0;

  busstopPosiDeviation=busstopPosiDeviate;//0.0

  firstBusstopNos=firstStop;
  lastBusstopNos =lastStop;//0;
  lastSignalNos =lastSig;//0;

  passAlready = passStart;//0;
  finalDist=finalp;
  finalDistD=8640.0;

  linkNumber1 = 0;
  busstopNumber1 = 100;
  signalNumber1 = 100;
  signalNumber111=100;
  greenTimeStart=0;
  signalNumber11=100;
  frontCarNumber=0;
  totalFrontCarNumber=0;
  nextLinkNos =firstLink;//0;
  nextBeaconNos=0;
  busstopArriveTime =0;
  signalArriveTime=0;
  busstopNumber11=100;
  busstopStopTime=0.0;
  totalBusstopStopTime=0;
  busHeadway=0;
  passAlight=0.0;
  passBoard=0.0;
  passWait=0.0;
  alreadyReserved=0;

  previousTimes=0;
  passengerJourneyTime=0;
  busJourneyTime=0;

  lastBusstopDeviation=0;
  lastBusstopSchedule=0;

  timeDuration1=0;
  endNo=1;

  gpsErrorDoorClose=0.0;

  busIsAtBusstop=0;

}
//--------------------------------------------------------------------------------
Bus::~Bus() {}//{delete[] name;}
int Bus::BusIdNumber()  {return busID;}
int Bus::ServiceNumber()  {return serviceNos;}
int Bus::nextLink()  {return nextLinkNos;}
```

```
int Bus::FirstBusstop() {return firstBusstopNos;}
int Bus::LastBusstop() {return lastBusstopNos;}
int Bus::LastSignal() {return lastSignalNos;}
float Bus::BusstopPosiDeviation() {return busstopPosiDeviation;}
//---------------------------------------------------------------------------
void Bus::GetGpsErrorAtDoorClose(float gpsErrorDC)  {gpsErrorDoorClose=gpsErrorDC;}
float Bus::GpsErrorAtDoorClose() { return gpsErrorDoorClose;}
//---------------------------------------------------------------------------
void Bus::getSpeed(float lSpeed)  //Changing bus speed in different links
{
 normalSpeed=lSpeed;
 linkSpeed=lSpeed;        //only used to calculate jrTime for detector distance
 speed=lSpeed;            //makes immediate speed change
 nextLinkNos++;
}
//---------------------------------------------------------------------------
void Bus::getSpeed2(float lSpeed2) //Changing bus speed between Detector-Stopline
{
 normalSpeed=lSpeed2;
 speed=lSpeed2;           //makes immediate speed change
}
//---------------------------------------------------------------------------
void Bus::changeSpeed(int times)
{
 int delayTimes= times;
 if (delayTimes!=0) {speed = 0.0;}
 else {speed = normalSpeed;}
}
//---------------------------------------------------------------------------
float Bus::BusSpeed ()      {return speed;}
//---------------------------------------------------------------------------
void Bus::ShiftOldPosition (float shiftedPosition)      {oldBusPosition=shiftedPosition;}
//---------------------------------------------------------------------------
float Bus::oldPosition ()      {return oldBusPosition;}
//---------------------------------------------------------------------------
float Bus::newPosition (float timeFactor)
{
 if (speed<0.0) {speed=7.2;}

oldBusPosition = newBusPosition;
 return newBusPosition;
}
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
int Bus::arriveSignalTime(int times,int signalNumber2)
{
 if (signalNumber2!=signalNumber1)
 {
  signalNumber1=signalNumber2;
  signalArriveTime= times;
 }
 return signalArriveTime;
}
//---------------------------------------------------------------------------
void Bus::getNextSignal(int signalNums)     {nextSignalNos=signalNums;}
int Bus::nextSignal()                       {return nextSignalNos;}
int Bus::totalSignalTime()                  {return totalSignalDelayTime;}
//---------------------------------------------------------------------------
int Bus::GreenTimeStart(int times,int signalNumber2)
{
 if (signalNumber2!=signalNumber111)
 {
  signalNumber111=signalNumber2;
}
//---------------------------------------------------------------------------
void Bus::FrontCarRemain(int signalNumber2,float frontCars,float delCars)
{

 if (signalNumber2!=signalNumber11)
```

```
 {
 }
 else {frontCarNumber=frontCarNumber-delCars;}
 if (frontCarNumber<0)frontCarNumber=0;
}
//---------------------------------------------------------------------------
int Bus::FrontCarRemainNumber()              {return frontCarNumber;}
int Bus::TotalFrontCarNumber()             {return totalFrontCarNumber;}
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
int Bus::arriveBusstopTime(int times,int busstopNumber2)
{
 if (busstopNumber2!=busstopNumber1)
 {
  busstopNumber1=busstopNumber2;
  busstopArriveTime= times;
  if (busstopNumber2!=firstBusstopNos)
  {
  passengerJourneyTime= passAlready*(times-previousTimes);
  busJourneyTime= times-previousTimes;
  totalBusJourneyTime=totalBusJourneyTime+busJourneyTime;

  }
  previousTimes=times;
 }

 return busstopArriveTime;
}
//---------------------------------------------------------------------------------------------------
int Bus::PassJourneyTime()     {return passengerJourneyTime;}
int Bus::BusJourneyTime()      {return busJourneyTime;}
int Bus::TotalBusJourneyTime()      {return totalBusJourneyTime;}
//int Bus::AvBusJourneyTime()      {return avBusJourneyTime;}
//---------------------------------------------------------------------------
void Bus::StopBusstopTime(int times,int busstopNumber2,int bHeadway,float pAlight,float
pBoard,float pWait,int reserveNos,float updateInterval)
{
 if (busstopNumber2!=busstopNumber11)
 {
  busstopNumber11=busstopNumber2;
  busstopStopTime= times;
  totalBusstopStopTime= times;
  busHeadway=bHeadway;
  passAlight=pAlight;
  passBoard=pBoard;
  passWait=pWait;
  alreadyReserved=reserveNos;
  busIsAtBusstop=0;
 }
 else
 {
    if (busstopStopTime>0) busIsAtBusstop=1;
 }
 if(reserveNos!=alreadyReserved)
 {
  alreadyReserved=reserveNos;
  busHeadway=bHeadway;
  passBoard=pBoard;
 }
}
//---------------------------------------------------------------------------
void Bus::GetBusIsAtBusstop(int busBusstop)  {busIsAtBusstop=busBusstop;}
int Bus::BusIsAtBusstop()                    {return busIsAtBusstop;}
int Bus::BusstopDwellTime()                    {return busstopStopTime;}
int Bus::TotalBusstopDwellTime()             {return totalBusstopStopTime;}
int Bus::BusHeadway()                        {return busHeadway;}
float Bus::PassAlighted()                    {return passAlight;}
float Bus::PassBoarded()                     {return passBoard;}
float Bus::PassWaited()                      {return passWait;}
```

```
void Bus::getNextBusstop(int busstopNums)     {nextBusstopNos=busstopNums;}
int Bus::nextBusstop()                        {return nextBusstopNos;}
//------------------------------------------------------------------------------
float Bus::totalPassInside (float passAlight,float passBoard)
{
  passAlready = passAlready-passAlight+passBoard;
  return passAlready;
}
float busOccupy = 1.0*totalPassg;
 return busOccupy;
}
//------------------------------------------------------------------------------
//------------------------------------------------------------------------------
void Bus::GetBusstopDeviation(int deviTimes)  {lastBusstopDeviation=deviTimes;}
int Bus::LastBusstopDeviation()               {return lastBusstopDeviation;}
void Bus::GetBusstopSchedule(int schTimes)  {lastBusstopSchedule=schTimes;}
int Bus::LastBusstopSchedule()              {return lastBusstopSchedule;}
//------------------------------------------------------------------------------
//------------------------------------------------------------------------------
float Bus::GpsErrorGenerateNorm(int gpsCounter)
{
   stream112 = fopen("C:\\SIBUFEM\\SimulationData\\aNormalTime1.txt", "r");
   float temp;

   {
    char msg[20];
    fgets(msg,20,stream112);
    temp = atof(msg);
   }
   fclose(stream112);
   return temp;
//------------------------------------------------------------------------------
```

```
//---------------------------------------------------------------------------
#ifndef BusStopH
#define BusStopH

#include <stdlib.h>
#include <iostream.h>
#include <stdio.h>

class BusStop
{
  public:
    BusStop(int busstopId,int linkNo,int busstopP,int busstopStartTime,
    float passRa,float passRb,float avDwTime,float sdDwTime);
    ~BusStop();
    int busstopLinkNumber();
    int busstopIdentity();
    int BusstopPosition();
    float AvDwellTime();
    float SdDwellTime();
    float AlightPercentage();
    float BoardPassRate();

    int alightPass(int passInside,int busArrive1);
    int boardPassGenerate(int times);
    int boardPassDischarge(int times, int busArrive, int alightPass);
    int DwellTime();
    int boardPassNos();
    int waitingTime(int timeFactor);
    int BusstopHeadway();
    void GetReserveBus1(int resNum);
    void GetReserveBus2(int resNum);

    void CalculateAvBusJourneyTime(float busJourneyTime);
    void CalculateAvBusSpeed(float bspeed);
    float AvBusJourneyTime();
    float AvBusSpeed();

    int numberOfBuses;
    int avBusJourneyTime;
    float cumuBusJourneyTime;
    float avBusSpeed;
    float cumuBusSpeed;

    int ReservedNum1();
    int ReservedNum2();

    int headwayCalculate(int times,int busArrive2,int passInside,int schTime,int reserved,int
serviceNo,int busAtStop);
    float AlightPass2();
    float BoardPass2();
    int DwellTime2();
    float WaitTime2();
    int AddBusAtBusstop();


  private:
    int busstopID;
    int linkNos;
    int busstopPosi;
    float alightPassRate;
    float boardPassRate;
    float avDwellTime;
    float sdDwellTime;
    int timeBus1;
    int startTime;
    int numberPass1;
    //int numberPass2;
```

```
    int passGenerateTime;
    int passGenerateFactor;
    int passDischargeFactor;
    int busAlready;
    int checkinTime;
    int alightPassTime;
    int numberBoardPass;
    int totalDwellTime;
    int passengerA;
    int busAlready1;
    int passengerB;
    int waitTime;
    float deadTime;
    float passAlightingTime;
    float passBoardingTime;
    int finalWaitTime;
    int busstopReserved1;
    int busstopReserved2;

    int headway;
    int busAlready2;
    float passengerA2;
    float passengerB2;
    int dwellTime2;
    int cumuHeadwaySquare;
    int cumuHeadway;
    float averageWaitTime;
    int addBusAtBusstop;


};
//---------------------------------------------------------------------------
#endif
```

```
#include "busstop.h"

FILE *stream132, *stream336;

BusStop::BusStop(int busstopId,int linkNo,int busstopP,int busstopStartTime,
        float passRa,float passRb,float avDwTime,float sdDwTime)
{
  busstopID = busstopId;
  linkNos = linkNo;
  busstopPosi = busstopP;
  alightPassRate = passRa;
  boardPassRate = passRb;
  avDwellTime=avDwTime;
  sdDwellTime=sdDwTime;
  timeBus1= busstopStartTime;
  startTime=busstopStartTime;
  numberPass1 =0;
  passGenerateTime=99999;
  passGenerateFactor=1;
  passDischargeFactor=1;
  busAlready =0;
  checkinTime=0;
  alightPassTime=0;
  numberBoardPass=0;
  totalDwellTime=0;
  passengerA=0;
  busAlready1=0;
  passengerB=0;
  waitTime=0;
  finalWaitTime=0;
  busstopReserved1=0;
  busstopReserved2=0;
  numberOfBuses=0;

  headway=0;
  busAlready2=0;
  passengerA2=0.0;
  passengerB2=0.0;
  dwellTime2=0;
  cumuHeadwaySquare=0;
  cumuHeadway=0;
  averageWaitTime=0;
  addBusAtBusstop=0;

  cumuBusJourneyTime=0.0;
  cumuBusSpeed=0.0;

  stream336 = fopen("C:\\SIBUFEM\\SimulationData\\zDwellVaryProfileOut.txt", "w+");
}
//-----------------------------------------------------------------------------
BusStop::~BusStop() {}
int BusStop::busstopLinkNumber()  {return linkNos;}
int BusStop::busstopIdentity()    {return busstopID;}
int BusStop::BusstopPosition()    {return busstopPosi;}
float BusStop::AvDwellTime()      {return avDwellTime;}
float BusStop::SdDwellTime()      {return sdDwellTime;}
float BusStop::AlightPercentage() {return alightPassRate;}
float BusStop::BoardPassRate()    {return boardPassRate;}
//-----------------------------------------------------------------------------------
int BusStop::headwayCalculate(int times,int busArrive2,int passInside,int schTime,int
reserved,int serviceNo,int busAtStop)
{
 if(serviceNo>100) {deadTime = 6.85; passAlightingTime=1.69; passBoardingTime=9.00;}
 else              {deadTime = 3.30; passAlightingTime=1.96; passBoardingTime=9.04;}
 addBusAtBusstop=0;
  int timeBus2=times;
  if(sdDwellTime<9999.0&&sdDwellTime>0.0)addBusAtBusstop=1;
```

```
  if(reserved==0)
  {
   headway = timeBus2-timeBus1;
   timeBus1 = timeBus2;
  }
  else
  {
  headway=0;
  }

  if(headway<0) headway=0;
  passengerA2 = passInside*alightPassRate;
  passengerB2 = headway/boardPassRate;
  if(passengerA2>0.0||passengerB2>0.0)
  {
   dwellTime2 = deadTime+passengerA2*passAlightingTime+passengerB2*passBoardingTime;
   if(reserved==0)
   {
    dwellTime2 =
deadTime+passengerA2*passAlightingTime+passengerB2*passBoardingTime;
    passengerB2 = (headway+dwellTime2)/boardPassRate;
    timeBus1=timeBus2+dwellTime2;
   }
  }
  else {dwellTime2=0;}

  busAlready2=busArrive2;
  cumuHeadwaySquare+=headway*headway;
  cumuHeadway+=headway;
  if ((headway+dwellTime2)>0)
averageWaitTime=0.5*headway*headway/(headway+dwellTime2);
  else averageWaitTime=0;
  //---------------------------------------------------------------------------
  //Calculating dwell time from mean and sd using Normal Distribution
  //---------------------------------------------------------------------------
  if (avDwellTime!=9999)
  {
   dwellTime2=avDwellTime;
   stream132 = fopen("C:\\SIBUFEM\\SimulationData\\aDwellVaryProfileWidth8.txt", "r");
   float temp;
   long curpos;
   curpos = (busAtStop-1)*10;
   if(sdDwellTime==9999)dwTimeDeviation=0.0;

   fprintf(stream336, "\n %5d %7d %7d %9.6f",busstopID,times,busArrive2,temp);
  }
 }
 return headway;
}
//---------------------------------------------------------------------------
int BusStop::AddBusAtBusstop()  {return addBusAtBusstop;}
float BusStop::AlightPass2()    {return passengerA2;}
float BusStop::BoardPass2()     {return passengerB2;}
int BusStop::DwellTime2()       {return dwellTime2;}
float BusStop::WaitTime2()      {return averageWaitTime;}
void BusStop::GetReserveBus1(int resNum)  {busstopReserved1=resNum;}
void BusStop::GetReserveBus2(int resNum)  {busstopReserved2=resNum;}
int BusStop::ReservedNum1()              {return busstopReserved1;}
int BusStop::ReservedNum2()              {return busstopReserved2;}
//---------------------------------------------------------------------------
//This section is for generating individual passengers (not required at this stage)
//---------------------------------------------------------------------------
int BusStop::alightPass(int passInside, int busArrive1)
{
rate
    busAlready1=busArrive1;
    finalWaitTime=waitTime;//storing the waiting time refered to arrival of bus
  }
  return passengerA;
```

```
}
//----------------------------------------------------------------------------
int BusStop::boardPassGenerate(int times)
{
  int addPass=0;
  int genTime = times-startTime;   //startTime is busstop start time
  if (genTime>=passGenerateFactor*boardPassRate)
  {

  int numberPass2 = numberPass1+addPass;
  if(numberPass2<1) numberPass2 =0;
  waitTime=waitTime+numberPass1*1;
  numberPass1=numberPass2;
  return numberPass2;
}
//----------------------------------------------------------------------------
int BusStop::boardPassDischarge(int times, int busArrive, int alightPass)
{
  int delPass=0;
  if(busAlready!=busArrive)
  {
    checkinTime = times;
    busAlready = busArrive;
    numberBoardPass=0;
  }
    delPass=1;
    passDischargeFactor++;
  }
  numberPass1=numberPass1-delPass;//0;
  numberBoardPass=numberBoardPass+delPass;
  totalDwellTime=times-checkinTime;
  return numberPass1;
}
//----------------------------------------------------------------------------
int BusStop::boardPassNos()     {return numberBoardPass;}
int BusStop::DwellTime()        {return totalDwellTime;}

int BusStop::waitingTime(int timeFactor)
{
  if(numberBoardPass==0) numberBoardPass=1;
  int outWaitTime=(waitTime/timeFactor)/numberBoardPass;
  waitTime=0;
  return outWaitTime;
}
//----------------------------------------------------------------------------

void BusStop::CalculateAvBusJourneyTime(float busJourneyTime)
{
 numberOfBuses++;
 cumuBusJourneyTime+=busJourneyTime;
 avBusJourneyTime=cumuBusJourneyTime/numberOfBuses;
}

float BusStop::AvBusJourneyTime()  {return avBusJourneyTime;}

//----------------------------------------------------------------------------

void BusStop::CalculateAvBusSpeed(float bspeed)
{
 //numberOfBuses++;
;
}

float BusStop::AvBusSpeed()  {return avBusSpeed;}

//----------------------------------------------------------------------------
```