Implementing a Hybrid Spatial Discretisation within an Agent Based Evacuation Model

Nitish Chooramun

A thesis submitted in partial fulfilment of the requirements of the University of Greenwich for the degree of Doctor of Philosophy

May 2011

DECLARATION

I certify that this work has not been accepted in substance for any degree, and is not concurrently being submitted for any degree other than that of Doctor of Philosophy being studied at the University of Greenwich. I also declare that this work is the result of my own investigations except where otherwise identified by references and that I have not plagiarised the work of others

____

Nitish Chooramun

____

Dr Peter Lawrence (first supervisor)

____

Prof. Edwin Galea (second supervisor)

ACKNOWLEDGEMENTS

I owe my deepest gratitude to my supervisors, Dr. P.J.Lawrence and Prof. E.R.Galea for all their support, suggestions and encouragement throughout the progression of this research and the writing of this thesis. Dr Lawrence's creative brain storming and code debugging sessions were very useful and motivating. Prof. Galea's constructive criticism has contributed significantly towards improving my research work and thesis. I would also like to thank Dr. S. Gwynne for his help and advice during the writing of the thesis.

I would like to give my special thanks to my father, mother and sister for all their love, help and constant encouragement which enabled me in completing this work.

I am deeply indebted to the University of Greenwich for providing the funding for this research.

Furthermore, I am really thankful to all my friends and colleagues who have supported me throughout my time at the university.

ABSTRACT

Within all evacuation and pedestrian dynamics models, the physical space in which the agents move and interact is represented in some way. Models typically use one of three basic approaches to represent space namely a continuous representation of space, a fine network of nodes or a coarse network of nodes. Each approach has its benefits and limitations; the continuous approach allows for an accurate representation of the building space and the movement and interaction of individual agents but suffers from relative poor computational performance; the coarse nodal approach allows for very rapid computation but suffers from an inability to accurately represent the physical interaction of individual agents with each other and with the structure. The fine nodal approach represents a compromise between the two extremes providing an ability to represent the interaction of agents while providing good computational performance.

This dissertation is an attempt to develop a technology which encompasses the benefits of the three spatial representation methods and maximises computational efficiency while providing an optimal environment to represent the movement and interaction of agents. This was achieved through a number of phases. The initial part of the research focused on the investigation of the spatial representation technique employed in current evacuation models and their respective capabilities. This was followed by a comprehensive review of the current state of knowledge regarding circulation and egress data. The outcome of the analytical phases provided a foundation for eliciting the failings in current evacuation models and identifying approaches which would be conducive towards the sophistication of the current state of evacuation modelling. These concepts led to the generation of a blueprint comprising of algorithmic procedures, which were used as input in the implementation phase.

The buildingEXODUS evacuation model was used as a computational shell for the deployment of the new procedures. This shell features a sophisticated plug-in architecture which provided the appropriate platform for the incremental implementation, validation and integration of the newly developed models. The Continuous Model developed during the implementation phase comprises of advanced algorithms which provide a more detailed and thorough representation of human behaviour and movement. Moreover, this research has resulted in the development of a novel approach, called Hybrid Spatial Discretisation (HSD), which provides the flexibility of using a combination of fine node networks, coarse node networks and continuous regions for spatial representations in evacuation models. Furthermore, the validation phase has demonstrated the suitability and scalability of the HSD approach towards modelling the evacuation of large geometries while maximising computational efficiency.

CONTENTS

FIGURES

TABLES

EQUATIONS

ALGORITHMS

Chapter 1 – Introduction

1.1 Motivation

Public areas and large enclosures for instance airports, stadiums, concerts, shopping malls, train stations comprise of increasing numbers of people. Therefore, in order to ascertain the safety of such enclosures, it is of paramount importance to demonstrate that these highly populated enclosures will be able to be emptied quickly and efficiently in the event of an emergency. Real incidents which have occurred in the past provide an insight on what can actually happen during egress situations. Examples of such incidents include; Manchester Woolworth store fire in 1979 (UK, 10 fatalities), King's Cross tube station fire in 1987 (UK, 31 fatalities), Gothenburg disco hall fire in 1998 (Sweden, 63 fatalities), World Trade Centre disaster in 2001 (USA, 2033 fatalities), Rhode Island station nightclub fire in 2003 (USA, 96 fatalities) [Galea 2004b]. Conventionally, two methods have been used to demonstrate the evacuation performance of building designs namely prescriptive building codes and evacuation trials.

Building codes consist of a rigid set of regulations to assess building safety. However, these codes, which predominantly consider configurational aspects of the buildings, are insufficient to cater to novel building designs. Evacuation trials comprise of the creation of full scale evacuation experiments based on real population within enclosures. However, this approach can become prohibitively expensive especially if the representative population and enclosures are large and if repeated runs for the trials are to be performed. In addition, evacuation demonstrations are typically conducted after the enclosures have been built and therefore, any important design changes deemed necessary for the enclosures might be too costly to implement. Moreover, real evacuation trials can pose serious ethical issues as they could expose the participants to risks of injury. In extreme situations, there have been unfortunate instances which have resulted in loss of lives of the participants. For instance, as stated in the Marine Accident Investigation Branch report [MAIB 2003], there was a fatality during a vertical chute evacuation drill from a ferry. The shortfalls of building codes and evacuation demonstrations have lead to a paradigm shift towards the use of computational egress tools for assessing building design safety.

Within all evacuation and pedestrian dynamics models, the physical space in which the agents move and interact is represented in some way [Gwynne 1999][Kuligowski 2005]. Current models typically use one of three basic approaches to represent space [Gwynne 1999], a continuous representation of space e.g. SIMULEX [Thompson 1995], a fine network of nodes e.g. buildingEXODUS V4.07 [Galea 1994] or a coarse network of nodes e.g. PEDROUTE [Buckmann 1994]. Each approach has its benefits and limitations, the continuous approach allows for an accurate representation of the building space and the movement and interaction of individual agents but suffers from relative poor computational performance, the coarse nodal approach allows for very rapid computation but suffers from an inability to accurately represent the interaction of individual agents with each other and with the structure. The fine nodal approach represents a compromise between the two extremes providing an ability to represent the interaction of agents while providing good computational performance.

Currently, the ability of existing evacuation tools to handle large and complex environments (e.g. large multi-storey buildings, underground environments, large sports arenas, urban environments) at maximal computational efficiency poses a considerable challenge. Moreover, this challenge escalates when the complexities of crowd phenomena in such large environments is required to be modelled on higher scales of granularity for instance when using the continuous spatial representation. In an ideal world, it would probably be best to use the continuous spatial representation everywhere to represent low level conditions in as much detail as possible, but due to very high computational costs involved such an option is often not viable. In addition, the ability to effectively handle very high population densities in evacuation simulations is another area of concern. Egress tools, especially those utilising the continuous space, are often prone to deadlock where agents inappropriately jam or become trapped in heavily congested regions and narrow passages [Thompson 1994][Pathfinder 2009].

Furthermore, since current egress tools utilise only one of the three approaches to represent the underlying physical space, the modeller is limited only to the advantages offered by the particular technique in use. Also, because of the inherent differences in the characteristics of each of these spatial representation techniques, users of evacuation tools are often faced with the dilemma of choosing the most suitable one for their modelling requirements.

Therefore, the key research questions and their corresponding ancillary questions which are addressed as a result of the above discussions are as follows:

1. **Can we derive a new approach that has the benefits and accuracy of the continuous model but without the computational overheads?**

    1.1. How can we represent agent navigation in continuous space?
    1.2. How can we represent agent perception and behaviours in continuous space?
    1.3. Can we derive a framework to allow dynamic integration of behaviours into the system?
    1.4. Can high densities of agents in continuous spaces be simulated without occurrence of deadlocks?
    1.5. What optimisation strategies can we use for improving the numerical performance of continuous models?
    1.6. How can we represent agent navigation in a coarse network model?
    1.7. How can we harness the benefits of the continuous and nodal approaches?
    1.8. What benchmark tests are available for validating evacuation models designed for the building environment?

2. **Can a hybrid representation which utilises the 3 fundamental spatial representation methods be used in such a way as to optimise accuracy and performance?**

    2.1. How can the transition of agents be represented in a hybrid model comprising of coarse networks, fine node networks and continuous regions?
    2.2. Can agents maintain their physical and experiential attributes when traversing the different spatial representation types?
    2.3. What type of software architecture is appropriate for a hybrid model?
    2.4. How do we map the different spatial representation types to different areas of an enclosure in a hybrid model?

3. **How suitable are each of the spatial representation approaches for modelling different terrain types and scenarios?**

4. **What are the benefits of a hybrid approach in terms of size, accuracy and performance of the problem that can be tackled?**

   4.1. What are the advantages of the hybrid approach over the spatial representation approaches

   4.2. How would such an approach be utilised to address for:

      a) Very large complex scale problems such as urban environment

      b) Moderately large complex structure such as tunnel station

      c) Complex buildings with multiple compartments.

1.2 Research Goal and Methodology

This research is an attempt at developing a technology which will allow enclosures to be modelled using a combination of all current techniques of representing space, within a single integrated model, while maximising computational efficiency. This dissertation comprises of the following objectives:

1. To generate an integrated computational model that is able to simulate people behaviour and movement using various techniques of modelling physical space, namely fine node, coarse node and continuous regions

2. To develop a computational model utilising continuous space that can simulate some key aspects of human behaviour and movement

3. To investigate the transitional movement of people in between the three spatial types and identify where their use is most applicable

These objectives are then segmented into a series of tasks in order to aid in the management and progression of the research. These activities are described as follows:

- Review different methods of modelling people movement and become familiar with recent data concerning people movement in crowds. The aim of this review is to gain an understanding of the primary aspects of crowd flow and the related crowd dynamics.

- To review the literature concerning the current state of knowledge of computer people evacuation and circulation model. The purpose of this review is to gain an understanding of the capabilities of the tools currently available for simulating egress. This would allow the identification of potential areas which might need further enhancements.

- Develop a familiarisation with evacuation modelling in general and a working knowledge of the buildingEXODUS [Galea 2000] evacuation model. Some of the research outcomes entail the development of behavioural and movement algorithms which require a computational shell in order to operate. The buildingEXODUS evacuation model was found to be appropriate for technical reasons and also for its availability. buildingEXODUS features a sophisticated plug-in architecture which would be conducive towards the implementation, testing and integration of new functionalities into the core model. Moreover, buildingEXODUS includes 2-D and 3-D visualisation tools which can be very beneficial towards the qualitative validation of the newly added features. For instance, such tools can aid in the tracking of the paths and attributes of the occupants as they navigate from their starting locations in the geometry to their desired exit locations. Hence these tools can allow the investigation of the level of realism demonstrated by the underlying behavioural mechanisms.

- Develop a computational framework for simulating the behaviour and movement of people in continuous space. For this objective, a multi-agent paradigm is adopted, whereby each agent is modelled as an autonomous character which exhibits some forms of adaptive behaviour. In other words, the agent has the ability to sense the constantly evolving conditions in its environment and react to them accordingly in a life-like manner. An interesting feature of this approach is that the underlying microscopic properties can contribute to the emergence of macro crowd phenomena.

- Develop a methodology on how people move between coarse, fine node and continuous regions. This is indispensable to allow the three spatial representations to inter-operate within a single integrated software tool. Each of the spatial representations techniques are situated on different levels on the scale of granularity and therefore effective mechanisms are required to enable the transition of the agents across the different spatial types.

- Identify where the various spatial representations are appropriate for use. This objective is crucial as it will provide an evaluation on the respective capabilities of the various approaches and how these can be fully exploited in the integrated model.

- Apply newly developed features to example test cases. Since these models aim at simulating realistic behavioural and movement characteristics of people, therefore model validation and calibration are indispensable phases within the development lifecycle. The test cases will cater for the various aspects of validation, namely Component Testing, Functional Validation, Qualitative Validation and Quantitative Validation.

1.3 Contributions

The primary contributions of this thesis towards the field of human behavioural and movement modelling are summarised as follows:

- The development of a novel approach called the Hybrid Spatial Discretisation (HSD) featuring all current techniques of spatial representation (coarse network, fine network and continuous regions). This model incorporates advanced behavioural and movement algorithms which facilitate the movement of people between the various spatial types.

- The investigation and guidance on the suitability of different spatial representation types namely fine network, coarse network and continuous for modelling different types of scenarios and terrains. A list of recommendations relating to this has been outlined in the thesis.

- The extension and calibration of a computer model using a coarse spatial representation. This model uses a flow to density equation in conjunction with the novel concept of a navigational graph.

- The development and calibration of an agent based computer model using a continuous spatial representation. This model uses original and advanced behavioural algorithms (listed below) in conjunction with the novel concept of a navigational graph. The agents have been developed as autonomous characters demonstrating adaptive behaviour. The following key advanced features were developed.
    - An advanced navigational system.
    - A method for linking continuous regions to other spatial representations.
    - Crowd avoidance behaviours allowing occupants to maintain a desired separation from each other and enabling them to avoid collisions with occupants moving in unaligned directions.
    - Behavioural anticipation – the ability to look ahead by a constant time. The distance projected ahead is proportional to travel speed.
    - A behavioural framework using a component oriented approach which facilitates the implementation and integration of new behaviours
    - Optimising the searching of neighbours using a Grid Method. This makes use of an underlying grid of cells (of any size) constructed at pre-simulation time. The querying and maintenance of the cells (allocation and de-allocation of agents to cells) is performed during simulation time. The discovery of neighbouring agents is considerably faster as the search space is reduced to only the current cell (to which the agent is registered) and the immediate neighbouring cells, thereby eliminating the need to search the entire geometry.

1.4 Dissertation Overview

This section represents a road map of the project and describes the organisation of the material into several chapters. The objectives of all the chapters are summarised.

Chapter 2 documents a review on evacuation models and spatial representations. It describes the different ways of representing space in evacuation models and evaluates on their respective capabilities. It also illustrates the applicability of the spatial representations to different types of scenarios. Furthermore, this chapter includes a review of some of the key factors influencing human behaviour coupled with an analysis of the primary components of crowd flow.

Chapter 3 describes the proposed architecture of the hybrid model and evaluates the core technologies which will be required for the implementation of the model. It also discusses possible strategies which could be employed to facilitate the interoperability of the various key components within the hybrid model.

Chapter 4 describes the EXODUS model which is the core of the hybrid model. Moreover, it includes a description of the interacting sub-models within EXODUS and places emphasis of on the sub-models which form an integral part of the hybrid architecture. In addition, this chapter documents the evaluation of the buildingEXODUS fine nodal model using a series of experiments. The results of those experiments are used as input in the calibration of the continuous model.

Chapter 5 describes the design and implementation of the movement and behaviour mechanisms of people moving in continuous regions. The structure of the framework, algorithmic procedures and computational methods as regards human behaviour and movement modelling is described and critically evaluated. It also includes an evaluation of the navigation system used in the continuous model followed by an overview of the macro and micro phenomena which take place within the program. In addition, this chapter describes the optimisation techniques for the algorithmic procedures employed in the implementation of various aspects of the continuous model. Besides, a series of validation test cases are incorporated in this chapter.

Chapter 6 describes the implementation of the coarse network approach for representation of physical space. It includes a description of the navigation system employed in the model.

Chapter 7 describes the techniques used in developing a model encompassing all essential characteristics of fine network, coarse network and continuous regions using a novel approach called Hybrid Spatial Discretisation (HSD). Furthermore, it discusses the challenges faced when interfacing models found on different scales of granularity. Moreover, the algorithms which allow the transition of the agents across the various transitions regions Coarse Node ↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous are described and evaluated. Furthermore, this chapter documents a series of test cases used for verifying various aspects of the hybrid model.

Chapter 8 describes the validation of the HSD model and demonstrates the efficacy of the methodologies used for integrating all the spatial representation techniques. In addition, it describes the application of the HSD model to different test cases to demonstrate its capabilities namely:

- Ability to tackle extremely large problems on an urban scale (City block case)
- Ability to tackle moderately large problems were size of domain is a challenge (Large tunnel station complex)
- Ability to tackle problems where accuracy and speed of performance is an issue (Complex building)

The impact of the different discretisation schemes on accuracy of results and numerical performance of the simulations is also evaluated.

Chapter 9 summarises the important aspects of the dissertation and describes the evaluation of the different aspects of the research namely its contributions in the evacuation modelling field and also the evaluation of the methodologies undertaken during the various phases of the project.

Chapter 10 includes a discussion on areas of potential future research which could be conducive towards the further advancement of the newly developed technologies.

The goal of this dissertation is to develop a hybrid approach named HSD, to represent the discretisation of space in circulation and evacuation simulation models. The aim of the HSD approach is to encompass the benefits of the three spatial representation methods and maximise computational efficiency while providing an optimal environment to represent the movement and interaction of agents. An ideal application of the HSD approach would utilise the fine nodal approach to map the majority of the geometry, providing good computational speed and an ability to model agent interaction. In parts of the geometry where greater precision is required to model detailed interaction between agents, the continuous approach is used and in regions where knowledge of detailed agent interaction is not essential the coarse nodal approach can be used, providing improvements in speed and computational efficiency. In such an application, the HSD would benefit from the speed of the fine and coarse approaches and the precision of the continuous approach.

As such there are two primary applications for the HSD, the first are large complex structures such as airport terminals and high-rise buildings or systems of structures such as long tunnels with complex interchanges. In these types of applications, the majority of the structure would primarily consist of a combination of Fine and Coarse nodes. The Fine node structure would be used throughout the bulk of the structure where detailed analysis was required while the Coarse nodes would be used in the far field to represent parts of the structure which are not central to the analysis. The Continuous approach would only be utilised in special areas to represent regions such as pinch points or exits. The second type of application involves urban environments such as a town or city. In applications involving urban scale geometries, the bulk of the geometry would be represented using the Coarse node approach with key areas such as assembly points or interchanges represented using the Fine node approach.

Chapter 2 – Review of Computer Models and Spatial Representation

The aim of this chapter is to review the background information assimilated during the conception process of the research and identify any blank spots from the perspective of the developing a hybrid egress tool. This chapter is divided into three key sections. The first section evaluates the different approaches for representing space in existing evacuation models. There is an array of evacuation models to date namely; EXODUS [Galea 1994, Galea 2004a, Galea 2010], PATHFINDER [Pathfinder 2009], GRIDFLOW [Bre 2008, Purser 2009], FDS-EVAC [Hostikka 2007], MASSEGRESS [Pan 2006a], LEGION [Berrou 2005], PEDGO [Klupfel 2003], ASERI [Schneider 2001], EVACNET4 [Kisko 1998], EVI [Evi 2001], SIMULEX [Thompson 1995, Thompson 1996], PEDROUTE [Buckmann 1994], WAYOUT [Shestopal 1994], EGRESS [Ketchell 1994], CRISP [Fraser-Mitchell 1994], EXITT [Levin 1989] and BFIRES [Stahl 1982]. However, only the evacuation models which have had an influence on this work will be discussed. Furthermore, the first section of the review focuses on the relative strengths and weaknesses of the various spatial types and elicits a list of recommendations on the suitability of using the various spatial types for modelling different types of terrains and scenarios. Currently, there are some evacuation models which are described as being hybrid based on their capabilities of using a combination of techniques for egress modelling. In this respect, a discussion on the different interpretations of the word 'hybrid', when applied to evacuation modelling tools, is provided.

The second section of this review presents some key methodologies which have been used for implementing motion behaviours in agent based models.

The third section of this chapter provides a review of the previous research into crowd movement in terms of the quantitative and qualitative aspects of crowd flow. Figure 2-1 depicts how the various elements of this review relate to the development of the hybrid evacuation model.

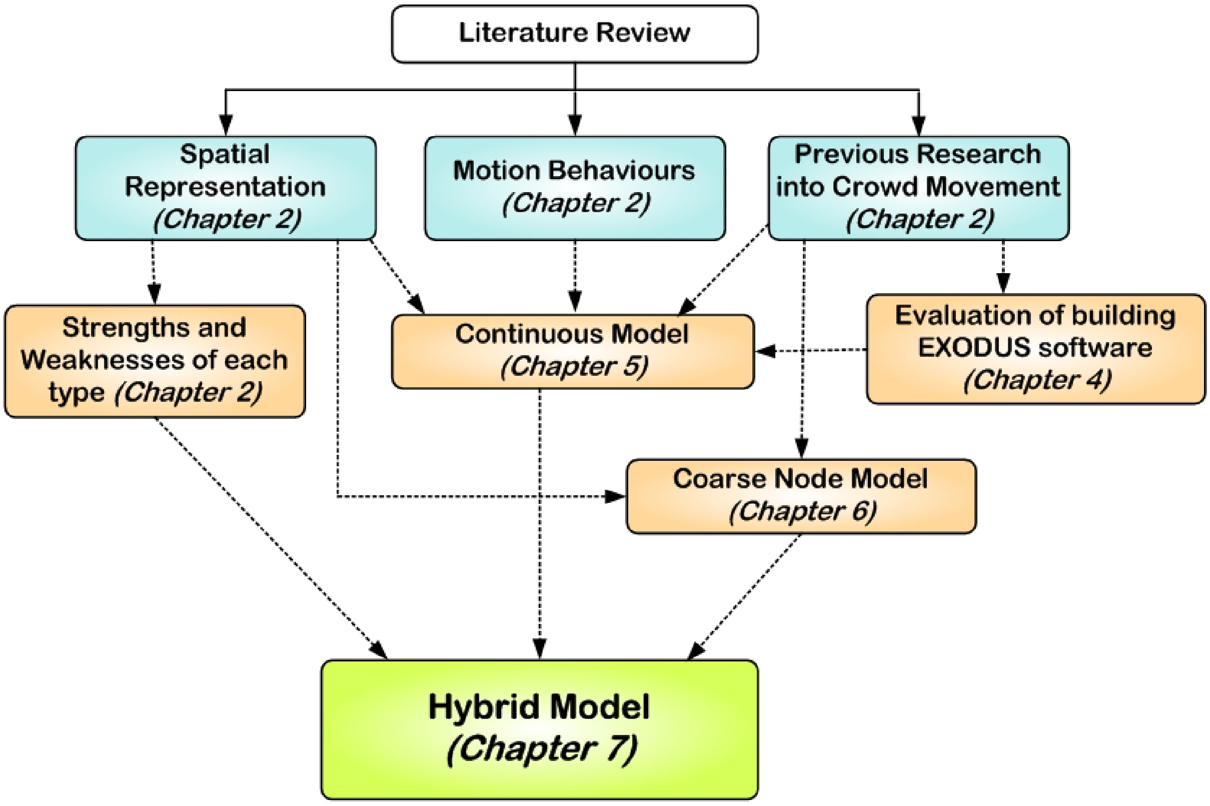


Figure 2-1 Components of the Literature Review and their interactions with other sections of the dissertation

2.1 Current techniques used in evacuation modelling

The level of safety of constructed structures is specified by rules which are known as building codes. In the past, many building codes and standards were prescriptive and relied solely on the configurational aspects of the structures for instance number of exits, exit widths, travel distance. The adherence to traditional codes meant that structures were only 'deemed to be safe'. However, the fixed criteria of such codes were too restrictive and insufficient to establish the level of safety provided by buildings. Moreover, the prescriptive approach made it very difficult to cost-optimise building construction [Foliente 2000]. This led to the development and use of codes and standards which are more performance based. The performance based approaches fall into two categories namely; hand calculations and the use of evacuation models. Hand calculations are used to determine the mass flow evacuation from buildings and are primarily based on the equations from the Society of Fire Protection Engineers (SFPE) Handbook [Nelson 2002]. In order to achieve more realistic evacuation calculations, designers, architects and engineers have shown increasing interest in computer models.

Quantifying and modelling human behaviour and movement is an area of research which has been in progress for at least 30 years. This research progressed into two routes as follows:
(1) Movement of people under normal non-emergency conditions.
(2) Predictions of people movement under emergency conditions.

Some of the earliest work in the quantification of human movement in non-emergency conditions were performed by Predtechenskii and Milinskii in 1969 [Milinskii 1969] and Fruin in 1971 [Fruin 1971]. The outcomes of this research into the movement capabilities of people lead to the development of movement models for example PEDROUTE. Research into evacuation is more recent whereby one of the first papers describing the modelling of emergency egress during fires appeared in 1982 [Stahl 1982]. Over the years, attempts of simulating evacuation progressed into the following categories of models namely; Movement, Partial Behavioural and Behavioural [Kuligowski 2005]. Another category which represents a more advanced version of behavioural models is Adaptive [Galea 2004b].

Movement models are also commonly known as ball bearing models whereby the occupants are treated as unthinking objects or particles which respond to external stimuli. These models are derived based on the analogy between fluid and interacting particles and crowd motion. Some variations of these models treat the occupants as a mass of people. One example is EVACNET 4 [Kisko 1998] which is used to describe the optimal evacuation from a building. In other words, the model minimises the time to evacuate the building such that the occupants are moved in the most optimal way throughout the space. Recent studies have shown that the validity of some of the methods used by such models cannot be justified. Keith Still [Still 2000] mentions in his thesis that people do not follow the laws of physics and that they can alter their direction and speed at will and that an accurate representation of people cannot be reduced to some equations representing the motion of ball bearings through fluids.

Partial Behavioural models are primarily movement models but which show some basic representation of human behaviour. For example, Simulex [Thompson 1994] uses the inter-person distance in order to specify the walking speeds of occupants. It also includes features such as body rotation, overtaking behaviour and side stepping. The movement of occupants throughout the space is based on observed human data.

Behavioural models incorporate both movement and behaviour of occupants whereby individuals are treated as active agents which respond to stimuli. The occupants have individual characteristics and can perform specific actions based on decision making processes (also known as decision trees). For example, buildingEXODUS [Galea 2000] uses a rule based behavioural approach and the simulation is controlled by rules and heuristics.

Adaptive models represent a more sophisticated version of behavioural models. The occupants are modelled as agents that have the ability to sense and detect changes within their local environment. The agents use the sensory information available to them in their decision making processes and react accordingly. Consequently, such models require fewer rules to be explicitly defined for behavioural representation. An interesting feature of adaptive models is that they are capable of providing a more realistic representation of human behaviours [Reynolds 1999, Pan 2006]. However, due to their nature, such models require far more computational resources as compared to the other model categories.

The different categories of models can be represented by a variety of modelling approaches which in turn comprise of different techniques for the representation of the population, behaviour and enclosure. The wide assortment of available approaches resulted in the development of several egress models which can be categorised depending on several factors namely [Gwynne 2000]:

- Nature of model application
- The perspective of the population
- The behavioural perspective
- The methodology used for representing the enclosure

2.1.1 Nature of Model Application

There are 3 fundamental approaches which evacuation models utilise in order to simulate egress. These are namely Optimisation, Simulation, and Risk Assessment. The underlying principles associated with each of the approaches determine the capabilities of the models.

Optimisation models rely on the assumption that all occupants evacuate in the most efficient manner during egress. Such models do not consider activities which could have an impact on the overall evacuation performance for instance delayed response of the occupants to the evacuation call. Moreover, the paths adopted by the occupants from their initial starting locations to the external exits are assumed to be optimal. Furthermore, optimisation models are designed to cater for large population sizes whereby the occupants are treated as a homogeneous ensemble rather than from an individualistic perspective. This limits the ability of such models to represent the individual attributes and behaviour of the occupants. Examples of optimisation models include EVACNET4 [Kisko 1998] and Takahashi's Model [Takahashi 1989].

Simulation models possess the capability of representing both the movement and behaviour of individuals during egress. Therefore, such models tend to provide more accurate results while demonstrating more realistic occupant behaviours and evacuation paths. Examples of simulation models include EXODUS, LEGION, ASERI, SIMULEX, EGRESS, PEDROUTE, MASSEGRESS, PATHFINDER and FDS-EVAC.

Risk Assessment models attempt to quantify risk by identifying any potential hazards which can hinder the performance of the evacuation. This is achieved by running repeated runs of the model in order to assess the statistical variations as a result of changes applied to enclosure designs. Examples of risk assessment models include WAYOUT and CRISP

2.1.2 Population Perspective

The nature of models is also dependent on the population perspective adopted. There are two ways in which the model can view the occupant; globally and individually. When the model views the occupants from an individual perspective, it tracks the movement of the individuals and provides detailed information about the exact location and other attributes of each individual such as age, gender and movement speeds. Examples of such models include EXODUS, SIMULEX, LEGION, FDS-EVAC, PATHFINDER, MASSEGRESS and ASERI. On the other hand, when the model views the occupants globally, the occupants are treated as a homogeneous group of people moving towards specified locations and their individual attributes are not modelled. This category of models includes PEDROUTE, WAYOUT and EVACNET4.

Although, the simulation from an individual perspective is more detailed, the choice of the individual or global perspective will depend on the purpose of the simulation [Kuligowski 2005]. Models adopting a global population perspective provide simulation results in terms of the initial numbers of occupants in the various compartments of the enclosure and the total numbers of numbers of occupants who have successfully evacuated, without indication of which of the occupants have specifically managed to evacuate. This approach can significantly improve the computational speed of the models but at the expense of the details at an individual level. For instance, models using a global population perspective are unable to represent the effects of toxic gases on individual occupants. Although such models can provide the facility of simulating the hazardous gaseous effects on a proportion of the entire population, the level of detail of the corresponding simulation results might not be sufficient to illustrate an accurate and realistic representation of the population behaviour.

2.1.3 Behavioural Perspective

The nature of evacuation tools is also dependent on the mechanisms used to simulate the decision making processes of the occupants during the evacuation process. The behavioural perspective is influenced by the choice of strategies for the representation of the enclosure and population and is distinguished as one of the most complex aspects. The various subcategories of behaviour perspective approach include *No Behaviour, Implicit Behaviour, Rule Based Behaviour, Artificial Intelligence* and *Functional Analogy* [Gwynne 2000].

No Behaviour indicates that the model do not include any behavioural rules such that only the physical movement aspects of the population are considered, for example WAYOUT and EVACNET4. In this respect, the occupants are unable to demonstrate complex decision making strategies.

Implicit Behaviour denotes that the model attempts to simulate behaviour by assigning certain population attributes and response delays which affect the movement of the occupants. Such models are highly dependent on the reliability of the data used. Examples of such models include PEDROUTE, SIMULEX and GRIDFLOW.

Rule Based Behaviour signifies that the model represents the behaviour of the occupants using a rule based approach (EXODUS, ASERI, BFIRES, CRISP and EGRESS). In other words, the model utilises a set of rules which dictate the behaviour of the simulated individuals. These rules are triggered by events occurring in specific scenarios. However, the limitation of this behavioural representation is that the occupants exhibit the same behaviour when faced with the same circumstances in a deterministic fashion. In this respect, repeated runs of simulations based on the same initial conditions do not provide a variation in the outcomes. This problem is often counteracted by the use of stochastic elements within the simulation.

Artificial Intelligence refers to models which attempt to represent human intelligence in the evacuation simulations, whereby the techniques are derived from the field of artificial intelligence. Examples of such techniques includes the use of the search algorithms such as the A* algorithm [Finlay 1996] for conducing spatial searches. Examples of such models include LEGION, MASSEGRESS and PATHFINDER.

Functional Analogy represents models which apply sets of equations which are derived from disciplines that are not necessarily related to the field of evacuation modelling. For example, the Magnet Model [Okasaki 1993] utilises principles which are based on Physics. These equations drive the behavioural response of the entire population and therefore, all the occupants are affected in the same way in a deterministic fashion. Such an approach limits the level of realism that can be applied to the population behaviour.

2.1.4 Spatial representation in Evacuation Models

All evacuation and circulation models need some method to represent the geometry of the enclosure and the representation of the internal space through which the occupants would be moving, irrespective of the perspective of the occupant or model. The two methods usually used for spatial representation are discrete and continuous spaces. The discretised spatial method involves the segmentation of the available space into sub regions commonly known as nodes, and each node can be connected to its neighbours via arcs or links. This method allows the representation of paths or routes which can be taken by occupants to move from one location to another and can be applied to solve complex geometrical problems. The discrete spatial method can be further divided into two categories namely Coarse [Kisko 1998] and Fine network [Galea 2000] and the size of the subdivision distinguishes between these two approaches. Whereas in the Continuous spatial method [Thompson 1994], space is represented as a polygon of any geometrical shape and there are no nodes as such. The various spatial representation approaches are shown in Figure 2-2. The models which have been reviewed in this section were chosen on the basis of the availability of documentation and also because their features were representative of the model development requirements for this dissertation.

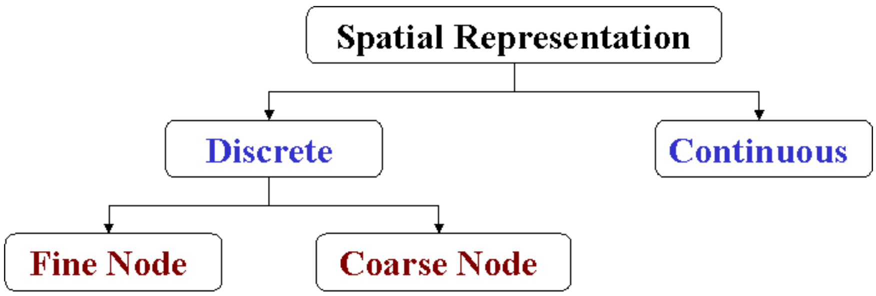


Figure 2-2 Types of Spatial Representation

2.1.4.1 Coarse Network model

In coarse network models (PEDROUTE, WAYOUT, EVACNET4, EXITT) , the geometry is defined in terms of segments or partitions which can represent different sections of the geometry for instance, corridors, staircases, escalators etc. Each partition is a node which is connected via arcs or links to represent doorways or other forms of connectivity in the structure. For example, a hypothetical building plan in **Figure 2-3** consisting of 6 rooms and 1 corridor could be modelled as a network of nodes and arcs as is shown in **Figure 2-4**.

Each node usually has a limit on the number of occupants it can contain (maximum capacity) while the arcs have a maximum flow capacity, i.e., the maximum number of people that can traverse the arc per time period. The occupants in coarse networks transit from room to room such that the physical movement from one area inside a room to another is not represented. The travel speeds of the occupants are dictated by mathematical equations. For example, PEDROUTE is a coarse network model which allows the simulation of the passage of travellers through public transport stations whereby the station plans are segmented into blocks to represent corridors, stairs, platforms, escalators and other station areas. Each block has a specific curve based on a speed-flow relationship [Cunningham 1993] to describe occupant movement. These curves are derived from data obtained from past surveys and observations at stations.

The nature of the coarse network approach restricted the applications of early coarse node models to geometries with simple geometrical shapes e.g. squares and rectangles and could not be used to model the complexities of all buildings [Thompson 1994]. However, there are some more recent implementations of coarse networks which allow the generation of complex geometries. For example, the coarse node implementation in buildingEXODUS allows the creation of non-convex regions, which is described in Chapter 6.

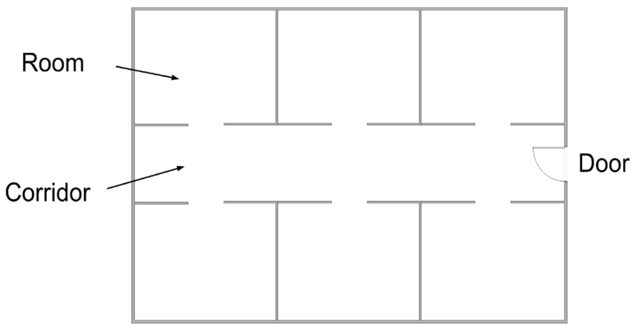


Figure 2-3 Hypothetical building plan

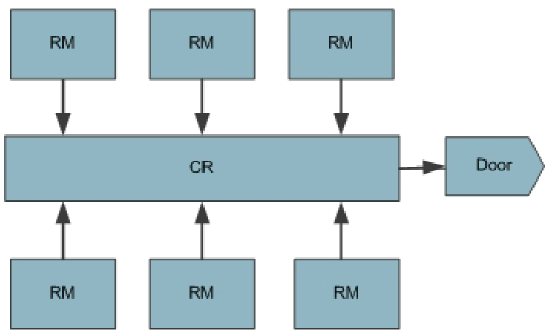


Figure 2-4 Representing the sample building plan in Figure 2-3 using a coarse network approach

Key:

RM : Node representing room

CR : Node representing corridor

→ : Arc

2.1.4.1.1 PEDROUTE/PAXPORT

PEDROUTE is a simulation model that has been developed by Halcrow Fox Associates in the UK. The purpose of PEDROUTE is to simulate the passenger movements, congestions and capacities in public transport stations and has been used for the modelling of approximately 100 underground stations in London [Kuligowski 2005]. A derivative of this model is called PAXPORT which can be used to model airport and railway terminals.

The physical space of the enclosures is represented using a coarse network approach whereby different sections of the stations can be represented by different types of blocks namely; Passageways, Moving walkways, Junctions, Concourses, Stairs, Escalators, Platforms, Lifts and UTS Gates. Blocks that are physically adjacent can be interconnected by links which determine whether the flow of occupants through the block is one-way or two-way. The flow of occupants is dictated by speed/flow curves which are associated with the different types of blocks. The equations are given by:

$$t=t_0+A*(V/C)^n$$ **Equation 2-1**

Where:

t : Travel time (seconds) at flow level V
t_0 : Free-flow travel speed (seconds)
V : Occupants demand (pass/m/hr) or density (pass/m^2)
C : Max Capacity (pass/m/hr or pass/m^2)
A and n : Constants estimated in the model fitting process

The results from **Equation 2-1** are then transformed into speed/flow relationships using the following equation:

$$s = \frac{l}{t}$$ **Equation 2-2**

Where s: Estimated speed (m/s)
l: Length of facility (m)

Moreover, PAXPORT views the occupants from a global perspective whereby the occupants are modelled in groups. The model includes 16 different types of groups which are defined based on the nature of the required service for instance domestic or long haul flights. Also, the model includes the capability of predicting the 'Level of Service' [Fruin 1971] for the different blocks used to represent the enclosure. However, the model does not track the locations and the trajectories of the individual passengers.

An interesting feature of this model is that the speed/flow curves have been derived from data based on past surveys and observations at underground stations and therefore can be considered appropriate for modelling passenger movement. In this respect, the underlying principles of the speed/flow curves have been employed in the development of the coarse spatial representation (described in Chapter 6) which is one of the objectives of this dissertation.

2.1.4.1.2 EXITT

EXITT is a simulation model that has been developed by B.M. Levin from the National Bureau of Standards, USA. This model has been designed to simulate residential occupancies and the decision making processes of occupants in response to fire hazards.

EXITT uses a coarse network approach to represent the physical space bounded by the enclosures. The buildings are represented by a combination of nodes and arcs whereby the nodes represent rooms or secondary locations within the rooms while the arcs indicate the distances in between the nodes. The occupants within the enclosure navigate to their destinations using the shortest paths. However, the underlying algorithms can also assign penalties to the paths depending on the smoke concentrations causing the occupants to use the paths with the least penalties. The equations used to calculate the response of occupants to sensory cues (seeing flame, smelling smoke, seeing smoke) is based on the research work of Nober [Nober 1981].

Furthermore, EXITT views the occupants from an individual perspective such that the population is made up of individuals with defining attributes such as age, gender, normal travel speed, response time and individual capabilities (e.g. tolerance to smoke). Moreover, the movements of the occupants can be tracked throughout the simulation. In addition, the model is capable of simulating various forms of behaviour namely investigation, rescue, offering assistance, exiting and limited communication. The resultant behaviour of the occupants is dependent on their individual characteristics as well as environmental effects such as smoke concentration.

Interesting features of this model include its capability of viewing occupants from an individual perspective and tracking their movement throughout the simulation. One key objective of this dissertation involves the development of a hybrid model capable of utilising a combination of coarse nodes, fine nodes and continuous regions for enclosure representation. In this hybrid representation, it is likely that the occupants would be transiting across different spatial types, and therefore the model should be capable of tracking the attributes and positions of the occupants irrespective of the spatial type in which they are located. In this respect, it was identified that an individualistic approach would be suitable for the development of the coarse network component of the hybrid model. Also, the technique of applying penalties to paths within enclosures was adapted for the redirection of occupants navigating in coarse nodes in the hybrid model (described in Chapter 7).

2.1.4.2 Fine Node model

With advancements in computational processing power, the adoption of fine networks for modelling became increasingly popular. In the fine node approach, each section of the geometry is made up of a collection of nodes which are interconnected via arcs (unlike the coarse network in which each section is represented by a single node). The size and shape of the node and the connectivity varies from model to model, for example, buildingEXODUS typically uses a 0.5 m x 0.5 m mesh and each node has 8 neighbours whilst EGRESS [Doheny 1995] uses hexagonal nodes of size equal to the minimum area occupied by an occupant, with each node having 6 neighbours. **Figure 2-5** illustrates the two dimensional grid of cells for the enclosure representation of a multiple compartment geometry using the fine network implementation of buildingEXODUS. The fine network representation allows geometries and internal obstacles to be accurately represented. Furthermore, only one agent can occupy a node and therefore, this allows the location of each individual to be accurately tracked throughout the simulation. The EXODUS egress software is described in greater detail in Chapter 4.

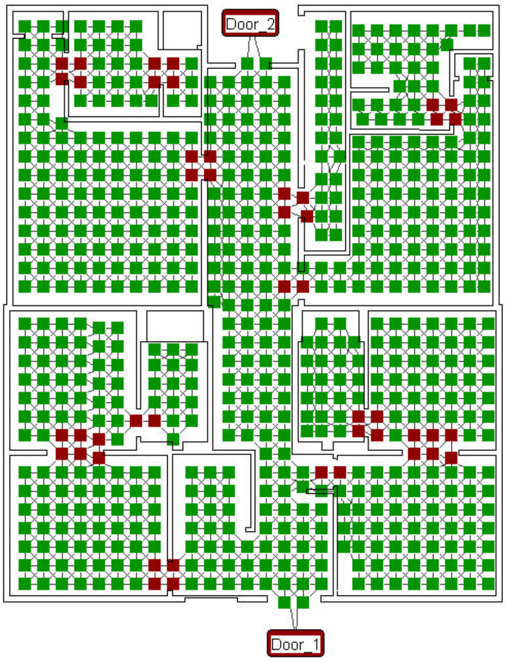


Figure 2-5 Representing a multiple compartment geometry using a two-dimensional grid of nodes in buildingEXODUS.

The next section describes the representation of physical space in evacuation models using a continuous modelling approach. Since a key objective of this dissertation entails the development of a continuous model, the focus of the next section is on the evaluation of the modelling approaches which have been implemented in current continuous models and the respective capabilities of such models.

2.1.4.3 Continuous Model

In continuous models, enclosures can be represented by polygons of any shape. The continuous model also views the occupants from an individual perspective, and, unlike the fine network approach where there is only one agent to a node and they occupy the centre of that node, any number of people can occupy a continuous region and they will all have a unique location within it. Also, occupants can navigate to any point (coordinate) in the continuous space without any restriction on their directions of motion. This is in contrast to the fine nodal representation where the directions of the occupants are limited by the numbers of arcs leading to adjacent nodes. **Figure 2-6** illustrates a multiple compartment geometry modelled in continuous space approach in Simulex. The contours provide an indication of the distance of different sections of the geometry from the exit.

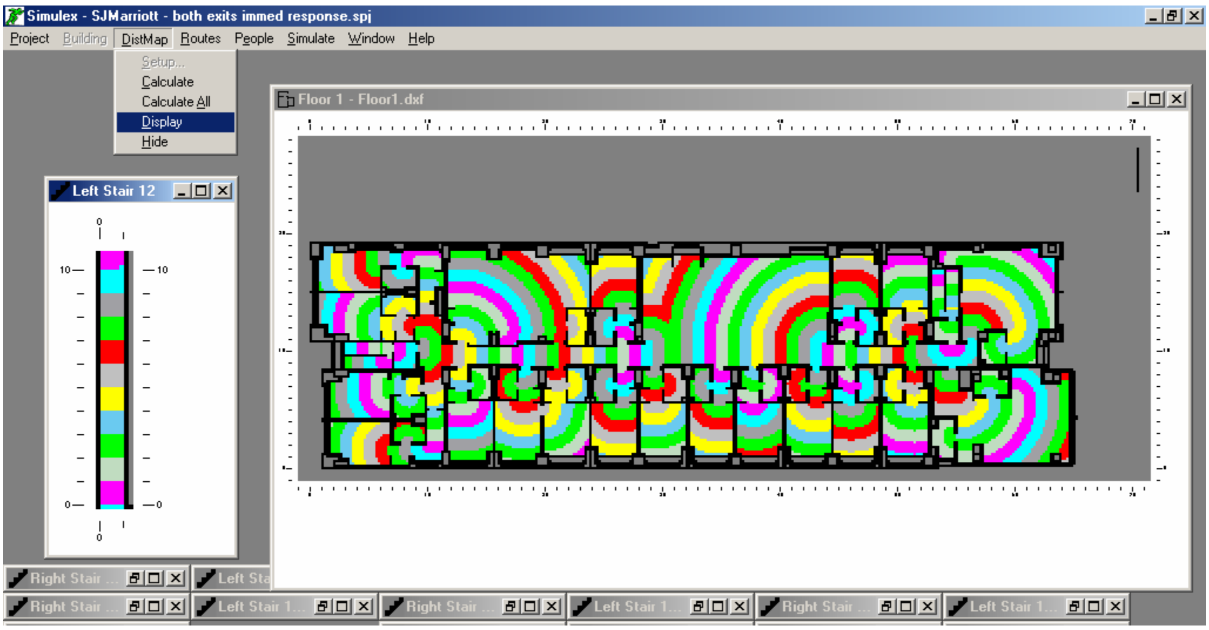


Figure 2-6 Continuous spatial representation in Simulex

Some of the current evacuation models which utilise a continuous space for representing agent movement are namely Simulex [Thompson 1994, Thompson 1995, Thompson 1996], ASERI [Schneider, Schneider 2001], Legion [Legion 2008a, Legion 2008b], GridFlow [Bre 2008, Purser 2009], MASSEgress [Pan 2006a] and Evi [EVI 2001, EVI 2008a, EVI 2008b].

2.1.4.3.1 Simulex

Simulex is a software tool developed by Peter Thompson from Integrated Environmental Solutions Ltd, UK. It is capable of modelling evacuation of large populations through multi-storey buildings. It can also model large, geometrically complex buildings with multiple floors and staircases. The model views the occupants from an individual perspective and tracks their positions throughout the simulation. Each occupant is modelled as a combination of three circles linked together rather than an ellipse in order to simplify the mathematical calculations of collision detection between the occupants. The representation of an individual occupant in Simulex is as shown in **Figure 2-7**. This body shape provides an accurate representation of an occupant's physical space occupancy and has been adapted for use in the bEX-Continuous Model (Chapter 5).

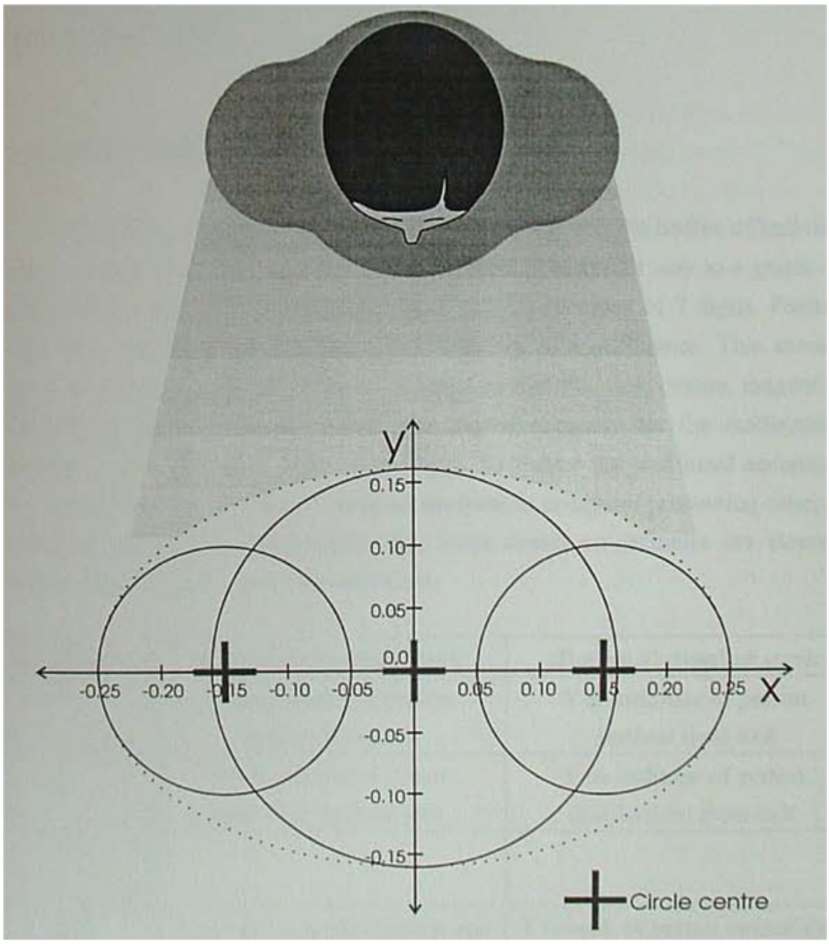


Figure 2-7 Representation of the body shape in Simulex

The enclosures in Simulex are represented as continuous spaces. The building plan is input and processed by DRAWPLAN which is then transformed into a distance map using GRIDFORM. The latter segments the available building space into a grid of 0.25 m by 0.25 m blocks. Each block of space is represented as a single point which has a numerical value denoting its distance from the nearest exit. For instance, exits and solid objects (e.g. walls) are assigned a value of zero and -1 respectively. The distance maps are used for assessing travel distances in buildings and also used by the route finding functions. The representation of the continuous space by the spatial blocks points is as shown in **Figure 2-8**. However, the use of the spatial block points in this fashion could result in an unnecessarily large number of points for large enclosures which need to be searched by the agent. On the other hand, methods such as the Visibility Graph (described in Chapter 5) could significantly reduce the number of points generated in a region of continuous space. Reducing the number of points will typically improve the computational performance of the route searching algorithms.

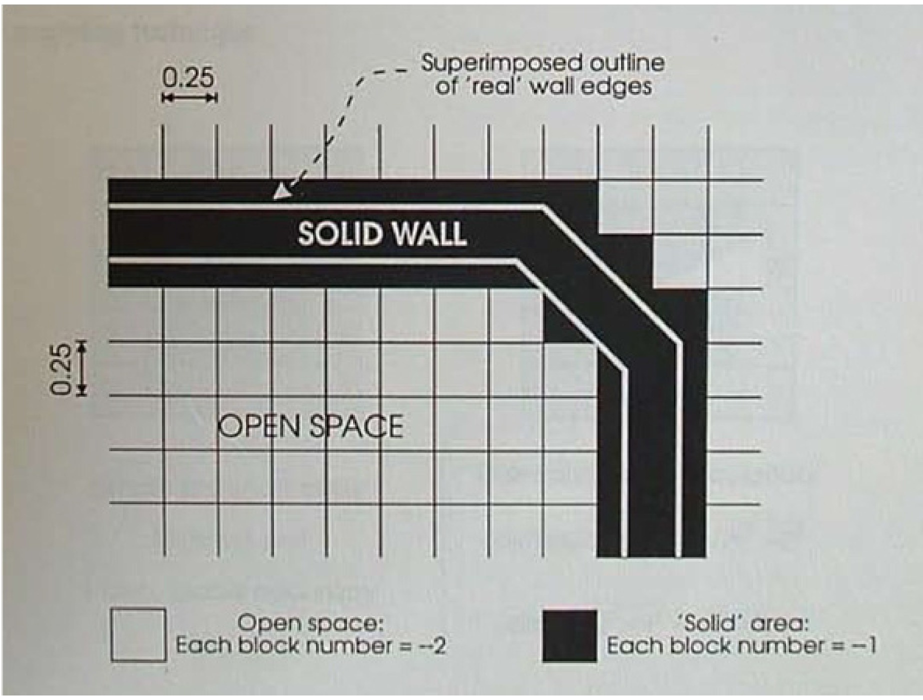


Figure 2-8 Representing the building space using spatial block points in Simulex

Simulex is a partial behavioural model which relies on inter-person distance (which is translated in population density) in order to assess travel speeds. Moreover, the movement model is based on a concept called 'body twist' which signifies that individuals can only rotate up to certain extent within a particular time frame without causing self-injury [Thompson 1994]. Simulex can model body rotation whereby the occupants are allowed to rotate by a maximum of 100 degrees per second. This value was obtained from many video-based analyses of individual movement at different venues [Thompson 1994]. Furthermore, Simulex incorporates localised route deviation algorithms which allow the occupants to demonstrate overtaking behaviour.

2.1.4.3.2 ASERI

ASERI (Advanced Simulation of Evacuation of Real Individuals) is an evacuation model developed by I.S.T. (Integrierte Sicherheits-Technik) GmbH, Germany [Kuligowski 2005]. It is capable of simulating the egress process of high-rise office buildings, trade fairs, hotels, stadiums, airports, railroad stations with very large populations.

ASERI views the occupants from an individual perspective such that the location and movement of every individual can be tracked throughout the simulation. Each occupant is characterised by a series of attributes which determines the behavioural capabilities during normal or emergency conditions. Some of the attributes are static (for instance sex, special incapabilities, special knowledge of the building layout, former experience, social interdependencies), while some attributes are dynamic (for example, response to smoke and exposure to toxic combustion products). Each occupant within the enclosure has an exit, which can be either the nearest exit or user-defined. During the simulation, the occupants can adjust their routes based on the occupant congestion around them and the external impact from evolving conditions in the building, for instance smoke conditions. The behaviour of the occupants is modelled using a rule-based approach. The perception of cues is modelled using individual reaction times. ASERI uses a matrix of delay times [Kuligowski 2005] which depends on the initial activity which an occupant is engaged in. Examples of such activities include watching TV, social activity and sleeping. ASERI is also capable of demonstrating behaviours such as queuing, merging of flows and congestion.

ASERI uses a continuous space system such that the instantaneous location of an occupant is represented by a pair of coordinates related to a floor or staircase. **Figure 2-9** illustrates the visualisation of an example simulation. An individual's body size is represented by shoulder and chest width which can be assigned individually or by specifying a distribution for a population. This approach allows the modelling of occupants with increased space requirements such as occupants with limited mobility and occupants carrying children or briefcases [Friedman 1992]. The movement of the occupants is defined by a combination of their travel speeds and the orientation of their corresponding velocity vectors. Furthermore, the movement is driven by two levels of goals namely local (internal exits, corners) and global goals (external exits).

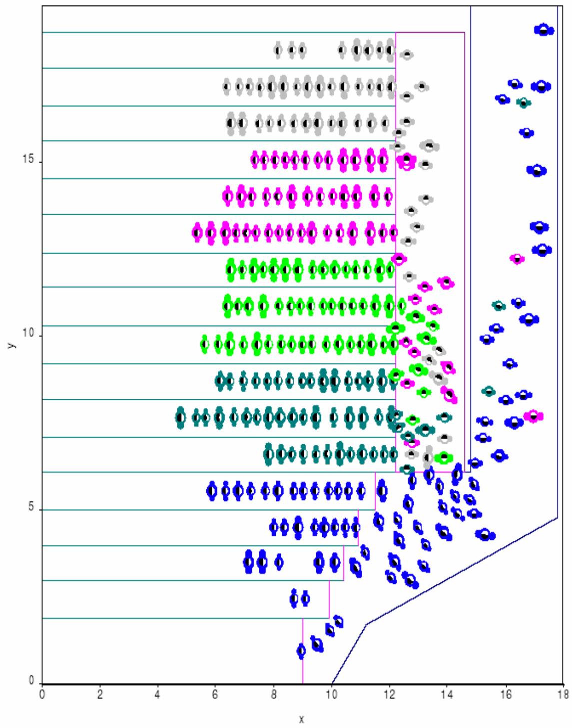


Figure 2-9 Visualisation of a simulation in ASERI

Although the underlying space is represented in a continuous fashion, the model is based on discrete event simulation techniques such that time is advanced in discrete time steps of 0.5 seconds. However, models such as Simulex and the bEX-Continuous Model (Chapter 5) utilise time step intervals of 0.1 seconds and 0.08 seconds respectively. It can be argued that the time interval of 0.5 seconds in ASERI might not be sufficient to represent highly detailed person-person and person-structure interactions. Also, this time step value is not adequate to represent the body rotations of the individuals.

2.1.4.3.3 Legion

Legion is a pedestrian model developed by the Legion International Ltd, UK. It is used to optimise the design and operation of public spaces which include transport terminals, sport, entertainment and leisure venues and shopping centres [Legion2008a]. Legion uses a two dimensional continuous space to model the structural configuration and movement of people. The model can handle multiple floors and various terrain types such as stairs, elevators, escalators and moving walkways.

The model views the occupants from an individual perspective and they are considered as intelligent agents with physical, behavioural and social characteristics [Legion 2008b]. These characteristics have been derived based on video footage of actual pedestrian movement.
The physical characteristics are defined by the body size; the behavioural characteristics include personal space, preferences for unimpeded walking speeds, willingness to adapt and the social characteristics include gender, age, culture and pedestrian type (e.g. commuter or tourist).

Each agent is modelled as a circular two dimensional entity with a circular body navigating in the continuous space in discrete time steps of 0.6 seconds. The occupants move by selecting a step which will minimise the perceived objective cost function whereby the cost is a weighted sum of three components namely inconvenience, frustration and discomfort (spatial). Each agent has a perception which is bounded by a distance and an angle, and is capable of identifying neighbours located within the perceptual boundaries. As the agents navigate towards their goals, they adapt their weights of the three components according to local conditions such as availability of space, population density and speed of neighbouring agents [Berrou 2005].

The representation of the personal space requirements of each agent is particularly interesting whereby the personal space lies in the direction of movement of the agent and is dependent on the travel speed. This approach has been adapted for use in the bEX-Continuous Model for the representation of the Perception System (Chapter 5). However, Legion simplifies the behavioural representation of the agents by utilising relatively few parameters and only one decision rule which is based on the least effort algorithm [Pelechano 2008b]. Moreover, using time steps of 0.6 seconds for discrete event simulation of a pedestrian model signifies lesser detail in the representation of person-person and person-structure interactions.

2.1.4.3.4 GridFlow

GridFlow is a building evacuation model developed by Purser and Bensilum of the Building Research Establishment (BRE) research and consultancy group in the UK. It was originally developed for research on human behaviour and egress. This model places a lot of emphasis on pre-movement interactions and pre-movement times in order to simulate the evacuation of building structures. These pre-movement times can be represented by explicit values or probability distributions.

The building structures in the model are represented using continuous space. However, a two dimensional grid of cells, each of size 0.5m x 0.5m, is overlaid on the floor plan to construct a distance map which maps the distance from each cell to all the exits, whilst working around obstacles present on the floor. This approach allows for the assessment of travel distances from the different sections of the enclosure. A visualisation of the distance mapping in GridFlow is as shown in **Figure 2-10**.

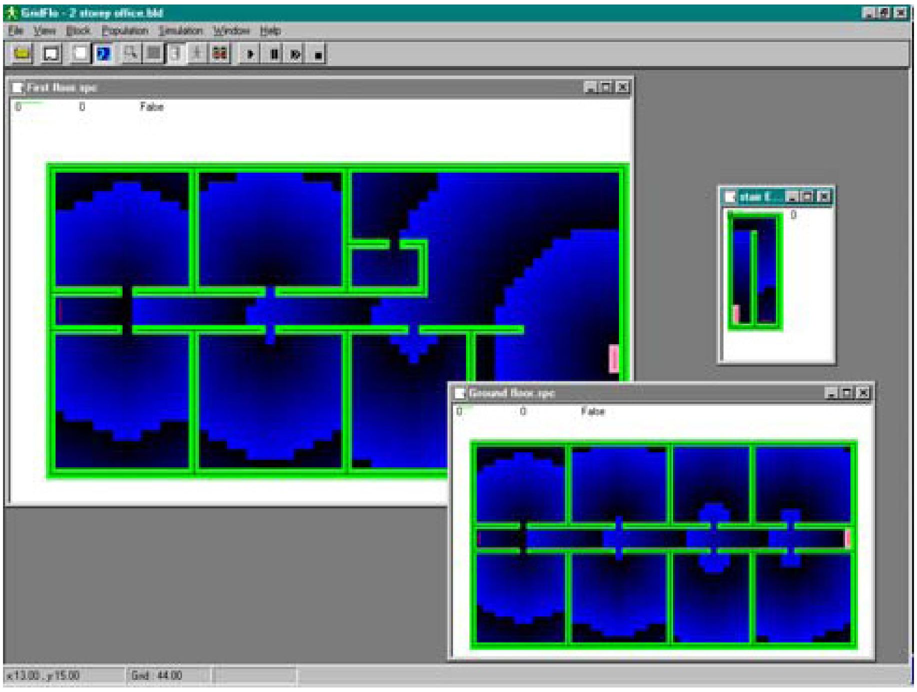


Figure 2-10 Distance mapping technique used in the continuous space system in GridFlow

The human behavioural model in GridFlow is simplified such that all the pre-movement activities are represented by a single time delay which differs for each individual. Moreover, it relies on pre-movement times and occupancy density to determine the movement of the population and is thus considered to be a partial behavioural model [Kuligowski 2005]. When the pre-movement activities have been completed, the occupants move to the nearest, preferred or randomly chosen exit.

The model views the occupants from an individual perspective such that the occupants are assigned individual attributes such as location, pre-movement time, exit goal and unimpeded walking speed. Behaviours such as overtaking and merging of flows can also be modelled.

2.1.4.3.5 MASSEgress

MASSEgress is a computational framework that has been developed by Xiaoshan Pan from Stanford University in USA. It is used for modelling human and social behaviours for emergency egress analysis. The physical space of the enclosures is represented by continuous space and spatial information such as obstacles, exits, assembly points and signs can be defined by CAD tools.

MASSEgress views the occupants from an individual perspective such that each individual occupant is modelled as an autonomous character and their locations can be accurately tracked throughout the simulation. Each occupant is represented as an agent comprising of an individual behaviour model which, in turn, is composed of three sub-systems namely Perception System, Behaviour System and Motor System. This approach of behavioural decomposition is particular interesting as it enables the different aspects of agent behaviour to be decoupled from each other, thereby facilitating the implementation as well as improving the maintainability of the model. The agents use a concept of View Volume [Pan 2006a] for detecting other agents and gathering spatial information such as location of wall boundaries, obstacles and signs. The View Volume represents a region of space bounded by a perception range and a view angle as shown in **Figure 2-11**.

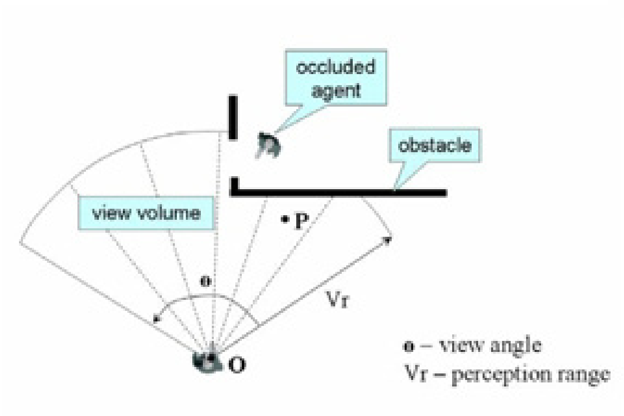


Figure 2-11 View volume of an agent in MASSEgress

However, relying solely on the agent's view volume for determining the visibility of wall boundaries and other obstacles in the enclosure is not computationally efficient. This is because the agents are required to perform computationally expensive collision detection checks with the walls and obstacles at every time step in the simulation.

Furthermore, the agents' decision making capabilities are modelled using decision trees which are triggered depending on the agents' urge to exit and the environmental conditions to which they are exposed. The combination of these decision trees can be used to model a variety of psychological and sociological characteristics of human behaviour for instance, exit familiarity, urge to exit, stress threshold type and herding factor. The social behaviours modelled are those which are commonly observed during emergency evacuation for instance queuing behaviour, herding behaviour, bi-directional crowd flow and competitive behaviour [Pan 2006a].

An interesting feature in MASSEgress is that the model makes use of a Grid Method in order to optimise the collision detection in between the agents. This approach entails the creation of an underlying grid of cells during the pre-processing phase such that each agent in the simulation is registered to a cell. The implementation of the Grid Method is as shown in **Figure 2-12**. During the simulation, the agents are de-allocated from their current cell and registered to the neighbouring cell depending on their locations. Therefore, instead of computing collision detection checks against all agents in the enclosure, the checks are limited only to the occupants within the neighbouring cells. Consequently, the Grid Method saves considerable computation time. The author claimed that this method is a solution for computing collisions of large numbers of agents with an O(N) time complexity. This method has been adapted for use in the agent search algorithms within the bEX Continuous Model (Chapter 5).

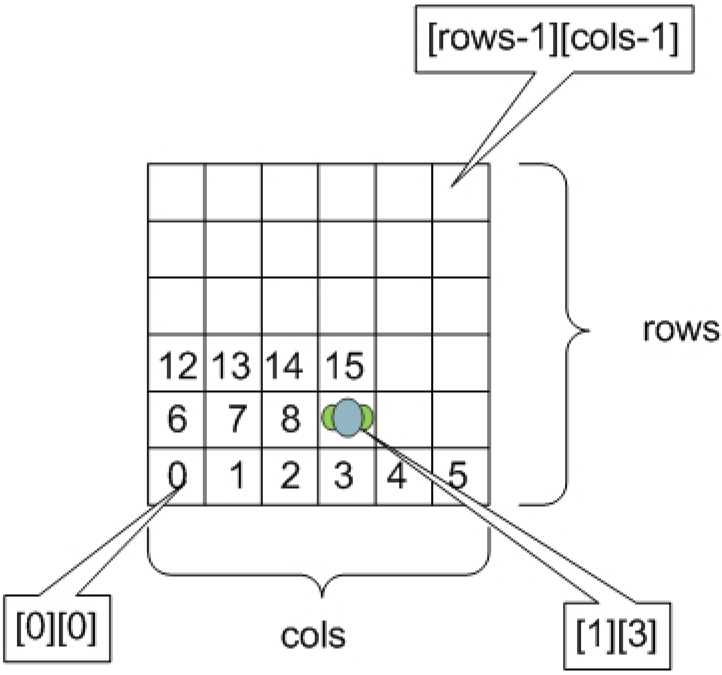


Figure 2-12 Construction of 2D grid in MASSEgress

2.1.4.3.6 Evi

Evi (Evacuability Index) is an evacuation model developed by the Evacuation Simulation Group of the Ship Stability Research Centre (SSRC, University of Strathclyde, UK) in collaboration with Deltamarine Ltd, UK. Evi has been specifically designed for the marine environment and its main areas of application include passenger ships and large cruise liners. It can be used to model passenger mustering and evacuation involving various scenarios (transfer to refuge areas, abandon ship) in emergency situations (fire, progressive flooding, cargo shift and collision). The model also takes into account the motion of the ship in its calculations [EVI 2001].

The environment model in Evi is modelled using a two dimensional continuous space. The ship area is segmented into multiple regions and region of space in between any two adjacent regions are defined as gates. In other words, two regions are said to be directly connected to each other if they have a common gate. The regions in the enclosure are defined as cabins, corridors and public areas and comprise of their own coordinates system and connectivity. The combination of the gates and their interconnectivities constitutes a gate graph which can be searched by the agents when navigating from their initial starting locations towards the embarkation stations. Evi uses a depth first exhaustive search over the gate graph to select optimal routes during high level planning processes. During the pre-processing phase, the path planning algorithm computes the distance information from each door to the embarkation station. Therefore, when situated in a region, agents can retrieve distance information of all the doors in the region and head for the shortest path leading to the embarkation area. The methodology behind the gate graph is particularly interesting as it reduces the complexity of the ship's continuous physical space to a much simpler graph. **Figure 2-13** illustrates a gate graph comprising of a network of 12 gates in Evi. However, the gate graph, unlike the Visibility Graph method [Doyle 1995], does not provide indication of the location of the wall boundaries. Consequently, agents navigating towards their target locations will be required to constantly check for wall boundaries at a local level, thereby impacting computational speed.

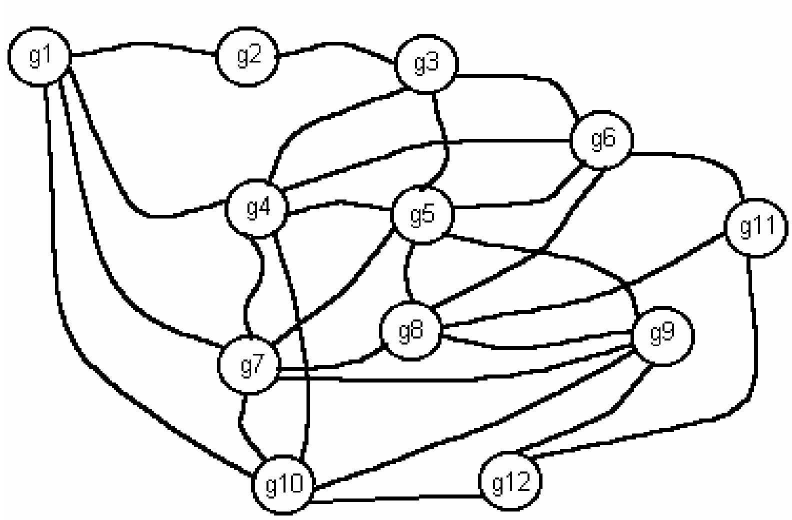


Figure 2-13 Gate graph in Evi

The developers describe EVI as being a mesoscopic model as it has the capability of representing both the macroscopic (planning, high level decision making) and microscopic (steering and movement of occupants) features of the evacuation process. The occupants are viewed from an individual perspective whereby each of them is treated as an individual agent. Each individual agent is defined by a collection of attributes such age, gender, familiarity and mobility. The low level movement is achieved by steering behaviours [Reynolds 1999] which attempt to direct the agents from their current positions to their targets. The travel speed of the agents is dependent on the sea/ship environment, the associated scenarios and parameters such as age, size, gender and crew guiding. Moreover, the agent travel speeds is also dictated by the surrounding population density. Because of the non-uniformity of population density and its dependence on the size of the area used for the density calculation, Evi utilises a concept called *Perceived Density.* This concept entails projecting a density rectangle of fixed dimensions ahead of the agent for enhancing the accuracy of density calculations. The travel speed values corresponding to the calculated perceived densities are based on the International Maritime Organisation (IMO) specifications [IMO 2002]. Furthermore, although the paths of the agents are pre-planned, the agents are capable of re-planning during evacuation depending on the local conditions, for instance crowd density and presence of smoke.

2.1.4.4 Hybrid Approaches

This section discusses the use of hybrid modelling approaches in evacuation models. It also sheds light upon the various perspectives from which the term 'hybrid' has been applied in different evacuation models.

Hybrid approaches rely on the combination of two or more subsystems for simulating new elements [Antsaklis 1997]. The subsystems can either be selectively employed, whereby certain components are used from each subsystem simultaneously or employed at different times depending on the conditions and situations.

The developers of Evi [EVI 2001, EVI 2008a, EVI 2008b] describe the model as being a hybrid as it utilises a combination of macroscopic and microscopic modelling techniques (mesoscopic). The macroscopic techniques pertain to the use of high-level planning techniques while the microscopic ones refer to the modelling of the individualities of the agents and their associated behaviours. When Evi is viewed from the perspective of the multiplicity of underlying technologies involved, it can be considered to be a form of hybrid. However, since the environment in which the agents navigate is modelled using continuous space, it can be argued that Evi is primarily a continuous model.

In [Borrmann 2010], the authors describe the implementation of a bidirectional coupling of macroscopic and microscopic pedestrian evacuation model. The macroscopic model is based on dynamic network flow theory while the microscopic model is based on cellular automaton. The network flow approach is based on the assumption that each pedestrian will be adopting optimal routing strategies towards their destinations. Therefore, this allows the identification of the lower bounds of evacuation times. On the other hand, the cellular automaton approach enables human behaviour to be captured and thereby predicts the heuristic upper bounds. The modelling strategy employed entails the coupling of the microscopic and macroscopic models which allows output from one model to be fed as input in the other model and vice versa. This coupling method reduces the gap between the predictions of evacuation times between simulation and optimisation. Based on the symbiotic relationship between the two models, the overall methodology can be defined as being hybrid. However, from a spatial representation perspective, this coupled model is essentially a fine network model.

Furthermore, based on the review of MASSEgress (described in Section 2.1.4.3.5), it can be argued that the egress tool is primarily a continuous model as it utilises continuous space for representing the physical space in which the agents navigate. This system uses a discretisation strategy whereby the enclosure consists of an underlying two dimensional grid of cells for optimising the spatial search of surrounding agents. Both the continuous spatial representation coupled with grid approach are employed simultaneously throughout the progress of the simulation. In this respect, MASSEgress can be described as a model with a hybrid nature.

Moreover, in [Cisek 2010], the authors describe the implementation of a fine-coarse network model for simulating building evacuation with an embedded information system. This model aims to address the problem of optimising the evacuation of a building consisting of multiple egress routes. Considerable emphasis is placed on the dynamic changes which occur as a result of people-structure interactions. The system is capable of providing full information about the various evacuation possibilities and provides recommendations on the most favourable routes. The agents are modelled from a collection of attributes which influence their decision making processes. Although the authors claim that the representation of model environment uses a combination of the fine network and coarse network model simultaneously, the actual spatial representation is based on a two-dimensional grid of nodes.

In the context of this model, the term 'coarse' is used simply to describe an underlying graph which allows the computation of optimal routes, which in turn, are used by the agents navigating in the fine network model. In this respect, the term 'fine-coarse network model' does not signify that the model is capable representing the physical space using a combination of fine nodes and coarse nodes within the same software tool. The use of such terms for characterising evacuation models is not appropriate as it can lead to ambiguities as regards the true capabilities of the models under consideration.

However, the term 'hybrid' used for defining the model developed as part of this dissertation is used from an entirely different perspective. In this work, the term 'hybrid' is used to define a model capable of representing the entire physical space of enclosures using a combination of fine node networks, coarse node networks as well as continuous regions, within a single integrated software tool. Based on the perspective of spatial representation, this modelling strategy is novel in the field of evacuation modelling and is described in Chapter 7.

2.1.4.5 Evaluation of the suitability of the various spatial representation techniques

This section evaluates the suitability of the coarse network, fine network and continuous region approaches towards modelling different types of scenarios and terrains in egress simulations.

2.1.4.5.1 Coarse Network Approach

The representation of enclosures using coarse nodes greatly simplifies the building layout and such an approach can make it easier to manage large enclosures in simulations, both from the perspective of the user and the model. This representation gives such models the ability to handle large population sizes at a relatively low computational cost.

However, since the exact location of an occupant is not represented, it not possible to perform detailed calculations on individual movement and therefore, the interaction between the occupants (e.g. overtaking and local conflict resolution) and the interaction between the occupants and the structure (e.g. obstacle avoidance) cannot be accurately modelled. In addition, the structure of coarse network approach assumes that all exit paths will be used at maximum capacity, throughout the entire simulation. In this respect, such models tend to provide optimistic evacuation time predictions.

Moreover, the equations used in coarse node models represent a simplified version of egress movement, rather than behaviour. Therefore, such models fail to incorporate the complexities of human behaviour during egress [Gwynne 2008]. Also, coarse node models rely on the assumption that human behaviours can be averaged out, but this is often not the case. Furthermore, PEDROUTE [Halcrow 1987] has different speed-flow data corresponding to different terrain types such as escalators, junctions of passageways, stairways, concourses and platforms. In the case of the usage of escalators, it has been observed that there is more than one type of occupant behaviour. This has been observed by J.O'Neil et al [ONeil 2008] in their analysis of escalator related injuries. For example, some people prefer to ride while others prefer to walk on the escalators [ONeil 2008]. As there are multiple behaviours which could be involved, relying on a speed-flow equation for describing movement along the escalator might not be sufficient. Modelling combinations of complex behaviours requires the dynamics of the occupant body size and location to be included in the simulation which is not feasible in coarse network approaches [Gwynne 2000].

Coarse network models are most applicable for modelling scenarios where the occupants are expected to show uniform behaviour, for instance behaviour of occupants moving in the same direction across a corridor. Although coarse network models, for instance, PEDROUTE includes speed-flow data as regards junctions of passageways [Cunningham 1993]; the accuracy of the model for representing phenomena such as cross-flows is questionable. This is because accurate modelling of such phenomena requires the occupant body size, location and person-person interactions to be represented.

Furthermore, it is difficult to represent the interaction of occupants with objects in coarse networks, for instance, pushing, pulling or carrying objects. The presence of objects increases the overall space occupied by the individuals and depending on the nature and size of the object, there could be a significant impact on the dynamics of the overall motion of the occupants. For instance, the manner in which a person pushing a wheelchair slowly across a doorway can affect the motion and behaviour of the surrounding occupants, cannot be modelled accurately. [Thompson 1994].

Nevertheless, coarse network models are useful to identify areas of particular interest for example, areas where maximum flows can be achieved or regions which might be prone to congestion and bottlenecks. Some models also have the facility to analyse the bottlenecks that would occur during the evacuation and also to find the average time spent by people in congestion e.g. EVACNET4 [Kisko 1998]. Also these models can be used to handle very large geometries e.g. whole towns or cities because the intricate details regarding the geometries and people to people and people to geometry interaction need not be modelled. In other words, such models can be used to assess the potential for evacuation for relatively simple geometries, but cannot be fully utilised to investigate scenarios where the identification and the understanding of the complex physical interactions between individuals, is of prime importance.

2.1.4.5.2 Fine Network Approach

The fine nodal approach allows the creation of geometrically complex layouts and the representation of barriers and internal obstacles which can have a significant impact on the choice of route of the occupants. Furthermore, occupants can be modelled from an individual perspective in the sense that each occupant has a unique location (the occupant occupies the centre location of the node) and their movement can be accurately tracked throughout the simulation. Therefore, it allows the representation of a more diverse population, with each occupant having different traits.

Nevertheless, such representations make use of grids of predetermined sizes and therefore cannot accurately represent the dynamic space occupied by an individual. This is because the dynamic space occupied by an individual fluctuates depending on the circumstances to which the individual is exposed to [Thompson 1994]. For instance, in congested areas or openings, people have a tendency to move as close as possible. In models such as buildingEXODUS, the flow rates of occupants can be adjusted at high density regions in order to reflect the situation. Moreover, this restriction in space occupancy by an individual limits the ability to model objects of varying sizes which the occupants might be interacting with for example, wheelchairs, trolleys etc. However, there are some models for instance, buildingEXODUS and EGRESS, which have the facility to allow the users of the model to increase the grid spacing if the occupants are expected to carry large objects. Nevertheless, this approach has some drawbacks; firstly the accurate implementation of this feature will depend on the users specifying the correct grid spacing sizes and secondly, increasing the grid spacing will alter the whole population rather than just those with objects.

Fine node models have good computational speeds and can be scaled to model large population sizes in complex multi-storey buildings. This is because the discrete spatial representation allows for faster computation of people to people (e.g. queuing) and people to structure (e.g. avoidance of walls and internal obstacles) interactions. Fine node models demonstrate superior computational performance as compared to continuous models. Since agents in fine node networks can only navigate to regions occupied by nodes, boundaries and obstacles within the enclosure can easily be modelled by removing relevant nodes and creating gaps of empty spaces around the wall boundaries and obstacles. In this respect, the agents are not required to explicitly perform computationally expensive collision detection checks with the boundaries of the enclosure and internal obstacles. Furthermore, agents in fine node networks can readily retrieve the neighbouring agents by identifying agents located on surrounding nodes and therefore, computationally expensive spatial search routines are not required. In addition, the fine nodal structure prevents any form of overlapping of agents and therefore, agents are not required to perform expensive collision detection and avoidance strategies with surrounding agents.

However, the use of fixed node sizes forces the maximum densities to be limited. This is because the maximum packing density is restricted by the node spacing and the number of neighbouring nodes around each node. For instance, if a node size is defined with an area of 0.25 m^2 and considering that each node can only by occupied by one agent, the maximum density which can be attained at any point within the simulation is 4 persons/m^2. However, literature in the field has shown that population densities beyond 4 persons/m^2 can be attained while still allowing people movement [Predtechenskii 1978, Ando 1988]. In addition, fine network models are less accurate in predicting flow values when the packing densities become very high in heavily congested areas.

In addition, the modelling of enclosures with fine network models can lead to inaccuracies if the grid is not well aligned with the geometry [Pelechano 2008b]. This can lead to scenarios where only one agent is allowed to pass through an opening, when in reality; the width opening could be adequate to allow multiple people through simultaneously. Also, the representation of the occupant movement in fine network models involves the transition of the occupants from the centre of their current cell to the centre of their neighbouring cells. The methodology is typically based on the use of potential maps which map out distances from the various points of the geometry. In certain areas of the geometry, the potential map could favour the adoption of certain trajectories which might be unrealistic. However, models such as EXODUS include additional features such as the use of Attractor and Discharge nodes to counteract such artefacts [Galea 2004a]. These special nodes can be used to manipulate occupant movement within the enclosure by allowing the flow rate to be capped and by overriding the global potential map. **Figure 2-14** shows an illustration of an example use of attractor and discharge nodes in a corridor. Without the Attractor/Discharge nodes, the agents have the tendency to follow path A, whereas with Attractor/Discharge nodes, the agents favour path B.

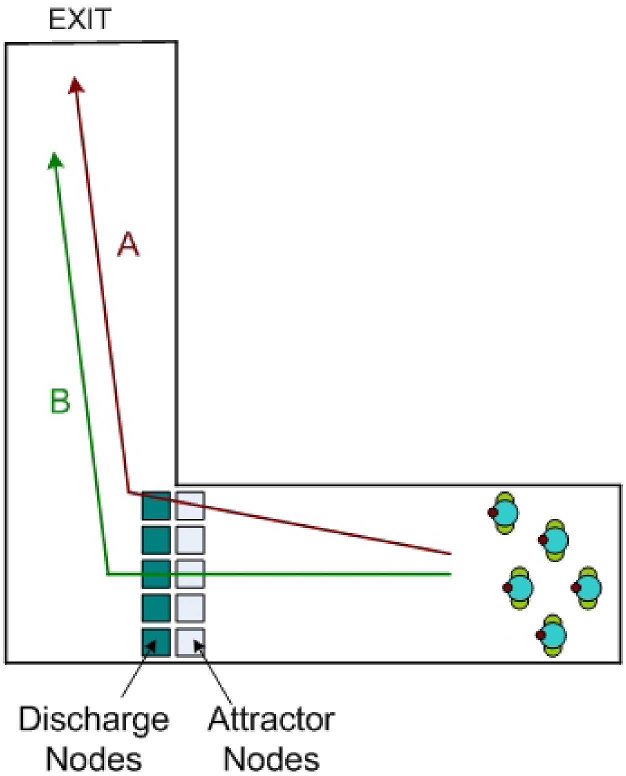


Figure 2-14 Using Attractor/Discharge nodes in a corridor modelled in buildingEXODUS

2.1.4.5.3 Continuous Region Approach

People modelled in continuous space have physical attributes which are not present in the other nodal types; e.g. varying body widths, turning rate of occupants (the rate at which each occupant changes its direction). People modelled in continuous space can also exhibit additional behaviours, such as overtaking, collision avoidance with walls, people and internal obstacles. Also, the representation of the physical space and the personal spaces of each individual agent makes the continuous model appropriate for modelling scenarios involving a multitude of behaviours; for instance merging of flows, contra-flows, and behaviours on staircases and escalators. Such models also possess the capability of demonstrating the emergence of certain behaviours and are particular suited for the investigation of behaviours such as lane formation in densely populated cross-flow scenarios.

In addition, due to fact that the physical space occupied by an occupant in a continuous model is not limited to a node of fixed size, it is better suited for modelling the interaction of occupants with objects. The continuous space allows for an accurate representation of the shape of the objects and the resultant speeds produced by the interaction of the agent with the objects can be represented. Examples of such interactions include occupants pushing, pulling or carrying objects such as shopping trolleys, luggage, suitcases and stretchers (hospital evacuations).

Moreover, continuous models are more sensitive to small variations in the enclosure. For example, small changes in the corridor width and the resultant flow can be better modelled in continuous models than fine node networks which are locked to a grid size. In this respect, continuous models are better suited for predicting flow rates at openings. However, this approach requires far more computational processing power than the fine and coarse network approaches, and is therefore less scalable. Nevertheless, there are several strategies which will be discussed in later chapters that can enhance the scalability and efficiency of such models.

From a qualitative perspective, the movement patterns and paths of agents navigating in continuous spaces appear more realistic when compared to fine network models. However, from a quantitative perspective, although continuous models are suitable for predicting flow rates, they are also harder to calibrate than models utilising the other spatial representation types. This is because of the increased number of attributes in the continuous models. Due to the high level of granularity involved, such models are also prone to locking, as a result of conflicts amongst agents in high population densities. Therefore, continuous models require additional mechanisms to address any potential deadlock situations.

2.2 Factors influencing human behaviour

One of the key aims of this project involves the development of a Continuous Model for simulating egress. Before proceeding to the implementation of any evacuation model, it is imperative to consider the fundamental factors which impact human behaviour and movement. Also, it would be unrealistic to assume that human behaviours can be represented to the exact same level of detail and accuracy as they occur in real life. Nevertheless, it should be possible to develop a distribution of behavioural responses which can be representative of the expected outcomes. Based on literature, there is an array of factors which are known to impact human behaviour during egress [Proulx 2001, Shields 1999, Gwynne 2000, Sime 1983, Pan 2006a]. This section presents an overview of the key factors which were considered in the development of a behavioural engine (mainly focusing on motion behaviours) for the Continuous Model (Chapter 5).

2.2.1 Configuration

The configuration of the enclosure represents the geometrical boundary within which the event causing the evacuation takes place. The various aspects of configuration are generally defined in the traditional building codes and include the number of exits, exit widths, travel distances and building layouts [Gwynne 2000]. The essence of these configurational considerations relies on the assumption that the building regulations are adhered to before and after the construction of the enclosure. Fire disasters in the past have illustrated the catastrophic effects due to poor adherence to building codes. For instance, in the Beverly Hills Supper Club fire incident [Best 1977], the building did not have a sufficient number of exits to be able to cater for the safe evacuation of large numbers of people that it could accommodate.

The configurational influences on movement have also been demonstrated in studies conducted by Peschl (Section 2.4.5). His studies showed that the flow through a door could be increased if the doorway had curved corridor walls funnelling people into the exit. The impact of configuration on people movement rates in described in more detail in Section 2.4.

In these respects, in order to be able to represent the effects of geometrical constraints on the population within the Continuous Model, the agents need the capability of sensing and detecting walls and obstacles. Moreover, algorithms related to the assessment of travel distances from various sections of the geometry (represented by the continuous space) are required.

2.2.1.1 Procedure

The procedures implemented within an enclosure are employed to guide occupant behaviour during evacuation. These include the use of egress systems such as alarms and signage. Such systems, if implemented properly, can be really effective towards the success of an evacuation. The initial phases of egress can be prone to uncertainty and confusion which can ultimately lead to significant delays in occupant response. For instance, in experiments conducted in [Proulx 1994], Proulx noted that some people took as long as 25 minutes before starting to evacuate. In this respect, the usage of appropriate alarm systems in enclosures can significantly reduce response times. However, the success of such systems depends on the clarity of the warning and the believability of the alarm [Galea 2004b]. Moreover, the use of signage systems can help in the wayfinding process especially when the occupants are situated in unfamiliar structures. The effective use of signage systems can be used to guide evacuees towards appropriate emergency routes. Consequently, such systems help to minimise the time spent by the agents in wayfinding, thus improving the evacuation efficiency.

In addition, building familiarity is an essential factor in determining the likely actions of occupants. As such, it can have significant impact on the behaviour of occupants in the sense that instead of using the nearest exits, the evacuees could end up travelling further to use a less obvious exit. As noted by Sime in [Sime 1985], people have a tendency to head in a familiar direction in an emergency. For instance, in an unfamiliar building most occupants are seen to have the preference of evacuating via the route they came in [Shields 1999]. This behaviour could have significant influences on evacuation efficiency due to the fact that egress routes are conventionally designed to be used in emergency only, and consequently could be totally unfamiliar to occupants in public buildings.

These findings suggest the need for developing a mechanism in the Continuous Model (described in Chapter 5) to allow exits to be assigned to the agent population, rather than them navigating only to their nearest exits. In addition, algorithmic procedures regarding the assignment of response time distributions to occupants are necessitated in the model to simulate the effect of delayed occupant response on evacuation efficiency.

2.2.1.2 Social Characteristics

An occupant rarely exists on his/her own but instead is usually surrounded by other individuals in the enclosure. Therefore, it is vital to consider the social characteristics which can impact human behaviour. An occupant acts differently when alone and when located in a small group. In [Sime 1983], it was noted that individuals have a tendency of maintain close proximities with others to whom they have emotional ties. For instance, members of the family would try to adopt strategies which would benefit the group rather than for individual survival. This was referred to as Affiliative Behaviour [Sime 1993] whereby individuals would refer to primary members of the crowd for resolving ambiguities and move towards familiar persons. Moreover in studies involving unannounced evacuation of four retail stores [Shields 2000], it was found that customers, who had separated in the stores at the time the alarm sounded, tried to locate and rejoin their companions before starting to evacuate. Such forms of behaviour can invariably trigger delays in the evacuation process, thus impacting the overall evacuation performance.

Previous studies have shown the effects of crowd density on the behaviour of people. A crowd is a collection of individuals. Crowd density is defined as the number of individuals per unit area. Still [Still 2000] defined crowd density as 40 people in 10 square metres for a moving crowd (i.e. 4 people/m^2 and 47 for standing areas (4.7 people/m^2). The change in behaviour which occurs when a person moves from a region of low population density to one of high population density can be explained through the concept of personal space.

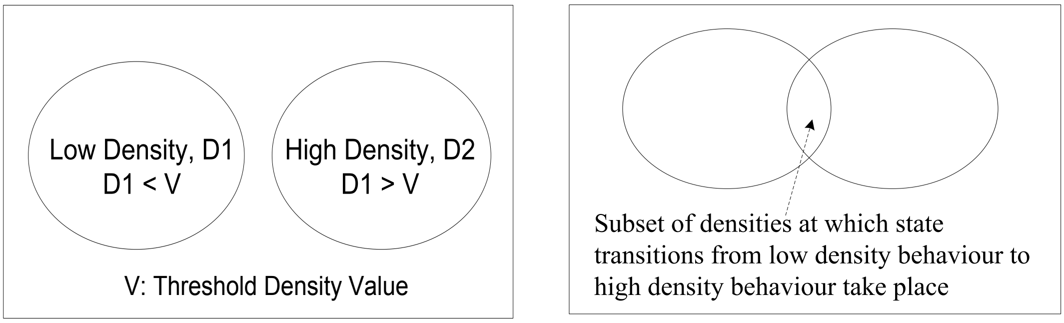
Pan [Pan 2006a] defines the concept of personal spaces whereby individuals conform to a social rule and aim to maintain a safe distance from others. This social norm does indeed vary according to the gender, culture and social structures. Furthermore, the personal space concept suggests that people refrain from intruding into the personal spaces of other people while trying to maintain their own personal space. Consequently, when individuals find that their personal space is being invaded (for instance in a region of high population density); their behaviour is likely to change. A person whose space is invaded will slow down or take evasive action, such as side-stepping or overtaking as long as it is possible to do so. However, even when the crowd density builds up, the individuals try to regain their personal space and avoid contact with others until the safety limit proposed by Still [Still 2000] is exceeded and maintenance of personal space becomes less feasible. Also, in [Thompson 1994], Thompson describes the concept of inter-personal distance as being the spacing between individuals in a crowded situation and explains how this spacing can be representative of the density of the population.

Hall [Hall 1966] provides a classification of interpersonal distances as shown in **Table 2-1**. These distances have been obtained from observation and are based on the sensory shifts of sight, smell, hearing, touch and thermal receptivity. In this respect, these interpersonal distance criteria are indicative of levels of separation between people when situated in different population densities and therefore could be included in the development of the behavioural engine of the Continuous Model.

Table 2-1 Classification of Interpersonal distances [Hall 1966]

Type of interpersonal space	Description	Distance
public distance (far)	Represents the 'flight zone' – indicating that an individual can take evasive or defensive actions	> 7.5m
public distance (near)		3.6-7.5m
social distance (far)	Zone of potential vulnerability – distance used for formal meetings	2.1-3.6m
social distance (near)	Zone of potential vulnerability – distance used for casual meetings	1.2-2.1m
personal distance (far)	Individual's circle of trust – interpersonal spacing found in a spacious waiting area	0.75-1.2m
personal distance (near)	Commonly encountered in denser waiting areas and queuing situations.	0.45-0.75m
intimate distance	Crowding occurs - leading to the 'touching' situation found when travelling in a lift car.	<0.45m

Furthermore, to model the movement of individuals, exposed to different crowd densities, it is vital to incorporate behaviours which are sensitive to the densities present. For instance, in Simulex [Thompson 1994], overtaking behaviour is applied only when the population density in the forward projected area of the assessing agent is less than 2 persons/m^2. However, instead of having a direct behaviour transition from low density to high density, it would be desirable to include a subset of densities which would relate to behaviours common to both low and high densities as shown in **Figure 2-15**(b). This criterion will be taken into consideration in the development of the agent's perception mechanism in the Continuous Model.



(a) Distinct density sets **(b) Subset of densities**

Figure 2-15 Proposed representation of potential population density sets in agent model

2.2.1.3 Human Physical Attributes

A crowd consists of a collection people who have diverse sets of physical attributes. Based on the literature, some of the key attributes which affect human movement in a crowd are namely gender, age, level of disability and body size.

2.2.1.3.1 Gender

The impact of gender on decision-making is dependent on the scenario, specifically the role structure in place and whether this structure is sensitive to gender. For example, social structures in domestic situations result in gender being a significant factor in egress. Previous studies have shown that males and females respond differently in emergency situations whereby men tend to be more action oriented while women tend to be safety oriented. For instance, males have been identified to be predominant in fire fighting activities while females are more likely to warn others [Bryan 1996, Wood 1990]. Moreover, studies have shown that gender has an influence on the travel speeds of individuals. For instance, in [Ando 1988], results have shown that the travel speeds of males were higher than that of females at all ages.

2.2.1.3.2 Age

Age is an important factor which needs to be considered as regards evacuation movement. Age is related to the mobility of the occupants. For instance, children of very young age require assistance by the adults during egress. Also, age has an effect on the behaviour of occupants. For instance, in [Proulx 1996], the findings showed that those aged over 65 were less likely to hear the fire alarm and had to be informed in order to evacuate, whereas those in the age groups of 18 to 40 had a better perception of the risks. Moreover, the people aged over 65 were less inclined to try to evacuate by themselves and therefore required assistance.

2.2.1.3.3 Level of disability

It is unrealistic to assume that all occupants in an enclosure have equal movement abilities. Movement abilities of disabled people impose additional concerns on evacuation efficiency. For instance, studies conducted by Shields [Shields 2000] regarding the unannounced evacuation of retail stores, illustrated the impact of disability on movement rates. In these studies, it was found that presence of wheelchairs users within the population caused congestion and delays at certain locations within the buildings, for example, when the wheel chairs users were faced with stairs during their egress. Moreover, in enclosures such as nursing homes and hospitals, the occupants might require additional assistance in the event of an evacuation, and therefore, such structures need to have special procedures in place [Galea 2010]. In addition, the level of disability can have an effect on the occupant behaviour during egress. For example, a visually impaired evacuee would face difficulties in the wayfinding process. This proposes the need for introducing a mobility attribute in the Continuous Model will make it possible to differentiate between able-bodied and disabled persons.

2.2.1.3.4 Body Size

The body size of individuals in a crowd helps determine the crowd density which in turn affects the crowd movement. For instance, Predtechenskii and Milinskii [Predtechenskii 1978] describe crowd density as the ratio of occupied space to available space. In this respect, the density of a crowd with a specific number people within a defined area will vary according to the season. During winter, the increase in the layers of clothing for each individual result in an increase in the ratio of occupied space to the available space, for a defined area. This would result in a lesser packing of individuals within the area. The measurements of Predtechenskii and Milinskii show a maximum flow of 1.14p/m/s at density $0.75m^2/m^2$ under normal conditions and mid-season dress, and a maximum flow of 1.40p/m/s at density of $0.72m^2/m^2$ under emergency conditions. Still [Still 2000] describes the difference in body sizes and how these should be considered for high density environments, especially during ingress and egress. Thompson [Thompson 1994] describes the use of 3 circles for the representation of the varying body dimensions of agents and the corresponding benefits of using such an approach. The same technique is applied in the Continuous Model for agent modelling (Chapter 5).

2.3 Motion Behaviour

The term behaviour can have many meanings, for instance, it can refer to the actions or reactions of an object or organism, under specified circumstances. It can also be used to describe a person's conduct, that is, how a person's actions fit within the social environment. In the context of this research, the type of behaviour which is mainly under consideration is motion behaviour, which refers to the path or trajectory undertaken by a character in response to stimuli within its environment [Reynolds 1999].

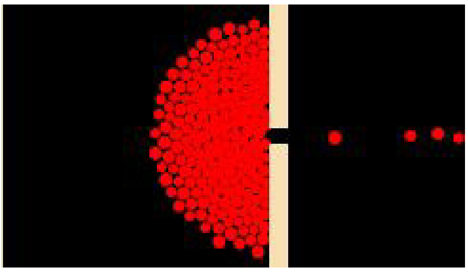
2.3.1 Helbing's Social Forces Model

The underlying principle behind Helbing's Social Forces model is that pedestrians are used to the situations to which they have been exposed previously (based on their past experiences) and as a result, they try to identify how to 'best' react to a certain situation [Helbing 1997]. The social forces model simulates crowd motion by treating the pedestrians as particles which are subject to social and physical forces. The forces can be attractive or repulsive in nature and inherent differential equations determine where the agents should move to and at what speed. Helbing's equation of motion [Helbing 2000] describing changes in velocity for a particular pedestrian is as shown below:

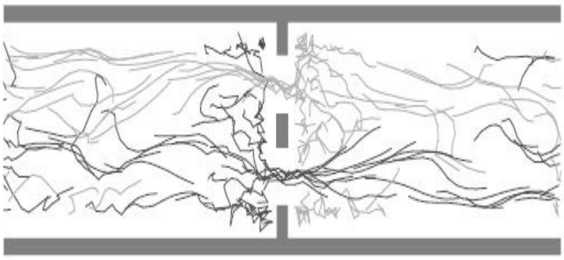
$$m_i \frac{dv_i}{dt} = m_i \frac{v_i^0(t)e_i^0(t) - v_i(t)}{\tau_i} + \sum_{j(\neq i)} f_{ij} + \sum_{W} f_{iw} \qquad \textbf{Equation 2-3}$$

Where a pedestrian of mass m_i moves at a desired speed of v_i^0 within a time span τ_i. The second term on the right hand side represents the repulsive forces due to the presence of neighbouring pedestrians and the third term represents the represents the repulsive forces due to the presence of wall boundaries [Helbing 2000].

Moreover, this model is capable of demonstrating queuing, blockages at exits and self organisation phenomena which are often seen in real crowds. Examples of such phenomena include lane formation as a result of high densities of agents moving in opposite directions. Example behaviours from Helbing's model are illustrated in **Figure 2-16**. However, a drawback of this model is that the pedestrians are seen to vibrate continuously, especially at high crowd densities. Such vibrations appear unrealistic and are not truly representative of human movement in crowds. Moreover, this model does not represent the differences between the pedestrians such that they are all assigned with the same behavioural set.



(a) Pedestrians crowding at an exit

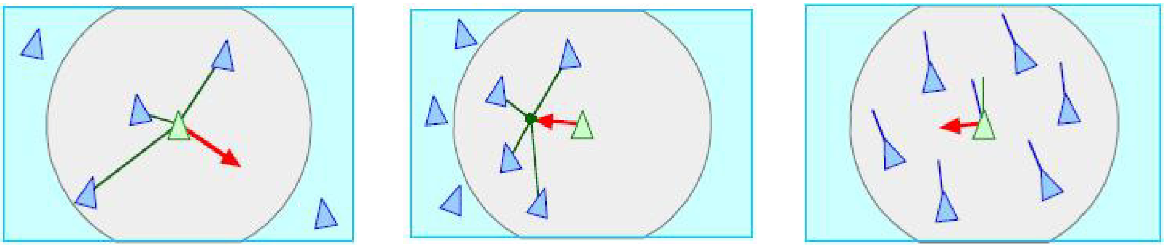


(b) Formation of lanes

Figure 2-16 Examples of behaviours demonstrated in Helbing's model

2.3.2 Reynolds's Boids System

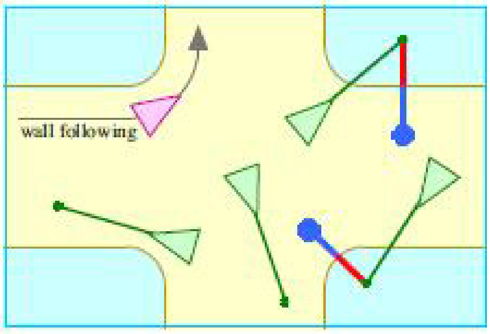
Reynolds's Boids System consists of autonomous characters which navigate in their environments in an improvisational manner [Reynolds 1999]. The behavioural component of this model consists of simple rules which can be combined to produce complex motion behaviours. One such example is in the simulation of characters known as boids in which 3 simple behaviours namely Separation, Cohesion and Alignment are used to demonstrate flocking characteristics of birds and fishes [Reynolds 1987]. Separation keeps the boids a certain distance from each other, Cohesion allows them to form groups with other nearby boids and Alignment allows them to move in the same direction.



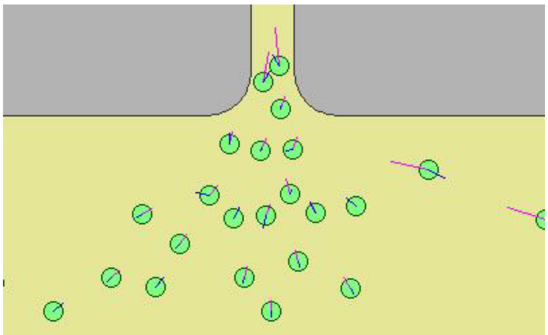
(a) Separation **(b) Cohesion** **(c) Alignment**

Figure 2-17 Component behaviours used for simulation of boids

In [Reynolds 1999], Reynolds describes the implementation of motion behaviours such as Containment and Unaligned Collision Avoidance which can be adapted for the simulation of pedestrians. Examples of behaviours simulated in Reynolds's model are shown in **Figure 2-18**. However, these behaviours operate on the assumption that the rate at which the agents move forward is faster than the rate at which the agents turn. Consequently, these behaviours do not work particular well for slow moving agents. Furthermore, although Reynolds's behavioural model is capable of simulating collision detection and avoidance of agents, however it cannot perform collision response. Consequently, the agents may overlap with each other at higher densities. In addition, the agents show oscillatory motion when the agents are navigating very slowly in high density environments. These drawbacks imply that the Reynold's model is not particularly suitable for modelling the egress of agents in high densities. Moreover, none of the examples showing the application of Reynold's model show geometries with square corners, which might indicate a potential issue. However, the approach involving the combination of simple behaviours for the simulation of more complex behavioural routines is particularly interesting as it adds extensibility to the Reynold's behavioural model. This component oriented approach has been adapted for use in the design of the behavioural system of the Continuous Model presented in this thesis (Chapter 5).



(a) Containment: Agents remain within the boundaries of the geometry



(b) Queuing: Agents demonstrating queuing on their approach to exits

Figure 2-18 Examples of behaviours demonstrated in Reynolds's model

2.3.3 Reciprocal Velocity Obstacle

The Reciprocal Velocity Obstacle (RVO) technique has been derived from a concept in robotics called Velocity Obstacle [Fiorini 1998]. This approach has been applied for the simulation of multi-agent navigation in crowded environments [Van den Berg 2008]. The aim of the RVO approach is to enable agents to navigate from their starting locations to their goal locations without colliding with other agents moving in the same environment. This approach relies on the assumption that the agents are dynamic obstacles whose future motions are predicted by extrapolating their current velocities. The RVO approach is applied as follows:

- Each agent identifies the extrapolated velocities of its surrounding agents. This set of velocities constitutes an obstacle.
- Each agent attempts to select a new velocity outside the region bounded by the obstacle.

Figure 2-19 illustrates the application of the RVO approach to two agents A and B moving in opposite directions. **Figure 2-19**(b), agent A selects a new velocity (nearest to its current velocity) which lies outside the velocity obstacle induced by agent B, while in Figure 2-19(c), agent B selects a new velocity (nearest to its current velocity) which lies outside the velocity obstacle induced by agent A.

The RVO approach is capable of demonstrating smooth movement paths and the emergence of lane formation. However, it is not particularly suitable to congested environments involving cross-flows in narrow passages, as it causes the agents moving in opposite directions to perform lengthy oscillations before they are able to pass each other.

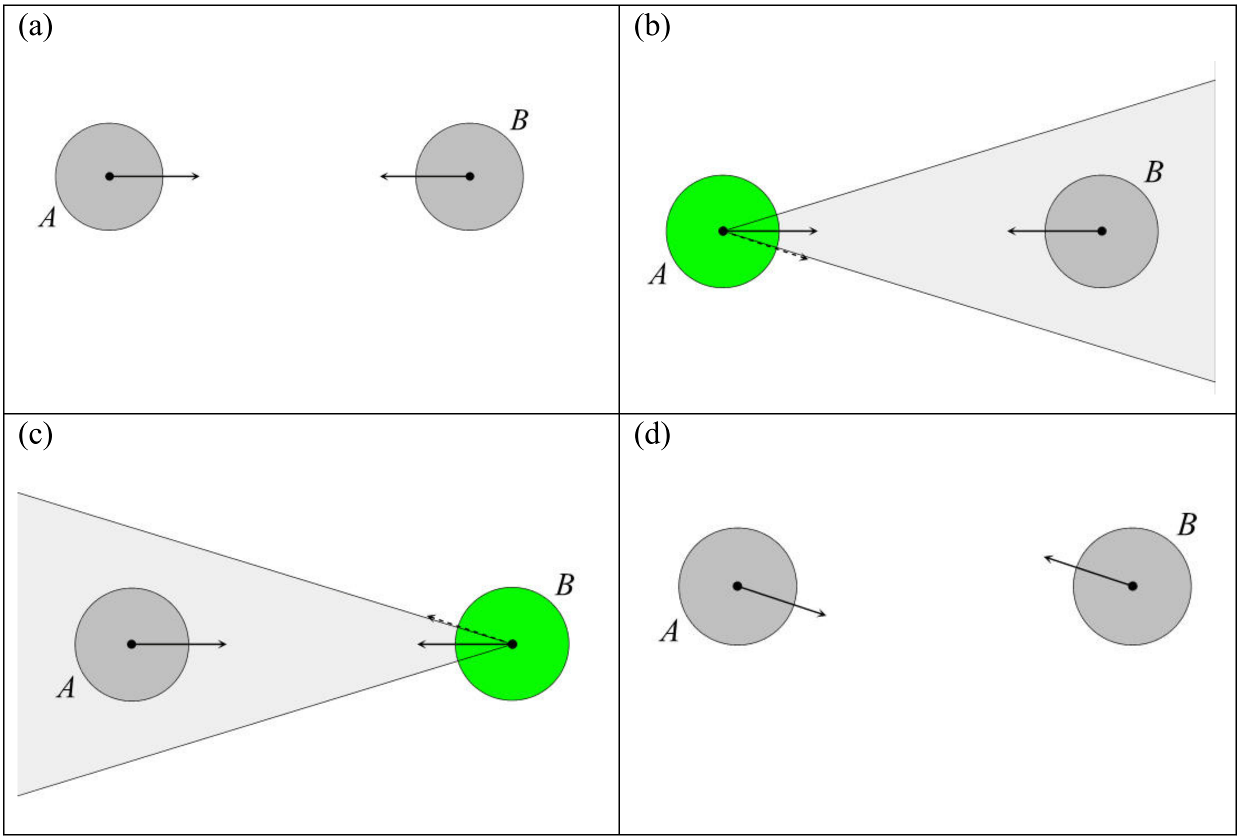


Figure 2-19 Applying the RVO approach to agents A and B moving in opposite directions

2.4 Previous Research into Crowd Movement

This section discusses and highlights important aspects of data which have been gathered in previous research. This review will provide a comprehensive guide on the various crowd characteristics which need to be considered when developing the Continuous Model presented in this thesis (see Chapter 5). However, this does not imply a direct mapping of empirical observations into model development as the source of the data and circumstances under which they have been collected needs to be taken into consideration.

2.4.1 Fruin (Fruin 1971)

Fruin's book "Pedestrian Planning and Design" is recognised and widely accepted as a seminal guide for pedestrian planning. His research has defined numerous criteria for establishing safety standards for areas where people congregate. The book deals with numerous aspects of crowd movement; namely flow of pedestrians on stairs and flat spaces, relationship between flow speed and passage width, variation of travel speed with occupant density, shape and size of occupants' bodies, pedestrian queuing and level of clothing.

Moreover, Fruin describes the level of service concept for pedestrians. This defines the relationship between flow density and speed of crowd. This is based on the assumption that as population density increases, then the ability of a person to select normal locomotion speed decreases. The level of service standard comprises 6 levels A to F, with A and F relating to the minimum and maximum congestion respectively. Fruin's approach was developed only for people circulation and therefore is not necessarily applicable for emergency situations; however it is frequently applied in emergency calculations.

Fruin also describes the body buffer zone i.e. the quantity of space occupied by one person plus the amount of free space around the body. He states that any density increase beyond the touch zone will lead to frequent, unavoidable contact between people. He also noted that there is a gap of 0.3 to 0.5 m of free space between moving people and solid walls and this gap could be dependent on the purpose of the individual's presence and type of enclosure.

In his observations, Fruin used time lapse photography to quantify movement parameters. Those time lapse studies were used to obtain the relationship between crowd density and average speed which is shown in **Figure 2-20**. Although Fruin's data regarding the population densities corresponds to a maximum of 1.80 persons/m^2, it should be noted that enclosures such as stadiums and train stations have population densities in excess of densities measured by Fruin [Still 2000]. This is because Fruin's data is based on measurements conducted on pedestrian high streets where the crowd has more avenues for dispersions therefore, resulting in lower crowd concentrations.

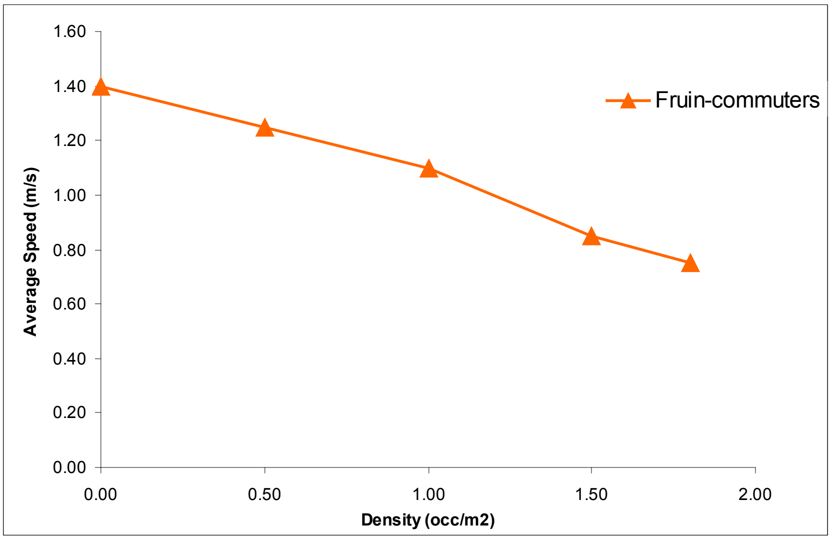


Figure 2-20 Relationship between walking speed and crowd concentration [Fruin 1971]

The crowd flow rate was calculated using Equation 2-4.

$$Crowd\ Flow\ Rate = Crowd\ Speed\ x\ Crowd\ Density \qquad \textbf{Equation 2-4}$$

The relationship between crowd density and flow rate is shown in Figure 2-21.

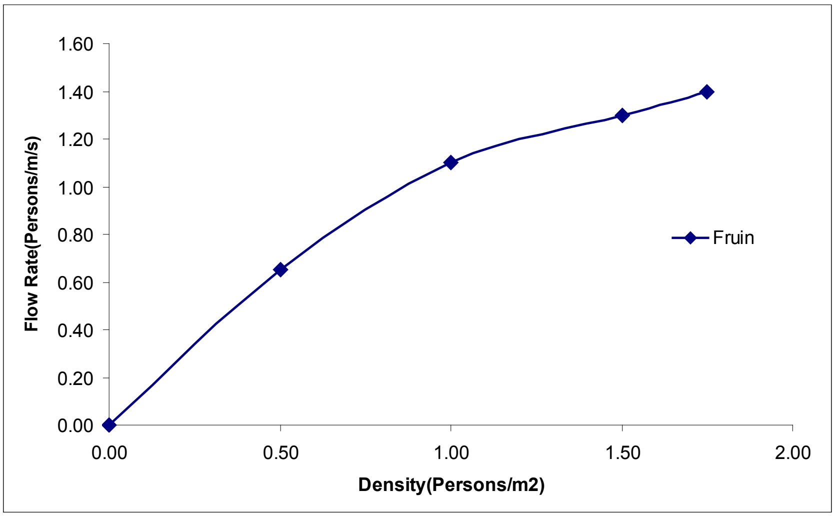


Figure 2-21 Relationship between flow rate and crowd concentration derived from Figure 2-20

2.4.2 Predtechenskii and Milinskii (Predtechenkii 1978)

The work of Predtechenskii and Milinskii was primarily concerned with the merging and combination of flows involving 7000 observations in the Soviet Union. Their method of calculation consisted of calculating the number of people in a specific area, thus obtaining the resulting density and then deriving the speed and flow rate from data points on the Speed/Density graph and Flow Rate/Density graph as shown in **Figure 2-22** and **Figure 2-23** respectively.

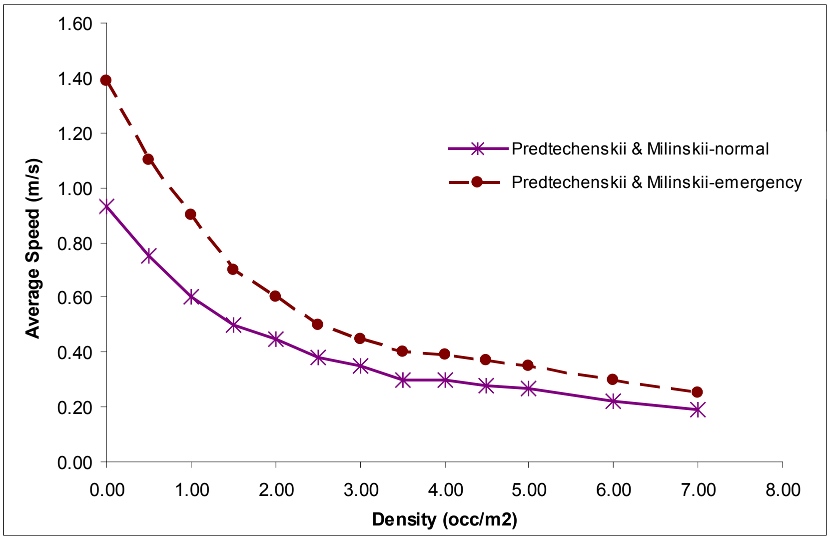


Figure 2-22 Relationship between average walking speed and density

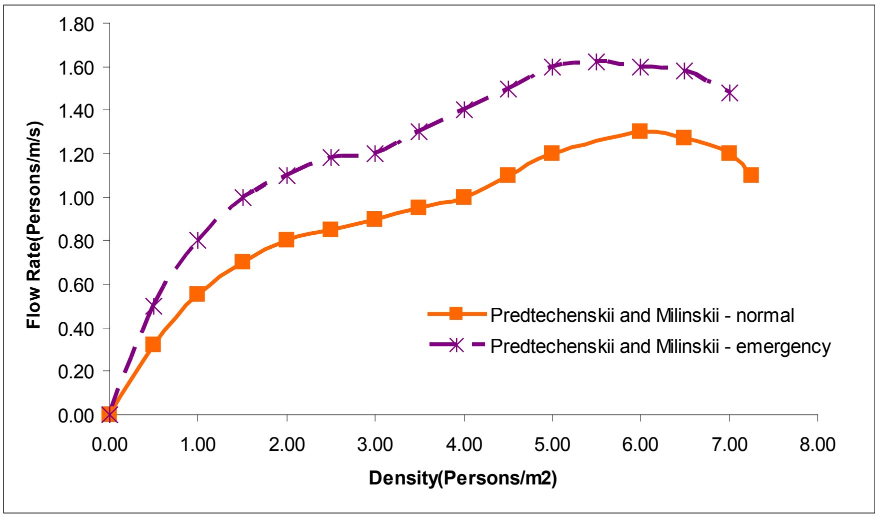


Figure 2-23 Relationship between flow rate and density

One important difference between their work and any other previous work was the way in which they defined density as being the ratio of the projected area of people in crowd and the area of the walkway surface (metres^2/ metres^2). The projected area represented the horizontal projection of the space occupied by that person on the building plan. Measuring density in this fashion allowed the investigation of the variations of the body size of the occupants depending on the season. Unlike Fruin, Predtechenskii and Milinskii did not include the edge effect in their analysis. This was possibly due to the fact that they were investigating movement of crowds at extremely high densities. The resulting crowding would cause the maintenance of the space from the wall boundaries to be less feasible. The following equations were used to assess crowd flow.

$$D = \frac{\sum f}{\partial \times \ell} \quad - (1)$$

$$Q = D \times v \times \partial \quad - (2)$$

$$q = D \times v \quad - (3)$$

$$f = \frac{\pi}{4} \times a \times c \quad - (4)$$

Where

∂ = width of traffic stream or passageway (m)

ℓ = length of flow of people (m)

f = projected area of ellipse of each person (m^2)

v = average locomotion speed of people in flow (m/min)

D = density (sum of projected areas per m^2) (m^2/m^2)

Q = traffic capacity of flow path (m^2/min)

q = intensity of movement (flow concentration) (m/min)

a = body breadth

c = body depth

(Predtechenkii 1978)

2.4.3 Hankin and Wright (Hankin 1958)

Studies were conducted for the London Transport in which two series of observational tests were carried out. The experiments were based on 200 schoolboys circulating around a circular passageway 1.3m wide and 9.1m internal diameter. The method involved measuring travel speeds at different densities and also by varying the passage width.

The second series of experiments were conducted on London Underground in which two observers mixed with the crowd and logged data regarding the crowd flow by using stopwatches. The data obtained from the observations helped to derive more speed/density and flow/density graphs. The speed was obtained by logging the time taken for one observer to walk a set distance within the crowd and the flow rate was obtained at an end location of specific width.

The data obtained from the experiments were used to derive the graphs showing the relationship between person travel speed and population density (**Figure 2-24**) and unit flow rate and population density (**Figure 2-25**).

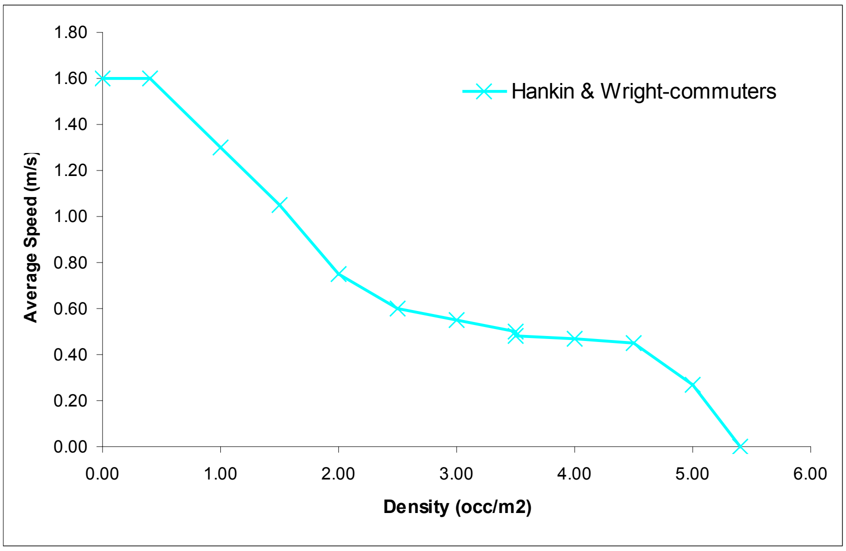


Figure 2-24 Relationship between average walking speed and density

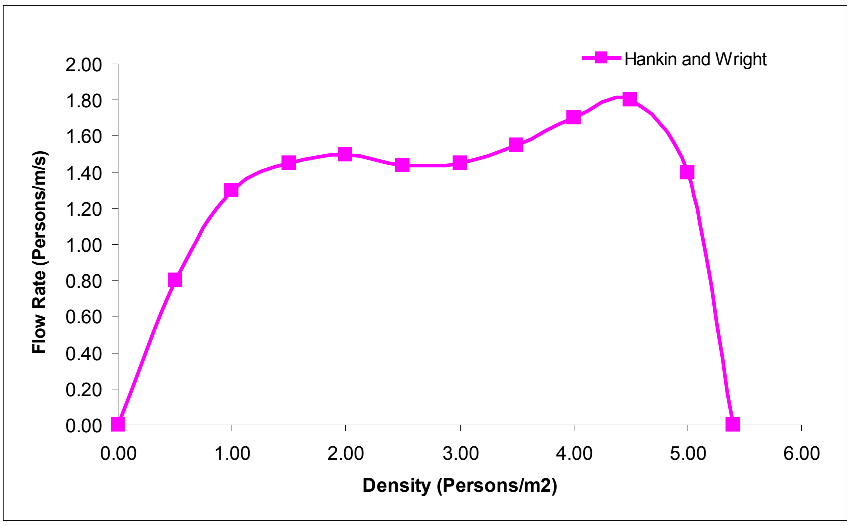


Figure 2-25 Relationship between flow rate and density

2.4.4 Ando et al (Ando 1988)

The research was concerned with crowd movement in common direction in densely populated railway stations. The objective of their work was to improve the flow of passengers through railway stations at peak times. During their study, they observed that the crowd would almost come to a halt at 4 persons/ m^2; however restricted movement was possible even above this density. Under extreme and dangerous conditions, really high densities up to 15 persons/m^2 were seen to develop. Moreover, Ando et al also identified from the data collected that occupant travel speeds were dependent on the age and gender, whereby males have higher travel speeds than females and that peak travel speeds for both genders were observed for young adults of around 20 years of age. The data generated from the experiments were used to generate the graphs as shown in **Figure 2-26** and **Figure 2-27**.

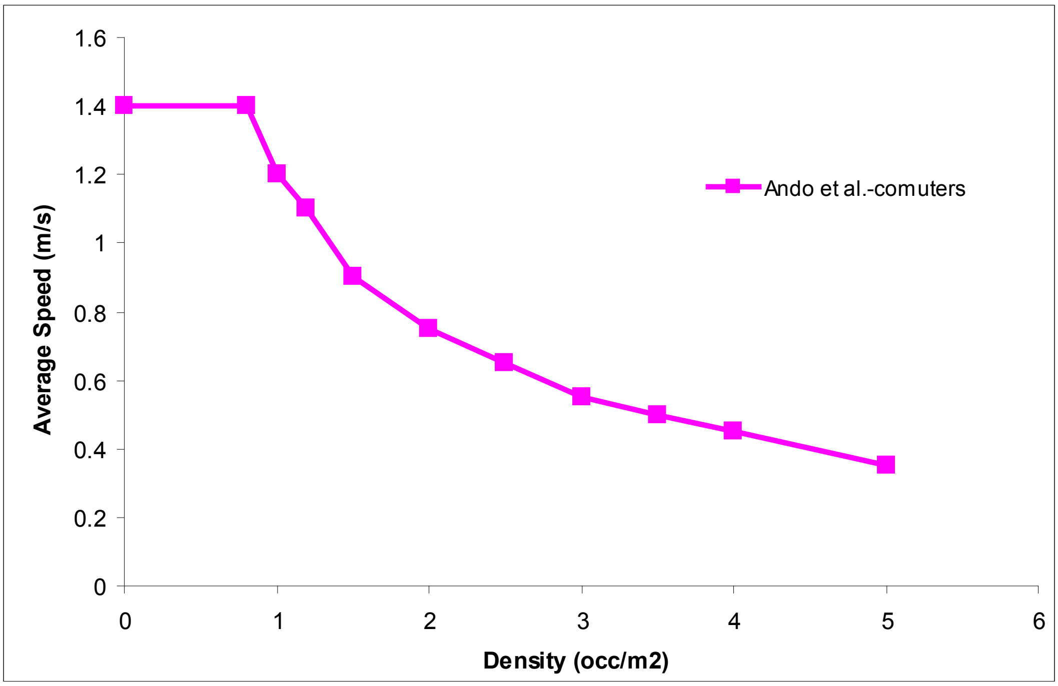


Figure 2-26 Relationship between average walking speed and density

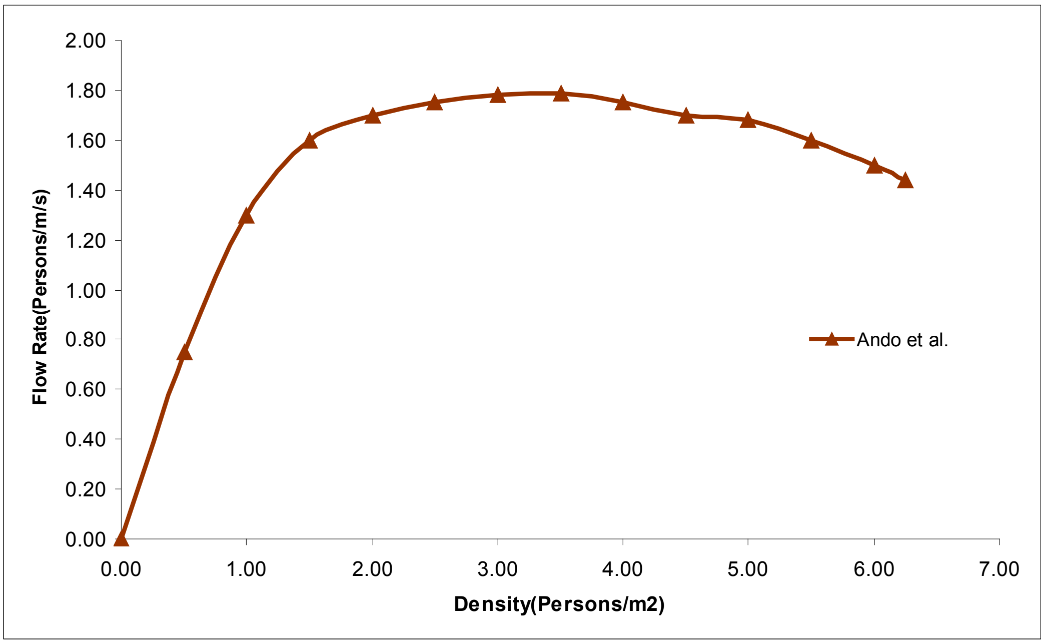


Figure 2-27 Relationship between flow rate and density

2.4.5 Peschl (Thompson 1994)

Peschl's investigation was on the capacity of doors during simulated emergency conditions. As part of the research, experiments were conducted involving volunteers who were asked to move towards an opening and force their way into it. It was observed that at high localised densities, many volunteers attempting to go through the door simultaneously lead to the formation of arches around the opening as shown in **Figure 2-28**. This would cause a drastic reduction in the flow rate through the opening.

Moreover, there was considerable pressure exerted on the people who were nearer to the opening. Upon passing through the opening, people stumbled as a result of the force applied by others behind them. It was also noted that the size of the arch was dynamic and that it was dependent on the size of the opening, such that the wider the doorway the smaller the probability of arch formation. The formation of arches not only takes place at exits but also along corridors. For instance, the significant increase in population density as a result of a corridor width being constricted would result in arch formation around the constricted region.

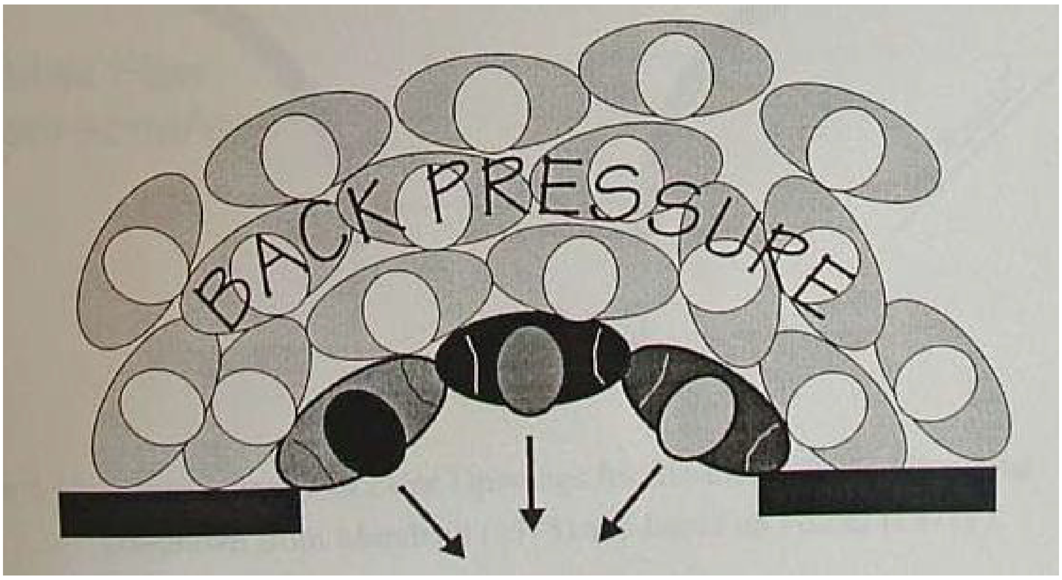


Figure 2-28 Formation of body arches around exits (Adapted from Thompson 1994)

2.4.6 Other Research Work

Weidmann [Weidmann 1993] compiled an extensive literature survey comprising of 25 different investigations (based on field studies and experimental research) to determine the relationship between crowd speed and density. The underlying data captured as part of the survey dealt with both uni and bidirectional crowd flows in open and closed boundary conditions. The data from the survey was used to generate the fundamental diagram in **Figure 2-29**.

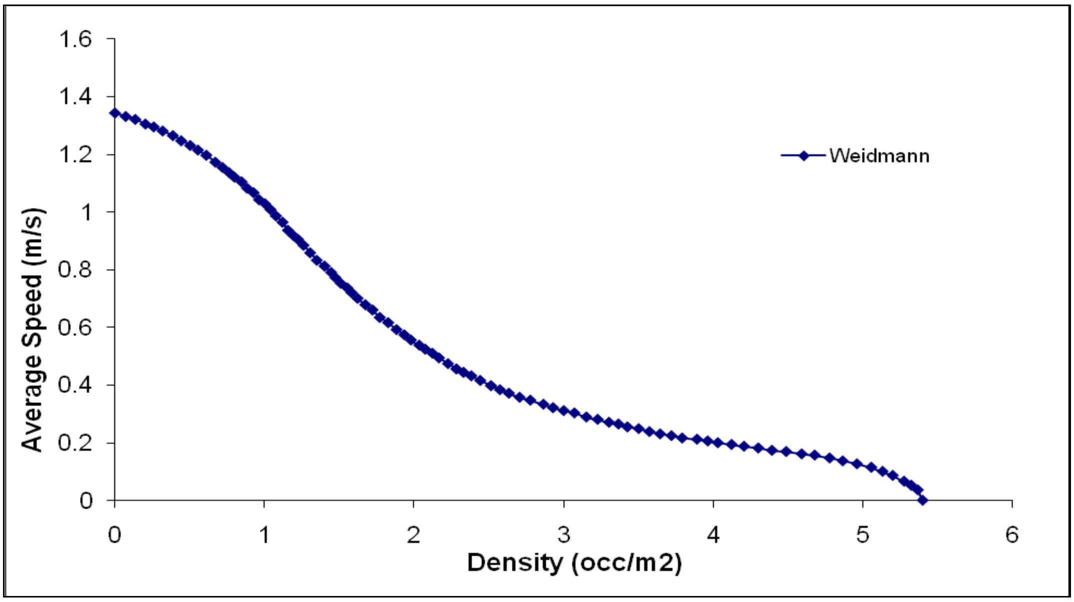


Figure 2-29 Relationship between crowd travel speed and crowd density

In [Seyfried 2009], the authors describe experiments which were performed under laboratory conditions in order to study the unidirectional pedestrian flow through bottlenecks. In this study, the variation of quantities (individual travel speeds, density, and individual time gaps) in bottlenecks of different widths was investigated. The data was based on video recordings of cameras above the bottleneck and the flow was calculated as the ratio of the number of people in the flow and the difference in time of crossing of the first and last pedestrian. Similar studies involving the measurement of flow rates at varying widths have been shown in [Kretz 2006] and [Nagai 2006]. Because of the empirical evidence associated with these studies, the results were considered appropriate for validating the flow characteristics predicted by the buildingEXODUS model (Chapter 4) and the bEX-Continuous model (Chapter 5).

2.5 Summary

The review conducted in Section 2.4 has elicited some key findings which are applicable towards the implementation of the Continuous Model. The physical dimension of individuals in a crowd is really important when considering the physical nature of crowd movement. In studies conducted by Predtechenskii and Milinskii, it was observed that maximum flow rate of a crowd comprised of people wearing light summer clothing was 1.70 p/m/s. However, the maximum flow rate dropped to 1.36 p/m/s for people wearing thick winter clothing. Fruin's observations have shown the existence of an amount of free space between the edge of a crowd and an adjacent wall. However, this gap was not observed in the Predtechenskii and Milinskii studies where the crowd densities were extremely high. This highlights that as the crowd density increases, the edge effect becomes negligible. The studies conducted by all the researchers have shown the effect of the invasion of personal space on the travel speed, and that the speed decreases even prior to the occurrence of any bodily contact. In addition, the studies carried out by Peschl have provided qualitative data in terms of the formation of arches around exits.

Furthermore, the review of evacuation models demonstrated some current methodologies used for modelling the physical space of enclosures in egress models. This review has provided a discussion on some hybrid techniques used in evacuation models and attempted to formulate a better understanding of the term 'hybrid' in relation to egress tools. Nevertheless, none of the existing evacuation models in the literature were seen to utilise a combination of all the spatial representation approaches for modelling the space in which the agents navigate and interact. This highlights the novelty of the technology being developed as part of this dissertation.

Moreover, this review has evaluated the respective capabilities of the various spatial types. This provides an indication on the potential advantages that could be harnessed by encompassing the benefits of the spatial types within a single integrated model. In addition, the outcome of this review has lead to the elicitation of a list of recommendations as regards the applicability of each spatial type towards modelling different types of scenarios and terrains. Moreover, these guidelines have laid the foundation towards the development of the HSD approach in the sense that the recommendations can be used to guide the choice of using fine nodes, coarse nodes or continuous regions for defining different areas within enclosures.

Also, this chapter has provided a review of the techniques used for implementing motion behaviours in agent based models utilising continuous space. The component oriented architecture used for the behavioural model in the Reynold's model is particularly interesting as it enables the development of simple behaviours followed by their incremental integration within the system. In addition, an interesting feature of the Helbing's model was its ability to demonstrate self-organisation phenomena e.g. the formation of lanes. Nonetheless, none of these approaches incorporate behaviours which can be used for simulating the transition of agents from one spatial type to another. This is also due to the existing agent based models not utilising a hybrid spatial representation technology as yet.

Chapter 3 - Proposed Architecture for Hybrid Spatial Discretisation (HSD)

One of the main aims of this dissertation is the development of a technology which can utilise all the three spatial representation methods namely, fine nodes, coarse nodes and continuous regions, for representing the physical space of enclosures within a single integrated software tool. This chapter describes a proposed architecture for this novel approach called the Hybrid Spatial Discretisation (HSD) and discusses the core technologies which are required for the development of the model. Moreover, it evaluates the strategies which could be employed to facilitate the interoperability of the various integral components within the model.

3.1 Hybrid Spatial Discretisation

The HSD approach involves the mixing of macroscopic and microscopic modelling methodologies and the various components of this approach are as shown in **Figure 3-1**.

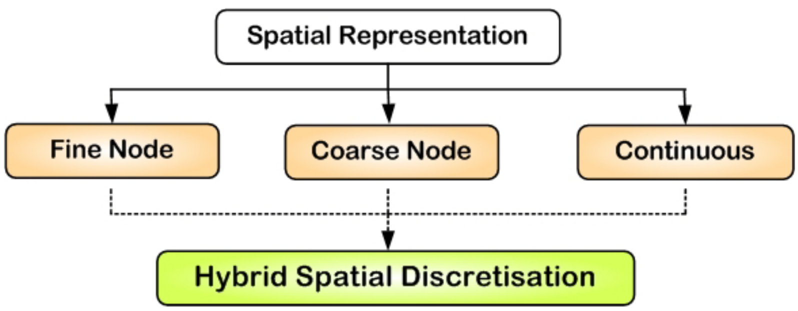


Figure 3-1 Definition of Hybrid Spatial Discretisation

The development of the HSD entails the implementation of various spatial representation, behavioural and movement algorithms, all of which require a computational shell in order to operate. However, the development of a new computational shell would be impractical. In this respect, the use of an existing shell was favoured.

3.2 Choice of Computational Shell

The buildingEXODUS software was selected as a development and deployment platform for its availability in-house and also for technical reasons. buildingEXODUS is an egress tool which utilises fine nodal networks for representing physical space within enclosures. This model has been modified to allow plug-in modules to be included into the core software using a component oriented engineering approach and is identified as the buildingEXODUS-Hybrid or bEX-H. This sophisticated architecture provides a platform whereby new functionalities can be independently developed, tested and incorporated into the model as required.

In addition, EXODUS provides both graphical (2-D and 3-D) and textual output. The 2-D output is in the form of a graphical user interface which illustrates the various dynamics of the evacuation simulation. It allows the user to interrogate the occupants and events during the progression of the simulation. Also, a data output file containing all the relevant information regarding the evacuation processes is generated for each simulation. In order to enhance the interpretability of the results, EXODUS also incorporates a post-processing 3 dimensional visualisation software tool called vrEXODUS [Galea 2004a].

Furthermore, the EXODUS software has undergone various forms of validation namely Component Testing, Functional Validation, Qualitative and Quantitative Validation [Galea 1997a]. These include comparison of model predictions with past experimental data and comparing the nature of human behaviour with behavioural simulations of the occupants in the model.

The key features of the buildingEXODUS software and the capabilities which make it suitable as an integral component of the HSD are described in greater detail in Chapter 4.

3.3 Choice of Programming Language

The buildingEXODUS computational shell supports the integration of new functionalities in the form of modules and therefore, the new software artefacts could be developed in a different programming language other than the one in which the shell was implemented. The class of language which was considered for the implementation was one which would support object orientation. This is primarily because the object oriented approach takes into account fundamental features such as modularity, encapsulation and abstraction. These characteristics are conducive towards the implementation of a high quality system, that is, a system which is usable, flexible, maintainable and reliable. As a result, the C++ programming language was selected for the implementation phase. Moreover, the other benefits of using C++ is that well designed C++ software can be memory and performance efficient, and can also allow further tuning at a lower level.

3.4 Development of Continuous Region and Coarse Network Components

The development of the HSD approach entails the implementation of two new models namely a Continuous and Coarse Model. These models will be independently developed as components which can be used as plug-in modules for the bEX-H model. This methodology will allow the newly developed models to be individually tested and also facilitate their integration into the core of the bEX-H model. An overview of the architecture of the bEX-H model is illustrated in **Figure 3-2**.

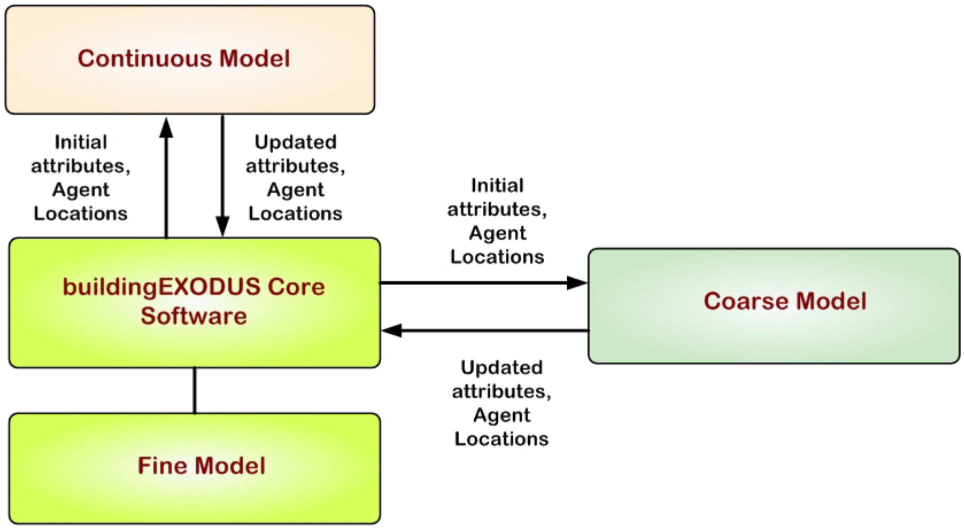


Figure 3-2 Proposed architecture for Hybrid Spatial Discretisation in bEX-H model

3.4.1 Continuous Region Component

When using a continuous approach for the discretisation of space, it is possible to take into consideration a larger number of agent attributes allowing for a wider range of agent behaviours to be modelled. The Continuous Region component will necessitate an advanced navigational system which can interface with the fine node and coarse node networks. Moreover, this component will be based on a multi-agent system whereby each agent is modelled as an autonomous agent which exhibits some forms of adaptive behaviour. The implementation of the Continuous Region Component is described in Chapter 5

3.4.2 Coarse Network Component

Using this approach, the available physical space can be segmented into partitions whereby each partition can represent a section of the geometry such as a room or corridor. Each partition is a node which is connected via arcs or links to represent doorways or other forms of connectivity in the structure. The Coarse Network component will require an advanced navigational system which can interface with the fine node network and continuous regions. The development of the Coarse Network component is described in Chapter 6.

3.5 Development of Transition Regions

The mixing of the macroscopic and microscopic modelling methodologies in the HSD approach presents several challenges. The agents navigating in an enclosure modelled using the HSD approach will at some point in their trajectories transit from one type of spatial representation to another. Therefore, effective mechanisms on how people move between coarse, fine node and continuous regions are indispensable within the bEX-H model. Moreover, in order to be able to fully exploit the capabilities of the HSD approach, all six possible interface transition regions are required to be modelled. These transition regions are namely: Coarse Node ↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous. The Transition Regions are described in greater detail in Chapter 7.

3.6 Summary

This chapter provides a blueprint of the architecture of the HSD approach and puts into perspective the core technologies which are indispensable for implementing the HSD approach in the bEX-H model. The architectural design is vital as it can be used to identify any potential problems at early stages in the development process.

Chapter 4 – An Overview of buildingEXODUS Evacuation Model

The primary aim of this thesis is the development of a hybrid model capable of representing the physical space of enclosures using a combination of fine nodes, coarse nodes and continuous regions. The previous chapter discussed the reasons for choosing the buildingEXODUS evacuation model as the core of the hybrid model (bEX-Hybrid). buildingEXODUS has been frequently described in other publications and will not be discussed exhaustively in this dissertation [Galea 1994] [Galea 1997a] [Galea 1997b] [Galea 1997c] [Galea 2000] [Galea 2001] [Galea 2002] [Galea 2003] [Galea 2004c] [Grandison 2007] [Galea 2010]. However, an overview of some of the mechanics and the key features of the model will be presented in this chapter.

In addition, this chapter includes a detailed evaluation of the buildingEXODUS software through the investigation of some fundamental relationships namely; variation of occupant travel speeds with occupant density, variation of flow rate through corridors with occupant density and variation of flow rate with exit width. The results from these tests were used as input in other objectives of this research for instance, the calibration of the Continuous Model.

4.1 Model overview

EXODUS is a suite of software tools designed to simulate the evacuation and movement of large numbers of people from a variety of enclosures. It is developed by the Fire Safety Engineering Group at the University of Greenwich, UK. The EXODUS family of evacuation models consists of buildingEXODUS [Galea 1994, Galea 1997a, Galea 1997b, Galea 2004a, Galea 2004b, Galea 2010], maritimeEXODUS [Boxall 2005, Galea 2003, Galea 2004c], airEXODUS [Galea 1997c, Galea 2002] and railEXODUS [Galea 2001]. buildingEXODUS is designed to for applications in the built environment for instance high rise buildings, airport terminals, rail stations, hospitals, supermarkets and schools. It can be used to assess the evacuation efficiencies of different types of structures and demonstrate compliance with building codes. maritimeEXODUS is designed for applications in the marine industry. It can be used to simulate the circulation and evacuation of various types of ships including passenger ships and warships. It can also simulate the use of life saving devices such as lifeboats during ship abandonment. airEXODUS is designed for applications in the aircraft industry for instance aircraft design, accident investigation and compliance with 90 second certification requirements. railEXODUS is designed to simulate evacuation from rail carriages. This model possesses rail specific human performance data.

EXODUS views the occupants from an individual perspective and addresses 3 types of interactions namely people-people (e.g. overtaking), people-structure (e.g. behaviour on staircases) and people-environment interactions (e.g. physiological response to narcotic gases). The EXODUS software has been written in C++ using Object Oriented techniques and utilises a rule based approach to control the simulation. In this respect, the movement and behaviour of the occupants is defined by a series of heuristics or rules which are grouped into 6 interacting sub core models namely OCCUPANT, MOVEMENT, BEHAVIOUR, TOXICITY, HAZARD and ENCLOSURE, as shown in **Figure 4-1**. The OCCUPANT, MOVEMENT, BEHAVIOUR AND ENCLOSURE sub-models are most applicable to the purpose of this dissertation and therefore are discussed in more detail.

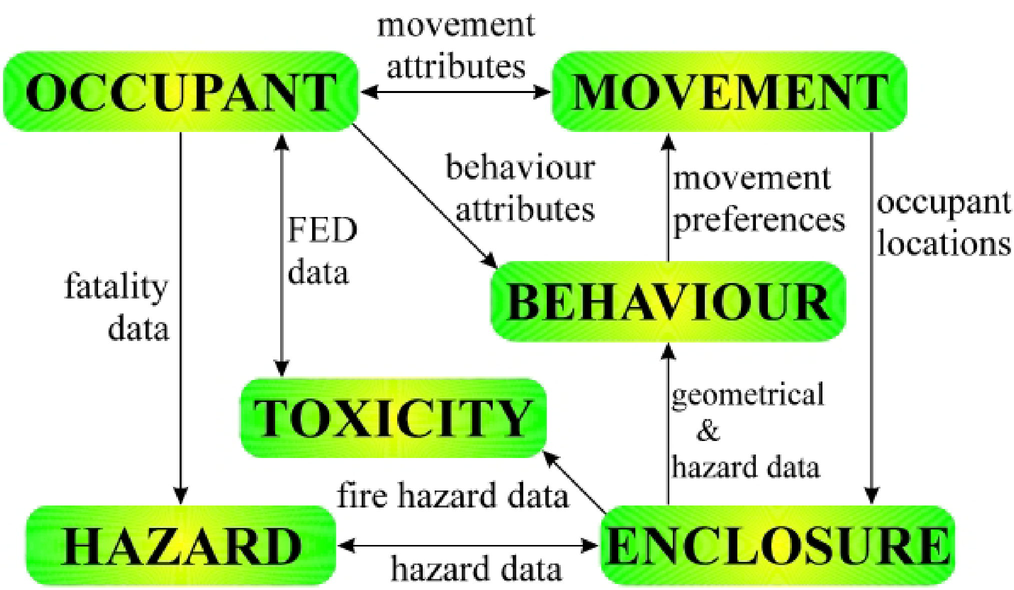


Figure 4-1Interacting sub-models in EXODUS

4.1.1 Enclosure sub-model

The Enclosure sub-model defines the geometry of the enclosure and the representation of the internal space. Within buildingEXODUS, the entire internal space can be represented using a two-dimensional fine network of nodes which are interconnected by a series of arcs. Each node has a dimension of 0.5 metres by 0.5 metres and represents a region of space which can be occupied by a single occupant. **Figure 4-2** illustrates the spatial representation of a building using a mesh of nodes and arcs. During a simulation, the occupants travel from node to node along the arcs. Each node has a set of attributes which define the terrain type e.g. free-space, internal exits, external exits, stairs and seats. These attributes influence the travel speeds and behaviours of occupants passing over them. Moreover, the arcs interconnecting the nodes have two attributes namely Length and Obstacle. The Length represents the actual physical distance between nodes and the Obstacle is indicative of the degree of difficulty of passing over nodes.

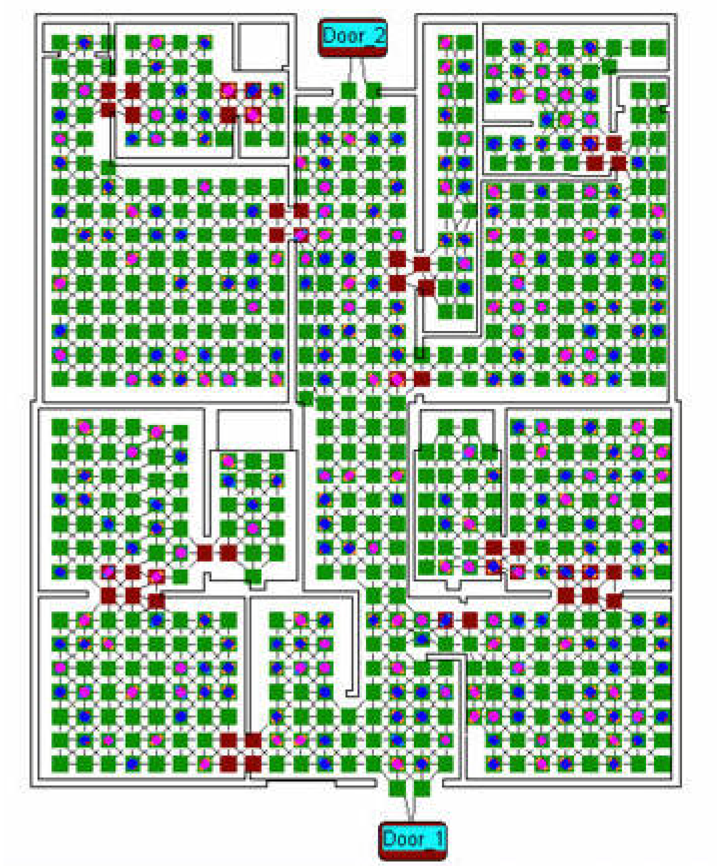


Figure 4-2 Outline of a building showing nodes and arcs

A key data structure within the buildingEXODUS fine node model is the Potential Map which is generated automatically in the initialisation phase prior to the running of the simulation. During the Potential Map generation, each node is assigned a potential value which is a measure of the node's distance from the nearest exit. When the Potential Map option is enabled, the occupants attempt to move to nodes of lower potential until they reach their final locations. During the initialisation phase, a Distance Map is also generated which allows each node to keep track of its relative distances from each of the exits in the geometry. The Distance Map is used to allow occupants to navigate to pre-assigned target exits rather than the nearest ones. **Figure 4-3** illustrates the Potential Map for the building shown in **Figure 4-2** whereby the arrows are directed towards nodes of lower potential.

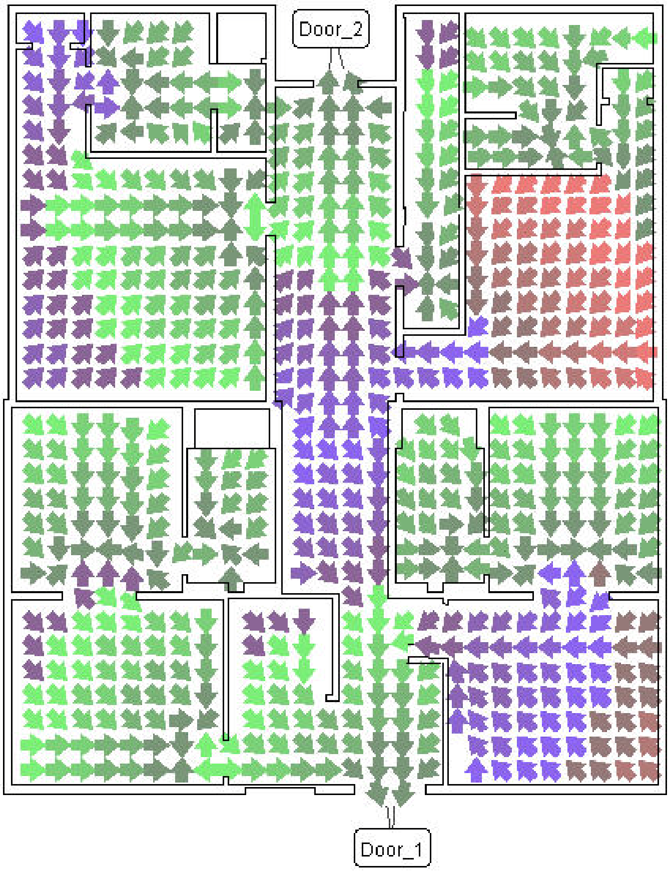


Figure 4-3 Potential map for the building shown in Figure 4-2

4.1.2 Occupant sub-model

The Occupant sub-model describes an individual as a collection of attributes. These individual attributes influence the behaviour of the occupants during the simulation and have been determined according to data from empirical and experimental sources. Some of these attributes remain constant throughout the entirety of the simulation, while others change dynamically as a result of inputs from the other sub-models. In the model, the attributes are categorised as physical, psychological, experiential and hazardous effects.

4.1.2.1 Physical attributes

The physical attributes are used to distinguish each individual and govern the assignment of other attributes as well. Examples of physical attributes include age, gender, maximum travel speeds and mobility. The mobility attribute is used in conjunction with travel speed to represent movement disabilities.

4.1.2.2 Psychological attributes

The psychological attributes are used to determine the occupant's response to situations. Examples include response time, patience and drive. The response time determines the time required by the agents before they start to evacuate. The drive attribute is used to resolve conflicts when agents are competing for the same nodes. Patience is a measure of how long the occupants are prepared to wait before seeking alternative actions.

4.1.2.3 Experiential attributes

Experiential attributes are used to keep track of the various experiences of the agents throughout the simulation. Examples of experiential attributes include Personal Elapsed Time (time spent by occupant in simulation), distance travelled and distance remaining to exits.

4.1.2.4 Hazard Effects Attributes

These attributes are used to measure the overall effect of narcotic gases, the effects of temperature and the volume of gases breathed.

4.1.3 Movement sub-model

The Movement sub-model controls the physical movement of individual occupants from their current position to the most favourable neighbouring location until they reach their final destination in the enclosure. If an occupant cannot find a favourable location, it exhibits waiting behaviour. Moreover, the movement model controls the speed at which the occupants navigate, which is dependent on the type of motion and the terrain type. There are six types of movement speeds defined in the model namely Fast Walk, Walk, Leap, Crawl, Stairs-Up and Stairs Down. These represent the maximum unimpeded travel speed of the occupants under a variety of conditions.

4.1.4 Behaviour sub-model

The Behaviour sub-model determines an individual's response to the current situation based on the personal attributes. This model functions on 2 levels: local and global, local would imply determining response to the local situation of occupant, and global would be the overall strategies employed by the occupant for example, exit via the nearest serviceable exit or via most familiar exit. The global and local behaviour strategies can be triggered using various methods.

Global Behaviour can be implemented using 3 methods. The default method can be used to direct occupants to their nearest serviceable exit. This is achieved through the use of node potentials whereby the occupants attempt to move onto nodes of lower potentials until they reach their destination. The second method is used when guiding occupants towards an assigned exit, which could be representative of the exit to which the occupants are most familiar with. This method enables the occupants to travel towards user defined target exits irrespective of the underlying potential map. The third method involves the occupants relying on their personal knowledge of the enclosure whereby the occupants attempt to move to their nearest exit which is selected from their own list of familiar exits.

Local Behaviour is strongly influenced by the individual attributes of the occupants coupled with behavioural rules for instance conflict resolution. There are 2 regimes in which the local behaviour operates namely Normal and Extreme behaviour. In normal behaviour, the occupants will attempt to lower their potential and not increase it whereas in extreme behaviour, the occupants can accept to move to a higher potential resulting in alternative and potentially indirect routes. The Behaviour model also includes strategies to resolve conflicts which could arise when multiple occupants are attempting to occupy the same location in space. The Conflict Resolution strategies entail the use of the occupants' Drive attributes to determine which of the occupants should be given the priority to step into the next favourable node. Occupants involved in conflict resolution also incur a time penalty which is representative of the time lost during the interactions. Furthermore, the model includes behaviours which can allow the occupants to adapt their trajectories in relation to the population densities around them. The occupants choose their next favourable location not just based on the distance to the target but also based on the population densities. Thus, agents prefer nodal locations with lower population densities. Moreover, the occupants can also adapt their exit selection based on congestion levels around their target exits. This behaviour allows agent to redirect from their target external exit towards a more favourable one. This behaviour involves the use of signage to redirect occupants towards the exits to which the signs correspond.

The approach towards modelling the behaviour of the occupants from a local and global perspective is particularly interesting. This approach provides a platform which allows the different levels of behaviours to be independently tested and combined in order to provide a more naturalistic representation of human behaviours. In this respect, this behavioural modelling approach has been adapted in the implementation of the Continuous Model (Chapter 5).

4.1.5 Hazard

The Hazard sub-model controls the atmospheric and physical environment and distributes pre- determined fire hazards such as heat, smoke and toxic products. It also controls the opening, closing and the availability of exits.

4.1.6 Toxicity

The Toxicity sub-model determines the effects on an individual exposed to toxic products distributed by the Hazard sub-model. These effects are passed on to the Behaviour sub-model which, in turn, influences the movement of the occupants.

4.2 Evaluation of buildingEXODUS - Investigating fundamental aspects of crowd flow

In the light of the literature described in Chapter 2, some fundamental relationships related to crowd flow were identified. These are namely the relationship between travel speed and population density, relationship between flow rate and density and relationship between flow rate and corridor width. This section describes a series of tests for validating the buildingEXODUS model. These tests entail the comparison of buildingEXODUS predictions against empirical data collated from previous studies (reviewed in Chapter 2).

4.2.1 Relationship between people travel speed and population density

The term crowd density is commonly defined as the number of persons present per unit area of space in a specified region. Based on the literature in Chapter 2, it has been noted that many researchers for instance Fruin, Predtechenskii and Milinskii, Ando et al. and Hankin and Wright have observed the influence of population density and travel speeds. According to field experiments, it has been observed that a higher crowd density leads to an increasing number of interactions in between individuals, and therefore causing the individual walking speeds to decrease. The combined effect of this reduction in individual walking speeds leads to a slower crowd.

4.2.1.1 Experimental set up to investigate relationship between people travel speed and density

The tests were performed on a corridor of length 50 m and width 2.5 m. The corridor width is chosen such that passing is possible. A group of agents (Group A) was generated in a section of the corridor as shown in **Figure 4-4**. In order to investigate the relationship between population density and average occupant travel speeds, the number of occupants was increased in the tests and the corresponding average speed attained was obtained by measuring the distance covered by an occupant in the centre of the group during a set measurement time. Moreover, an important consideration during the set up of the simulations was to ensure that the density of the group of agents must remain uniform or almost uniform during the measurement period. Therefore, in order to prevent the agents in Group A from dispersing as soon as the simulation was started, two additional groups of agents (Group B and Group C), each containing the same number of occupants as Group A, were generated in front of Group A. This set up ensured that the target occupant in Group A was exposed to the same constant density during the measurement time period.

The set up in buildingEXODUS was as follows:

- To ensure that the results were comparable against empirical data, no external influences, for instance, flow rate limitations or reduction in travel speeds were imposed on the exit. Flow Rate at exit was set to Free-Flow [Galea 2004a].

- A population in buildingEXODUS is comprised of individual occupants whereby each occupant has a set of attributes used to define and track each individual throughout the simulation [Galea 2004a]. For the tests, the population was generated manually as some of the attributes such as travel speeds and response times needed to be defined specifically. The default values [Galea 2004a] for the attributes of the occupants are as shown in **Table 4-1**.

Table 4-1 Evacuee profile attribute default values in buildingEXODUS [Galea 2004a]

Category	Attribute	Unit	Default Value	
			Min	**Max**
Physical	Mobility	-	1.0	1.0
	Age	Years	20	60
	Weight	Kg	50	183
	Gender	-	Male	Male
	Agility	-	3	7
	Height	M	1.54	1.83
Psychological	Response Time	s	0	30
	Patience	s	1	5
	Drive: [female] [male]	-	1 5	10 15
Travel Speeds	Fast Walk	m/s	0.80	1.5
	Walk	m/s	0.72	1.35
	Crawl	m/s	0.16	0.30
	Leap	m/s	0.64	1.20
	Stair-Up	m/s	*	*
	Stair-Down	m/s	*	*
Experiential	Target Exit	-	Nearest Exit	
	Familiarity		Nearest Exit	
	Itinerary		Null	

- The initial travel speed of the occupants was set to 1.4 m/s. This is because most of the curves based on empirical data contained a data point which showed that the initial travel speed of occupants was 1.4 m/s when the population density was 0 persons per m^2. In this respect, the initial travel speeds of all group members were set to a common value.

- To ensure that the occupants would start to move without any delay, the response times of all the occupants was set to 0 seconds.

- buildingEXODUS has the capability of implementing two behavioural regimes: NORMAL and EXTREME behaviour [Galea 2004a]. Under NORMAL behaviour, occupants will patiently queue if they cannot achieve their immediate goal or make their desired move. Under the EXTREME regime, occupants will wait for a period of time which is equal to their PATIENCE, which is user defined, before taking actions which lead to their jostling around or recommitting to another course of action. Since the empirical data under analysis comprised of normal locomotion of occupants as opposed to emergency movement, the behaviour of the occupants in the tests was set to NORMAL.

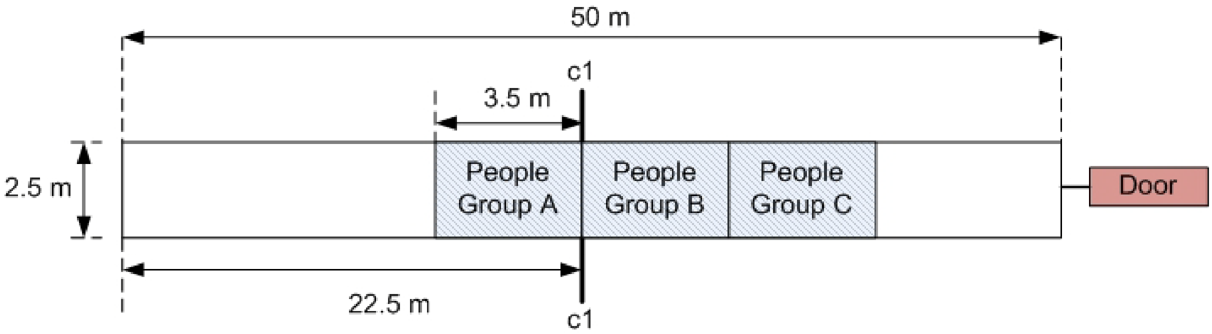


Figure 4-4 Set up in buildingEXODUS

4.2.1.2 Measurement of occupant travel speed and density

A fixed marker c1 was generated in the corridor as shown in **Figure 4-4**. The population group was generated in such a way that the front of the group was aligned with the marker. Moreover, it was ensured that a target individual in the centre of the group had sufficient people around itself to impact its travel speed. Once a target individual was identified, the travel speed of that agent was calculated using **Equation 4-1**. The time interval used in the speed analysis was calculated by the difference in time between the front of the group passing the fixed marker and the last agent in the group passing the same fixed marker. The distance travelled was determined by the distance between the target individuals' starting locations and locations they reach at the end of the time interval.

$$\textit{Average Speed of occupant } (m/s) = \frac{\textit{Distance travelled by target individual } (m)}{\textit{Time Taken } (s)} \quad \textbf{Equation 4-1}$$

Furthermore, during the course of these tests, it was found that the choice of the methodology for measuring the population density could have a significant impact on the accuracy of results. For instance, the initial set up for these tests consisted of generating the population across the entire corridor and measuring the time taken for the occupants to travel a set distance between two markers. However, results from these simulations showed that travel speeds of the occupants were unreasonably high. Upon further analysis, it was highlighted that this was due to the fact that the average population density in the entire length of the corridor was not indicative of the local density to which the individual occupants were exposed to. In this respect, the localised density measurement methodology shown in **Figure 4-4** was favoured. This measurement methodology ensured that the travel speeds of the target individuals were impacted by the presence of other occupants in their immediate vicinity. The population density was calculated as shown in **Equation 4-2**. The number of occupants in the segment was gradually increased and the corresponding travel speeds for target individuals was determined.

$$Population\ Density\ (persons/m2) = \frac{No\ of\ people\ in\ segment\ (persons)}{Area\ of\ segment\ (m2)}$$ **Equation 4-2**

The results are as shown in **Figure 4-5**. The buildingEXODUS curve is plotted along with the graphs relating to empirical data. These graphs were extracted from P.A.Thompson's thesis [Thompson 1994]. The Weidmann curve was extracted from [Weidmann 2010].

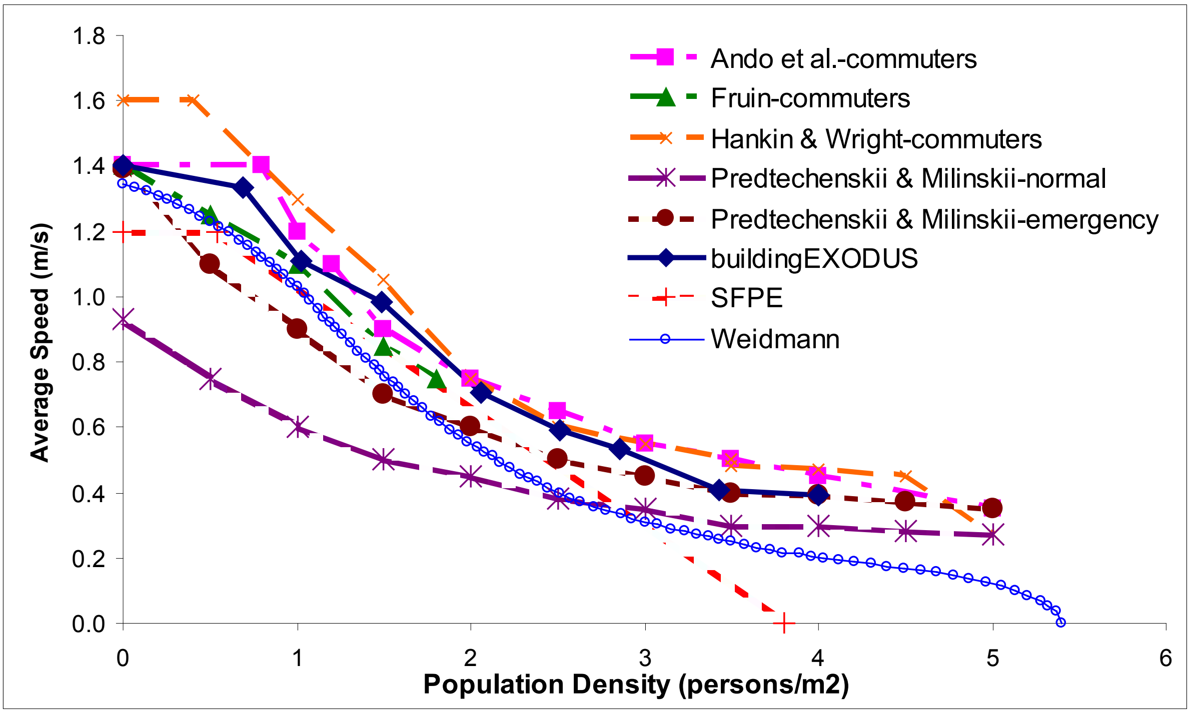


Figure 4-5 Relationship between crowd travel speed and crowd density.

Since the curves in **Figure 4-5** pertaining to the different sets of empirical data illustrate different initial travel speeds for zero population density, the curves were normalised for better comparability of results. For every curve, each of the data points was normalised using **Equation 4-3**. The normalised curves are as illustrated in **Figure 4-6**.

$$\text{Normalised value of travel speed} = \frac{\text{travel speed}}{\text{travel speed at zero density}} \quad \textbf{Equation 4-3}$$

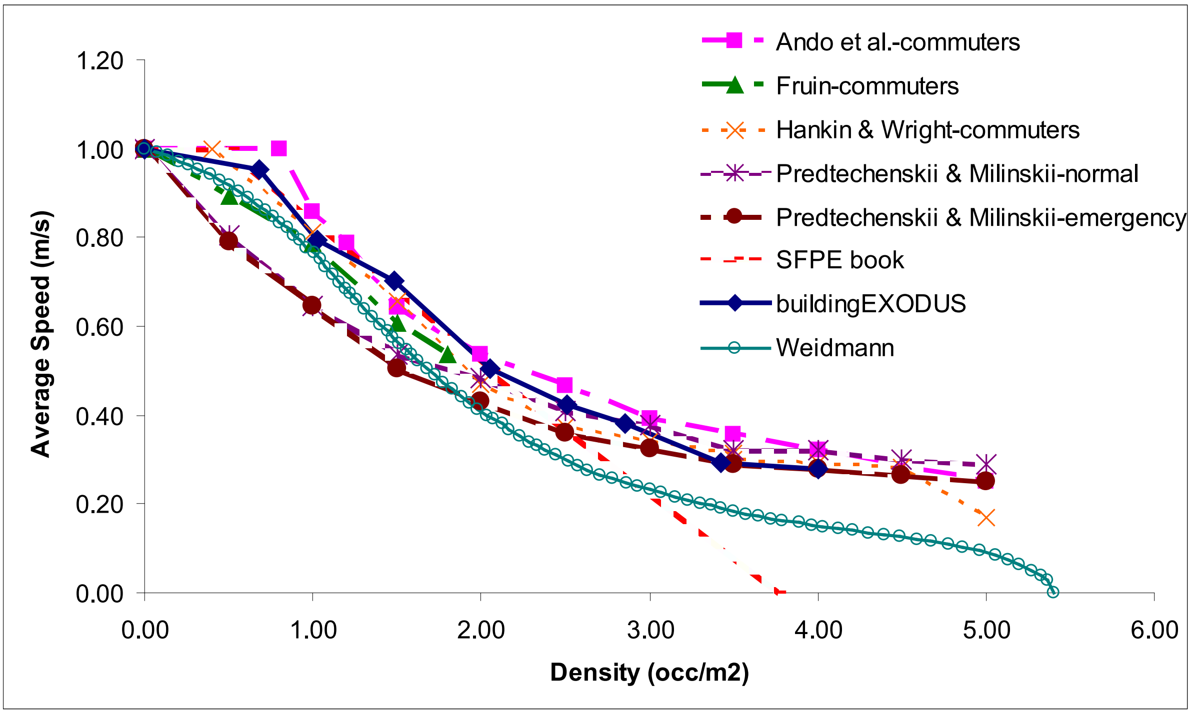


Figure 4-6 Relationship between travel speed and crowd density (Normalised)

From **Figure 4-6**, it can be noted that buildingEXODUS results are in good agreement with empirical data. However, the Fruin curve does not contain data points for densities above 1.8 persons per square metre which inhibits comparison with the buildingEXODUS curve at those higher densities.

At lower densities less than approximately 1 person per square metre, buildingEXODUS results are in agreement with the curves from Hankin and Wright and Ando et al in the sense that, at these low densities, the decrease in buildingEXODUS occupant travel speeds is not significant. Above these densities, the travel speeds pertaining to buildingEXODUS, Hankin and Wright and Ando et al curves decrease significantly with increase in population density. In addition, at higher densities above 2.5 persons per square metre, the buildingEXODUS curve shows similar trends with the Predtechenskii and Milinskii, Hankin and Wright and Ando et al curves.

The results indicate that the buildingEXODUS model adequately represents the speed-density relationship under the conditions examined.

4.2.2 Relationship between Unit flow rate and population density

The term Unit Flow Rate for an exit is a measure of the number of occupants per metre of exit which can pass through the exit per second (occupants/metre/second). The Flow Rate through the exit (occupants/second) is calculated by multiplying the exit width by the Unit Flow Rate [Galea 2004a].

In order to investigate the relationship between unit flow rate and population density, the set up was as shown in **Figure 4-4** was used. The measurement of flow rates was achieved by using an important feature in buildingEXODUS called **Census Lines** [Galea 2004a]. These are designed to enable the retrieval of information relating to important sections of the geometry without interfering with the behaviour of the occupants. In order to capture flow rate information, a census line c1 was generated in the middle section of the corridor as shown in **Figure 4-4**. The set up in buildingEXODUS was as follows:

- There were no flow rate restrictions imposed on the exit i.e., the exit was set to Free Flow conditions,

- The population was generated using the Manual Population Generator Panels and the default attributes were assigned as shown in **Table 4-1**. The response times were set to 0 s to ensure instantaneous movement of the occupants.

buildingEXODUS outputs the flow rate (Average PPM: Average People per Minute) at census line c1 over the total flow time period through that section of the corridor. Therefore the average PPM (people per minute) can be converted into Unit Flow Rate (people per metre per second) as follows:

Unit Flow Rate (occupants/m/s) = Average PPM / Corridor Width / 60 **Equation 4-4**

For each simulation run, the ratio of the number of people in Group A prior to running of the simulation and the area of section in which the people were located, i.e. the average population density was recorded. Moreover, the corresponding flow rate value at census line c1 was obtained. For each density value, the simulation was repeated 10 times and each time the locations of the occupants in Group A were randomised.

The buildingEXODUS curve and results related to other analyses are as shown in **Figure 4-7**.

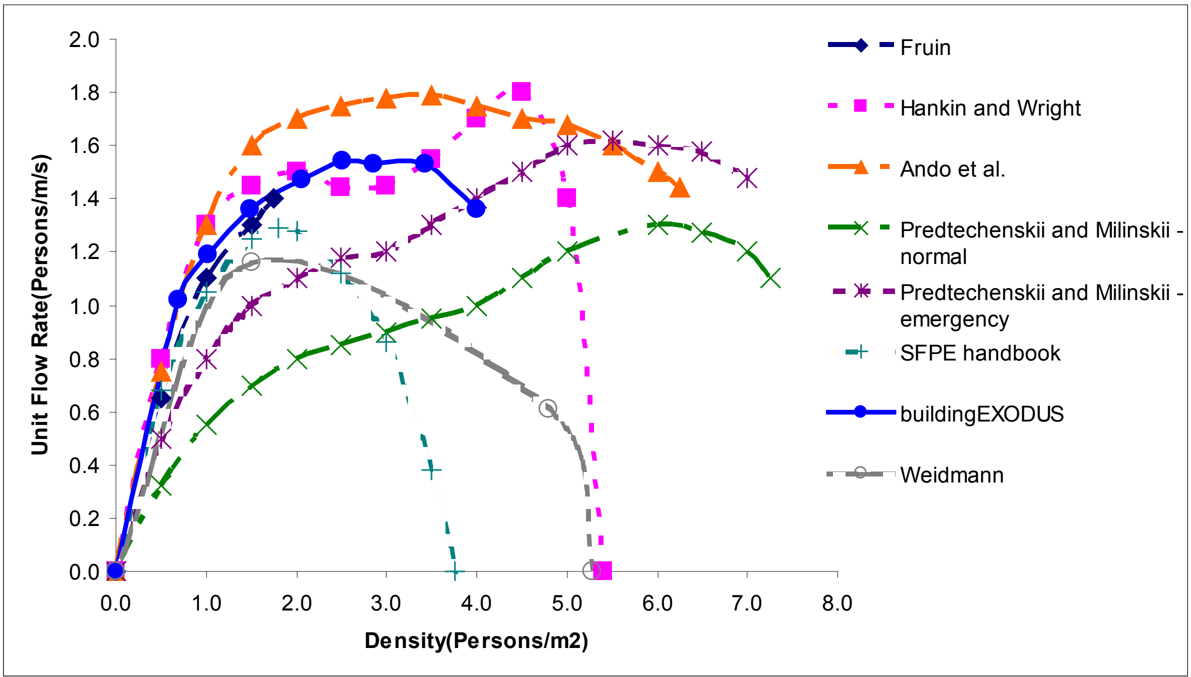


Figure 4-7 Relationship between Unit Flow Rate and Population Density

All the curves follow a similar trend whereby the unit flow rate increases to a peak value and then decreases at higher densities. The increase in the unit flow rate across the census regions is due to the increase in the number of people passing through the corridor until the corridor is being utilised to its full capacity. Further increases in density results in a decrease in unit flow rates as the number of inter-person conflicts increases. However, the differences between the data points of the curves are quite significant. The discrepancies in the data pertaining to the different empirical curves could be the result of cultural differences or psychological factors given by the incentive for the movement [Schadschneider 2009].

The unit flow rate value corresponding to data points on the Fruin curve in **Figure 4-7** are derived values from the crowd speed and crowd density from the Fruin's curve on **Figure 4-5**. The relationship between Crowd Flow Rate, Crowd Speed and Crowd Density is as shown in **Equation 4-5**.

$$\textit{Crowd Flow Rate} = \textit{Crowd Speed x Crowd Density} \qquad \textbf{Equation 4-5}$$

The buildingEXODUS and Fruin curves follow similar trends for low densities (up to 1 person/m2). However, the Fruin curve does not show data points for densities above approximately 1.90 persons per square metre. This inhibits comparison with the buildingEXODUS curve at those higher densities. Moreover, because of the lack of data points, the Fruin curve fails to illustrate the decrease in unit flow rates above certain densities.

The Predtechenskii and Milinskii curve (normal circulation) shows lower flow rate values when compared to the other curves. The reason behind this could be based on the fact that the Predtechenskii and Milinskii curve is based on observational results in the Soviet Union where the temperatures could be usually lower. Thus, the people within those experiments would have had winter clothing on them. Winter clothing results in an increase in the body ellipse size such that the density of a specific number of people would increase [Thompson 1994]. The increase in density could result in lower unit flow rate values as depicted by the Predtechenskii and Milinskii curve.

The buildingEXODUS curve shows similar trend to the Hankin and Wright curve for densities up to approximately 1.0 person per square metre, after which the buildingEXODUS curve shows an increase in unit flow rate values up to a peak value of 1.55 persons/m/s. On the other hand, the Hankin and Wright curve rises to a peak value of 1.80 persons/m/s and then drops significantly. However, the buildingEXODUS curve does not show flow rates beyond a density of 4 persons per square metre. This is due to the fact that fine nodal configuration in buildingEXODUS allows a maximum packing density of 4 persons per square metre.

When compared to the Ando et al. curve, the buildingEXODUS curve shows good agreement for population densities up to 1 person per square metre, after which the Ando et al. curve rises to a peak value of approximately 1.79 persons/m/s. It can also be noted that flow rates on the Ando et al. curve are almost constant for a range of densities beyond 2 persons per square metre, which then decrease gradually at higher densities.

Predtechenskii and Milinskii (normal circulation) and Hankin and Wright curves illustrate a sudden increase in the flow rate value at very densities. This could be due to the effect of crowd pressure at those high densities which had the effect of pushing the people forward. The effect of pressure exerted on people moving forward in a crowd was also noted by Peschl as described in Chapter 2.

4.2.3 Relationship between crowd flow rate and exit width

This section describes the investigation of the relationship between crowd flow rate and exit width. The geometry used is as shown in **Figure 4-8**. This geometry is commonly used for validating evacuation models, the reason being, its ability to illustrate the movement and behaviour of occupants attempting to move from a large available space through a narrow constricted region. In order to capture the flow rates, a census Line C1 was generated at the exit opening as shown in **Figure 4-8**.

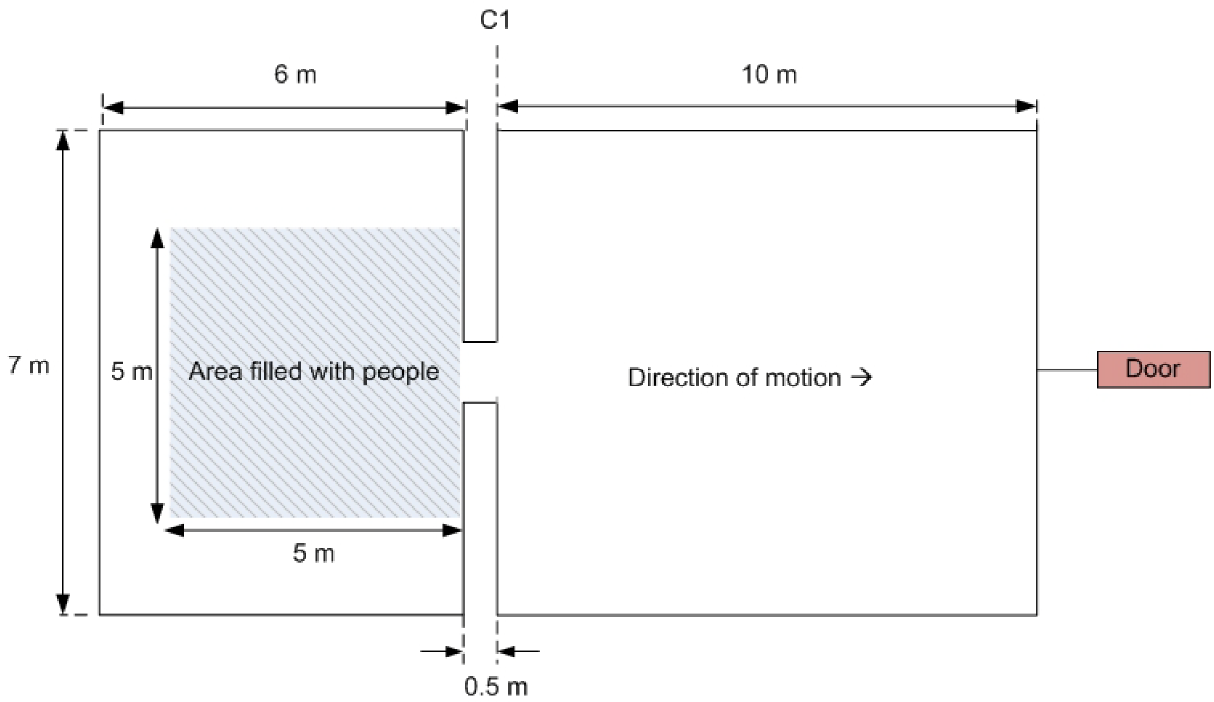


Figure 4-8 Geometry used for investigating relationship between crowd flow rate and exit width.

The set up in buildingEXODUS was as follows:

- The population was generated using the Manual Population Generator Panels and the default attributes were assigned as shown in **Table 4-1**. The response times were set to 0 s to ensure instantaneous movement of the occupants.

The variant in the simulations was the width of the opening in the range of 1 m to 3 m. buildingEXODUS outputs the average PPM at the census line C1 for the period of flow time at that region. These results were then converted into flow rate values as using **Equation 4-4**.

10 simulations run were conducted for each width specification. The corridor width was gradually increased in denominations of 0.5 m and the corresponding average flow rates were determined. **Figure 4-9** illustrates the buildingEXODUS results and how it compares against empirical data collected by other researchers [Kretz 2006] [Predtechenkii 1978] [Seyfried 2009] [Nagai 2006].

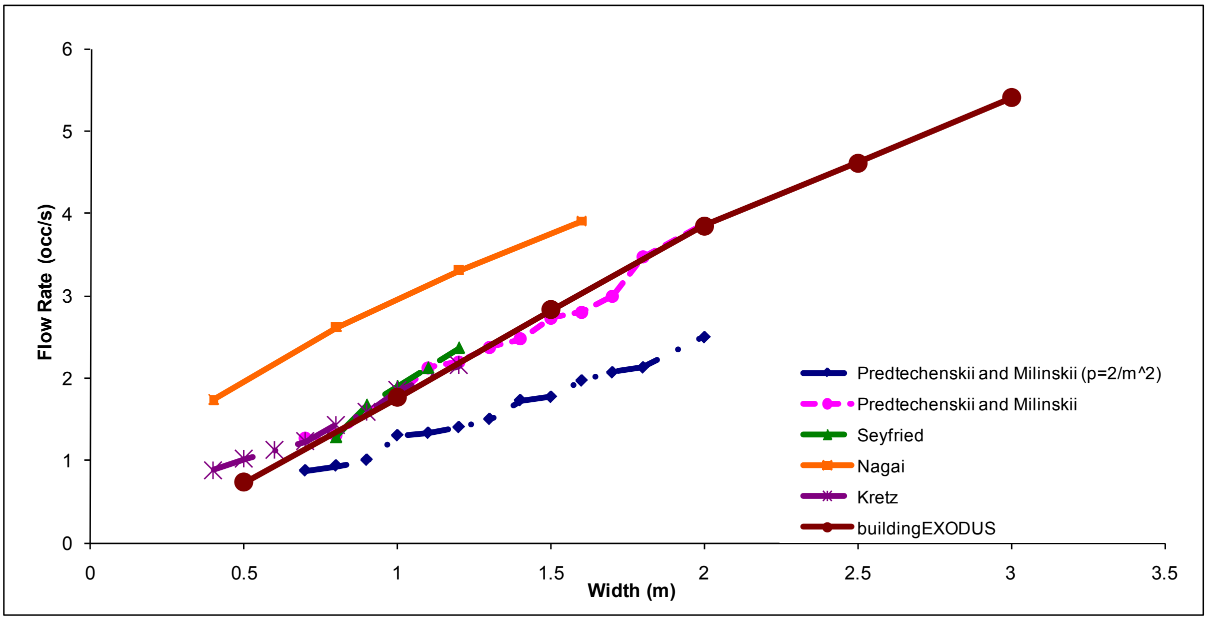


Figure 4-9 Variation of flow rate with exit width

In the light of the results shown in **Figure 4-9**, it can be noted that there is an almost linear relationship between flow rate and exit width. Moreover, the buildingEXODUS curve lies within the empirical envelope of results. The increase in passage width results in an increase in available space which can be utilised by the moving occupants, thereby increasing the flow rate. The buildingEXODUS curve is in close agreement with the Predtechenskii and Milinskii, Kretz and Seyfried curves.

4.3 Summary

This chapter describes some of the key mechanisms and capabilities of the buildingEXODUS evacuation model, which is the core of the bEX-Hybrid model. Furthermore, it includes an evaluation of the buildingEXODUS software through a series of tests. These tests were also performed with a view to consolidate an understanding on some fundamental aspects of crowd flow such as variation of occupant travel speeds with population density, variation of flow rates with population density and variation of flow rates with exit width. The outcome of these tests will be more specifically geared towards the calibration and further validation of the Continuous and Coarse models (See Chapters 5 and 6). During the course of these tests, it was found that, when performing comparative analyses against empirical data, although it may not be possible to recreate the exact scenarios used in the empirical data capture, it is imperative to understand the capabilities of the model used for the analysis. In the same breath, it is crucial to use the available features and apply the correct parameters and set up the simulations as accurately as possible.

Chapter 5 - Development of the Continuous Model

The focus of this chapter is on the development of a computational model, bEX-Continuous, which utilises continuous space for simulating pedestrian evacuation. bEX-Continuous is a core component of the hybrid architecture, namely the buildingEXODUS-Hybrid (bEX-H), which will be presented later in this thesis (Chapter 7). When using a continuous approach for the discretisation of space, it is possible to take into consideration a larger number of agent attributes allowing for a wider range of agent behaviours to be modelled.

bEX-Continuous is based on a multi-agent system whereby each agent is modelled as an autonomous agent which exhibits some forms of adaptive behaviour. This chapter describes the algorithmic procedures for simulating advanced agent behavioural and movement mechanisms. Moreover, it depicts the implementation of a novel navigation system called the Navigational Graph coupled with the planning approach of the agents. In addition, this chapter discusses the optimisation strategies which were formulated during the development of the bEX-Continuous model with a view to reducing the run times of the simulations. Moreover, it includes a discussion on how deadlock situations can be avoided during the simulation of high densities of agents in confined continuous spaces.

In addition, this chapter describes the validation of the Continuous Model using a series of test cases specified in the IMO guidelines **[IMO 2002].** Although these test cases have been designed for evacuation models used within the marine environment, the majority of the set ups used in those test cases are designed for small-scale, non-specific, flat surfaces and are therefore applicable for validating evacuation models used for building environment as well.

Furthermore, in the current model, the word continuous is used in relation to the space occupied by the enclosure such that occupants can move from any point of the space (x and y coordinates) to another throughout the enclosure. The simulation however is based on discrete events, that is, it keeps track of the current simulation time and skips to the next event start time as the simulation proceeds [Hakonen 2003].

5.1 Navigation within the continuous space

This section describes and evaluates the techniques which have been used to simulate navigation in the continuous space. Navigating within the continuous environment involves the process of planning, recording and controlling the movement of the occupants from one place to another during the simulation. In the context of people circulation and evacuation models, the occupants need to navigate from the starting locations within the enclosure to their goal locations which could be exits, assembly points or any other prescribed locations. The path of an agent within continuous space from a start location to an end location is a continuous map [Doyle 1995]. However, the complexity of this map can increase significantly in large geometries for example, buildings with multiple internal rooms and floors. Moreover, the presence of static obstacles within the enclosure makes the path planning process of the agents even more complicated. This topic has been a subject of research in areas such as computer graphics, robotics and modelling of characters in virtual environments. The complexity of the continuous map can be reduced by the use of roadmap methods which can translate the map into a network of paths which can be iterated through by the use of searching algorithms. Examples of roadmap methods include the Visibility Graph [Doyle 1995] and Sub Goal method [Doyle 1995] [Green].

5.1.1 Visibility Graph

The visibility graph is a commonly used technique in the field of robotics. This graph is a roadmap that can be used in a two dimensional continuous space. The roadmap is constructed by including all the vertices of the obstacles in the space and the start and goal locations of the agents. Each vertex is treated as a node, and each node is connected to another node via a link as long as the link does not intersect with another obstacle. In other words, a link is only created when two nodes are "visible" to each other, thus the name visibility graph. The combination of the visibility graph and the MinPath theorem can determine the shortest path between two locations [Feurtey 2000]. **Figure 5-1** illustrates a simple visibility graph with O and G being the start location and goal location respectively.

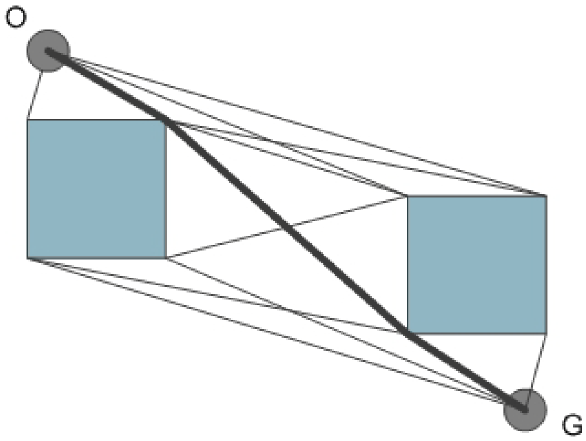


Figure 5-1 Visibility graph [Feurtey 2000]

Key:

Obstacle

Link between two "visible" vertices

Shortest Path

5.1.2 Sub Goal Method

This method consists of the generation of intermediate locations, also known as sub goals, in between the start and goal locations. This method is applied when the final goal is not directly reachable by the agent (the final goal might not be visible from the starting location of the agent) and therefore intermediate visible locations have to be visited prior to reaching the final destination. However, in case these intermediate locations are not directly reachable for example because of the presence of obstacles, additional locations can be generated from the existing intermediate locations. This technique has been applied in many areas including animation of virtual characters. For instance, in [Green], the author uses a behaviour called Follow Path to animate movement from start to target locations. The path in between the start and goal locations has points called waypoints, and the lines in between those waypoints are path segments, and the combination of these path segments are sub paths. Each character increments along the path by using the waypoints until reaching the final target location. This is illustrated in **Figure 5-2**.

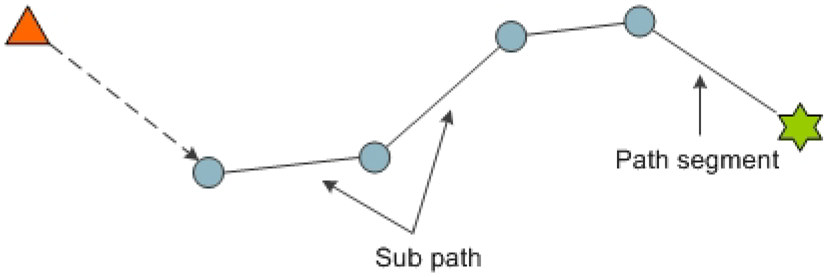


Figure 5-2 Using waypoints in Sub Goal method [Green]

Key:

 Starting location

 Waypoint

 Target location

 Initial direction of travel

5.1.3 Navigational Graph in Continuous Region

In bEX-Continuous, the space bounded by the enclosure is modelled using Continuous Regions. These regions are represented by polygons of any shape which can be drawn manually using the standard polygon drawing functions provided by buildingEXODUS. Moreover, buildingEXODUS also provides an automatic method of generating Continuous Regions from the outline of the geometry's walls coupled with the location of the external doors.

A key data structure in bEX-Continuous is its Navigational Graph which is illustrated in **Figure 5-3**. This graph is based on a novel approach which utilises the benefits of the techniques used in the visibility graph and the sub goal methods in sections 5.1.1 and 5.1.2 respectively. The Navigational Graph is a network of waypoints and path segments which is automatically generated in the pre-processing-phase. Each waypoint is a two dimensional coordinate used as a point of reference to guide the occupants towards a particular target and the path segment is the line between each waypoint. A collection of such segments from the starting location to the ending location represents the actual path that can be taken by an occupant. However, unlike the visibility graph where the links (path segments) were connected to each and every visible vertex, in the navigational graph, the waypoints are generated only at locations where the internal angles are concave. This ensures that there are far fewer links generated for every polygon and results in the agent performing fewer searches for target waypoints. The reason for placing the waypoints at concave corners is that they are sufficient to direct the agents away from the location of the walls of the enclosure and around corners. In this respect, when the agents are heading for the waypoint, they do not need to perform computationally expensive collision detection tests with the walls. In addition, the mechanism used for adjusting the paths of the agents away from the wall locations also takes into account the agents' widths. The path adjustment procedure is described in more detail later in this chapter (Section 5.3.1.2.2).

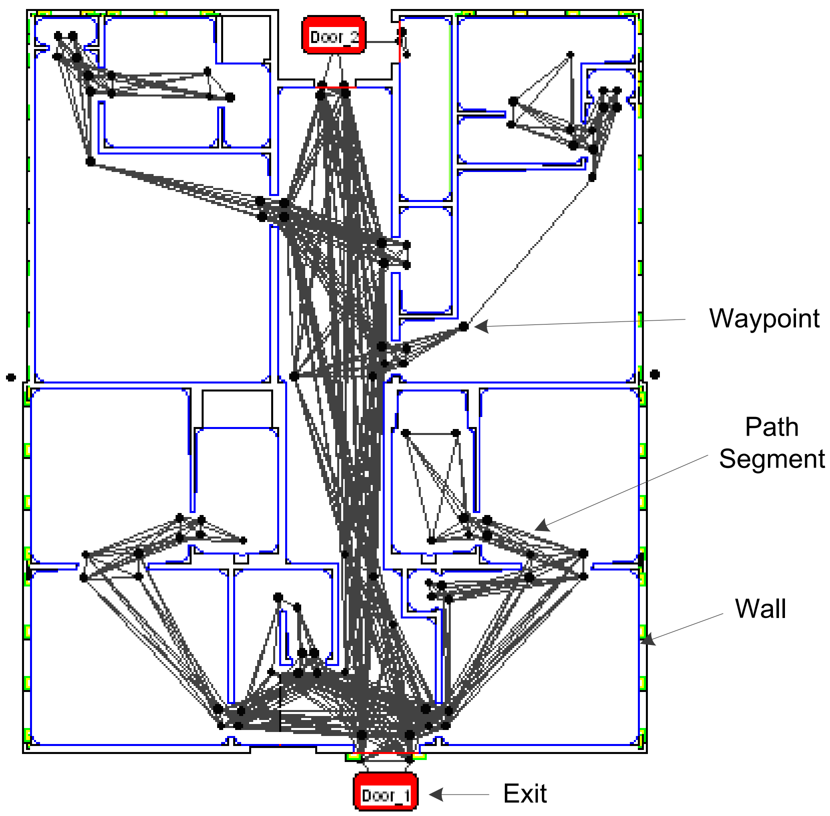


Figure 5-3 Navigational Graph in bEX-Continuous

The waypoints are automatically generated once the polygon has been converted into a continuous region.

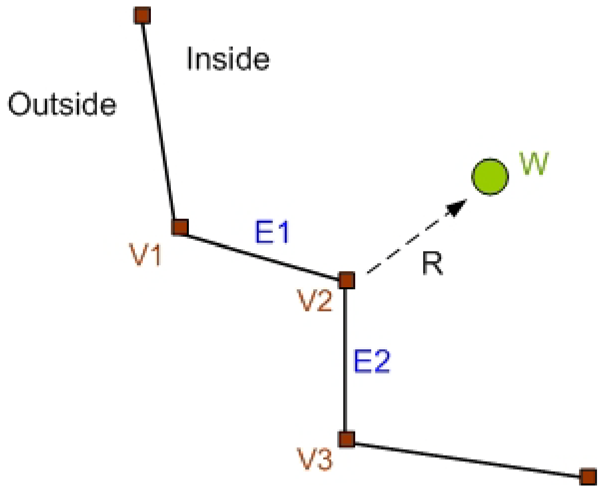


Figure 5-4 Calculation of the internal waypoint

Key:

Waypoint

Vertex

Edge (Boundary)

Resultant Vector used to calculate location of waypoint

The algorithm considers two consecutive edges and determines whether the internal angle is greater than or less than 180 degrees. For internal angles which are greater than 180 degrees, the algorithm uses vector arithmetic to calculate the coordinates of the waypoint. **Figure 5-4** illustrates two edges namely E1 and E2 which form part of a polygon bounding a continuous region. E1 has vertices V1 and V2 while E2 has vertices V2 and V3. Treating E1 and E2 as vectors, the addition of vectors E1 (V2-V1) and E2 (V2-V3) results in a resultant vector R.

The location of the waypoint would lie on the resultant vector R, and thus the (x,y) coordinates can be calculated as follows:

```
x  = P2x + t*[(P2x-P1x) + (P2x-P3x)]
y  = P2y + t*[(P2y-P1y) + (P2y-P3y)]
```

where t is a constant to scale the position. By default, the waypoint is placed at a distance of 0.25 m from the concaved vertex V2 as shown in Figure 5-4. This is because the default average width of the agents is 0.25 m.

After the generation of the waypoints, the algorithm proceeds by connecting each waypoint to their visible neighbouring waypoints through arcs. This is followed by the generation of a potential map whereby the waypoints are assigned a potential which is the shortest distance from the exit. This map is implemented using ***the breadth-first heuristic*** search, which is outlined in the next section.

5.1.3.1 Implementation of the potential map for the navigational graph

The potential map implementation is based on the breadth-first search algorithm and is used as a means of navigation by the agents. Upon instantiation, all the waypoints are assigned very high potential value (`999999.0f`). The algorithm starts off by firstly considering all the waypoints which are connected to the doors. The algorithm then visits each neighbouring waypoint in turn assigning them a potential value. This potential value is the shortest visible arc distance from the door. Then for each of the neighbouring waypoints, the algorithm explores the neighbouring unexplored ones, and the algorithm proceeds until all the waypoints in the graph have been visited. The order in which the waypoints are visited is shown in **Figure 5-5**.

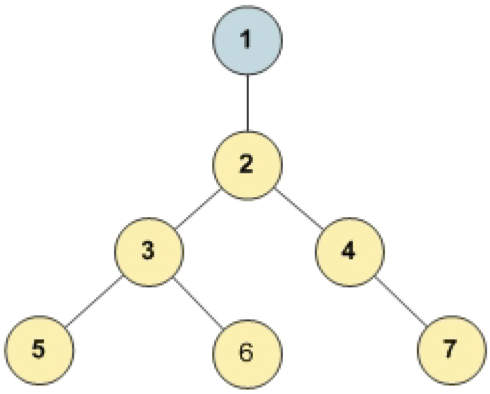


Figure 5-5 Order in which waypoints are visited

The pseudo-code for the algorithm is as shown below:

```
List of waypoints to process, ProcessList
FOR each waypoint connected to external exit
    Add first waypoint to ProcessList
    WHILE ProcessList is not empty
        Retrieve first element from Processlist, Wc
        Get list of arcs, arcList connected to Wc
        Erase first element from ProcessList
        FOR each arc in arcList
            Find waypoint Wa which is adjacent to Wc
            Get length of arc L which connects Wa and Wc
            Set the new potential, newPot = Potential of Wc + L
            IF newPot is less than potential of Wa
                Set potential of Wa to newPot
                Add Wa to ProcessList
            END IF
        END FOR
    END WHILE
END FOR
```

Algorithm 5-1 Procedure for implementing potential map

5.1.3.2 Location of waypoints

The location of the waypoints is an important aspect of the path planning process. Waypoints placed closer to the corner end causes the agents to be directed too close to the walls and requires them to adjust their positions earlier than if the waypoints were placed further. Waypoints placed further from the corner results in the agents travelling further before negotiating the corner. In [Still 2000], Still uses two images showing the flow of a crowd around a corner to illustrate that the density of people at the corner is higher than the surrounding region, which means that people have a preference of using the inner corner more often. In the model, the waypoints are placed at a fixed distance of 0.25m from the concaved corner by default. However, the occupants in the model have varied body widths and therefore if they were heading for a fixed waypoint, the higher the body widths of the occupants, the closer they would get to the inner corner. In this respect, a dynamic solution is preferred such that the agents would not need to head for the exact location of the waypoint. The position to which they head towards is dynamically altered depending on their body size and personal comfort space. This dynamic adjustment algorithm is described in Section 5.3.1.2.2.

5.1.3.3 Effect of the location of waypoints on flow rates

The exact location of the waypoints does have an effect on the flow rate especially around corners. When the waypoints are located close to the inner corner (less than 0.25 m away from the inner corner), the flow of agents narrows significantly and constricts around that region, thus leading to a reduced occupant flow around that region. However, this constriction is avoided due to the behavioural algorithm that is activated while a crowd of occupants attempt to undertake a corner. The agents have a behavioural trait called Agent Avoidance (Section 5.3.2.4). With this behaviour, the agents attempt to maintain a certain inter-personal space with each other. The resulting effect around corners is particularly ‘naturalistic’ in the sense the flow does not narrow down around the inner corner, but is rather spread over the overall cornering region, although the density of occupants around the inner corner is relatively higher. This is illustrated in **Figure 5-6**.

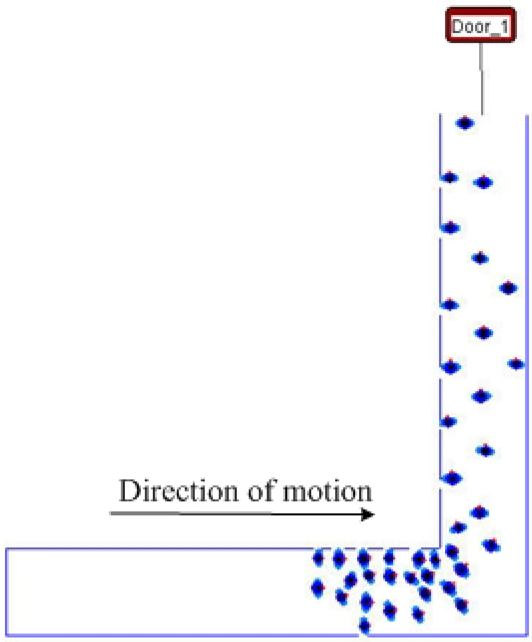


Figure 5-6 Agents demonstrating cornering with Agent Avoidance behaviour

5.2 Continuous Agent Model

This section describes the implementation of a continuous agent navigating within continuous space. The agent is modelled as an autonomous character which exhibits some forms of adaptive behaviour. In other words, the agent has the ability to navigate in a life-like manner and react to stimuli in its environment. The agent will at some point in its trajectory encounter obstacles. The obstacles can be static such as walls, tables, chairs or dynamic, for instance, other agents navigating in the same environment. In this respect, the agent is able to detect these obstructions and react to them accordingly.

A continuous agent is characterised by a series of individual attributes and behaviours which define its autonomy. Some of the basic attributes of a continuous agent are as shown in **Table 5-1**.

Table 5-1 Basic attributes of a continuous agent

Attributes	Description	Quantity
2-D Position	The location of the agent in continuous space	Vector
Velocity	Rate of change of displacement	Vector
Acceleration	Rate of change of velocity	Vector
Max Speed	Maximum walking speed of agent	Scalar
Max Acceleration	Maximum acceleration of a continuous agent	Scalar
Mass	This includes the mass of the agent coupled with the mass of objects being pushed, pulled or carried	Scalar
Orientation	The possible movement directions (headings)	N basis vectors
PET	Personal Elapsed Time in the simulation	Scalar
Drive	The urge for the agent to undertake a specific activity or action	Scalar
Body Frame Width	The width of the agent excluding the size of the shoulders	Scalar
Body Shoulder Width	The width of the agent's shoulders	Scalar

5.2.1 Continuous Agent Representation

The agent is represented in two dimensional space from the top view perspective. Based on Thompson [Thompson 1994] and Still [Still 2000], the average shape of a person can be represented as shown in **Figure 5-7**.

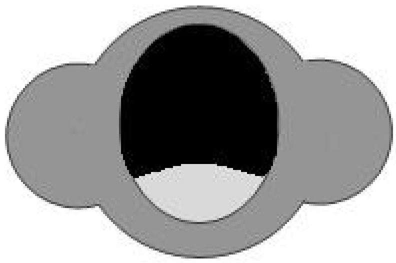


Figure 5-7 Top view shape of a person

However, using this shape as it is to represent agents in the model would lead to complex mathematical equations for the graphical representation as well as the computation of collision detection and avoidance with static and dynamic obstacles. A much simpler shape which can be used for such a representation is a circle; however, a circle does not accurately represent the space requirements of an agent. Moreover, in his thesis [Thompson 1994], Peter

Thompson describes the computational complications of representing persons as ellipses and proposes the benefits of the use of a combination of three circles merged together to represent each agent. The three circle representation was therefore used in this model and is illustrated in **Figure 5-8**.

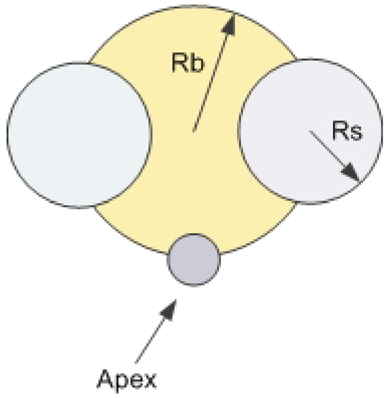


Figure 5-8 Representation of a continuous agent in the model

Key:

Rb:	Radius of Body frame
Rs:	Radius of Shoulder
Apex:	Used to illustrate the orientation of the agent (forward facing)

5.3 Autonomy of Continuous Agent

The autonomy of the continuous agent is modelled as a hierarchical structure comprising of three layers as shown in **Figure 5-9** .

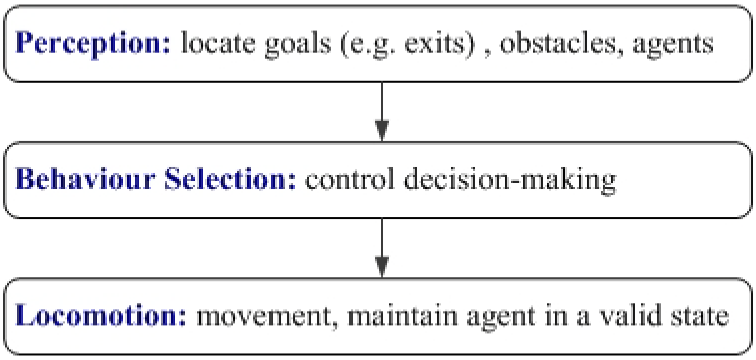


Figure 5-9 Decomposition of autonomy of continuous agent

A similar decomposition strategy is used by Reynolds in the representation of autonomous characters [Reynolds 1999] and Pan in the simulation of agents [Pan 2006a]. This strategy facilitates the design and implementation of the agent's autonomy as the layers are decoupled from each other.

A key module within the Continuous Model is the Crowd Simulation Engine which uses a discrete event simulation approach whereby the movement and behaviour of the agents is computed at discrete time steps. During each time step, the simulation loop iterates through each layer of the Continuous Agent model for every agent. The procedure is as follows:

```
Initialise all the agents in enclosure/region
WHILE there are agents present in the enclosure/region
Increment the simulation time
    FOR each agent
        IF simulation time is greater than agent response time
            Update Agent Perception
            Behaviour Selection
            Locomotion
        END IF
    END FOR
END WHILE
```

Algorithm 5-2 Crowd Simulation Engine procedure

5.3.1 Perception System

The perception mechanism of the autonomous agents consists of sensors aimed at gathering information and detecting changes in its immediate environment. The information gathered in the perception system is fed to the behaviour as input for the decision making processes. The sensory data includes the location of exits, obstacles and other agents within the enclosure.

5.3.1.1 Vision of autonomous character

The agents navigating in continuous space need some form of vision system in order to establish what they can see at any one time in the simulation. This capability has been implemented using a technique called Ray Casting [Klawonn 2008]. This method involves casting a ray from the centre of the agent to the point which needs to be tested for visibility. If the ray hits any static obstacles such as walls in its path, this implies that the agent does not have a direct line of sight to the point of interest.

This is illustrated in **Figure 5-10** whereby an agent attempts to locate a neighbouring waypoint during its path planning process. In this scenario, agent P performs a line of sight check for waypoints Wa and Wb. Waypoint Wb is not visible to P due to the presence of wall boundaries in between P and Wb.

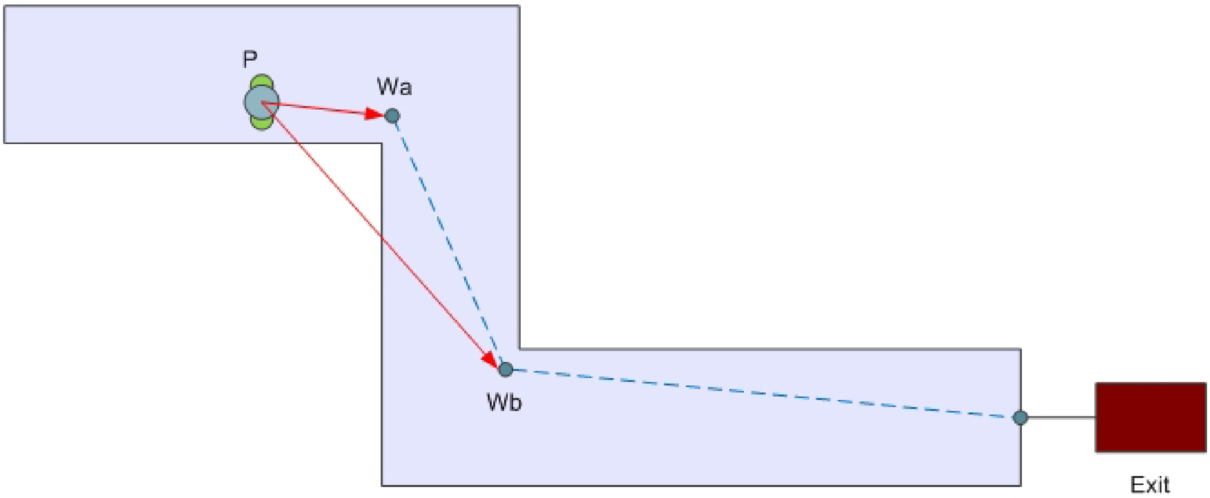


Figure 5-10 Locating elements in the continuous space (Choosing which waypoint to head towards)

Key:

(red arrow)	Ray used for collision detection
(dotted line)	Path Segment
(solid line)	Wall boundaries

The pseudo-code for this algorithm is as follows:

1. Cast ray emerging from agent towards intended target
2. **IF** ray intersects a static obstacle
3. Target not visible
4. **ELSE**
5. Target visible
6. **END IF**

Algorithm 5-3 Ray casting procedure

5.3.1.2 Path Selection in Navigational Graph

The Path Selection component of the Perception System allows the agents to search the Navigational Graph in order to determine the appropriate routes from their individual start locations to the external exits. An agent navigating in the continuous spatial environment can have up to three goal locations namely Short Term Goal, Medium Term Goal and Long Term Goal.

1) Long Term or Final Goal

The Long Term Goal refers to the final destination of an occupant. It is generated using global navigational rules and controlled by an occupant's knowledge of the structure and their choice of target for example, the exit of a building or a destination within it.

2) Medium Term Goal

The Medium Term Goal is a waypoint in the geometry which directs the agents towards a local exit. It is used to control the agent's route to their long term goal for example, which exit out of a room within a building.

3) Short Term Goal

The Short Term Goal is a sub goal of the Medium Term Goal and is generated using local navigational rules. The agents always attempt to move towards their Short Term Goal. This goal is controlled by the agent's physical fitness and dimensions and is also used to drive the agents away from the walls of the enclosure.

In some cases, all three goal points can be the same, for example when an agent is approaching its final destination.

5.3.1.2.1 Medium Term Goal Evaluation algorithm

At the start of the simulation, each agent performs a line of sight check for the nearest visible waypoint which is then set as its Medium Term Goal. This initial Medium Term Goal is used as a starting point in the Navigational Graph search. Once in motion, the agent then tries to update its current Medium Term Goal to the next visible waypoint, within the structure, that is nearest to the agent's Long Term Goal. The update takes place when the agent can see the next visible waypoint or when it is within a certain threshold distance (1.0 m) from the current Medium Term Goal. This threshold distance is required as the agents are not expected to pass exactly over the waypoints. Moreover, rather than checking for the visibility of the next visible waypoint at every time step, the agent performs the check after every 5 time steps. The frequency of visibility checks was chosen such that the agents would not go beyond their current Medium Term Goal without identifying their next visible waypoint, which could consequently lead to the agents taking rather unusual paths. The waypoint update mechanism is repeated until the agent locates its Long Term Goal. Furthermore, every time that the Medium Term Goal is updated, it is also adjusted to take into account the agent's width and personal space. Personal space is used to represent the extra space which people like to maintain between themselves and the walls or other people. The Path Selection process within the Perception System is as shown in **Figure 5-11**.

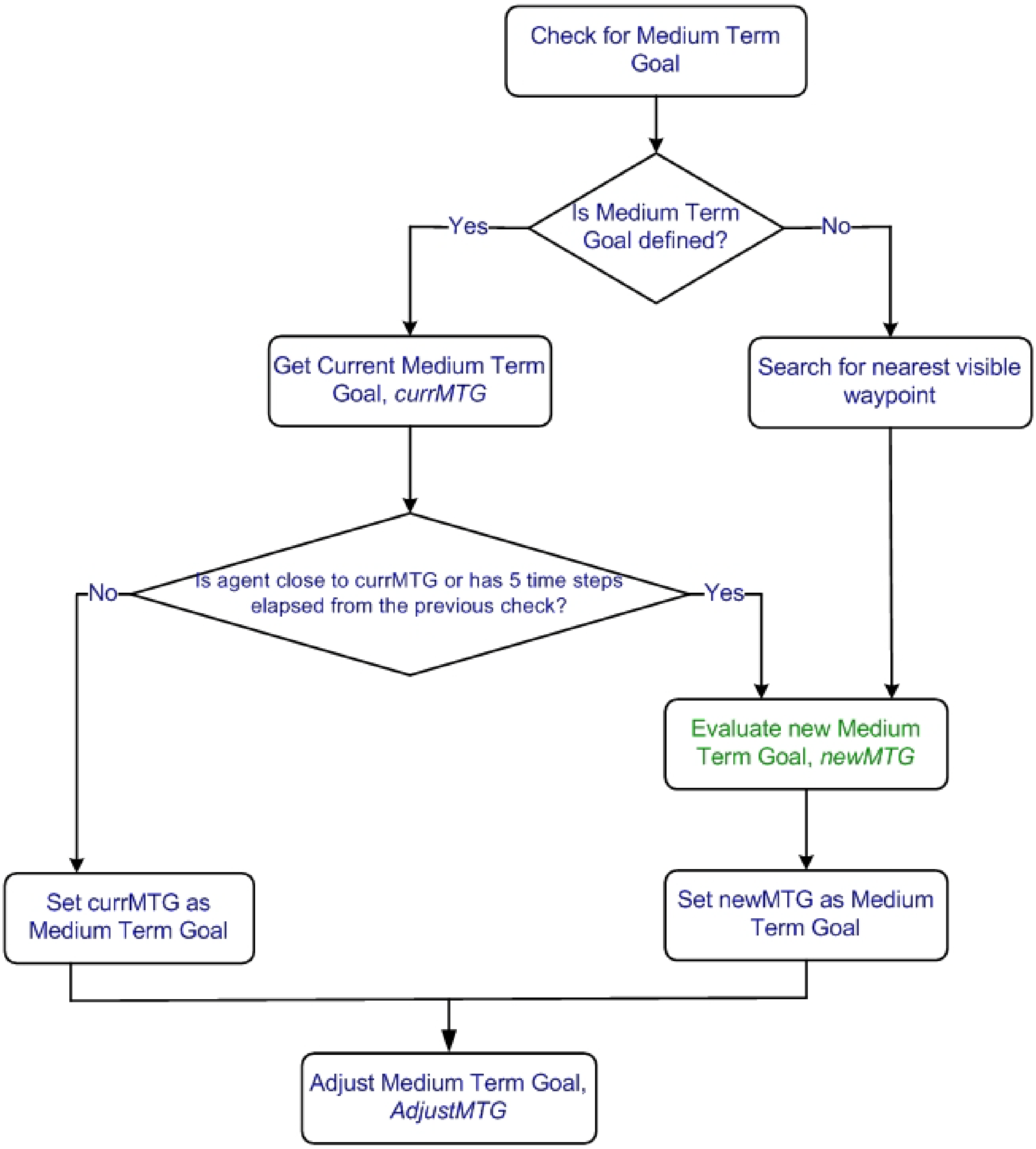


Figure 5-11 Path Selection Process in Perception System

The Medium Term Evaluation algorithm is further sub-divided into two components. The first component serves in finding the route to the ***nearest*** exit in the geometry using the

Potential Map (Section 5.1.3.1). This is achieved by the agents attempting to move towards waypoints of lower potential values.

However, it would be unrealistic to assume that all agents within a particular geometry would be moving to their nearest exit. In this respect, the functionality of the Medium Term Goal Evaluation algorithm has been extended into the second component which allows the assignment of different exits to different portions of the agent population. The second component allows agents to identify their routes to their ***specified*** exits using a Distance Route Map.

5.3.1.2.1.1 Implementation of Distance Route Map

The Distance Route Map is created in the pre-processing phase using the same waypoints which are generated in the Navigational Graph. The algorithm generates a Target Distance List for each waypoint. This allows each waypoint to keep track of its distance from every exit in the geometry as illustrated in **Figure 5-12**. In this geometry, the lengths of the arcs in between the waypoints are as shown in **Table 5-2**.

Table 5-2 Length of arcs in between individual waypoints

Arc connecting	**Length of arc (metres)**
Wa and Wb	3.0
Wb and Wc	2.0
Wc and Wd	3.0

Considering Waypoint Wa as an example (See **Figure 5-12**), its distance from Exit 1 is 0 metres and its distance from Exit 2 is equal to the sum of the arcs in between Wa/Wb, Wb/Wc and Wc/Wd, i.e. 8.0 metres. The size of the Target Distance List for each waypoint is dynamically adjusted to accommodate for increasing number of exits that are added to the geometry.

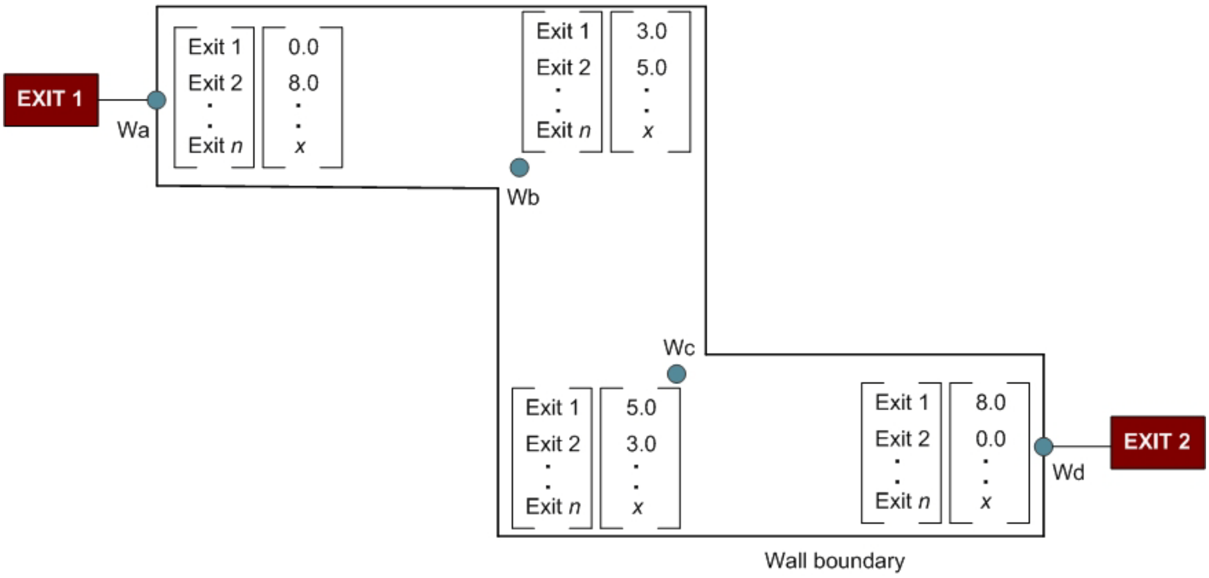


Figure 5-12 Implementation of Distance Route Map

Once an exit has been specified to an agent prior to the start of the simulation, the exit becomes a target for the agent. During movement, the agent with a pre-defined exit will attempt to navigate towards waypoints which have the smallest distance values corresponding to that specific exit. For example, an agent is located at the point Wc and has been assigned Exit 1 as its Target. During its next movement cycle, the agent will search the neighbouring waypoints of Wc which are Wb and Wd. The distance value corresponding to Exit 1 for waypoint Wb (3.0 m) is lower than the distance value corresponding to Exit 1 for waypoint Wd (8.0 m). Therefore, waypoint Wb is selected as the next Medium Term Goal. The algorithm proceeds until the agent reaches its target Exit 1. If a target exit has not been specified, the agent will use the Potential Map instead in order to navigate to the nearest exit. The two components of the Medium Term Goal Evaluation algorithm are as shown in **Figure 5-13**.

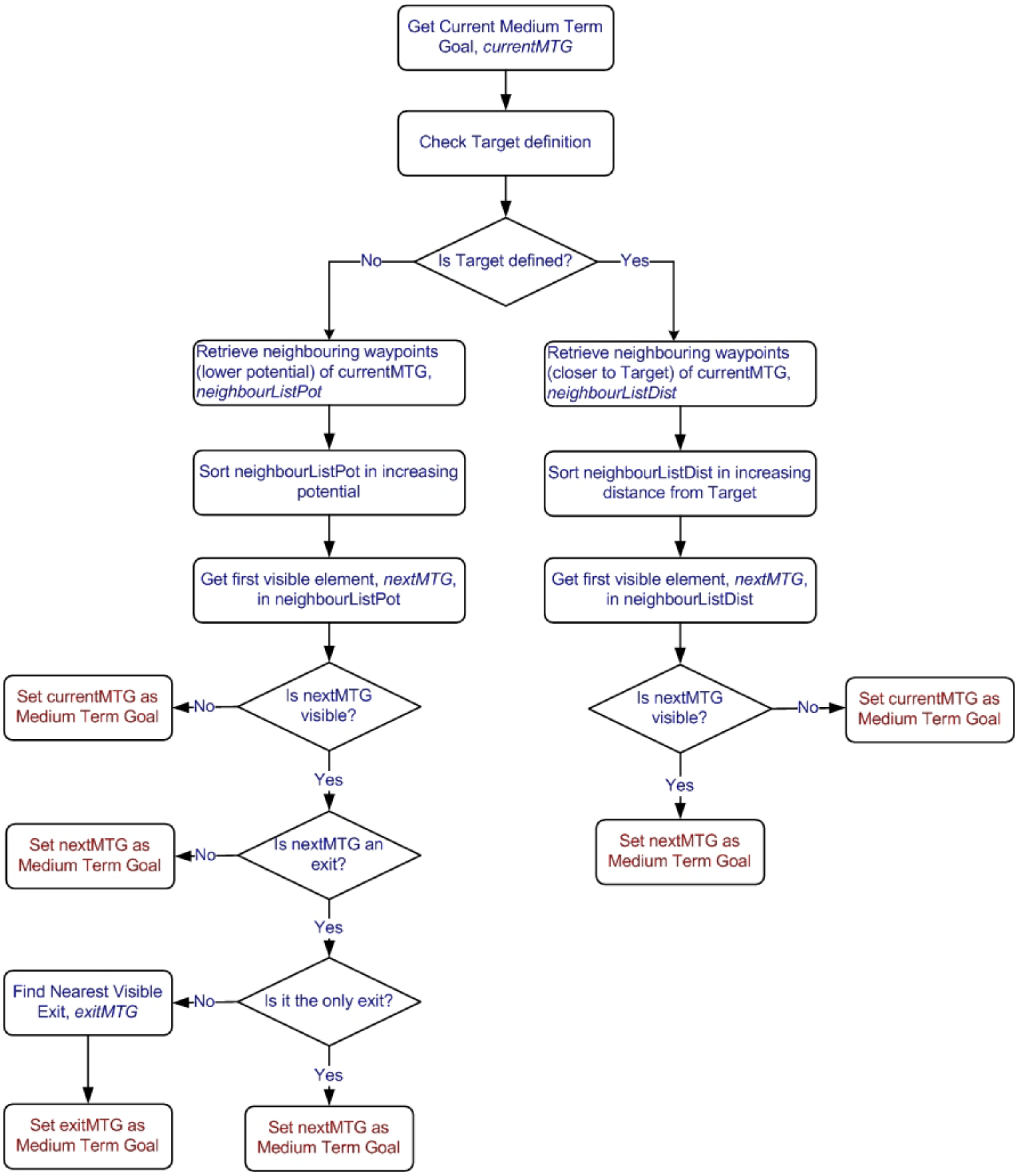


Figure 5-13 Evaluate Medium Term Goal algorithm

5.3.1.2.2 Adjusting the Medium Term Goal

Once the Medium Term Goal has been assigned, it is adjusted to take into account the agent's personal space requirements. There are two variations of this adjustment namely:

(1) Agents approaching a concave corner

(2) Agents approaching an exit

The Medium Term Goal adjustment algorithm makes use of the virtual width lines of the agents. These are two parallel lines which extend from the sides of the agent and are directed towards the agent's direction of motion as illustrated in Figure 5-14. The distance (Wsum) between the parallel lines S1 and S2, is the sum of the agent's body width (Wp) and an effective width (We). This effective width accounts for the agent's personal comfort space which allows the agent to maintain a certain minimum distance from corners and walls in the enclosure.

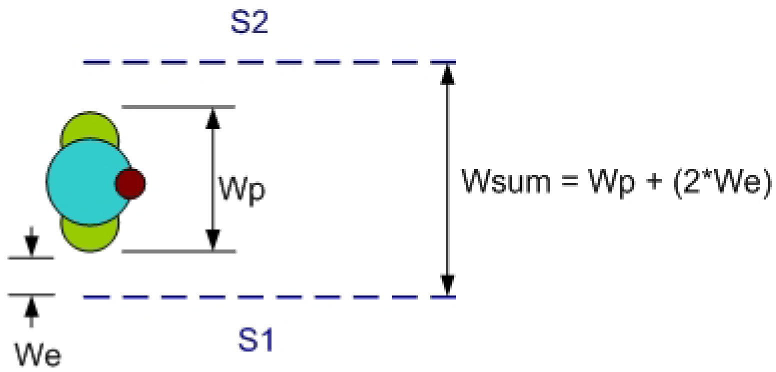
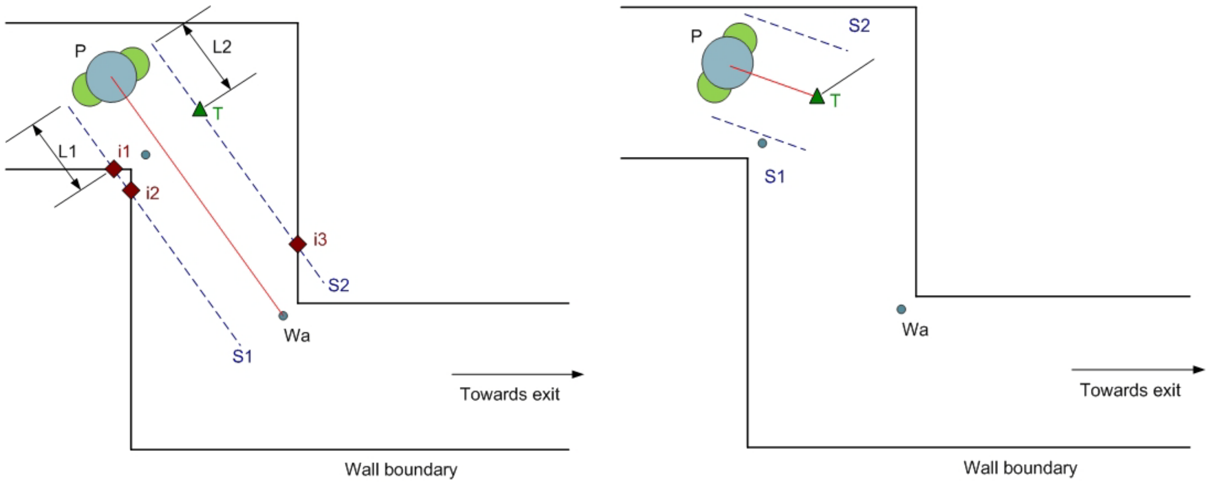


Figure 5-14 Virtual width line of agent

5.3.1.2.2.1 Agent approaching a concave corner

In **Figure 5-15**, the agent P is shown who has initially chosen waypoint Wa of the Navigational Graph as its Medium Term Goal. The agent then uses its virtual width lines to consider future collisions which could be encountered on its path to Wa. The agents virtual width line, intersects the wall boundaries at three locations, namely i1, i2 and i3. The algorithm chooses the nearest intersection point i1 for the adjustment calculation and computes the distance L1 on the virtual width line S1. The adjusted target T is placed on virtual width line S2 at a distance of L2 from the starting point of the line, whereby L2 = L1. The agent then selects point T as its new Medium Term Goal and the resultant path (**Figure 5-15**b) causes it to avoid the wall boundaries at the concaved corner.



(a) Initial path of agent **(b) Final path of agent**

Figure 5-15 Agent adjusting its Medium Term Goal due to location of walls

Key:

◆	Intersection point of virtual width with wall boundaries
▲	Adjusted target
●	Waypoint
———	Agent line of sight
— — —	Agent's virtual width lines (representing effective width and personal space)

5.3.1.2.2.2 Agent approaching an exit

When agents are on their approach to an exit, they would be heading for the waypoint which is connected to that exit. However, if all the agents were converging for the same waypoint around the exit location, it could lead to unnecessary congestion and also the agents would end up not utilising the entire width of the door, which is unrealistic. In this respect, the agents have to adjust their final location to avoid this convergence of paths.

In **Figure 5-16**, the agent initially selects waypoint Wa as its Final Goal. The agent then attempts to locate the nearest point, point T, on the exit edge by dropping a perpendicular line from its current location to the exit edge. The intersection point of the perpendicular line and the exit edge becomes the nearest point T. If point T can be reached without wall obstructions, then it is set as the new Final Goal, otherwise the agent follow its initial path to Waypoint Wa until it can locate T in the next movement cycles. This adjustment ensures that agents approaching an exit do not head directly for the waypoint but rather spread across the overall length of the exit edge.

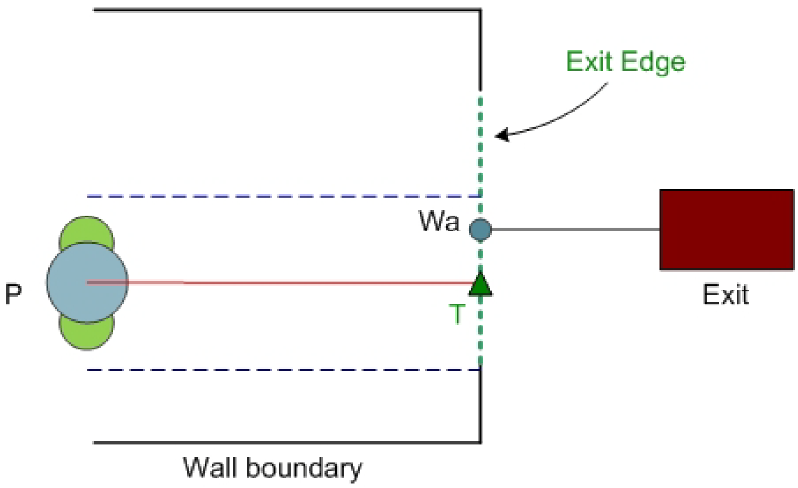


Figure 5-16 Agent adjusting its target on approach to exit

Key:

Adjusted target

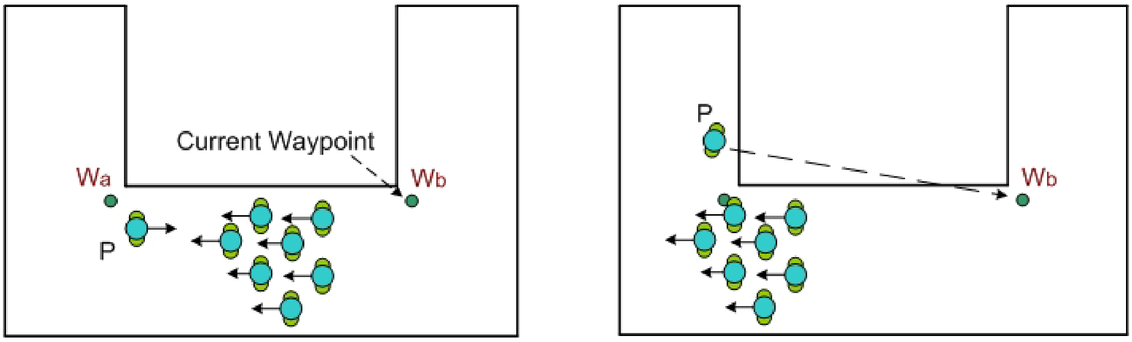
Waypoint

Agent line of sight

Agent's virtual width (representing personal space)

5.3.1.2.3 Agent Re-navigation

In addition, there are some situations where an agent would have reached its current Medium Term Goal and then incremented to the next visible waypoint, and then get pushed back due to the interaction of other agents moving in contra flows. This would lead to the agent tying to reach a waypoint which is in fact not visible via its line of sight and could result in the agent getting stuck. Such a scenario is illustrated in **Figure 5-17** (a) and (b).



(a) Agent selecting Waypoint Wb as its current Medium Term Goal

(b) Agent trying to get to its current Medium Term Goal Wb, although it is not directly Visible

Figure 5-17 Scenarios illustrating an agent getting stuck in the continuous space

In order to circumvent this problem, the agent possesses the capability to re-navigate once its current Medium Term Goal is not visible. This is achieved by re-assigning another nearest visible waypoint to the agent. In the current example in **Figure 5-17**(b), the waypoint Wa is re-assigned as the current Medium Term Goal of P.

5.3.1.3 Neighbourhood Search

An agent rarely exits on its own but exists in an environment shared by other entities. Therefore, an autonomous character is required to locate the other entities around itself while its navigating towards its target. Furthermore, since the surrounding entities are in motion, the assessor agent needs to constantly update its perception of the surrounding space. However, the agent does not require storing information about the totality of the environment as this would be a computationally expensive process as well as being unrealistic. Therefore, the agent only requires knowledge of its immediate surrounding. This can be simulated by restricting the search of neighbouring agents to a smaller region, such that size of the region would be indicative of the personal space of the assessing agent.

The initial approach used to eliminate the agents which are far from the assessor's personal space involved sorting all the neighbouring agents in ascending order of x coordinates. The differences in the x coordinates is indicative of the distances of the neighbouring agents from the assessor such that agents which are beyond a certain x coordinate value would be excluded from the search. However, this approach was found to be inefficient. This is because for large geometries consisting of large populations, the sorting process would end up being computationally too expensive. In this respect, a spatial hashing approach which consisted of subdividing the available continuous space into smaller continuous regions was implemented. For example, each room in a multi-compartment geometry could be represented by one continuous region. Such an approach would enable an agent to localise the search of neighbouring agents to its current region only, therefore improving the efficiency of the agent search. Nevertheless, this technique did not prove to be efficient in regions with large population sizes.

Consequently, in order to enhance the region based technique, another spatial hashing approach called the Grid Method was implemented. This method has been used by Pan [Pan 2006a] in the fast computation of collision detection amongst agents. In [Pan 2006a], an underlying grid of cells was used whereby each cell would accommodate one agent, and the size of the cells being set to the width of the largest occupant in the simulation. However, in the development of bEX-Continuous, the Grid Method was applied differently. In the model, each continuous region has an underlying grid of cells whereby each cell can accommodate multiple agents. Each cell has a length and width of 1.5 m and the choice of these dimensions is explained further in this section. The grid construction and the allocation of agents to the cells is done in the pre-processing phase whereas the querying and the update of the grid is done at runtime. At runtime, each agent searches its current and eight neighbouring cells (Moore Neighbourhood) to identify the neighbouring agents which are registered to those cells. Those selected agents are then stored and used as input in the agent's Behaviour System. This is as shown in **Figure 5-18** in which agent P is searching for its neighbouring agents whereby all the agents located outside the dotted box are excluded from the search. The combination of the region based approach and the Grid Method significantly improves the computational performance of the neighbourhood search algorithm. The use of this method brings the computation complexity from O(N^2) down to O(NM), whereby N is the total number of agents in the simulation, and M is the total number of agents within an agent's current cell and neighbouring cells of the underlying grid. M has a significantly smaller value as compared to N, especially in very large population sizes.

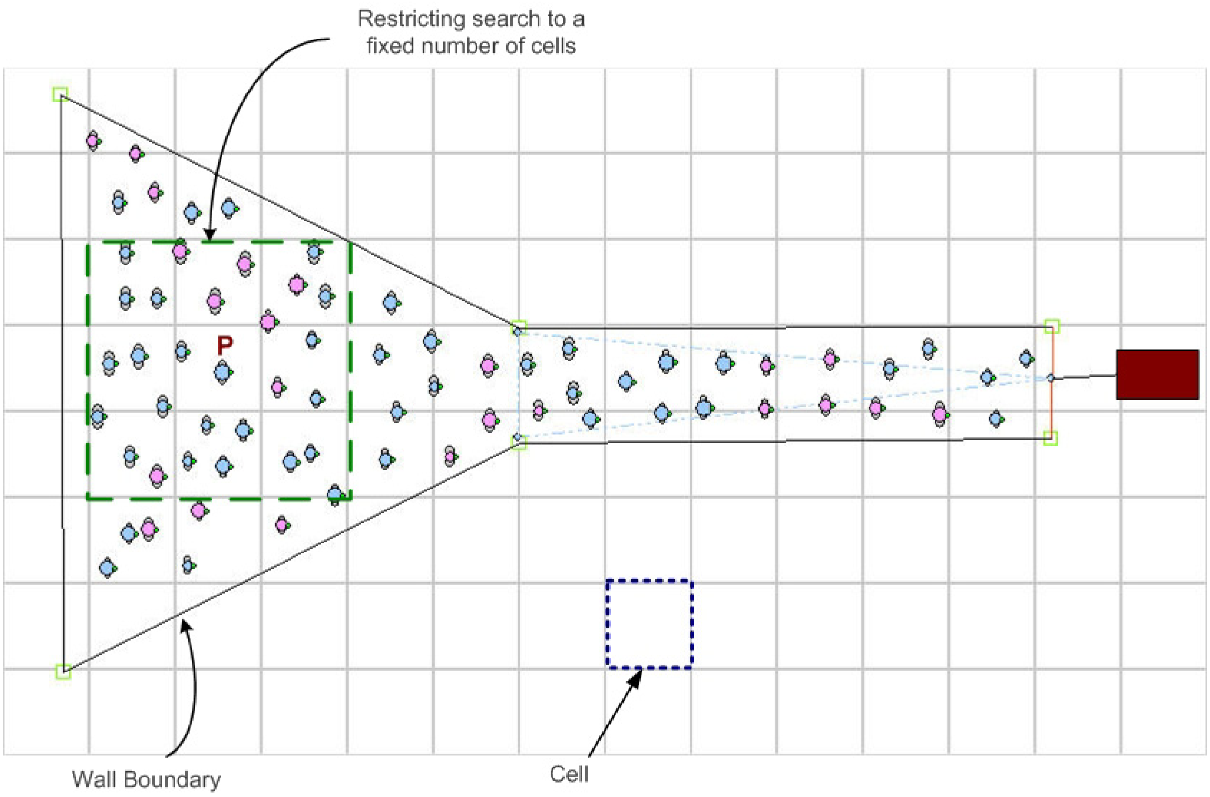


Figure 5-18 Neighbourhood Search restricted to a fixed number of cells

Moreover, an agent's behaviour is likely to be affected only by those neighbouring agents which are within its close range. Hall [Hall 1966] provides a classification of personal distance (far) and personal distance (near). The personal distance (far) covers a range of 0.75 to 1.2 m and is defined as the individual's circle of trust and can be found to be the interpersonal spacing found in a spacious waiting area. Whereas the personal distance (near) covers a range of 0.45 to 0.75 m and is commonly encountered in denser waiting areas and queuing situations. Therefore in the model, each cell was assigned a length and width of 1.5 metres. Using such cell dimensions, the coverage of the search area bounded by the agent's current and eight neighbouring cells is adequate to cater for the agent's personal space requirements. The performance of the search algorithm can be improved by reducing the cell size to a smaller value (e.g. less than 1.0 m). However, this would not be appropriate for modelling agent avoidance behaviours in the Behaviour System. This is because using a smaller cell size would mean that each agent will not be able to capture all the other agents within its personal space.

Furthermore, referring to **Figure 5-18**, not all agents in the surrounding cells are of importance to the assessor agent P. For instance, neighbouring agents which are located behind will not affect the forward motion of P. In this respect, the list of neighbouring agents can be further restricted by an angle in order to exclude those which are at the back of the forward facing direction of P. This angular restriction is described further in this chapter in the Agent Avoidance section (Section 5.3.2.4)

During movement, an agent registers itself to the cell corresponding to its new location and de-registers itself from the cell corresponding to its old location. There are also scenarios in which an agent might not be registered to a cell. This occurs when an agent (in the hybrid model) is transiting from the continuous space to the coarse or fine spatial types (Chapter 7). The allocation and de-allocation of the agents to the cells at run time is as shown in **Figure 5-19**.

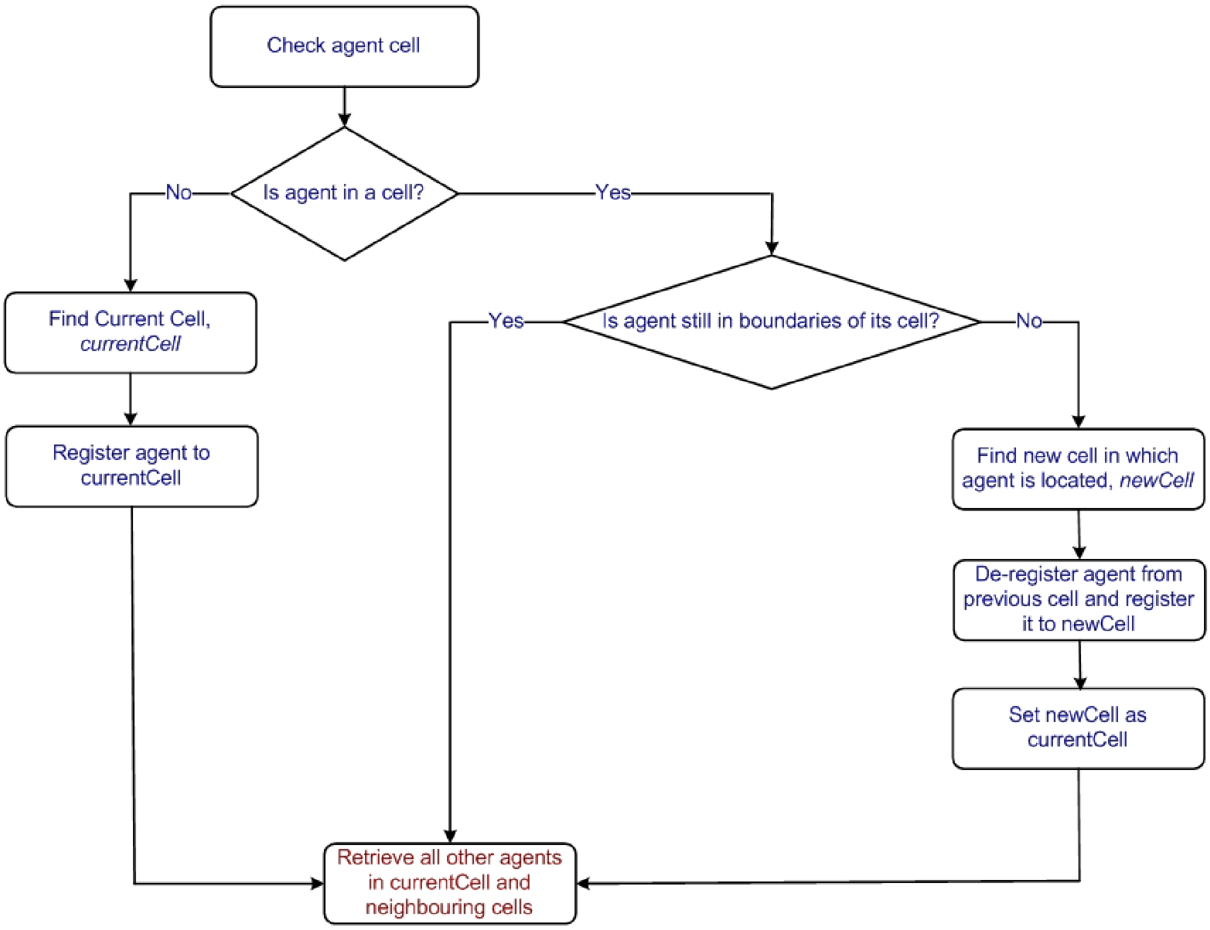


Figure 5-19 Allocation and de-allocation of agents to cells at runtime

5.3.1.4 Wall Search Optimisation

The walls within an enclosure delimit the areas where the agents are allowed to be situated within the continuous space. As a result, complex geometries can result in a significant number of wall boundaries to be generated. Although the waypoints generated within the Navigational Graph guide the agents away from the walls, there are scenarios which might lead to unavoidable contact between the agents and the walls, especially in crowded environments. Therefore, the agents are required to search for the nearest walls in their vicinity in order to compute wall detection and response (Section 5.3.2.3). In order to optimise the search of wall boundaries, the Continuous Model includes an optimisation technique which utilises the cells generated by the Grid Method (Section 5.3.1.3). This technique involves assigning the walls of the enclosure to the underlying cells, so that agents need not search the entire continuous space for wall boundaries, but instead limit their searches only to their current and neighbouring cells. The algorithmic procedure regarding the assignment of walls to underlying cells is performed in the pre-processing phase as follows:

```
1. FOR each wall 'W' in Continuous Region
2.     FOR each cell 'C' in underlying grid of Continuous Region
3.         Check if the wall 'W' intersects the boundaries of cell 'C'
4.         IF wall 'W' intersects cell 'C'
5.             Assign W to C
6.         END IF
7.     END FOR
8. END FOR
```

Algorithm 5-4 Optimisation Strategy: Assigning wall boundaries to underlying cells

The application of the wall optimisation strategy on a hypothetical geometry is as shown in **Figure 5-20**. The shaded cells are the ones to which wall boundaries have been assigned. Therefore, at run time, the wall search mechanism will be performed only by agents located within the shaded cells (within the inner bounds of the enclosure) as opposed to all the agents within the entire enclosure. In this respect, the use of the available computational resources is optimised.

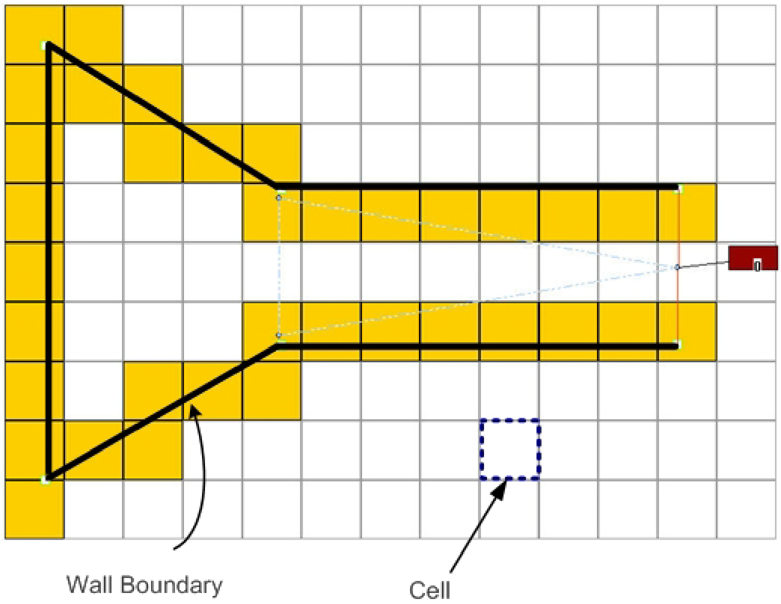


Figure 5-20 Assignment of wall boundaries to cells; shaded cells indicate the presence of wall boundaries

5.3.2 Behaviour System

The behaviour of an autonomous character refers to its improvisational and life-like actions in response to stimuli in its environment. The behaviour is governed by the Behaviour System which acts as the brain of the agent. The Behaviour System receives sensory information regarding the location of exits, obstacles and other agents from the Perception System. This information is then used in conjunction with the internal behavioural traits of the agent for controlling the decision making processes. Examples of internal behavioural traits of the agent include response time, building familiarity and urgency.

Response Time

Previous evacuation studies have shown that people do not always begin the evacuation at the time of the sounding of fire alarms. There is normally a time delay between the perception of the fire alarm by the occupants and the start of the evacuation [Proulx 1994]. This time delay is simulated by the Response Time attribute, which is a measure of the pre-movement time incurred by the occupant.

In bEX-Continuous, the agents do not start to move until the current simulation time equates or exceeds their Response Times. Stationary agents are modelled by assigning them very large Response Time values.

Familiarity

Familiarity can have an impact on the behaviour of occupants in the sense that it can lead to the agents moving towards the less obvious exits instead of the nearest ones. Previous evacuation studies have shown that people tend to use familiar routes in evacuation situations [Sime 1993]. For instance, during an egress, people would be inclined to move towards the door through which they entered the building albeit be a longer path instead of using a totally unfamiliar path which could be leading to a much nearer exit.

In the Continuous Model, the agents have the ability to maintain a localised understanding of the structure by storing a list of external exits. The agents assign more priority to familiar exits than others, however, when the list of familiar exits has not been specified, the agents select the nearest exit by default as their final destination. The navigation of the agents from their initial locations towards their familiar exits is achieved by use of waypoint information from the Distance Route Map described in Section 5.3.1.2.1.1.

Urgency

The Urgency attribute is used to differentiate between some the different forms of behaviours exhibited by the agents. For instance, agents with high urgency values will demonstrate competitive behaviour whereby the agents will tend to compete at the exits. This behaviour leads to frequent bodily contact amongst the agents and also illustrates the emergence of pushing. Whereas agents with low urgency values will demonstrate queuing behaviour, whereby the agents maintain a further separation from each other and evacuate in a more organised fashion.

The next section describes the implementation of behaviours of the autonomous character using a component oriented approach. The behaviours are modelled as simple components which can serve as building blocks for simulating more complex and life-like behavioural routines. Moreover, using this approach allows the behaviours to be implemented and tested individually, followed by an incremental integration into the Behaviour System. The integration process includes the verification of the combined effects of the individual behaviours. Examples of component behaviours include Seek, Evade, Wall Avoidance, Agent Avoidance, Queuing and Lane Formation.

5.3.2.1 Seek

Seek is used to steer an agent towards a specific goal, for instance an exit or a location within the environment. During seek, an agent attempts to move towards its current target at maximum travel speed. This behaviour takes into account the speed and turning rate of the agent in the sense that an agent moving at a slower speed has a tendency to turn faster. Moreover, there is a limiting factor as regards turning rates so that occupants cannot turn more than a certain angle within a time step of the simulation.

Figure 5-21 illustrates the current position and velocity Vc of agent P. The velocity required for P to reach the target is the Desired Velocity and is calculated as shown:

1. DesiredVelocity = Position of Target – Position of Agent
2. DesiredVelocity.Normalised()
3. DesiredVelocity.Scale(max_speed)
4. SteeringVector = DesiredVelocity - CurrentVelocity

Where *max_speed* is the maximum speed attainable by the agent

Algorithm 5-5 Calculating the desired velocity of an agent

The difference in the Desired Velocity and the Current Velocity results in the acceleration, Fs, needed to turn the agent towards the target as shown in **Figure 5-21**.

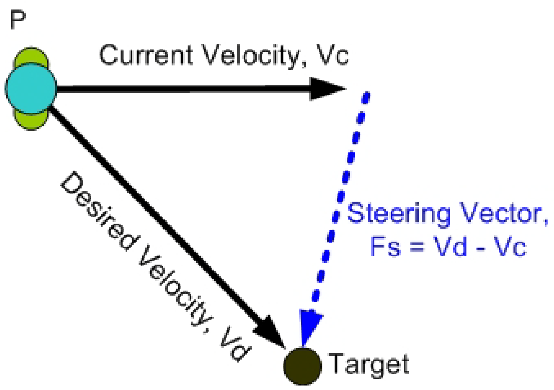


Figure 5-21 Calculating the desired velocity

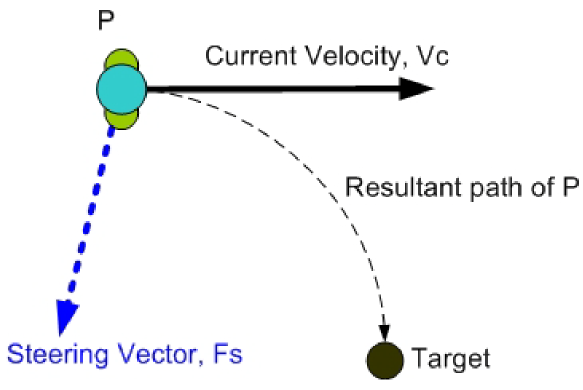


Figure 5-22 Adding the steering vector to the current velocity

When the steering vector Fs is added to the current velocity Vc, it causes P to change its direction of motion. The resultant path taken by P is as shown in **Figure 5-22**.

Moreover, the seek behaviour can be configured by altering the ratio R of *max_speed* to *max_acceleration*. The features that can be configured are as follows:

- The time taken for the occupant to accelerate to its maximum speed
- The time taken for the occupant to decelerate to rest.
- The rate of change of direction of the occupant (turning rate). In other words, R controls the turning radius of the agent, such that a higher value of R implies a larger turning radius. This is as shown in **Figure 5-23**.

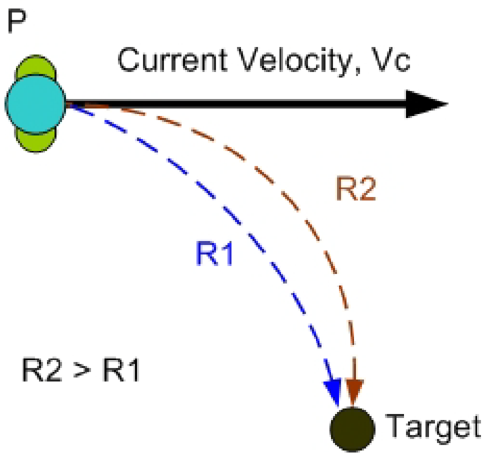


Figure 5-23 Rate of change of direction

5.3.2.2 Evade

Evade is opposite to the Seek behaviour whereby the agent steers away from the target. This can be used to steer an agent away from a location of danger, for instance, a fire, smoke etc. **Figure 5-24** illustrates an agent P with a current velocity Vc wanting to steer away from the target. In this case, the desired velocity of the agent is a vector which points away from the target.

1. DesiredVelocity = Position of Agent - Position of Target
2. DesiredVelocity.Normalised()
3. DesiredVelocity.Scale(max_speed)
4. SteeringVector = DesiredVelocity - CurrentVelocity

Where *max_speed* is the maximum speed attainable by the agent

Algorithm 5-6 Calculating the vector required to evade

The difference between the desired velocity and the current velocity results in a steering vector Fs as shown in **Figure 5-24**.

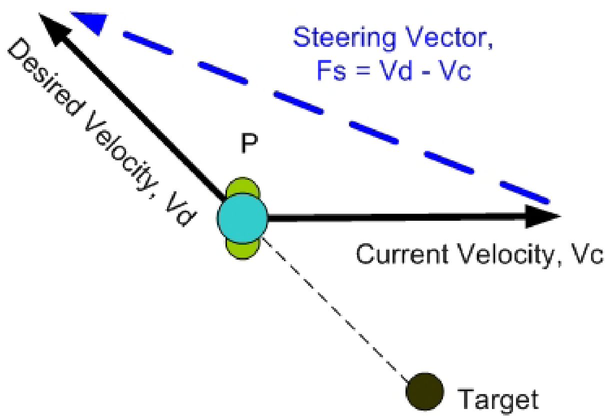


Figure 5-24 Desired velocity pointing away from target

This fleeing vector Fs is then added to the current velocity Vc, and the resultant vector causes P to change its direction such that it steers away from the target.

5.3.2.3 Wall Avoidance

An agent navigating in continuous space is required to detect and avoid collision with static obstacles such as walls in its immediate environment. Although the location of the waypoints in the Navigational Graph (Section 5.1.3) of bEX-Continuous guides the agents away from the walls, however there may be scenarios where the agents could end up navigating very close to the walls especially in crowded situations. In addition, agents could be set to wander around in their environment in search for visual cues, rather than following waypoints on the Navigational Graph. In these respects, a mechanism for avoiding walls is necessitated. This is implemented using the Wall Avoidance behaviour. In order for the collision avoidance to be effective, the character needs to detect collisions which could happen in the future on its current trajectory. To simulate this feature, the current velocity vector of the agent is projected by a constant time as shown in **Figure 5-25**. Any obstacle that intercepts this newly projected vector is a potential collision threat which the agent needs to avoid. To further enhance the wall avoidance mechanism, each agent has 2 probes namely P1 and P2 on its side as shown in **Figure 5-26**. A similar technique has been used by Reynolds for simulating the avoidance of obstacles [Reynolds 1999]. However, when Reynold's technique was applied, the motion of the agents with low travel speeds seemed to be abrasive as the agents got nearer to the walls, such that the agents' headings would oscillate abruptly. As this behaviour was not realistic, this technique was enhanced, as outlined further in this section, in order to remove the oscillatory effect.

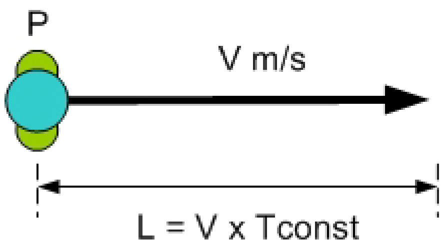


Figure 5-25 Projection of velocity by a constant time.

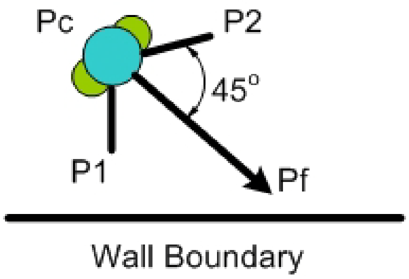


Figure 5-26 Using velocity projection for wall detection.

Key:

V : Current Velocity of P

Tconst : Constant time

Pc : Current position of P

Pf : Future position of P

P1 and P2 : Probes for detecting collisions on the agent sides

In the Continuous Model, the mechanism of the wall avoidance behaviour is implemented using vector projections. As shown in **Figure 5-27**, V1 is the current velocity which has been scaled by a constant time. The wall which is selected as the nearest potential collision threat is treated as a vector W.

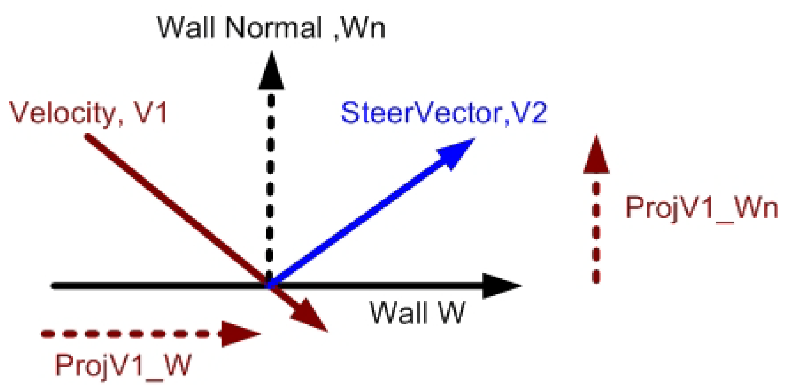


Figure 5-27 Calculating the steering vector for wall avoidance.

Key:

ProjV1_W : Vector projection of velocity V1 on wall W

ProjV1_Wn: Vector projection of velocity V1 on wall normal Wn

The steer vector V2 is the sum of the projections of V1 on W and V1 on Wn

Projection of V1 on W

x = [V1.Dot(W)/ W.Dot(W)]. Wx

y = [V1.Dot(W)/ W.Dot(W)]. Wy

Projection of V1 on Wn

This requires the vector V1 to be reversed so that the resultant
projection is in the correct direction.

Equation 5-1

V1.Scale(-1)

x1 = [V1.Dot(Wn)/ Wn.Dot(Wn)]. WnX

y1 = [V1.Dot(Wn)/ Wn.Dot(Wn)]. WnY

Steer Vector V2

V2 = ProjV1_W + ProjV1_Wn

Where *Dot* is the computation of the Dot Product of two vectors

The resultant steer vector V2 is used to steer the occupant away from the collision site with the wall and is proportional to the direction of travel, speed and angle with respect to wall. Moreover, since the calculation of the steer vector V2 ensures that V2 is at an angle to the wall, the resultant path of the agent is smoother during wall avoidance.

5.3.2.4 Agent Avoidance

People tend to react differently when located in regions of high population density. It is a social norm that people tend to maintain a desired separation (also known as inter-person distance) from each other for example when talking to each other, or moving in a crowd. However, when this inter-person desired distance is breached, the behaviour changes significantly [Pan 2006a]. To simulate this behaviour, an algorithmic procedure called Separation was implemented. Each occupant receives sensory information regarding the location of its neighbours within the local neighbourhood from the Perception System. As illustrated in **Figure 5-28**, the assessing agent, P, considers each neighbour P1 in turn and computes the vector, Fr, representing the difference in the locations of P1 and P. This vector is normalised and scaled to be inversely proportional to the squared of the inter-person distance so that distant neighbours would have a negligible effect on the assessing agent. The sum of the vectors, serves to steer agent P in such a way that it maintains a desired distance between its neighbours.

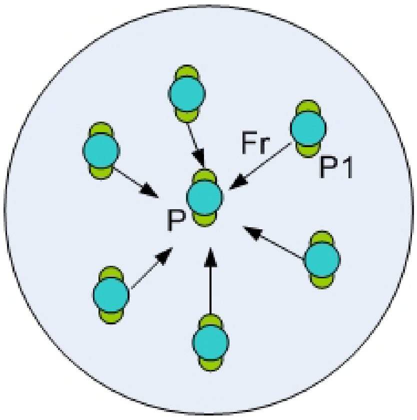


Figure 5-28 Computing the vectors for Separation Behaviour

However, the Separation Behaviour did not allow the close packing of agents. Moreover, this behaviour assumed that people moving directly ahead and on the side of someone's path have the same influence which was not realistic. In effect, people will have a preference of for avoiding others who are directly ahead of them rather than from those who are on their sides. Therefore, to circumvent these issues, another behavioural algorithm called Avoid Neighbours was implemented in which the angles of approach of the various agents in a crowd were taken into consideration.

Each agent receives sensory information regarding the location of its neighbours by using a perception box to scan the space ahead. The length of the box is set such that it is proportional to travel speed of the agent. This ensures that faster moving agents are more responsive in avoiding the agents ahead in their vicinity. As illustrated in **Figure 5-29**, the assessing agent, P, considers neighbour P1 and computes the steer vector Vr which is required to modify the trajectory of P in order to avoid collision. The magnitude of Vr is scaled as such that further neighbours will have less influence in the change of direction of P. When there are multiple neighbouring agents in the Perception Box, the sum of all the individual avoidance vectors due to each agent is used to steer agent P. The pseudo code for the algorithm is as follows:

1. Retrieve list of agents in Perception Box
2. Initialise vector Fsum
3. **FOR** each agent in list
4.   Vd = Position of P – Position of P1
5.   Calculate perpendicular projection of Vd on Vc , Vr
6   Add Vr to Fsum
7. **END FOR**
8. Add Fsum to Vc

Algorithm 5-7 Calculating the avoidance vector, Fsum of agent P, as sum of avoidance vectors due to neighbouring agents

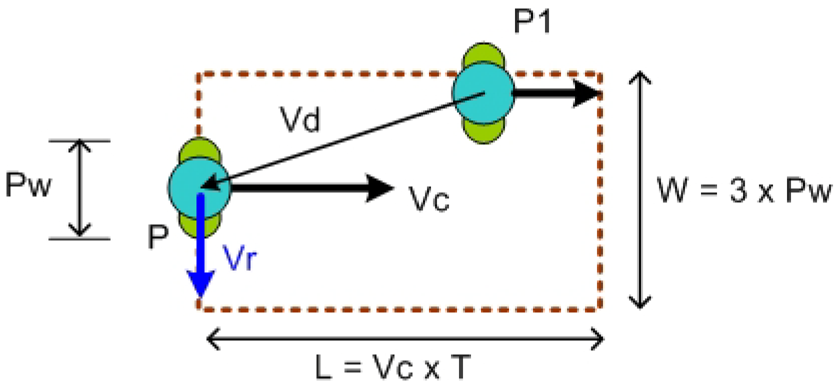


Figure 5-29 Using perception box (bounded by dotted line) to scan the space ahead

Key:

Vc : Current velocity vector
Vd : Vector from P1 to P
Vr : Resultant steer vector
Pw : Width of P
L : Length of Perception Box
W : Width of Perception Box
T : Constant time for projecting the current velocity Vc

The overall effect of the Agent Avoidance behaviour is that that the agents are seen to spread out from each other when the space is available but on the other hand, the agents demonstrate close packing in crowded areas and when the enclosure is constricted. In other words, the agent has the capability of dynamically adapting to constantly evolving conditions within its vicinity.

5.3.2.5 Queuing

Queuing is often observed in regions of high population densities, for instance, a crowd gathering around an exit and areas where people movement is restricted due to the nature of the surrounding geometry, such as, commuters waiting to pass through turnstiles at tube and train stations. This behaviour emerges as a result of the self-organisation of people in order to allow for movement in an orderly fashion and thus leading to a more effective circulation and evacuation. In the Continuous Model, this behaviour is simulated using the Queuing Behaviour which takes into account an agent's wait attribute. An agent considers neighbouring agents within its perception box (**Figure 5-29**) and identifies any agent which could potentially impede its forward movement as shown in **Figure 5-30**. This routine is applied only if the neighbouring agent is within the personal space of the assessor.

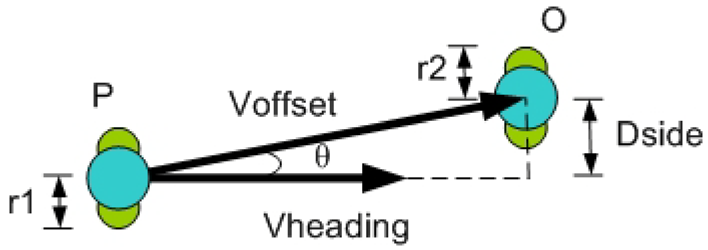


Figure 5-30 Checking if Agent O is a potential obstruction to forward movement of Agent P

Key:

Voffset:	: Vector from P to O (computed by the difference of locations of P and O)
Vheading	: Current heading of the agent
r1	: Radius of P

r2	: Radius of O
θ	: Angle between vectors Vheading and Voffset
Dside	: Distance of O to the side of P

The determination of whether a neighbouring agent is an obstruction or not is computed as follows:

1. Voffset = Position of O – Position of P
2. Dside = Length of Voffset x sin θ
3. **IF** Dside is less than sum of radius of P and O
4. O is an obstruction
5. **ELSE**
6. O is not an obstruction
7. **END IF**

Algorithm 5-8 Obstruction check procedure

If an obstruction is found, the assessing agent enters waiting mode and ceases movement. It then tries to move again after a certain number of time steps (as dictated by the wait attribute). Also, an agent in waiting mode can resume motion if there are no other agents within its personal space. The agents demonstrate Queuing Behaviour only if they have a low urge to exit as defined by the Urgency attribute. On the other hand, a high urge to exit will compel the agents to demonstrate competitive behaviour, whereby each agent attempts to reach their respectively destinations as soon as they possibly can.

The waiting procedure for the Queuing Behaviour is as follows:

```
IF an obstructing agent found
        Stop
        IF agent has waited for a number of time steps equal to waiting attribute
                Move
        ELSE
                Wait
        END IF
ELSE
        Move
END IF
```

Algorithm 5-9 Waiting Procedure for Queuing Behaviour

5.3.2.6 Emergent Lane formation

In high density crowd movement it is common to see the phenomenon of self organisation in bidirectional flows [Helbing 1997]. This self organisation is in the form of patterns comprising of long lanes of people moving in opposite directions. The lane formation occurs as result of the people trying to follow other people in the crowd who are moving in the same direction as them. In other words, the people are seen to follow paths of least resistance.

In bEX-Continuous, the Agent Avoidance behaviour (Section 5.3.2.4) of the agents is implemented such that the agents can identify not only the location but also the direction of flow of other neighbouring agents. This capability enables the agents to differentiate between agents moving in uni-directional flows or contra-flows. As illustrated in **Figure 5-31**, the assessing agent, P, scans the space ahead and identifies an agent O within its perception box. Agent P then computes the dot product of its heading (Pheading) and the heading of agent O (Oheading) in order to determine the direction of O.

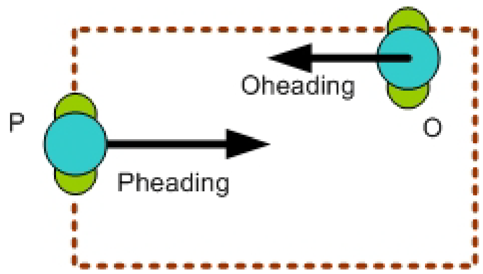


Figure 5-31 Identifying agent moving in uni-directional flow or cross-flow

The pseudo code for this algorithm is as follows:

```
1. Pheading = Heading of P
2. Oheading = Heading of O
3. Calculate dot product of Pheading and Oheading, DotP
4. IF DotP > 0
5.              P and O are moving in the same direction
7. ELSE
8.              P and O are moving in opposite directions
```

Algorithm 5-10 Direction identification Procedure

The agents in bEX-Continuous demonstrate emergent lane formation. The term emergent is used to denote that this behaviour has not been specifically programmed in the behavioural system, but which occurs as a result of the local interactions of individual agents moving in opposing flows.

The Agent Avoidance behaviour of the agents (Section 5.3.2.4) and the procedure for identifying the direction of flows (**Algorithm 5-10**) are implemented such that agents moving in contra-flows have higher avoidance criteria than those moving in unidirectional flows. The resulting effect is that the agents are seen to follow mostly neighbouring agents which are moving in the same direction as them. In other words, the agents tend to follow paths of least resistance which accounts for the emergence of lane formation. The lane formation behaviour is as illustrated in Figure 5-32 in which arrows have been superimposed to indicate the direction of the lanes.

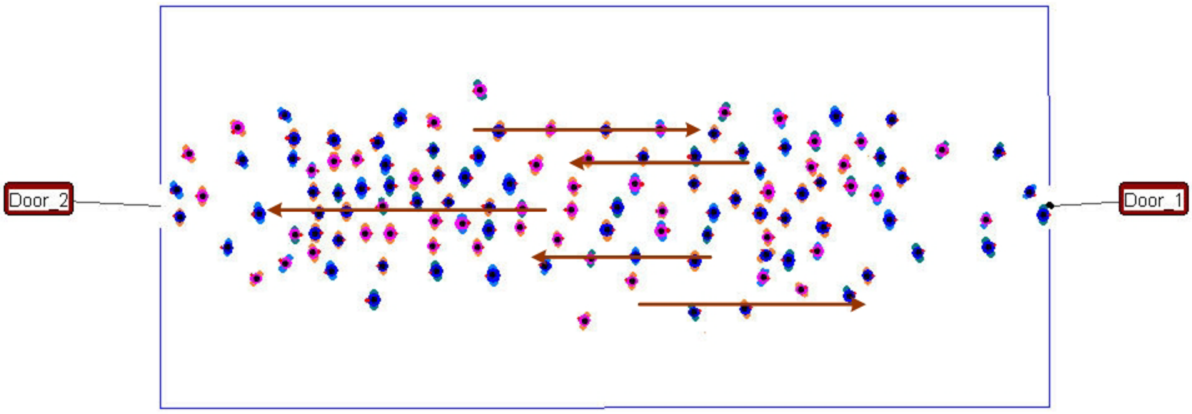


Figure 5-32 Emergence of Lane Formation behaviour

5.3.2.7 Agent Travel Speed Settings

One of the outcomes of the analysis conducted in Chapter 2 was the relationship between crowd density and travel speed. The Continuous Model includes both an adaptive and SFPE setting for agent travel speeds.

(1) Adaptive Speed Setting

In the adaptive setting, an agent considers neighbouring agents within its perception box (Section 5.3.2.4) and decelerates upon detection of nearby agents which are in front and are moving at a slower pace. The procedure for detection of potential obstructions in front of an assessing agent is as shown in **Algorithm 5-8**. Once a potential obstruction has been identified, a braking vector is applied on the assessor. This vector has a decelerating effect on the agent thereby reducing its travel speed. Moreover, the magnitude of the braking vector is inversely proportional to the distance in between the assessor and the obstruction such that distant agents will have a lesser influence on the assessor's travel speed. Furthermore, as the crowd density increases, the distance in between the agents decreases, which in turn leads to a reduction in their travel speeds. The resultant effect of this braking behaviour enables the agents to adapt their travel speeds according to population density in their surroundings.

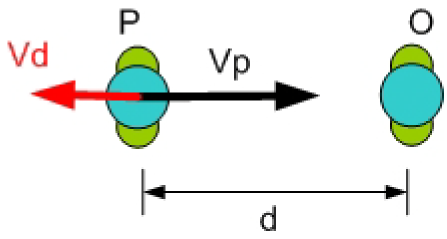


Figure 5-33 Calculating a braking vector

Key:

P : Assessing agent
O : Obstructing agent
Vp : Current velocity of P
Vd : Braking vector

(2) SFPE Speed Setting

In the SFPE setting, the travel speed of each agent is dictated by a speed density relationship such as:

$$S = k - akD$$ [Nelson 2002] **Equation 5-2**

Where:

S = Travel speed
D = Population density
'a' and 'k' are constants.

As the population density within a room or compartment may be non-uniform and also, since the population density in a room might not necessarily represent the density to which every agent is actually exposed, the agents adopt a localised density calculation approach. Using this approach, the agents project a density box of length and width of 1.5 m ahead of them, and identify the number of agents within the boundaries of the box. The number of people within the box area is indicative of the population density to which the assessor agent is exposed and the corresponding travel speed is calculated from the travel speed and density equation.

5.3.3 Locomotion System

The Locomotion System regulates the actual motion of the continuous agent. This system receives its input from the Behaviour System. The input is in the form of directional vectors which serve to execute the movement routines. In addition, this system ensures the agent is in a valid state, that is, it is not overlapping with wall obstructions and other agents.

5.3.3.1 Contact with Walls

Although the waypoints in the Navigational Graph and the Wall Avoidance behaviour will try to guide the agents away from the waypoints, there are situations in which contact with the walls is inevitable. For instance, when agent avoidance is carried out in close proximity to the location of walls, an agent may come into contact with those walls. If it was not for this behaviour, an agent would not require this test. In addition, the agents could come into contact with the internal corners as illustrated by agent P in **Figure 5-34**.

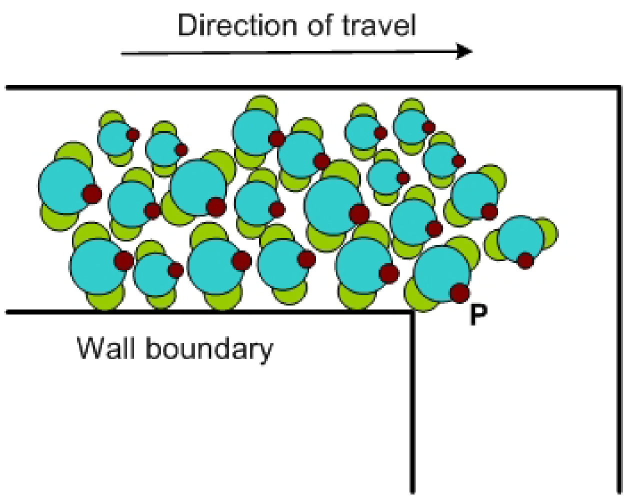


Figure 5-34 Agents in contact with walls

The solution for preventing the agents from overlapping the wall boundaries and the internal corners uses a combination of vector projections for the walls and the use of a body circle (**Figure 5-35**) around each agent. As illustrated in **Figure 5-36**, an agent's body circle is overlapping with a wall which is treated as a vector W and the vector Vadjust is used to shift the centre location of the body circle to counteract the overlap (the amount of overlap equals the length of Vadjust). This adjustment is applied when the agent's centre location lies in between the starting point and ending point of the wall. The pseudo-code for the adjustment is as follows:

1. Calculate vector V from starting point of wall 'S' to centre of body cicle 'C'
2. Calculate Vadjust, projection of V on the wall normal
3. Add C and Vadjust to obtain new location of centre 'C1' of body circle

Algorithm 5-11 Avoid wall overlap procedure

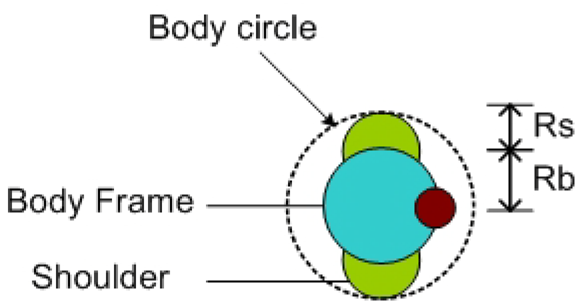


Figure 5-35 Body circle of an agent

Key:

Rs : Radius of shoulder
Rb : Radius of body

The mechanism for avoiding overlaps at the cornering regions in an enclosure is as shown in **Figure 5-37**. The pseudo-code for this procedure is as follows:

1. Determine the nearest vertex 'S' of the edge W from the centre of body circle 'C'
2. Calculate vector Vadjust from S to C
3. Calculate amount of overlap, L = Body Circle Radius – Length of Vadjust
4. Normalise and Scale vector Vadjust by the value of L
5. Add C and the scaled Vadjust to obtain new location of centre 'C1' of body circle

Algorithm 5-12 Avoid overlap at corners procedure

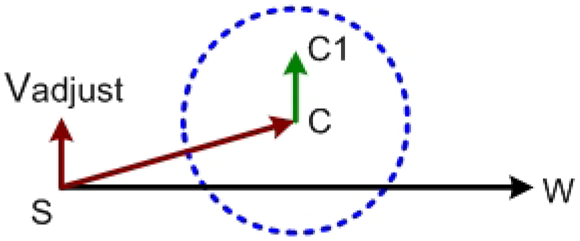


Figure 5-36 Adjusting the centre of the body circle

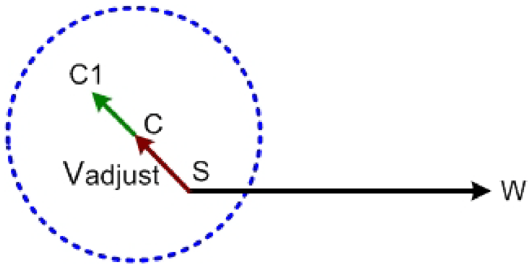


Figure 5-37 Adjusting body circle at corner

Key:

C	: Centre of body circle
C1	: New projected centre of body circle
S	: Starting point of wall
W	: Vector representing wall
Vadjust	: Vector from S to C used for adjusting location of body circle

5.3.3.2 Contact with Other Agents

The Agent Avoidance behaviour allows agents to maintain their desired interpersonal separation from each other. However, in heavily crowded scenarios, physical contact between the agents is inevitable. Such a situation is shown in Figure 5-38 where P is the assessing agent and O is the obstructing agent with velocities Vp and Vo respectively. The pseudo-code for an assessing agent P avoiding overlap with other neighbouring agents is as follows:

1. Identify nearest agent, O
2. Calculate the distance 'D' between the centres of P and O
3. **IF** D is less than sum of Rp and Ro
4. Calculate vector 'Vadjust' from centre of O to centre of P
5. Calculate amount of overlap, L = (Rp+Ro) – D
6. Normalise and Scale Vadjust by the value of L
7. Add the scaled vector Vadjust to the original centre location of P to obtain the new centre location
8. **END IF**

Algorithm 5-13 Avoid overlap procedure

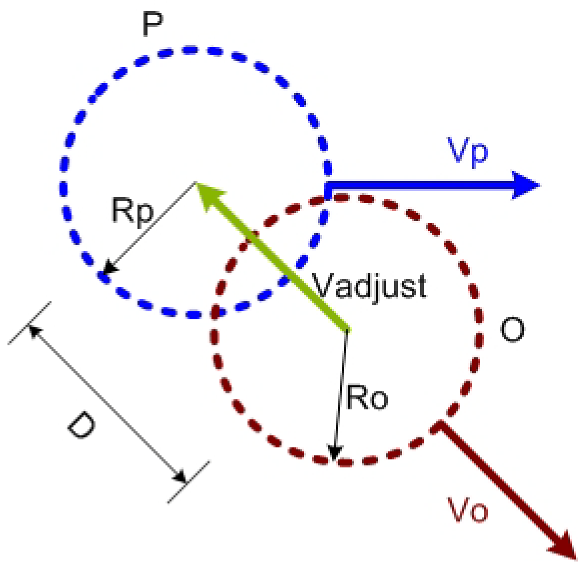


Figure 5-38 Determining overlap in between agents

Key:

Vp	: Velocity of P
Vo	: Velocity of O
Rp	: Radius of P
Ro	: Radius of O
Vadjust	: Vector from centre of O to centre of P
D	: Distance between centres of O and P

Continuous models are prone to the deadlock problem [Thompson 1994, Pathfinder 2009]. This is because the handling of collision detection and response amongst agents at such levels of granularity can lead the agents being unable to move past each other; therefore causing the simulation to stop prior to completion. This problem is more common when modelling heavily crowded scenarios and when the agents are trying to navigate in constricted areas of the enclosure. Therefore, such models require additional conflict resolution approaches to cater for different scenarios which could lead to deadlocks. Examples of such scenarios include:
(a) Multiple agents simultaneously moving towards a common target
(b) Agents moving in contra-flows in narrow corridors

However, in bEX-Continuous, the solution to this deadlock problem is an emergent property of the underlying behavioural and movement mechanisms. The combination of the agent overlap resolution coupled with the Seek and Agent Avoidance behaviours (Section 5.3.2) causes the agents to avoid overlap and slide past each other even at high population densities. Therefore, this mechanism acts as an effective conflict resolution mechanism without having to explicitly implement rules for avoiding deadlocks. Moreover, this technique accounts for the emergence of pushing behaviour amongst the agents.

5.3.3.3 Agent Update

The Locomotion System includes an update mechanism which computes the velocity and orientation of an agent in each movement cycle. This update takes place at every time step of the simulation. In bEX-Continuous, every time step constitutes a time span of $(1/12)^{th}$ of a second, which is equal to the time span used in buildingEXODUS. The Agent Update mechanism also takes into account the turning rate of the agents; that is, the maximum angle that an agent can turn in one movement cycle. In previous studies, the average turning rate of a person has been identified to be 100^0 per second [Thompson 1994]. The turning rate criterion has been implemented in bEX-Continuous whereby an agent is not allowed to rotate by more than a certain angle in one time step. The maximum turning angle per time step is calculated using Equation 5-3.**Error! Reference source not found.**

*Maximum turning angle per time step = Maximum Turning Angle in 1 s * Time Step* **Equation 5-3**

Since every time step constitutes a time span of $(1/12)^{th}$ of a second, the maximum turning rate of an agent per time simulation time step is assigned to be $(100/12)^0$. The turning rate limit ensures that the agents demonstrate smoother and more realistic paths when changing their directions of motion.

The pseudo code for the update mechanism is as follows:

1. Compute sum of resultant vectors, *Ftotal*, from the combination of selected behavioural Routines
2. Limit *Ftotal* to the maximum acceleration of agent
3. Update velocity of agent, *newVelocity* = *oldVelocity* + *Ftotal*
4. Limit the speed of agent to the maximum travel speed of agent
5. Limit the turning rate of agent to a maximum (100/12) degrees
6. Update location of agent, *newLocation* = *oldLocation* + *newVelocity*
7. Update agent body orientation

Algorithm 5-14 Agent update procedure

5.4 Exit Flow Capability (Interactive Doors)

An important factor affecting evacuation performance is the operating characteristics of external exits [Hankin 1958, Fruin 1971, Galea 2004a]. The exit flow capability is defined as the number of people passing per unit metre of the door width per unit time (Unit Flow Rate). Acceptable values for unit flow rates have been specified by previous studies and also by building regulatory authorities e.g. HMSO [HMSO 2010]. The Continuous Model includes the capability of capping the flow rates at exits to these acceptable values.

This functionality is achieved via an Interactive Door mechanism whereby each door constantly monitors the number of agents passing through it. When the unit flow rate exceeds the maximum allowed value, a delay timer is triggered which controls the passage of the occupants through the door. The delay timer value is computed by using the time of passage of the last occupant through the door as shown in **Equation 5-4**. The time delay must elapse before the next agent is allowed through the exit. This mechanism prevents the flow rate at the exit from exceeding the maximum specified UFR. In contrast, if the door is set to free conditions, the flow rate capping mechanism is not applied.

$$\text{Unit Flow Rate, UFR} = \frac{NumberOfPeople}{Width * Time}$$

Where:

Time = Time last agent through exit(Tlast)

– Time first agent through exit(Tfirst)

$$\text{Therefore, } Tlast = \frac{NumberOfPeople}{UFR * width} + Tfirst$$

Equation 5-4

Time Delay = Tlast – Current Simulation Time

5.5 Validation of the Continuous Model using IMO Validation Tests

Validation is defined as the systematic comparison of model predictions with reliable information [Galea 1997a]. Validation is an essential and ongoing phase within the software development life cycle of egress models. Validation can comprise of Component Testing, Functional Validation, Qualitative Validation and Quantitative Validation [Galea 1997a]. Moreover, it should be noted that no degree of successful validation will prove the correctness of a model. Nevertheless, confidence in the approaches employed is established the more the model is successfully validated in a wide range of scenarios.

This section describes the validation of the movement algorithms used in the bEX-Continuous model based on some of the cases in the International Maritime Organisation (IMO) document entitled "Interim Guidelines for Evacuation of New and Existing passenger ships**" [IMO 2002]**. The IMO test cases were selected as they are classified as an existing standard for the validation of evacuation simulation tools. Moreover, those test cases are devised for flat terrains which accounts for their applicability for validating egress tools used in the building environment. The IMO document includes cases for the purposes of Component Testing and Qualitative Validation which are described in Sections 5.5.1 and 5.5.2. The simulation set up and population characteristics used for the test cases were obtained from the IMO document specifications [IMO 2002].

5.5.1 Component Testing

Component testing involves checking the major software sub-components. This entails running the software through a series of tests to ensure that the various components are operating as intended.

5.5.1.1 Maintaining set walking speed in corridor

This test involves checking if an agent of walking speed of 1 m/s can maintain this speed in a corridor of length 40 m and width 2 m. The agent in the continuous model demonstrated to cover this distance in 40 s. Therefore, the continuous model passed this test.
For the test, it was ensured that as soon as the simulation started, the heading of the agent would be adjusted so that it faced its desired target (which is the exit in this case) and the

velocity vector of the agent was scaled to the maximum speed of 1 m/s. This is because when the agent is first created, it might be facing away from its desired target and by the time it would have rotated to its desired target and accelerated from rest to 1 m/s, the simulation would have advanced by a few time steps, resulting in the agents taking longer than 40 s to cover the entire distance. These adjustments ensure that the agents start to move instantly at their specified maximum speed towards their desired target.

5.5.1.2 Exit flow rate

This test involved generating 100 agents in a room of 8m by 5m with a 1 m exit as shown in **Figure 5-39** and recording the flow rates at the exit and the evacuation times. In order to obtain the actual flow rates predicted by the model, no external steering was imposed on the exit and the exit was set to Free-flow conditions. An average evacuation time of 63.8 s and average flow rate of 1.6 persons/m/s were obtained.

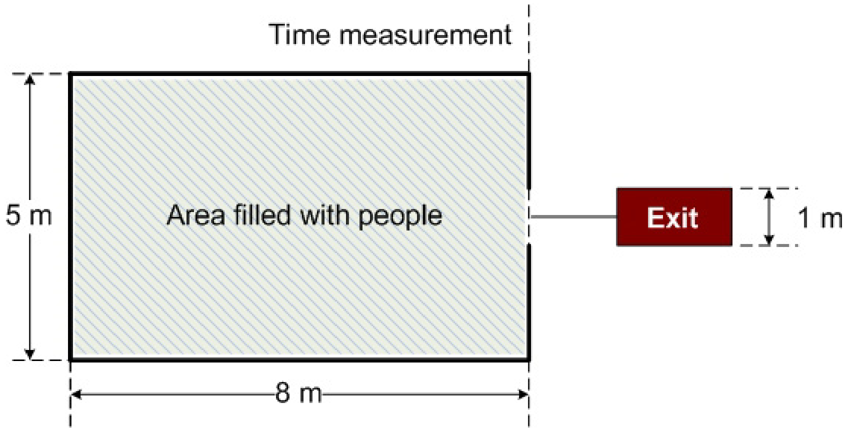


Figure 5-39 Rectangular room with a 1 m door

In order to investigate the validity of the results produced by the bEX-Continuous model, further experiments were performed on the same geometry. These experiments involved gradually increasing the number of occupants in the geometry and recording the corresponding evacuation times. The results were compared against the hand-calculation methods of Predtechenskii and Milinskii [Rogsch 2005] and are as illustrated in Figure 5-40. The Predtechenskii and Milinskii data set was chosen because it was representative of the conditions being modelled for the geometry presented in Figure 5-39. The bEX-Continuous results are seen to be comparable to those predicted by Predtechenskii and Milinskii.

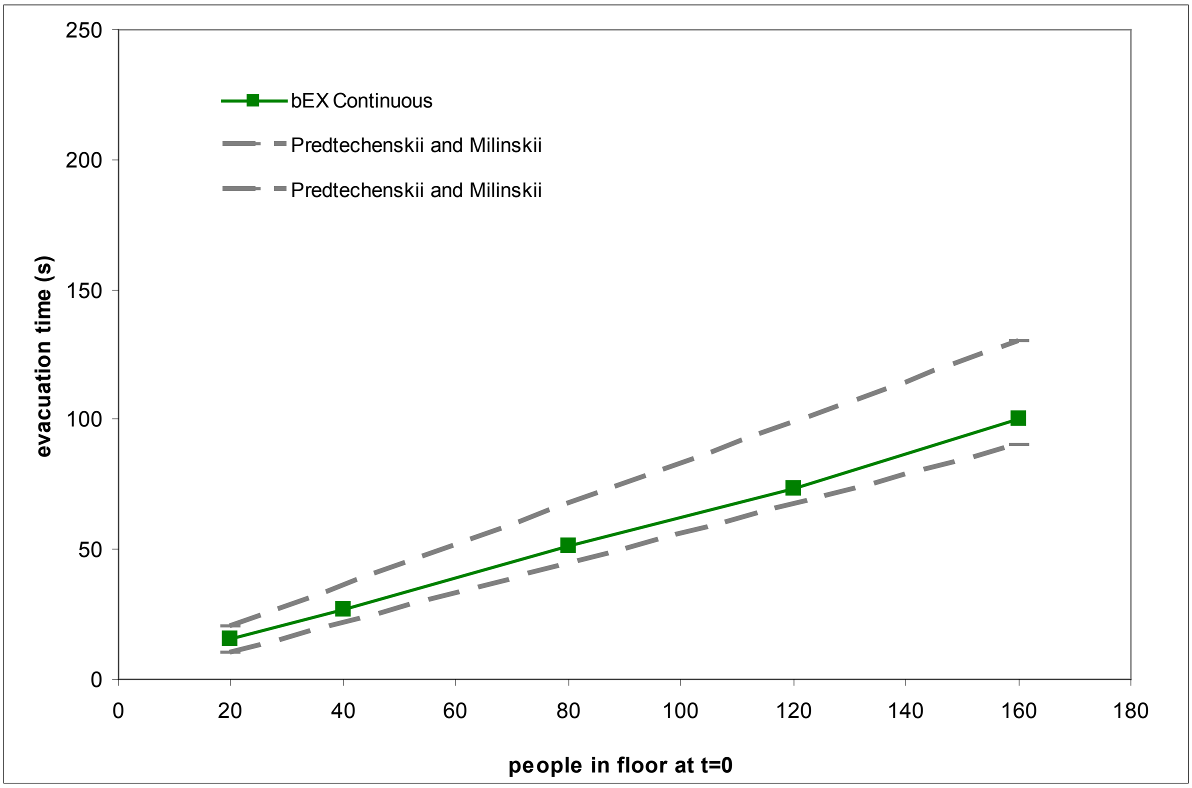


Figure 5-40 Time taken for a specific percentage of people to evacuate

5.5.1.3 Response Time

This test was based on the same geometry used in Section 5.5.1.2 but only with 10 randomly generated agents. Response times uniformly distributed in the range between 10 s and 100 s were imposed on the agents. This test involved verifying whether the agents started to walk at the appropriate times. During the simulation, the agents remained stationary until the simulation clock reaches their individual response times.

5.5.1.4 Rounding Corners

The aim of this test was to investigate whether 20 agents approaching a left hand corner will successfully navigate around the corner, without penetrating the wall boundaries. The geometry is as shown in **Figure 5-41**.

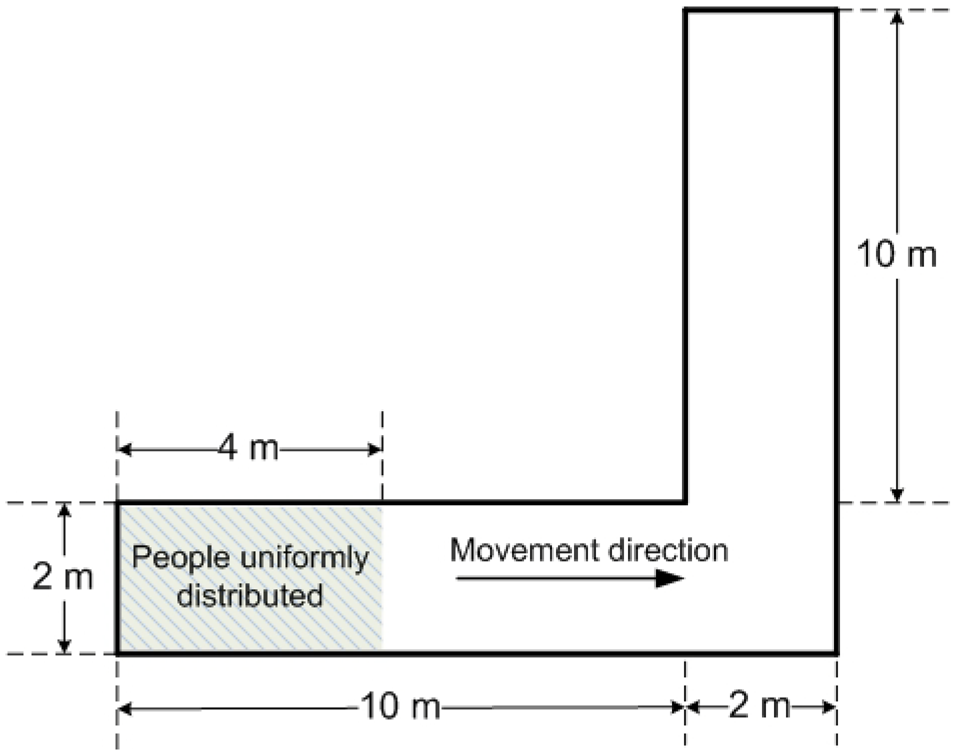


Figure 5-41 Transverse corridor

During the test, the Agent Avoidance Behaviour (Section 5.3.2.4) keeps the agents spread apart on their approach to the corner. Moreover, since the agents do not head directly for the waypoints, they do not converge at the inner corner. The *"Avoid Overlap with Wall"* algorithm (Section 5.3.3.1) also prevents the agents from penetrating the wall boundaries. **Figure 5-42** shows the agents navigating in the corridor without overlapping walls.

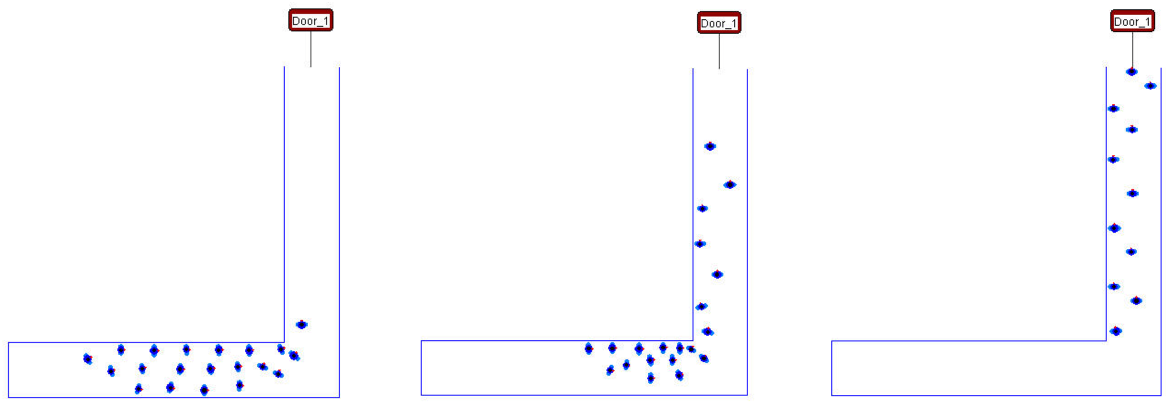


Figure 5-42 Locations of agents at different times during the simulation

5.5.1.5 Assignment of population demographic parameters

This test comprised of distributing the walking speeds of the agents over a population of 50 and verifying that the walking speeds are consistent with the distribution specified in the input. The population specifications for this test are as shown in **Table 5-3**.

The geometry which was chosen for this test was the one used in Section 5.5.1.2. The continuous model results are as follows; Minimum walking speed 0.97 m/s, Mean walking speed 1.3 m/s and Maximum walking speed 1.62 m/s, which illustrates the correct implementation of this feature in the Continuous Model.

Table 5-3 Walking speed on flat terrain used for IMO test 7

Population Groups - Passengers	Walking speed on flat terrain (e.g. corridors)		
	Minimum (m/s)	Mean (m/s)	Maximum (m/s)
Males 30-50 years old	0.97	1.3	1.62

5.5.2 Qualitative Validation

Qualitative Validation involves the comparison of predicted human behaviour with informed expectations. It is an important part of validation as it demonstrates whether the behavioural capabilities of a model are able to produce realistic behaviours.

5.5.2.1 Counterflow – two rooms connected via a corridor

The geometry specified for this test is as shown in **Figure 5-43** and the population based on **Table 5-3**. The initial distribution of the population is such that room 1 is filled from the left with the maximum possible density. Room 2 is filled with 0, 10, 50 and 100 occupants for different series of scenarios, and the occupants in both rooms start to move simultaneously, such that occupants in Room 1 move towards Room 2 and vice versa. The times taken for the last occupant in Room 1 to enter Room 2 are recorded and the expected result is that as the number of occupants is increased in Room 2, the time taken for the last occupant in Room 1 to enter Room 2 increases. The results are shown in **Table 5-4**.

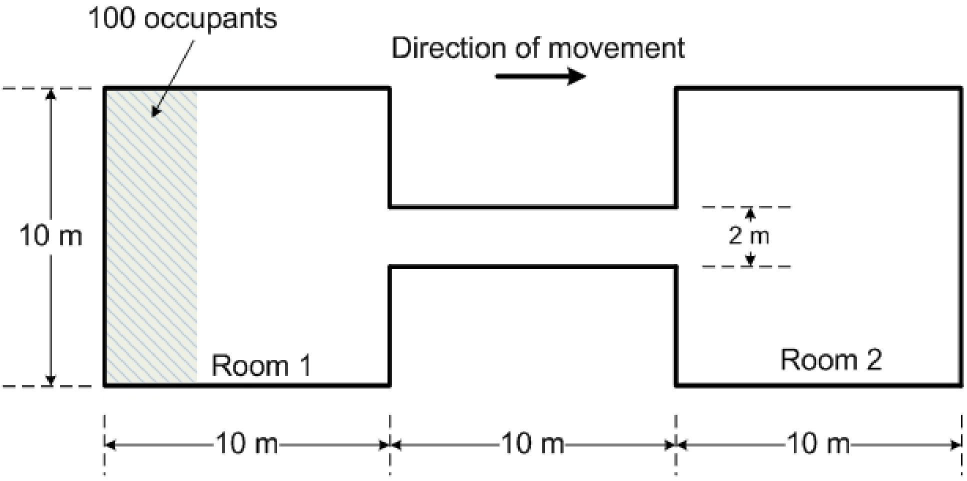


Figure 5-43 Two rooms connected via a corridor

Table 5-4 Time taken for last agent in Room 1 to enter Room 2

Number of people in Room 2	**Time for last agent in Room 1 to enter Room 2 (s)**
0	54.5
10	62.7
50	133.2
100	208.6

As expected, the time taken for the last agent in Room 1 to enter Room 2 increases as the number of agents in Room 2 increases. This illustrates that the agent avoidance strategies in place are operating as intended.

5.5.2.2 Crowd dissipation from a large public room

This test involves measuring the time taken for a crowd to dissipate from a large public room as shown in **Figure 5-44**. There are two scenarios examined: crowd dissipation through four doors; two of the doors (Doors 1 and 2) are shut with the crowd using the remaining two exits. The expected outcome is an approximate doubling of the time for all the agents to leave the room. The population specification is based on **Table 5-3**. The evacuation times for the two scenarios are as shown in **Figure 5-45**.

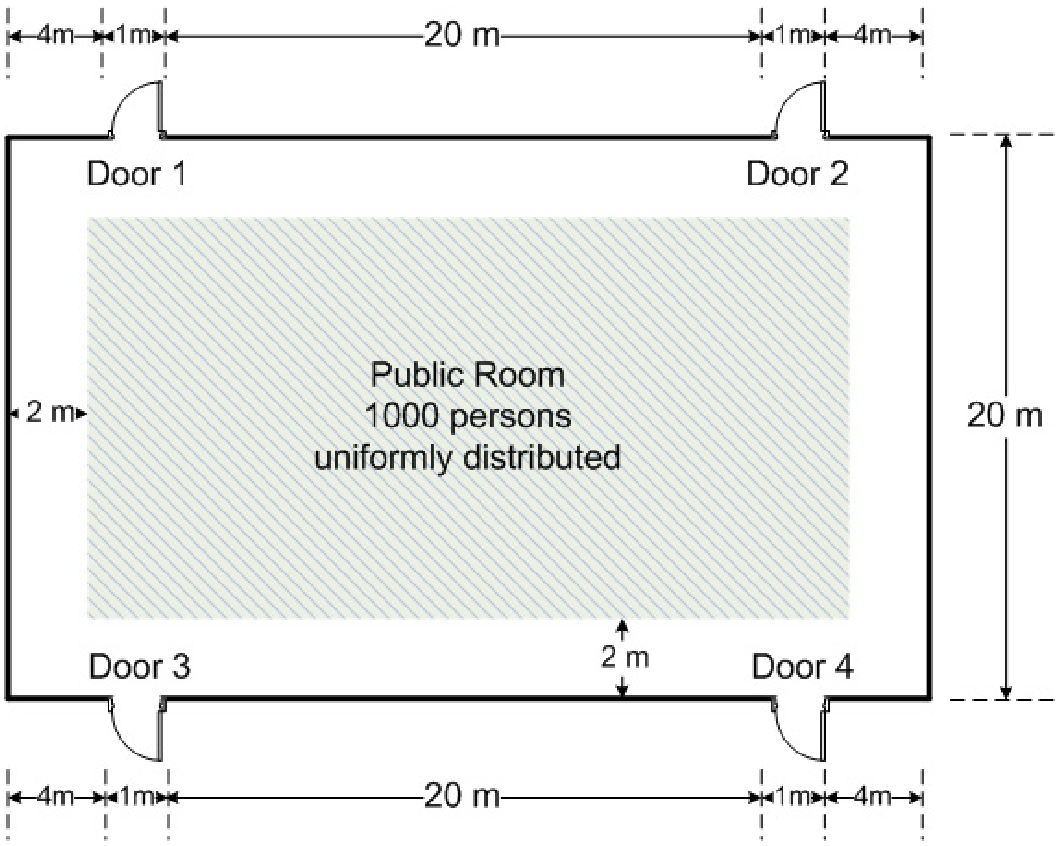


Figure 5-44 Exit flow from a large public room

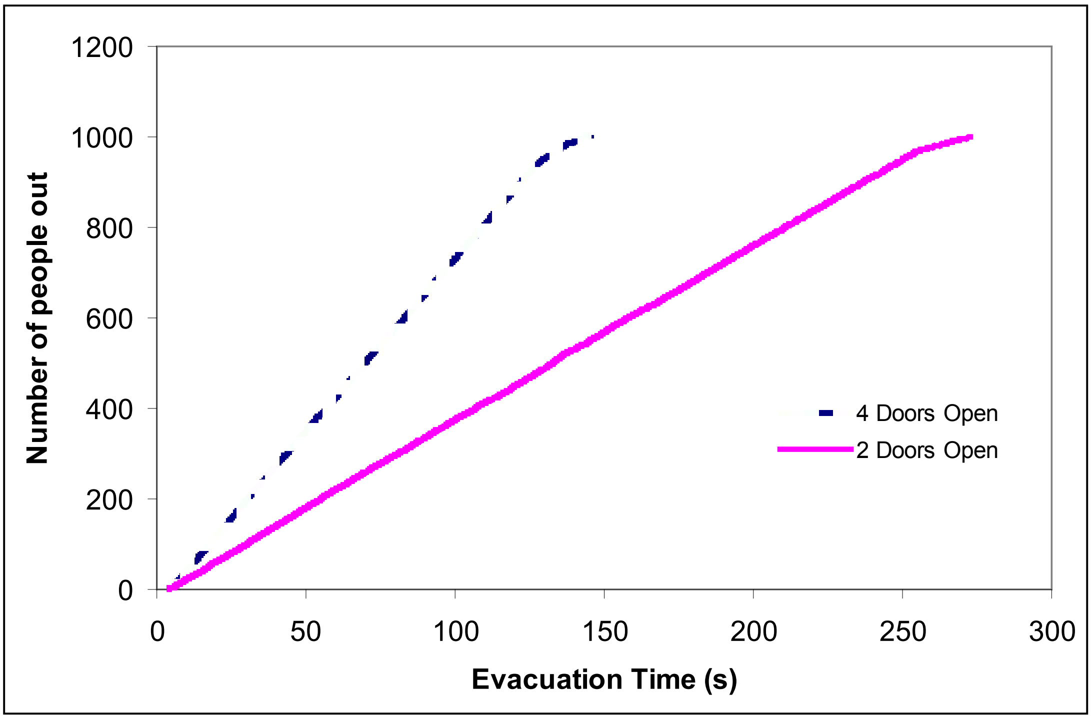


Figure 5-45 Comparison of evacuation times when the number of exits available is varied

Table 5-5 Evacuation times

Scenario	Evacuation Time (s)
4 Doors open	145.7
2 Doors open	272.7

The ratio of evacuation times of the scenarios *2 Doors Open* to *4 Doors Open* is 1.9 which illustrates the expected approximate doubling of the evacuation times when only half the number of doors is available.

5.5.2.3 Exit Route Allocation

This test was based on a cabin corridor as shown in Figure 5-46. The population comprised of 23 agents based on the specifications in **Table 5-3** and all agents were assigned instant response times. Agents in cabins 1,2,3,4,7,8,9 and 10 were assigned the main exit whereas the remaining agents were assigned the secondary one. The expected result is that the agents would navigate to the appropriate exits. When the simulation was run, the agents navigated to their respective target exits and the final paths are indicated by the arrows as shown in **Figure 5-46**. This illustrates a correct implementation of the Distance Route Map described in Section 5.3.1.2.1.1.

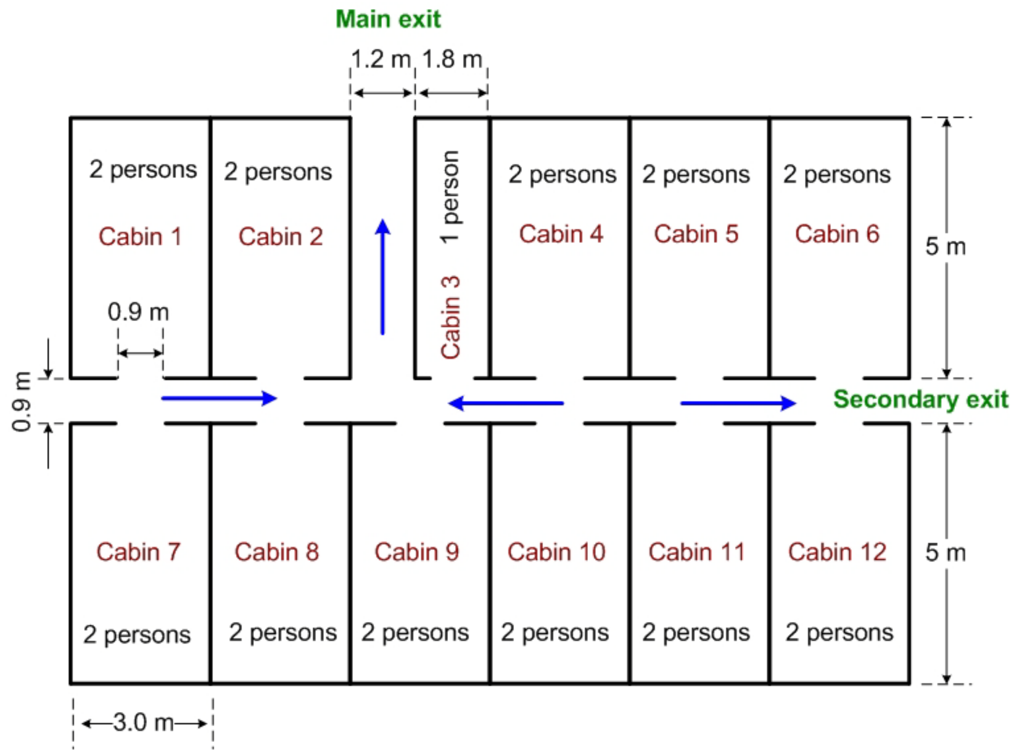


Figure 5-46 Cabin area

5.6 Validation of the Continuous Model with Fundamental diagrams

This section describes the evaluation of the Continuous Model through the investigation of some fundamental relationships related to crowd flow namely; variation of occupant travel speeds with occupant density, variation of flow rate through corridors with occupant density and variation of flow rate with exit width. These scenarios are based on the same experimental setup as the one used for the evaluation of the buildingEXODUS model (Chapter 4 Section 4.2) and therefore, will not be covered in detail in this section.

5.6.1 Relationship between people travel speed and population density

The results of the Continuous Model are compared against empirical data [Thompson 1994, Weidmann 2010] and are as shown in Figure 5-47. The Continuous Model curve shows good agreement with empirical data curves. Moreover, since the curves in Figure 5-47 show different initial travel speeds for zero population density, the curves were normalised using Equation 5-5. The normalised results are as shown in Figure 5-48.

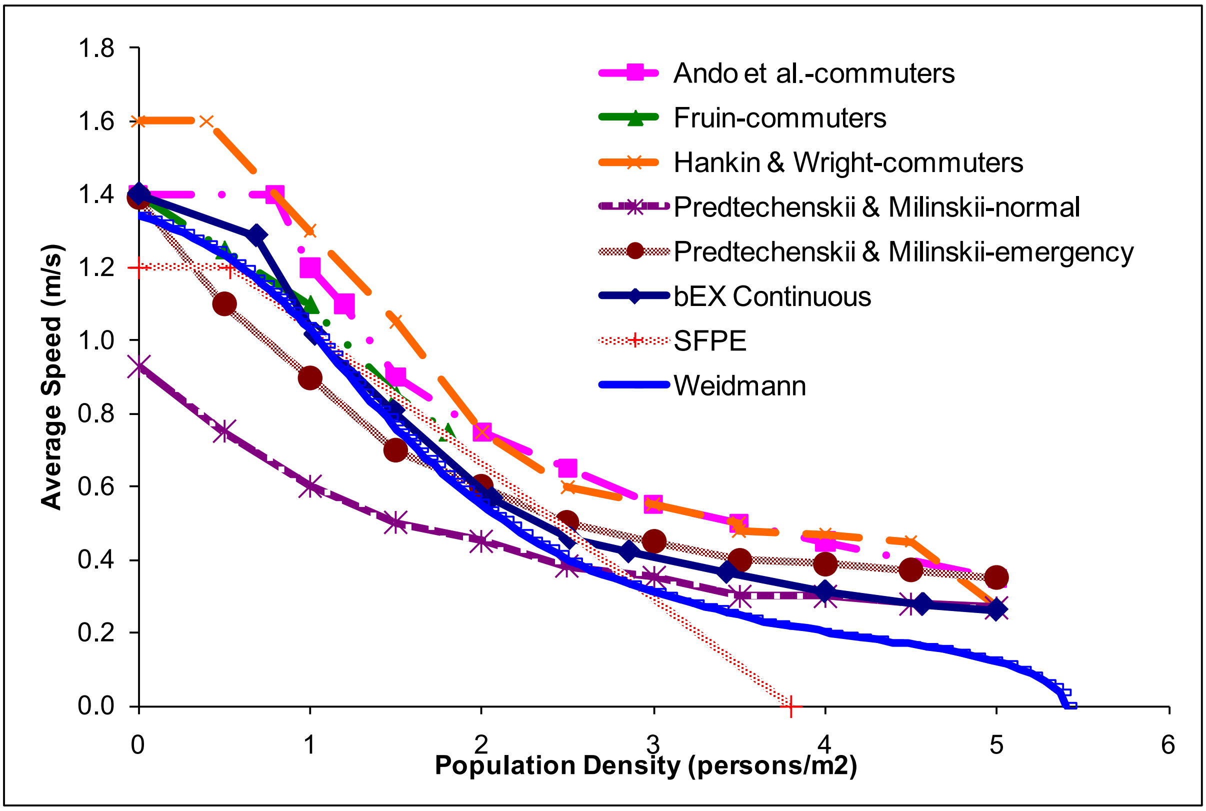


Figure 5-47 Variation of occupant travel speed with density

$$\text{Normalised value of travel speed} = \frac{\text{travel speed}}{\text{travel speed at zero density}} \qquad \textbf{Equation 5-5}$$

From Figure 5-48, it can be seen that the Continuous Model curve lies within the boundaries of the empirical data. In the light of the results, it can be stated that the agents in the model can appropriately adapt their travel speeds based on the densities of the surrounding population.

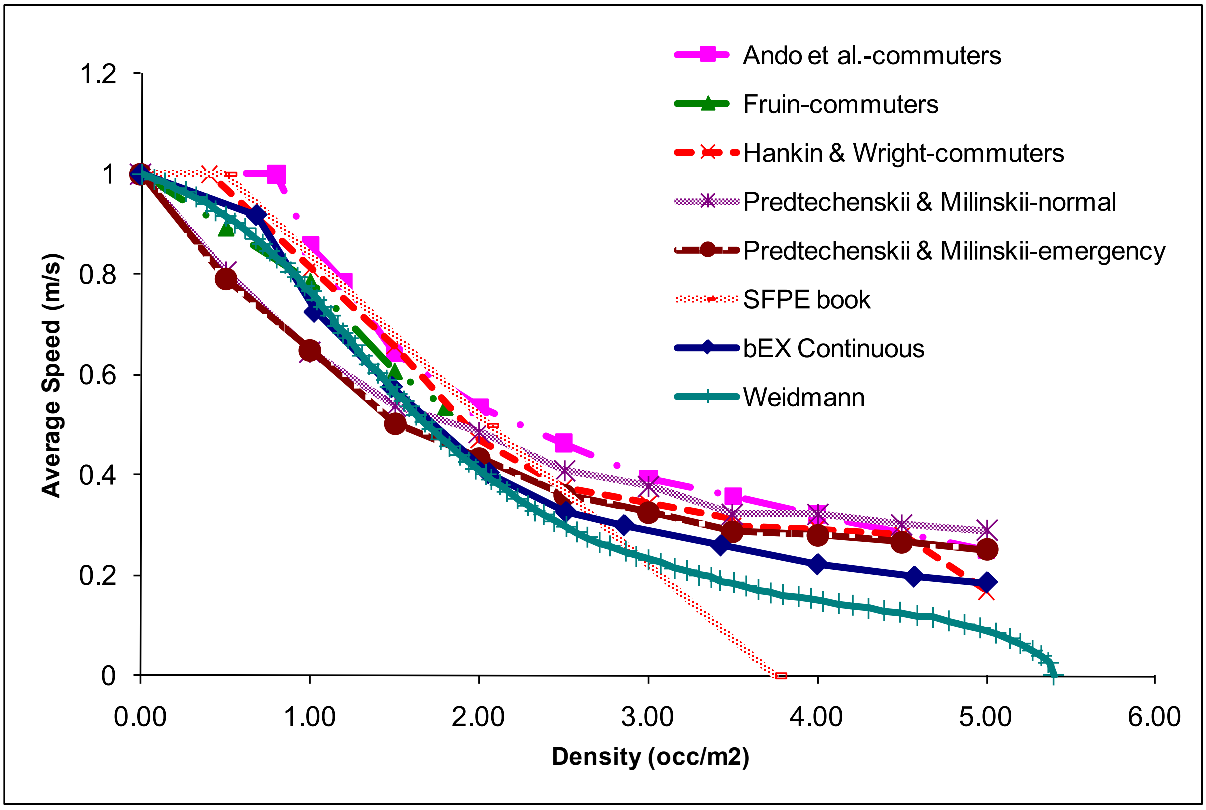


Figure 5-48 Relationship between travel speed and density (Normalised)

5.6.2 Relationship between Unit flow rate and population density

The results of the investigation of the variation of unit flow rate with population density are as shown in Figure 5-49. The population density at which the unit flow rate peaks varies significantly across the different curves. These differences can be due to the cultural differences and the urgency of the occupants during the experiments [Schadschneider 2009] along with other factors such as enclosure configuration. The Continuous Model predicts a peak flow rate of 1.3 occupants/m/s at a population density of 1.8 persons/m^2. The Continuous Model curve is in close agreement with the Hankin and Wright, Fruin and Ando et al. curves for up to population densities of 1.0 persons/m^2. Beyond this population density, the Continuous Model curve shows a similar trend to the Weidmann curve. In the light of the results, the Continuous Model predictions depict the correct trend whereby the flow rate increases to a peak value and then decreases.

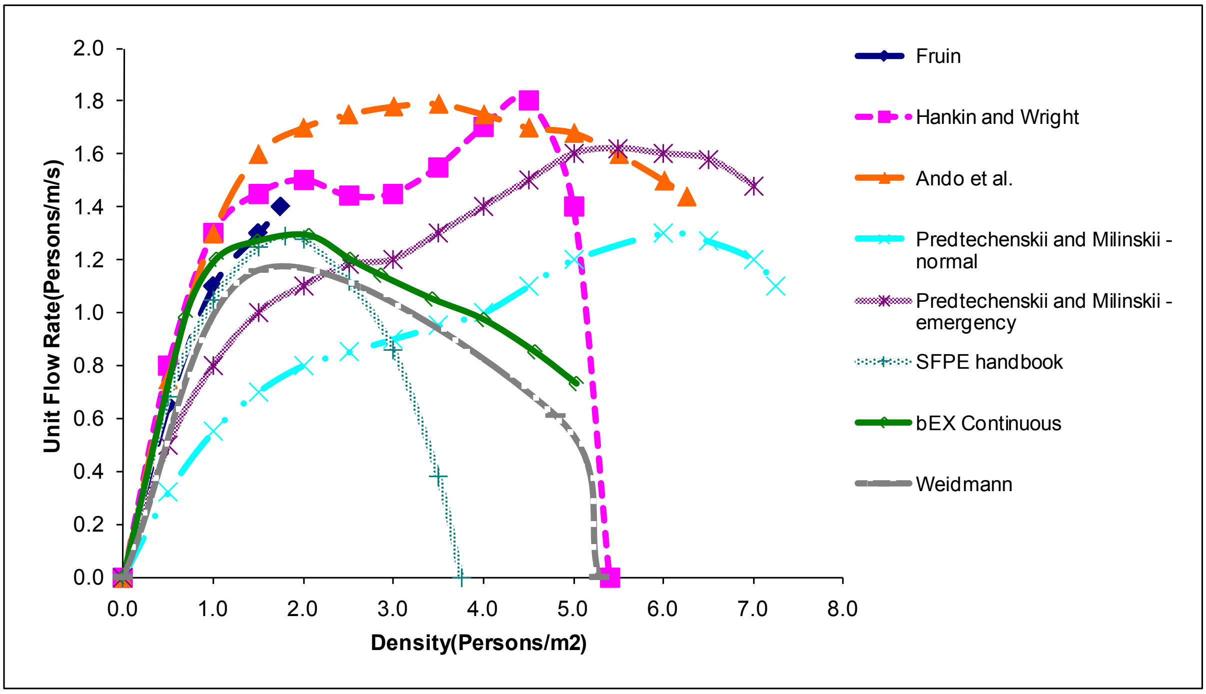


Figure 5-49 Relationship between unit flow rate and population density

5.6.3 Relationship between crowd flow rate and exit width

The results of the investigation of variation of crowd flow rate with exit width are shown in Figure **5-50**. The increase in exit width increases the available space which can be used by the occupants, which leads to an increase in flow rates.

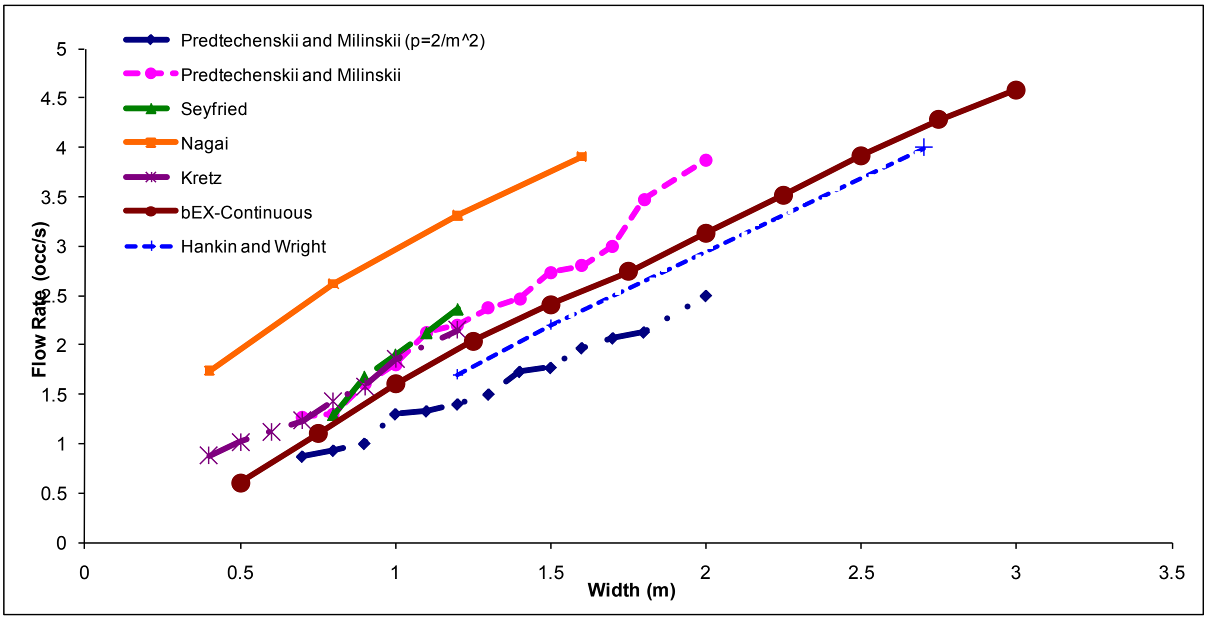


Figure 5-50 Relationship between crowd flow rate and exit width

Whilst all the empirical curves illustrate similar trends, however, the range of flow rate values corresponding to the exit widths vary significantly. This can be attributed to similar reasons mentioned in Section 5.6.2. The Continuous Model curve lies within the empirical envelope and therefore, it can be stated that the agent avoidance strategies have been appropriately implemented.

5.7 Summary

This chapter has described the approaches taken towards the design and implementation of a model (bEX-Continuous) utilising continuous space for conducting egress analyses. The aim was to implement a tool capable of simulating human behaviour and movement while demonstrating a high level of realism. bEX-Continuous has been developed as a plug-in to facilitate its integration in the component oriented architecture of the hybrid model, bEX-H (described in Chapter 7). The use of the continuous spatial representation was instrumental in allowing additional attributes and behaviours to be modelled which are not present in the other representation types for instance acceleration, varying body widths, turning velocity and agent avoidance strategies. In addition, bEX-Continuous is based on the multi-agent paradigm which accounts for its suitability for the exploration and modelling of complex and transitional behaviours for example cross-flows and merging of flows.

The task of simulating the navigation of agents in continuous spaces is particularly challenging especially in large geometries for example, buildings with multiple internal rooms and floors. Moreover, the presence of static obstacles within the enclosure makes the path planning process of the agents even more complex. In order to reduce the complexity of the continuous space, a novel and advanced navigational system called the Navigational Graph was developed. This graph is a network of waypoints and path segments which is automatically generated in the pre-processing phase and is used by the low level navigation algorithms of the agents. The locations of the waypoints are as such that they drive the occupants away from the location of the walls. This feature significantly reduces the computationally expensive wall detection checks by the agents. Furthermore, the nature of the graph makes it suitable for use in a mixed model, as described in the implementation of bEX-H in Chapter 7.

In bEX-Continuous, each agent is modelled as an autonomous character which has adaptive behaviour, thereby enabling the character to improvise actions and navigate in the space in a life-like manner. The autonomy of the agents is defined using a three layered structure comprising of subsystems namely Perception, Behaviour Selection and Locomotion. These subsystems allow the agents to receive sensory information, control decision making processes and execute movement routines accordingly. This tiered architecture minimised the level of coupling in the design, such that each tier could be independently updated without affecting the other tiers.

The behaviour of the agents is modelled as simple components which can serve as building blocks for simulating more complex behavioural routines. This approach allowed behaviours to be implemented and tested individually, followed by an incremental integration. bEX-Continuous is capable of simulating competitive, queuing, bi-directional crowd flow behaviours as well as demonstrating the emergence of certain crowd phenomena such as lane formation. In addition, this component oriented behavioural model provides a flexible platform for developing and integrating new behaviours. Moreover, due to the granularity of the representation of space, continuous models tend to be prone to deadlocks. However, the advanced agent avoidance strategies employed in the behavioural model of bEX-Continuous inhibits the formation of deadlocks. This makes the model suitable for simulating high density of agents even when they are navigating through narrow constrictions in the enclosures.

In addition, advanced strategies comprising of region based hashing and Grid Method were developed to optimise the spatial search of agents. The Grid Method makes use of an underlying grid of cells (of any size) constructed at pre-simulation time. The querying and maintenance of the cells (allocation and de-allocation of agents to cells) is performed during simulation time. As a result, the discovery of neighbouring agents is considerably faster as the search space is reduced to only the current cell (to which the agent is registered) and the immediate neighbouring cells, thereby eliminating the need to search the entire geometry.

In addition, this chapter described the methodology adopted towards the validation of the Continuous Model. This methodology entailed the identification and selection of appropriate test cases, followed by the application of the Continuous Model to those test cases. The test cases and guidelines presented within the IMO document [IMO 2002] were identified and used as a foundation for the purposes of Component Testing and Qualitative Validation of both models. Although these test cases had been devised for evacuation models used within the evacuation environment, however the fact that these test cases were based on flat terrains, suggested their suitability for the building environment as well. Furthermore, the Continuous Model was validated against empirical data from various sources. The results of the validation illustrated that the various behavioural and movement algorithms within the model were operating appropriately.

Chapter 6 – Development of the Coarse Network Model

This chapter describes the development of a coarse network approach for the representation of physical space within the bEX-Hybrid (bEX-H) model. It discusses the methodologies used for representing the space and occupant movement. Moreover, this chapter evaluates the choice of perspective used for representing the agents within the model.

6.1 Spatial Representation

When modelling enclosures using a coarse network approach, the available physical space can be segmented into partitions whereby each partition can represent a section of the geometry such as a room or corridor. Each partition is a node that is connected via arcs or links to represent doorways or other forms of connectivity in the structure. Each node has a maximum agent capacity while the arcs have a maximum flow capacity, that is, the maximum number of agents that can traverse the arc per time period. The agents within the coarse network model transit from one segment to another while the physical movement within the segment itself is not represented.

6.2 Flow Rate through Coarse Node

The flow rate through a coarse node in bEX-H is a function of travel speed, travel distance and population density in the region. The travel distance is defined as the distance the agent has to travel, within the node, from their entry point to their exit point. This is implemented using a “Flow to Density Equation” also known as the F-D Model [Buckmann 1994]. The equation for this is:

$$t = t_0 + A.\left(\frac{V}{C}\right)^n$$ **Equation 6-1**

Where:

t : Travel time (seconds) at flow level V

t_0 : Free-flow travel time (seconds)

V : Occupants demand (pass/m/hr) or density (pass/m^2)

C : Max Capacity (pass/m/hr or pass/m^2)

A and n : Constants estimated in the model fitting process

The travel time, *t*, is then converted into speed using the following transformation:

$$s = \frac{l}{t}$$ **Equation 6-2**

Where:

s : Estimated speed (m/s)

l : Travel distance (m)

The rate of flow, in a coarse node, is also subject to two other limitations, the maximum capacity of the coarse node and the maximum capacity of the coarse node's arcs connecting this node to other nodes. When an agent enters a coarse node, their path, travel distance and speed is fixed based on the Flow to Density Equation. The only dynamic events which can affect their dwell time in the node are the capacity of the arcs out and congestion in adjacent nodes.

6.3 Types of Coarse Nodes

There are seven types of coarse node implemented within bEX-Hybrid, whereby each type uses the same **Equation 6-1** but with different parameters, which are listed in the Appendix. These coarse nodes are namely; Compartment, Intersection, Interchange, Gates, Stairs, Escalators and Travelators.

6.3.1 Compartment

Compartment blocks are used to represent a general region in which agents enter or exit. After creation, such blocks can be made one-way or two-way by changing the direction of the arcs within them.

6.3.2 Intersection

Intersection blocks are used to represent complex regions in which occupant flows from different directions merge such that the resultant cross flows could lead to congestion. Such blocks are used in identifying localised congestion regions.

6.3.3 Interchange

These blocks are a larger version of intersection blocks and are used to represent large areas where occupant flows from several directions can merge and cause congestion.

6.3.4 Gates

Gates blocks can be used to represent narrow metered passages such as turnstiles. Since turnstiles are designed to enforce the flow of people in one-way, the Gates blocks allow flow only in one direction.

6.3.5 Stairs

These blocks are designed to allow the flow of occupants in both directions, but the direction of flow in these blocks can be altered by changing the direction of the arcs within them.

6.3.6 Escalators

Escalator blocks are designed to operate only in one-way and this requires that all the arcs leading to and from escalator blocks are set to one way towards the entry of the escalator and one way away from the exit of the escalator.

6.3.7 Travelators

Similar to Escalators, Travelators are designed to function in one-way. Such blocks are usually arranged in pairs to cater for movement in each direction.

6.4 Perspective of Occupant in Coarse Network

One key factor in the development of the coarse network was the choice of the perspective for the representation of the agent population. If the agent population is represented as a homogeneous ensemble, it has to be taken into account that when the agents in the coarse space encounter the space represented by the fine network (Chapter 4) or continuous regions (Chapter 5), they would need to be modelled from an individual perspective and their individual physical locations would need to be represented. In this respect, the problem we are faced with is the identification of agents which have reached the end of the coarse nodes and are about to transit in the microscopic fine and continuous models. In order to address this problem, the occupants in the coarse node model are modelled from an individual perspective as well. In other words, although the graphical output of coarse nodes in bEX-H only illustrates the total number of agents in the node at any point during the simulation, the underlying algorithmic procedures keep track of the individual identities of each agent. Furthermore, the locations of the agents along the arcs of the coarse node's navigational mesh are also recorded. Using this methodology, the simulation is able to keep track of the agents which have reached their exit points within the coarse node, and re-position those selected agents into the adjacent segment of physical space represented by the fine networks or continuous regions.

6.5 Advanced Coarse Nodes

The nature of the coarse network approach restricts the modelling of a region with simple geometrical shapes such as rectangles. However, these simple shapes might not be sufficient to accurately model all the complex segments within an enclosure. Unlike other coarse node models, the coarse node implementation in bEX-H allows the creation of non-convex regions as shown in **Figure 6-1**. The key data structure behind a coarse node is a mesh which is based on the same principle as the Navigational Graph used in the Continuous Model (Chapter 5). The waypoints and arcs within the mesh provide the information required to model the flow of occupants in non-convex coarse nodes.

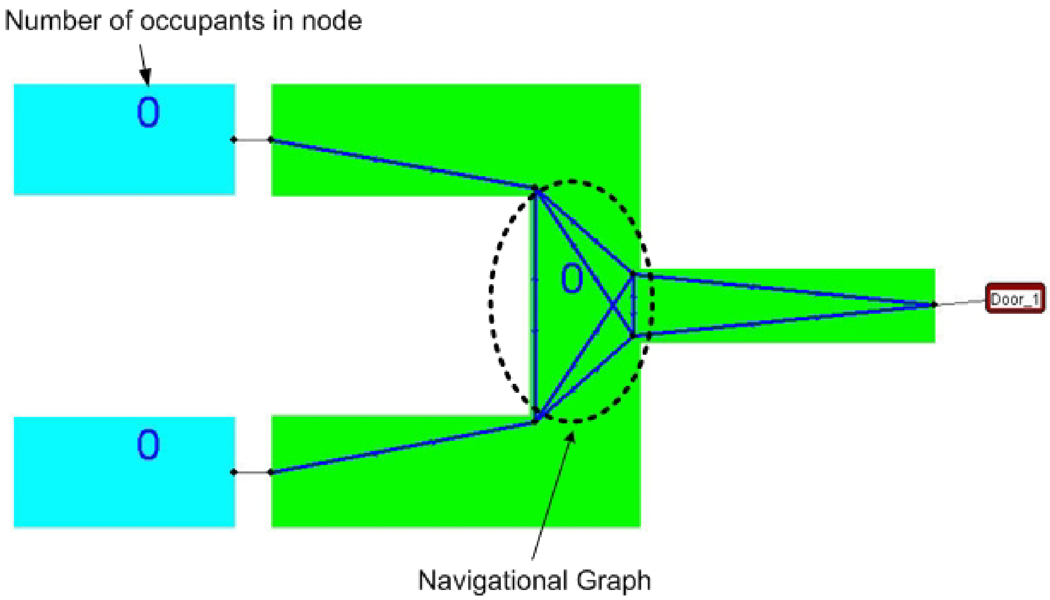


Figure 6-1 Representing non-convex coarse nodes in buildingEXODUS

6.5.1 Navigational Mechanism

Agents within coarse nodes move along the internal arcs and waypoints of the navigation mesh towards exit waypoints of lower potential. In addition, the coarse node implementation features a navigational mechanism which enables agents entering a coarse node to choose alternative paths depending on the evolving conditions of population density within the node. This feature is implemented through a load balancing algorithm. This algorithm dynamically adjusts the potential value of the waypoints to account for the number of agents heading towards them. When an agent enters a coarse node, they examine in turn all the adjacent waypoints, connected by valid arcs, which are nearer to their target location. Then, each of those adjacent waypoints, which direct the flow of agents nearer to their target locations are given a weight based on the following condition:

The minimum of the equation (Pe – Px) + D * A and Px

Where Pe > Px

Pe : Potential value at entry point of coarse node

Px : Potential value at a possible exit point of coarse node

A : Number of agents, in the coarse node, travelling to Px who are experiencing delay i.e. those having a wait count greater than zero.

D : Average depth of an agent

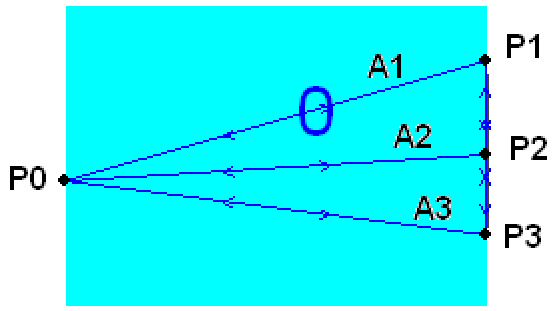


Figure 6-2 Coarse Node with three internal paths A1 to A3

For instance, in the example shown in **Figure 6-2**, agents travelling from the waypoint P0 to either of the waypoints P1 to P3 would consider the congestion along the paths A1 to A3. Without the application of the Load Balancing algorithm, the agents would use only path A2 in the coarse node, but when the algorithm is applied, the paths A1 to A3 are utilised more efficiently, thereby reducing the amount of time which the agents are likely to spend in congestion.

6.6 Summary

This chapter has described the implementation of the Coarse Network Model within bEX-H and evaluated the perspective used for representing the agents. This model forms an integral component of the Hybrid Spatial Discretisation approach and its application is described in Chapter 7.

Chapter 7 - Development of a Hybrid Space Discretisation Model

Egress models typically use one of three methods to represent the physical space in which the agents move, namely: coarse network, fine network or continuous. Each approach has its benefits and limitations:

- The continuous approach allows for an accurate representation of the building space and the movement and interaction of individual agents but suffers from relative poor computational performance.

- The coarse nodal approach allows for very rapid computation but suffers from an inability to accurately represent the interaction of individual agents with each other and with the structure.

- The fine nodal approach represents a compromise between the two extremes providing an ability to represent the interaction of agents while providing good computational performance.

This chapter presents a novel approach to represent space, called the 'Hybrid Spatial Discretisation' (HSD), in which all three spatial representations can be utilised to represent the physical space of the geometry within a single integrated software tool.

The aim of the HSD approach is to encompass the benefits of the three spatial representation methods and maximise computational efficiency while providing an optimal environment to represent the movement and interaction of agents. An ideal application of the HSD approach would utilise the fine nodal approach to map the majority of the geometry, providing good computational speed and an ability to model agent interaction. In parts of the geometry where greater precision is required to model detailed interaction between agents, the continuous approach is used and in regions where knowledge of detailed agent interaction is not essential the coarse nodal approach can be used, providing improvements in speed and computational efficiency. In such an application, the HSD would benefit from the speed of the fine and coarse approaches and the precision of the continuous approach.

The mechanics and the capabilities of the fundamental components of the HSD approach namely, the fine node model, continuous model and coarse network model have been described in Chapter 4, Chapter 5 and Chapter 6 respectively. This chapter describes the methodologies used for integrating all the spatial representations within the buildingEXODUS-Hybrid prototype model (bEX-H). Furthermore, the key regions representing the transition from one spatial representation to another are described and the challenges which result from the interfacing of the different spatial representation types are discussed.

7.1 Software Architecture

The bEX-H makes use of the core architecture of the buildingEXODUS software. The buildingEXODUS software has been modified to allow plug-in modules to be included into the core software using a component oriented engineering approach. This architecture provides a platform whereby new functionalities can be independently developed and incorporated into the model as required. The continuous region and coarse network are examples of two components which have been developed as plug-in modules for the bEX-H model. **Figure 7-1** illustrates an overview of the system architecture of the Hybrid Spatial Discretisation Model. The Interface Module is a key component within the architecture and acts as intermediate layer between the buildingEXODUS Core Software and the component modules. The Interface Module is divided into two sub modules namely a Geometrical Interface and a Population Interface which control the enclosure characteristics and the agent model respectively. Moreover, it regulates the transition of the agents across the interface regions namely; Fine Node ↔ Continuous, Coarse Node ↔ Continuous and Coarse Node ↔ Fine Node. Moreover, the plug-ins have been designed in such a way that multiple instances of the same plug-in can be assigned to different continuous regions or coarse nodes within the geometry. This feature is indispensable as the application of the HSD approach is likely to entail the use of multiple continuous regions and coarse nodes within an enclosure.

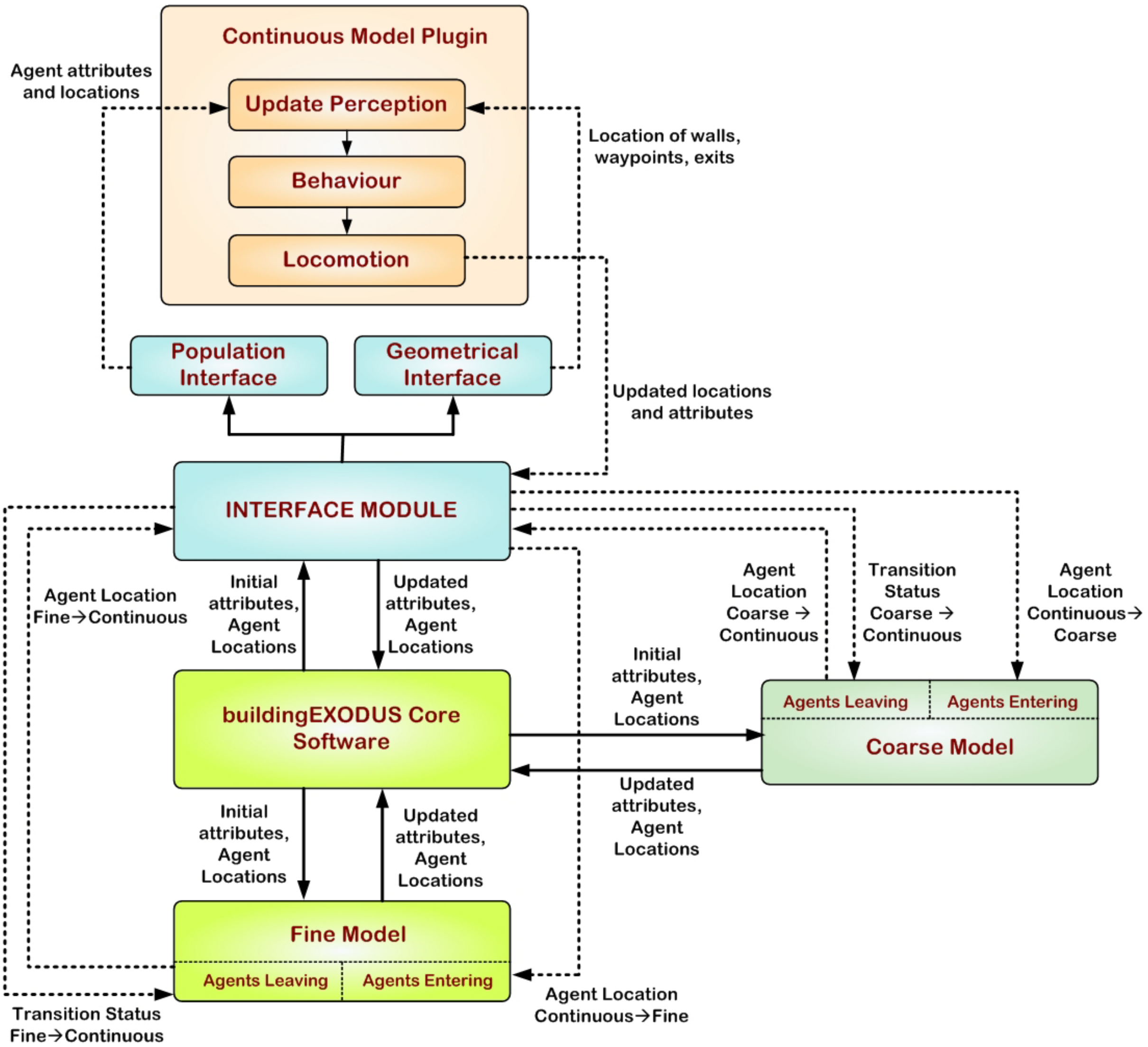


Figure 7-1 Hybrid Spatial Discretisation System Architecture

Key:

Agent Location A→B : Position of the agents before they transit from A to B

Transition Status A→B : Boolean flag which confirms whether the agents can physically transit from A to B where A and B represent Fine, Coarse or Continuous.

: Agents leaving and Agents entering signifies agents leaving and entering the Fine Node network

7.2 Transition Regions

The HSD approach involves the mixing of macroscopic (coarse node) and microscopic (continuous and fine) modelling methodologies which in itself presents several challenges. In the continuous and fine node models, the agents are modelled from an individual perspective whereby their movements, exact locations and behaviours can be tracked. However, in the coarse network approach, the agents are modelled from a global perspective whereby the population is treated as a homogenous ensemble, such that the locations and physical space occupancy of the agents are not represented. bEX-H incorporates behavioural and movement mechanisms to facilitate the transition of agents across the transition regions. The implementation of bEX-H features the representation of all six possible interface transition regions namely: Coarse Node↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous. **Figure 7-2** illustrates a multiple compartment geometry modelled using the HSD approach whereby different areas of the enclosure are modelled using the different spatial representation techniques.

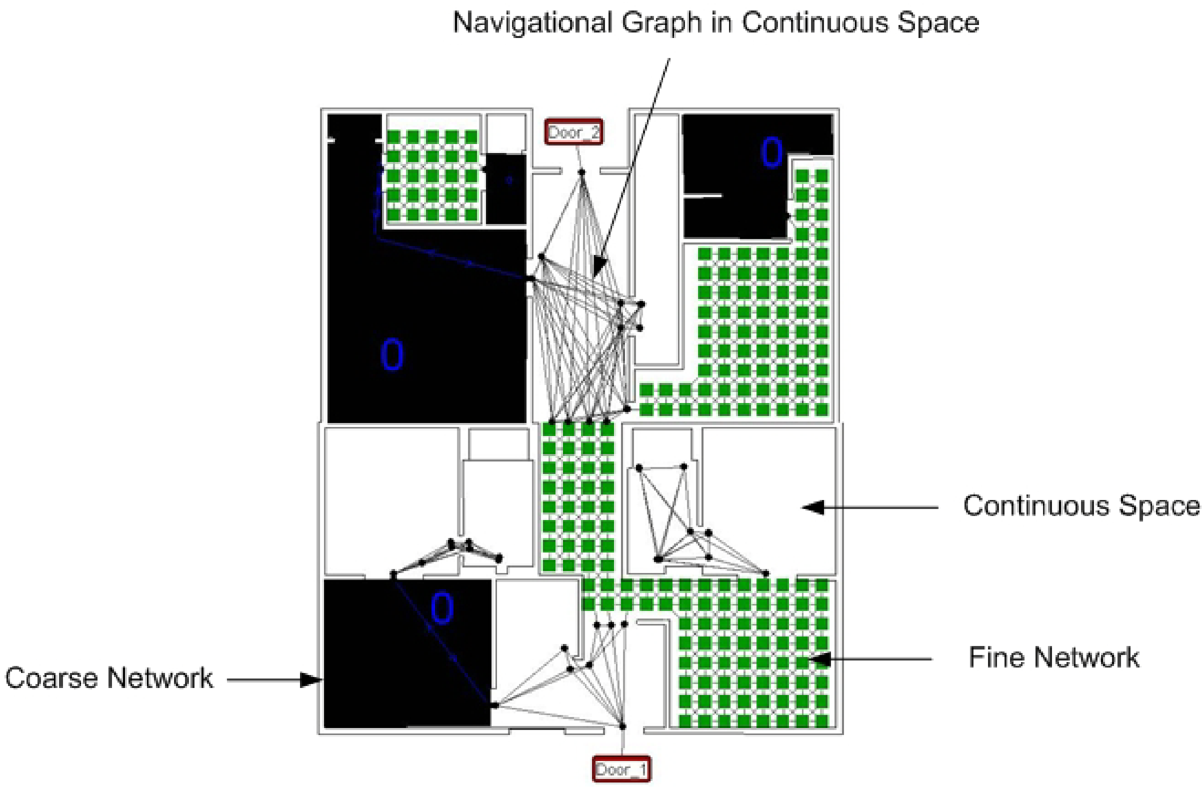


Figure 7-2 Entire geometry modelled using HSD approach in bEX-H

7.2.1 Coarse Node/Continuous Region Transition

Coarse → Continuous

This transition represents the two extremes on the scale of granularity. Connectivity between the coarse nodes and the continuous regions is achieved via arcs (generated manually), such that the endpoints of those arcs serve as waypoints. An example of the combination of arcs and waypoints within the transition region is as shown in **Figure 7-3**. The movement of agents across the transition region is monitored and controlled by the Interface Module (**Figure 7-1**). This module also maintains the agents in valid state, that is, it prevents agents exiting the coarse node from overlapping neighbouring agents in the adjacent continuous region.

The procedure for transiting an agent to a continuous region is as shown:

1. Retrieve list of agents who want to move to continuous region

```
FOR each agent in list
    Check if agent can move to continuous region (CanMoveTo)
    IF agent movement possible
        Remove agent from current spatial representation
        Initialise agent in continuous region
        Search for a cell in underlying optimisation grid of continuous region
        IF cell found
            Assign agent to cell
        ELSE
            Agent travelling in between region, assign agent to polygon
    ELSE
        Agent exhibits waiting behaviour in current spatial representation
END FOR
```

Algorithm 7-1 Algorithmic procedure for an agent transiting to a continuous region

The "CanMoveTo" procedure in **Algorithm 7-1** is detailed as shown in **Algorithm 7-2**. In this procedure, *pFuture* indicates the predicted future location of an agent on entering the continuous space and is set to the location of the nearest waypoint at the entrance of that space.

```
Get agent P who wants to move in continuous region
Get predicted future location, pFutureLoc of agent P in the continuous space
Get radius of P, pRadius
Retrieve list of other neighbouring agents present in region
FOR each neighbouring agent O in list
    Get radius of O, oRadius
    Compute distance, distOPfuture, between pFutureLoc and O,
    Compute sum of oRadius and pRadius as minDist
    IF distOPfuture ≤ minDist
        Agent P cannot move
    ELSE
        Agent P can move
```

13. **END FOR**

Algorithm 7-2 Algorithmic procedure for checking if an agent has adequate space for moving into continuous region

In the procedure shown in **Algorithm 7-2**, the distance criteria used for detecting the availability of physical space is *minDist*, i.e. the sum of the radii of the body circle of agents O and P (*oRadius* and *pRadius*). During experimentation, it was found that using values greater than *minDist* for the distance criteria check would lead to the agents being too far behind each other as they emerge into the continuous region, thus leading to very low and unrealistic flow rates around these regions. Using the sum of *oRadius* and *pRadius* as the distance check criteria would enable the agents to perform more accurate space availability detection while allowing for closer packing and more realistic representation of agents around the entrance of the continuous space.

The number of arc connections and resulting waypoints required in between the coarse node and continuous region is a function of the width of the transition region. When an agent traverses from the coarse node to the continuous region, the waypoints in the transition region are used to guide the entrance of the agents in the continuous space. Therefore, in order to ensure that the entire width of the transition region is being used by the agents, the waypoints need to be generated accordingly. After experimentation and consideration of the distribution of the agent widths in the agent model, it was found that 2 waypoints per metre of the edge leading into the entrance of the continuous region (Transit Edge B) are sufficient. Moreover, upon entering the continuous space, the agent avoidance mechanism is invoked by the agents. This causes them to spread out and make a more realistic use of the available continuous space.

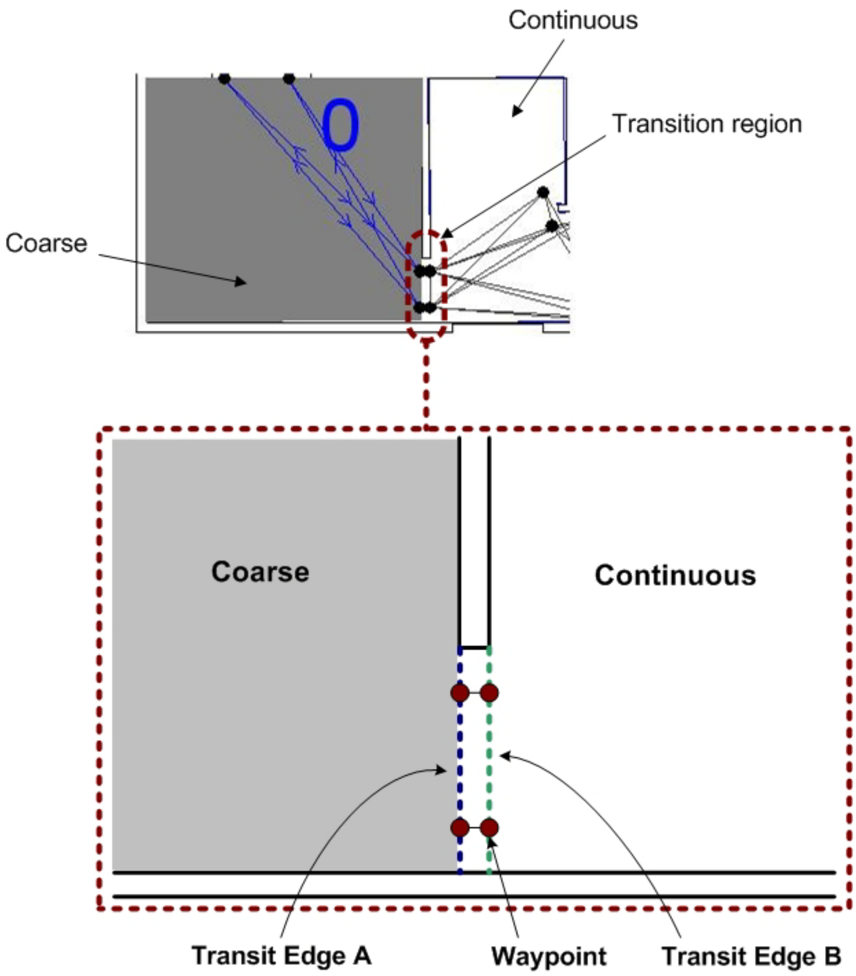


Figure 7-3 Transition Region in between Coarse node and Continuous region

Continuous → Coarse

When an agent attempts to traverse from the Continuous region to the Coarse Node, it uses the Interface Module to retrieve information about the total number of agents in the Coarse Node. When the capacity of the coarse node reaches its maximum limit, the transition of the agents is inhibited and their wait attributes is triggered in the continuous space. The continuous agents will keep on waiting until the capacity of the coarse node falls below its maximum value. The procedure for the transition of an agent from a continuous space to another spatial representation type is as shown in **Algorithm 7-3**. In this procedure, the distance of the agent from the transition region, *distanceFromTransit* can be calculated in various ways. For instance, this distance can either be computed from the location of the *centre of the agent* or the location of the *agent's apex* as shown in **Figure 7-4**.

Using d1 (distance from apex) as the distance measure criterion would imply that the agent would be removed slightly earlier from the continuous region than when using d2 (distance from centre of agent), thereby underestimating the distance travelled. Moreover, when an agent transits from a continuous region to a coarse node, it is the agent's *centre* location which is repositioned on the waypoint at the entrance of that coarse node, as opposed to the agent's apex location. Therefore, using distance d2 as the distance criteria is appropriate as it provides a more accurate representation of the distance travelled.

```
Retrieve list of agents in continuous region
FOR each agent P in list
    Calculate distance of agent from the transition region, distanceFromTransit
    IF P has reached transition region
        Check capacity or available space in adjacent spatial type
        IF space available
            De-allocate P from its underlying cell in continuous region
            Remove P from the continuous region
            Initialise P in the adjacent spatial type
        ELSE
            P executes waiting behaviour
    ELSE
        P proceeds on its current path in continuous region
END FOR
```

Algorithm 7-3 Algorithmic procedure for agents moving from continuous region to other spatial representation types

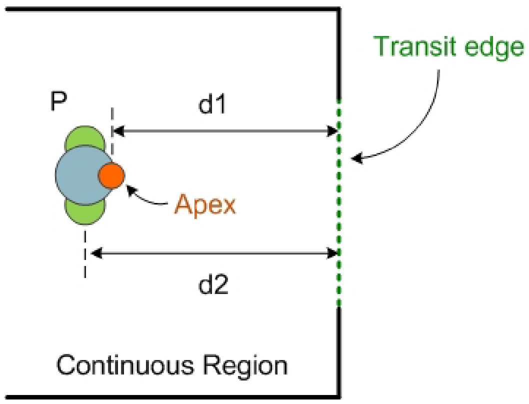


Figure 7-4 Measuring distance of agent from the transit edge

7.2.2 Coarse Node/Fine Node Transition

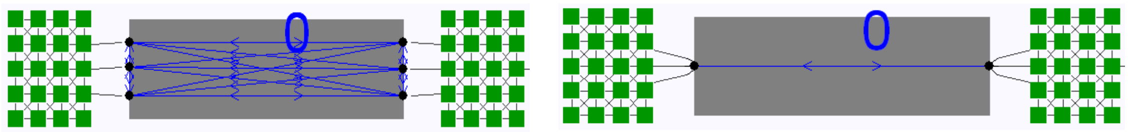
Coarse → Fine

When agents traverse from coarse nodes to fine nodes, they are re-positioned on the fine nodes which are connected by arcs to the coarse node region as shown in **Figure 7-5**. Prior to their transitions, the agents query the transition region to find the density in the space represented by the fine network. If the density is high such that there are no available fine nodes to transit to, the agent transitions are inhibited and consequently, the agents exhibit queuing within the coarse node.

Fine → Coarse

For the fine node to coarse node transition, the agents check the current capacity of the adjacent coarse node and transit only if the capacity has not been exceeded.

In addition, bEX-H features two methods for connecting fine nodes to coarse nodes as shown in **Figure 7-5**. The number of arcs generated using method A is much higher in comparison to the method B and therefore, agents entering the coarse node in method A will have to search for their target waypoint against a larger number of internal paths, thereby reducing performance. Moreover, due to the fewer number of internal paths in method B, this facilitates the management of arcs within the coarse node region. In these respects, when a group of fine nodes form a common source location for a coarse node, method B is preferred.



A B

Figure 7-5 Two methods of connecting a set of fine nodes (2D grid of nodes) to a coarse node (dark region)

7.2.3 Fine Node/Continuous Region Transition

Fine → Continuous

This transition makes use of the waypoints in the transition region to guide agents from the fine nodes to the continuous regions as shown in **Figure 7-6**. Prior to entering the continuous space, the agents query the Interface Module to obtain information about the locations of neighbouring agents at the entrance of the continuous region. If neighbouring agents are found, then the transition is inhibited.

Furthermore, the orientation of agents in fine network changes momentarily while they are waiting for available free space. For example, in Figure 7-6, an agent is waiting for the density at the entrance of the continuous space to decrease before transiting. However, at the moment of transition, the agent's orientation changes so that it now faces the vector *Pdirection*. If this new orientation is maintained during the transition, the agent would enter the continuous space at an angle which could lead to unusual paths being taken and cause unnecessary blockages. In order to circumvent this problem, all agents readjust their orientation to face their next visible waypoint upon entering the continuous space. This adjustment ensures that the paths of the agents and the resultant flows are consistent after their transitions. Moreover, this readjustment is appropriate as the heading of the agents in the fine nodal network does not affect their movement whereas in the continuous regions, the heading is an important factor which affects the rotation and movement of the agents.

Continuous → Fine

For the continuous region to fine node transition, the agents check for available fine nodes before transiting and invoke the waiting behaviour in the continuous space in case they are unable to do so.

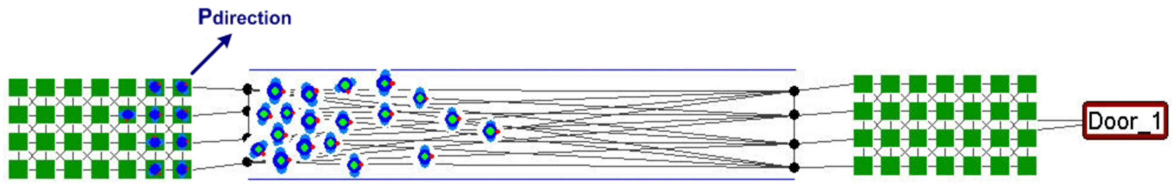


Figure 7-6 Connecting a set of fine nodes (2D grid of nodes) to a continuous region (white region)

In some scenarios, the location of the wall boundaries needs to be taken into account when connecting fine nodes to continuous regions. In the fine nodal network, the body widths of agents remain fixed and are specified by the arcs lengths whereas in the continuous regions, the body widths are varied. In this respect, when an agent transits from the fine node to the continuous region, its body width in the continuous might be bigger than that when it was in the fine nodal network. For instance, in **Figure 7-7**, if an agent has transited around the waypoint Wa, which is very close to the wall boundary, it might be overlapping with the wall as soon as it enters the continuous space, due to the increase in body width. This could cause the wall detection algorithm to fail as the agent could be in an invalid state even prior to the wall detection check. However, this would not occur if the continuous agent has a body width which is equal or less than the body width when on the fine node. This issue could be avoided by generating the arc from the fine node to the continuous region in such a way that the resultant waypoint Wa would be at a distance of D metres or more away from the wall boundary where D is equal to half the body width of the largest possible occupant that can be represented in the continuous agent model. This new location Wb would ensure that the agents would not be in contact with the walls as soon as they are initialised in the continuous space.

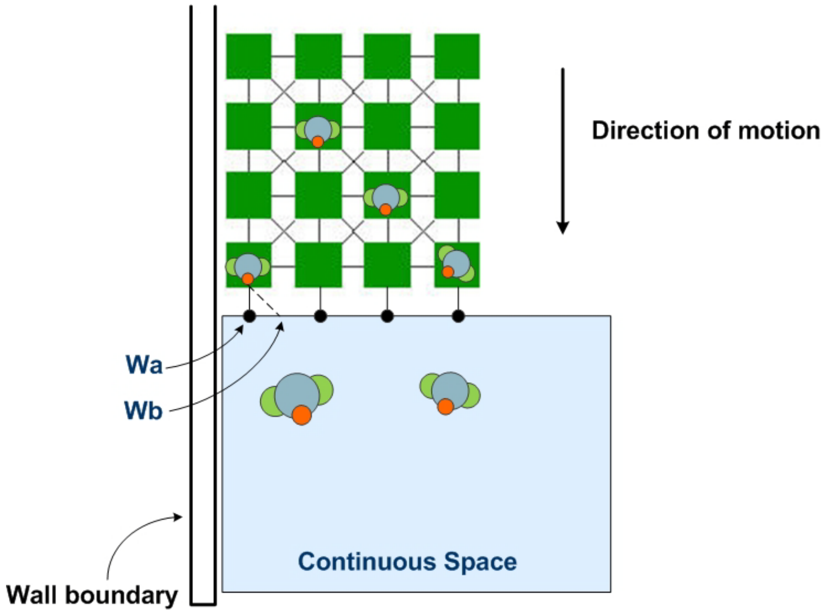


Figure 7-7 Transiting from fine nodes to continuous regions near wall boundaries

7.2.4 Attribute preservation across different spatial representation types

The agents in the HSD model retain their internal attributes even after they have transited across different spatial representation types. For instance, an agent initially created in the continuous component will have a body width attribute. During the agent's movement, if it transits to a region represented by fine nodes, its body width will be set to be equal to the arc lengths connected to the nodes on which it resides. However, when the agent transits back into continuous space, its body width is reset to its original value. This is based on the principle that all the agents generated in the HSD model are assigned with individual body width attributes irrespective of whether they have been generated in a fine, coarse or continuous model. Because of the level of granularity involved across the different spatial representation types, the body width attribute manifests itself only when an agent is moving through a continuous region.

Furthermore, in the coarse node network, the underlying algorithmic procedures keep track of the individual identities of each agent. In this respect, the agent attributes (e.g. age, gender, body width) are maintained when agents are traversing coarse nodes while the experiential attributes (e.g. distance travelled, cumulative wait time) are updated accordingly during the simulation. Also, the locations of the agents along the arcs of the coarse node's navigational mesh are also recorded. Using this methodology, the simulation is able to keep track of the agents which have reached their exit points within the coarse node, and re-position those agents into the adjacent segment of physical space represented by the fine network or continuous region.

7.3 IMO Validation Tests – HSD Model

This section describes the validation of the movement algorithms used in the HSD model using the IMO test cases. The reasons for choosing the IMO test cases have been discussed in Chapter 5.

7.3.1 Maintaining set walking speed in corridor

The geometry which was used for this scenario is as shown in **Figure 7-8** such that an agent with a maximum travel speed of 1 m/s was generated in the fine nodal network, on a node which is furthest from the exit. The agent took 40 s to cover the entire distance. Therefore, the HSD model appropriately demonstrated that the agent is able to maintain the appropriate travel speed across the transition regions.

This hybrid discretisation strategy shown in **Figure 7-8** was preferred as it was more appropriate to have the agent start in the fine nodal region rather than in the coarse node for this test. This is because agents created in the coarse node are assigned random locations within the node every time the simulation is reset, and therefore depending on where they are initially placed, they might not be travelling the full distance represented by the coarse node. This test would be successful as long as the agents were made to start up in the fine nodal or continuous regions. Examples of such configurations are as follows: Fine→Continuous→Coarse, Fine→Coarse→Continuous, Continuous→Fine→ Coarse and Continuous→Coarse→Fine.

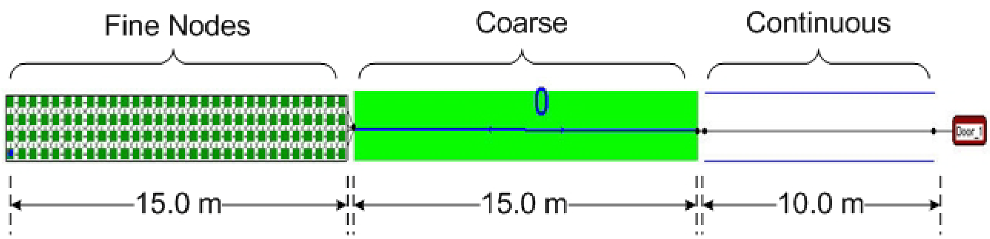


Figure 7-8 Verifying agent walking speed in corridor

7.3.2 Exit Flow Rate

This test involved generating 100 agents in a room of 8m by 5m with a 1 m exit as shown in **Figure 7-9** and recording the flow rates at the exit. In order to obtain the actual flow rates predicted by the model, the exit was set to Free-flow conditions. An average flow rate of 1.6 persons/m/s was obtained.

A further set of experiments was conducted on this geometry in order to investigate the validity of the results. This involved increasing the number of people in the geometry and recording the corresponding evacuation times. The results are as shown in **Figure 7-10**. The Predtechenskii and Milinskii data set (normal movement) was chosen as it was representative of the scenario being simulated. Moreover, these results were compared against the Continuous Model curve for that particular geometry.

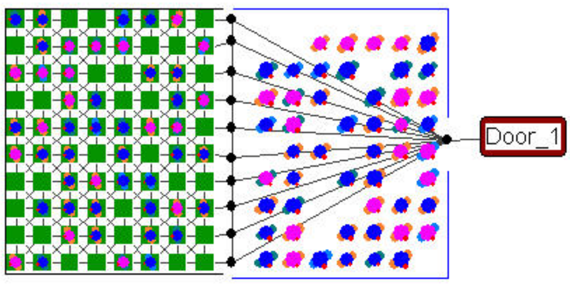


Figure 7-9 Exit flow rate test

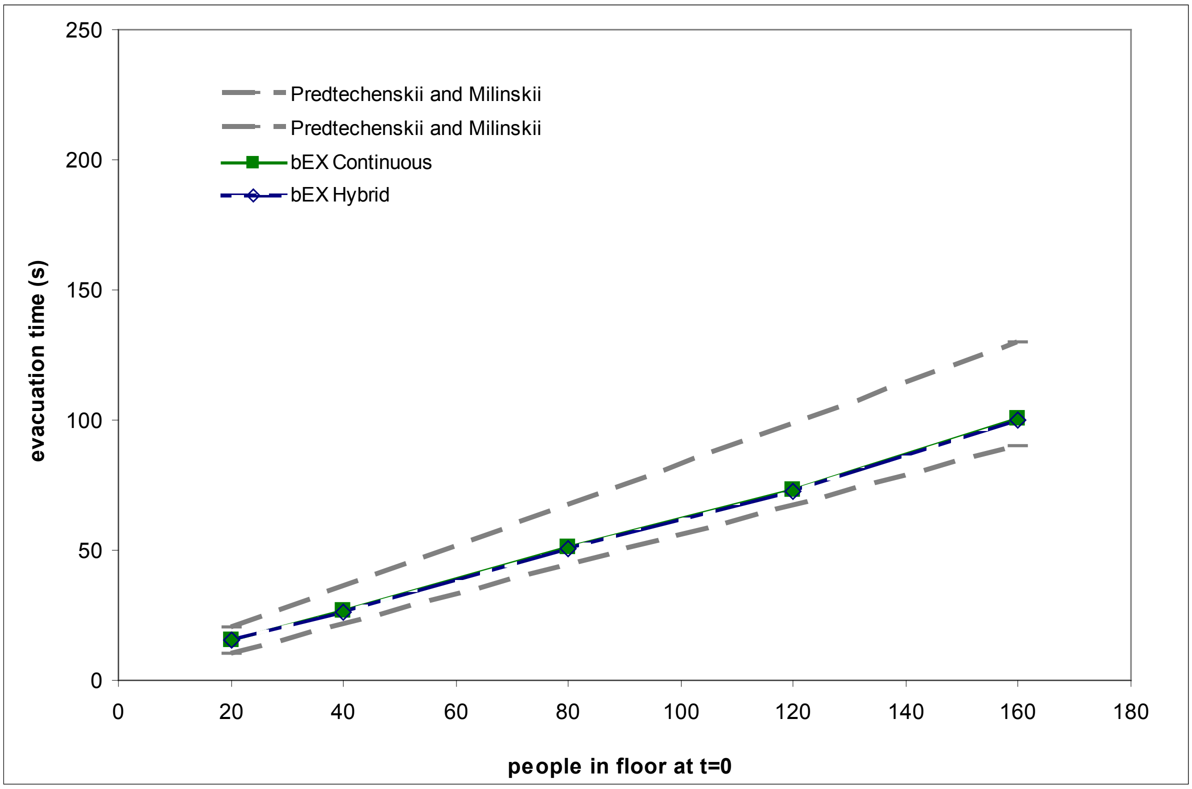


Figure 7-10 Time taken for a specific percentage of people to evacuate

As seen in **Figure 7-10**, bEX-Hybrid curve lies within the Predtechenskii and Milinskii empirical envelope [Rogsch 2005]. Moreover, the bEX-Hybrid and bEX-Continuous curves fall almost on top of each other showing that both models are generating similar evacuation dynamics. This also demonstrates a correct implementation of the transition of agents within the hybrid model.

7.3.3 Counterflow – two rooms connected via a corridor

The geometry and hybridisation strategy used for this test is as shown in **Figure 7-11**. The experimental settings were the same as described in the contra-flow test for the Continuous Model in Chapter 5 Section 5.5.2.1. The results in **Table 7-1** show that as the number of occupants in Room 2 is increased, the time taken for the last agent in Room 1 to enter Room 2 also increases, which is the expected outcome.

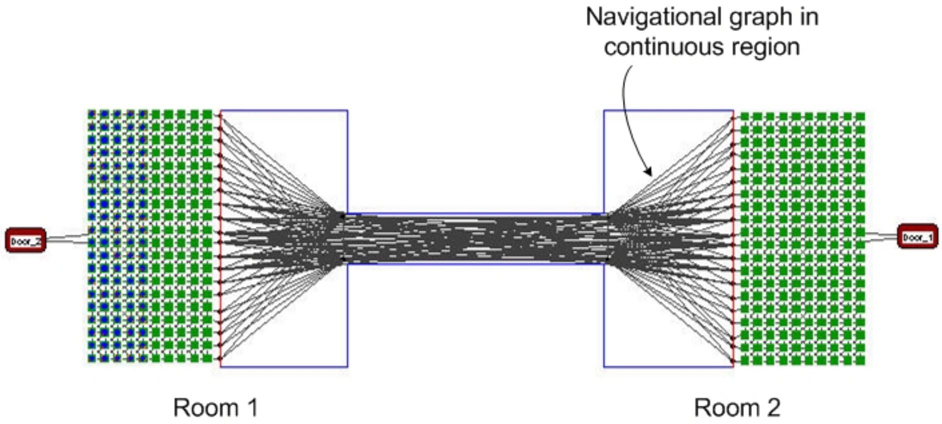


Figure 7-11 Two rooms connected via a corridor

Table 7-1 Time taken for last agent in Room 1 to enter Room 2

Number of people in Room 2	**Time for last agent in Room 1 to enter Room 2 (s)**
0	52.5
10	61.6
50	127.9
100	199.0

7.3.4 Crowd dissipation from a large public room

This test involved measuring the time taken for a crowd to dissipate from a large public room as shown in **Figure 7-12**. The agents were generated in the fine nodal region and set to evacuate via their nearest exits. The experiment was repeated by disabling two of the top doors and the corresponding evacuation time was noted. The expected result is an approximate doubling of the time required to evacuate the enclosure. The results are as shown in **Figure 7-13**.

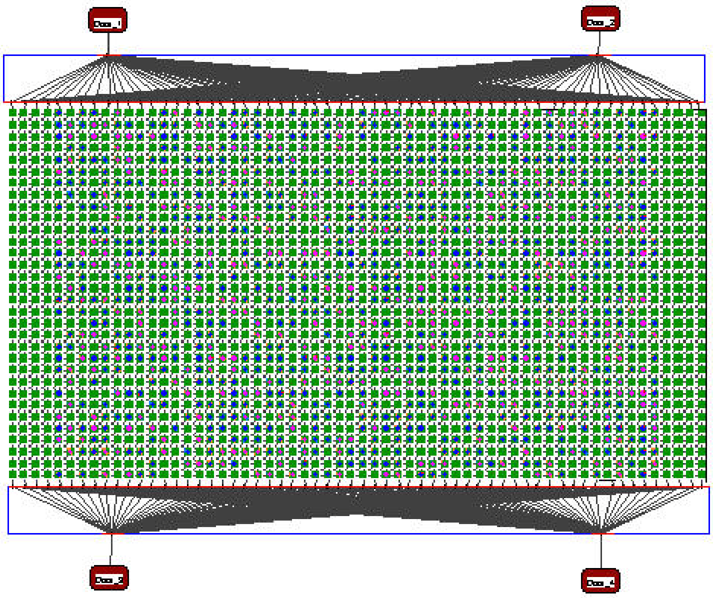


Figure 7-12 Exit flow from a large public room

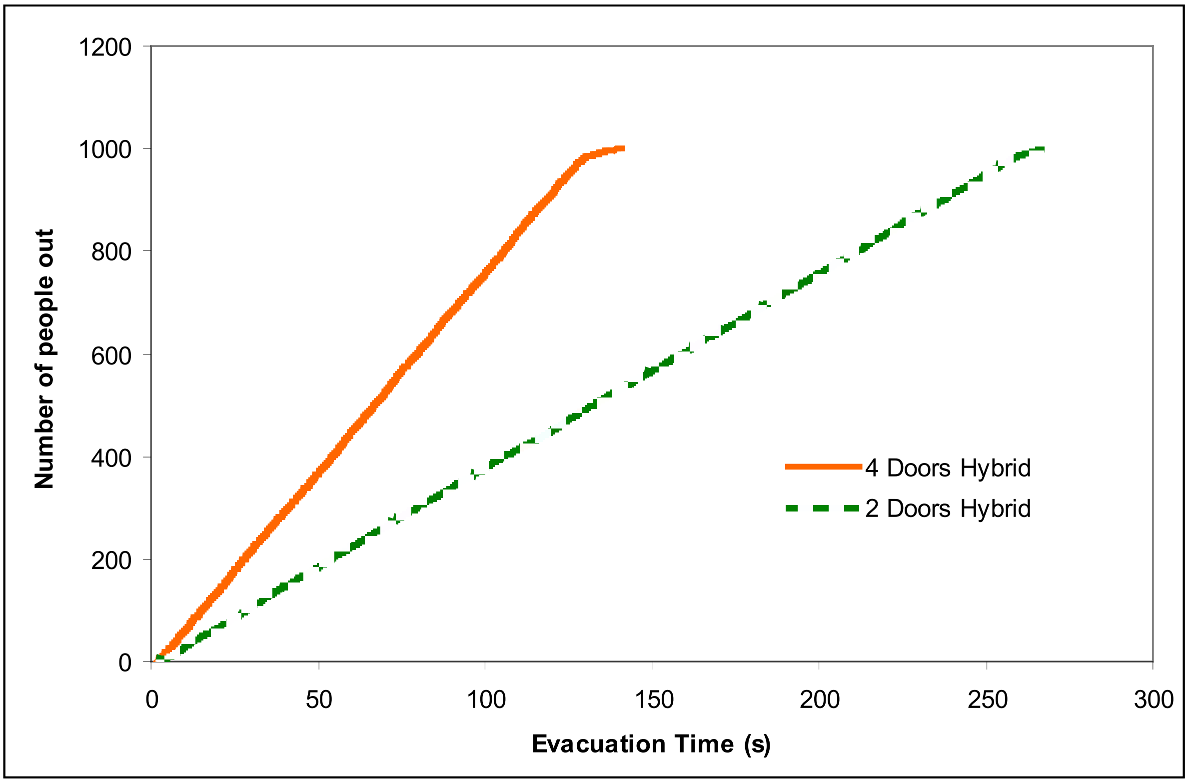


Figure 7-13 Comparison of evacuation times when number of exits available is varied

Table 7-2 Evacuation Times

Scenario	Evacuation Time (s)
4 Doors open	141.2
2 Doors open	267.2

The evacuation times in Table 7-2 illustrate that when using only 2 doors, the evacuation time is 1.9 times, or approximately double than when the simulation is run with 4 doors. This result closely matches the expected result and indicates that the hybrid model is appropriately simulating crowd dynamics.

7.4 Summary

This chapter has described the design and implementation of an approach called the Hybrid Spatial Discretisation (HSD), which is a novelty in the field of evacuation modelling. This approach allows enclosures to be represented using a combination of the current three techniques of spatial representation namely Fine Node networks, Coarse Node networks and Continuous Regions. Moreover, this chapter documents the intricate algorithmic procedures which enable all three models to inter-operate across all 6 possible interface transitions (Coarse Node ↔ Fine Node, Fine Node ↔ Continuous and Coarse Node ↔ Continuous). In addition, some of the issues which arise from interfacing different spatial types are evaluated followed by the discussion of some resolutions. Furthermore, the selection of where to use the different spatial types is user-based thereby providing users with the flexibility in defining geometries with a combination of all spatial types. However, the outcome of these selections would be more effective if the selection strategy is based on the recommendations elicited in Chapter 2.

In addition, this chapter includes a series of IMO test cases for the initial validation of the bEX-H model. The outcomes of this initial validation have demonstrated that the key behavioural and movement mechanisms in the model are operating as intended. In addition, these validation exercises demonstrate that the approaches used for interfacing the different spatial representation types have been appropriately implemented.

The HSD is particularly useful in modelling very large complex geometries such as underground environments, large multi-storey buildings, large sports arenas and urban environments. The further verification and application of the HSD to large environments is described in Chapter 8.

Chapter 8 – Verification of the Hybrid Spatial Discretisation Model

This chapter describes the verification of the Hybrid Spatial Discretisation (HSD) model and demonstrates the efficacy of the methodologies used for integrating all the spatial representation techniques. This chapter focuses on the application of the HSD model using different test cases to demonstrate its capabilities namely:

1) Ability to tackle problems where accuracy and speed of performance is an issue. This is demonstrated by the application of the HSD approach to a complex multiple compartment geometry. This test case is demonstrated using two different configurations; the first one involves representing the geometry using a combination of coarse nodes, fine nodes and continuous regions and the second one using a combination of fine nodes and continuous regions only.

2) Ability to tackle extremely large problems on an urban scale. This is demonstrated by the application of the HSD approach to an extremely large city block scenario (2 km by 2 km). This geometry is modelled using a combination of fine nodes and coarse nodes.

3) Ability to tackle moderately large problems where size of domain is a challenge. This is demonstrated by the application of the HSD approach to a large tunnel station complex using a combination of all three spatial representation methods.

The HSD approach utilises three core models, each offering different levels of computational performance. As a result, the different combinations of the three models are likely to impact the numerical performance of the hybrid approach. In this respect, this chapter includes an investigation as regards the scalability of the spatial representation methods as well as some of the key factors which need to be considered in order to maximise the computational efficiency of the HSD.

8.1 Multi-Compartment Geometry Case

This section describes the approach taken towards the quantitative validation of the HSD technique. This was achieved by the application of the HSD approach to the complex geometry depicted in **Figure 8-1**. The structure has overall dimensions of 13m x 16m and has a free floor space area of 186 m^2 and comprises of 17 irregular shaped rooms with two external exits.

Two different hybrid cases are presented. In the first example – Hybrid-1 - the multi-compartment geometry shown in **Figure 8-1** is modelled using all three spatial representations. The building geometry was setup so that 40% of the floor area was modelled using the Continuous approach, 30% of the floor area was modelled using the Fine Mesh approach and 30% was modelled using the Coarse Mesh approach. The particular location of the different discretisation regions is shown in **Figure 8-1**. The building population of 300 agents is initially distributed so that every compartment within the geometry is occupied. The particular distribution of the three different discretisation regions was selected to ensure that during the simulation all six possible interface transition regions: Coarse Node ↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous; would be utilised. In the second Hybrid example, the same multi-compartment geometry is modelled using only two spatial representations i.e. the Fine and Continuous approaches and is known as Hybrid-2. In Hybrid-2, the region either side of each internal exit and the region inside the external exits are represented using the continuous approach (**Figure 8-2**). The Hybrid-2 geometry was setup so that 32% of the floor area was modelled using the Continuous approach and 68% of the floor area was modelled using the Fine Mesh approach.

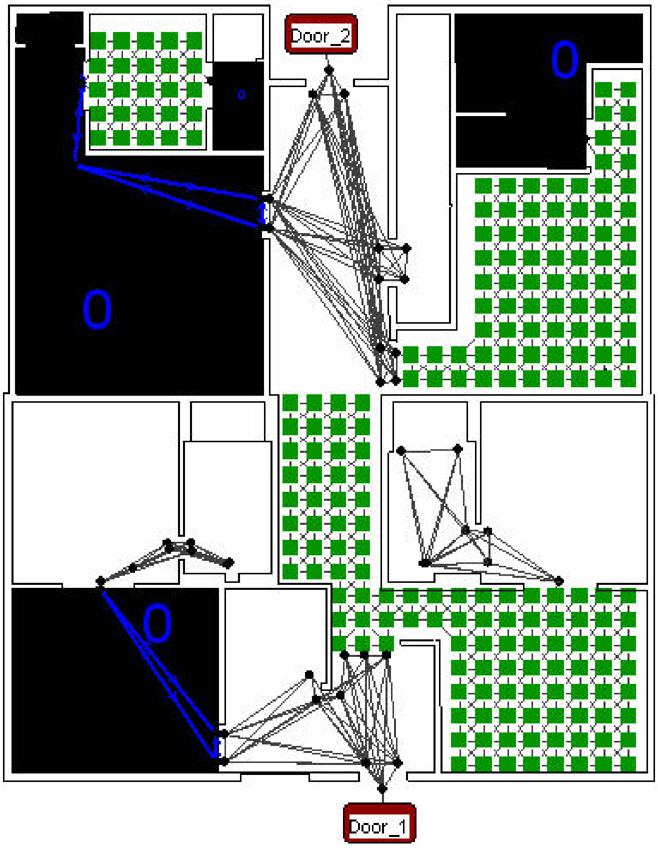


Figure 8-1 Complex geometry used in bEX-H Hybrid-1 demonstration example showing different discretisation regions, White region: Continuous approach; Dark region: Coarse node approach and Grid of Nodes region: Fine node approach.

The experimental set up for the simulations in bEX-H was as follows:

- 300 agents in total with average density of 2 persons/m^2 in each compartment. This average density was preferred, as it would allow enough room in the compartments for the agents to demonstrate their various behavioural capabilities. If the initial population densities in the compartments were chosen to be too high (e.g. 4 persons/m^2), the agents would be closely packed to each other, such that they would not have sufficient space available to demonstrate agent avoidance behaviours such as overtaking (in the fine and continuous models). On the other hand, if the initial population densities were too low (e.g. less than 1 person/m^2), the agents would not demonstrate any agent avoidance behaviours at all.

- To ensure that the results were comparable against results from the other spatial configurations (all-Coarse, all-Fine, all-Continuous), no external influences, for instance, flow rate limitations or reduction in travel speeds were imposed on the exit. Flow Rate at exit was set to Free-Flow.

- For the experiments, the population was generated using a feature called Manual Population Generation as some of the attributes such as travel speeds and response times needed to be defined specifically. The default values **[Galea 2004a]** for the attributes of the occupants are as shown in **Table 8-1**.

- The initial travel speed of the occupants was set to 1.5 m/s and also, to ensure that the occupants would start to move without any delay, the response times of all the occupants was set to 0 seconds.

- Both external exits were available throughout the simulation and were 1.0 m wide. The internal exits were between 1.0 m and 1.5 m wide.

Table 8-1 Evacuee profile attribute default values

Category	Attribute	Unit	Default Value	
			Min	**Max**
Physical	Mobility	-	1.0	1.0
	Age	Years	20	20
	Weight	Kg	50	183
	Gender	-	Male	Male
	Agility	-	3	7
	Height	M	1.54	1.83
Psychological	Response Time	S	0	30
	Patience	S	1	5
	Drive: [female] [male]	-	1 5	10 15
Travel Speeds	Fast Walk	m/s	0.80	1.5
	Walk	m/s	0.72	1.35
	Crawl	m/s	0.16	0.30
	Leap	m/s	0.64	1.20
	Stair-Up	m/s	*	*
	Stair-Down	m/s	*	*
Experiential	Target Exit	-	Nearest Exit	
	Familiarity		Nearest Exit	
	Itinerary		Null	

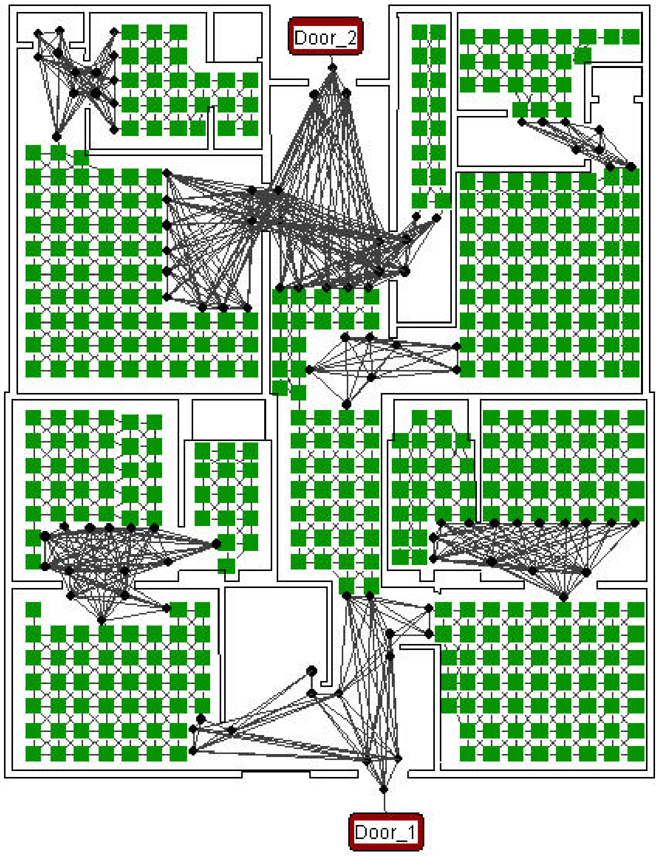


Figure 8-2 Complex geometry used in bEX-H Hybrid-2 demonstration example showing different discretisation regions, White region: Continuous approach and Grid of Nodes region: Fine node approach.

In order to demonstrate the differences and similarities between the various approaches, the demonstration case is repeated using; all Coarse Nodes, all Fine Nodes and all Continuous Regions. Ten simulation runs were conducted for each case. Each repeat simulation involved the same agents located within the same starting compartments, however; their location within the compartment was randomised for each repeat. In the All-Coarse Nodes case, the geometry was modelled by using a combination of compartment and intersection blocks as shown in **Table 8-3**. Regions 1 and 2 were set as intersection blocks as they would require modelling agent flows from different directions, for instance, regions 1 and 2 would encounter flows from exits of compartments labelled A, B, C and D, E respectively.

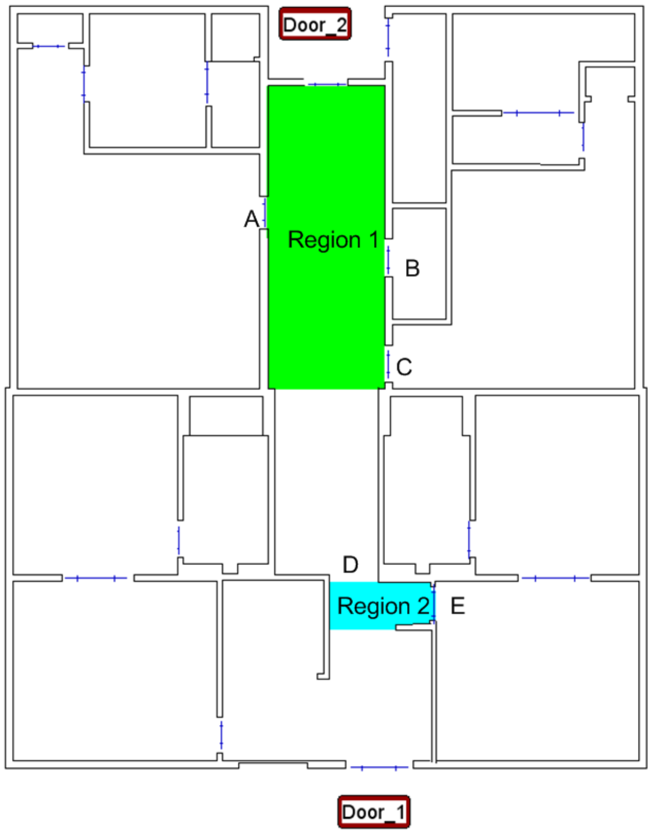


Figure 8-3 Complex geometry used in bEX-H demonstration modelled using all Coarse node approach showing location of Intersection Nodes (shaded regions 1 and 2) in the geometry. All the other areas modelled using Compartment Nodes.

8.1.1 Results and Discussion

The evacuation time curves for each case are shown in Figure 8-4. These curves represent an average for the 10 repeat simulations. As can be seen the four curves for the Hybrid-1, Hybrid-2, All-Continuous and All-Fine simulations fall almost on top of each other with the All-Coarse curve producing faster egress times (see Figure 8-4). This comparison benefits from each of the five cases being setup precisely in an identical manner and in having the core software for each case being identical. Nevertheless, differences in the overall egress times (see **Table 8-2**) and detailed evacuation dynamics do exist (see **Figure 8-5** and **Figure 8-6**).

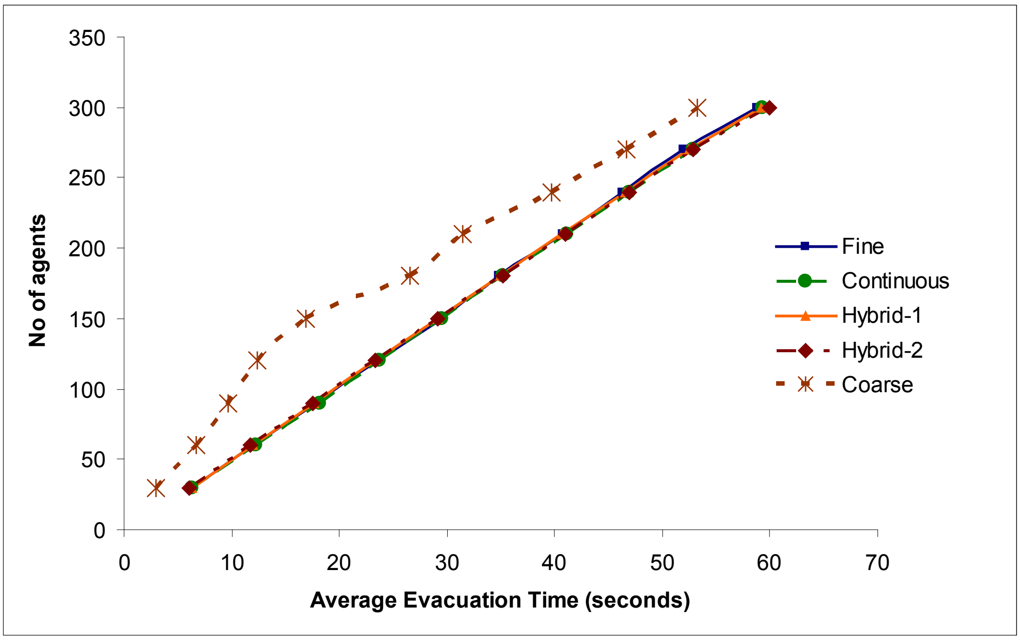


Figure 8-4 Time taken for building population to evacuate using All-Fine, All-Continuous, All-Coarse and Hybrid spatial representations

It is clear from **Figure 8-4** that the All-Coarse case consistently underestimates the egress times throughout the evacuation. This is due to the All-Coarse approach being unable to correctly incorporate the nature of the agent-agent interactions, especially in situations involving moderate to high crowd densities.

Compared to the average total evacuation time predicted by the All-Continuous model, the All-Fine model predicts a time which is 0.8% faster while the All-Coarse is 10.3% faster. The Hybrid-1 model produces an average total evacuation time which is 0.2% faster and the Hybrid-2 model produces an average total evacuation time which is 1.2% slower than that produced by the All-Continuous model (**Table 8-2**). The exit curves and the total evacuation times produced by the various models suggest that both Hybrid models are generating similar evacuation dynamics to the All-Continuous and the All-Fine models. This indicates that the intricate movement and behavioural algorithms operating at the various types of interface transition regions within the Hybrid Model are functioning appropriately.

Table 8-2 Summary of results averaged over 10 simulations

Spatial Representation	Average Total Evacuation Time (s)	Average Run Time (s)
All-Continuous	59.3	14.6
All-Fine	58.8	1.5
All-Coarse	53.2	1.4
Hybrid-1	59.2	7.1
Hybrid-2	60.0	8.4

Presented in **Table 8-3** are the average exit flow rates for each exit for each of the five cases. For each exit, the All-Coarse case produces the largest average flow rates. The maximum difference in average unit flow rates between the various cases for Door 1 is 12% and is between Hybrid-2 and the All-Coarse cases and for Door 2 the maximum difference is 9% between the Hybrid-1 and the All-Coarse cases. The maximum difference in average unit flow rates for the Hybrid-1, Hybrid-2 and the All-Continuous cases is 2%. It should be noted that the physical region just inside each of the external exits in these three cases are each modelled as continuous regions. Thus the evacuation dynamics in the regions just in front of the external exits and through the external exits would therefore be identical in these three models, contributing to the similarity in the average unit flow rates. The maximum difference in average unit exit flow rates for the All-Fine and the All-Continuous cases is 5%, with the All-Fine case producing greater average unit flow rates than the All-Continuous case.

Table 8-3 Average Unit Flow Rate for each case averaged over 10 simulations

Spatial Representation	Average Unit Flow Rate (persons/m/s)	
	Door 1	Door 2
All-Continuous	2.55	2.60
All-Fine	2.57	2.74
All-Coarse	2.87	2.83
Hybrid-1	2.60	2.58
Hybrid-2	2.52	2.64

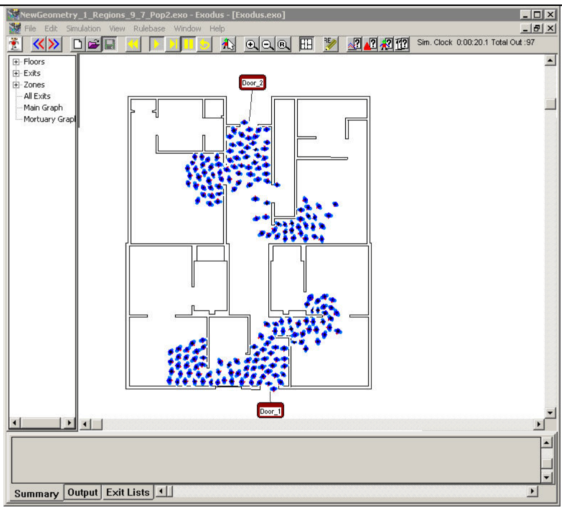
These results are consistent with the overall average total evacuation time, with the greatest average flow rates being associated with the shortest overall average total evacuation times. These results are also consistent with the egress problem being investigated being essentially driven by the queues at the final exits. This in turn was generated by using an instant response time for the agent population. This was deliberately selected so as to produce the maximum congestion at each of the internal and external exits in order to highlight the differences between the various approaches.

The accuracy of the predictions is also dictated by the granularity of the model used around the external exit locations. The hybrid discretisation strategy used as shown in **Figure 8-1** involved using the continuous space around the external exits. The experiment was repeated by using coarse nodes instead of continuous regions around the external exits and it was found that total average total evacuation time was 54.2 s which is 9.2% faster than the Hybrid-1 result in **Table 8-2**. This is to be expected as the people within the coarse node would be removed faster from the simulation due to the lack of representation of the complex agent to agent interactions around exit locations in the coarse node. In this respect, when using the HSD approach, it is essential to consider which sections of the geometry would be represented with the appropriate spatial representation types (see Chapter 2).

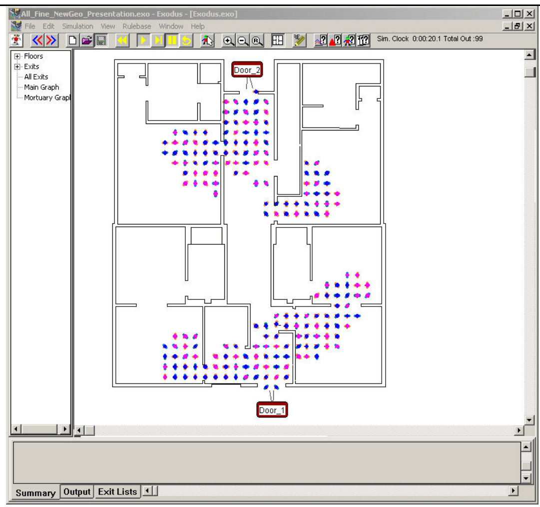
Presented in **Figure 8-5** and **Figure 8-6** are images representing the evacuation situation at 20 sec and 50 sec respectively, into the evacuation. These images allow comparison of the detailed evacuation dynamics produced by each of the models during the early and late stages of the evacuation. Note the congestion at most of the internal exits and each of the external exits. From **Figure 8-5** we note that the population distribution for all five cases are quite similar, in particular for the All-Continuous (**Figure 8-5**a), All-Fine (**Figure 8-5**b), Hybrid-1 (**Figure 8-5**d) and Hybrid-2 (**Figure 8-5**e). Consider two compartments, the large compartment located in the upper left and the large compartment located in the lower left of the geometry. It can be noted that the number of agents remaining in the large compartment in the upper left of the geometry after 20 sec is: 28, 30, 27, 29 and 53 for the All-Continuous, All-Fine, Hybrid-1, Hybrid-2 and All-Coarse models respectively. For the compartment located in the lower left of the geometry it can be noted that the number of agents remaining in the compartment after 20 sec is; 26, 19, 18, 18 and 0 for the All-Continuous, All-Fine, Hybrid-1, Hybrid-2 and All-Coarse models respectively. With the exception of the All-Coarse model, the other models are in close agreement. Even at 20 sec into the simulation, the All-Coarse model appears to be deviating away from the other four models while the other four models are in close agreement. This is consistent with the exit curve shown in **Figure 8-4**.

The population distribution at 50 sec into the simulation is depicted in **Figure 8-6**. Here we note again that all five cases are broadly similar, with the All-Coarse model having the smallest number of agents remaining within the geometry. This again is consistent with the exit curve shown in **Figure 8-4**. It can also be noted that the All-Fine model has agents spread over three compartments in the lower portion of the geometry (close to Door 1) whereas the other models have agents located in two compartments at most.

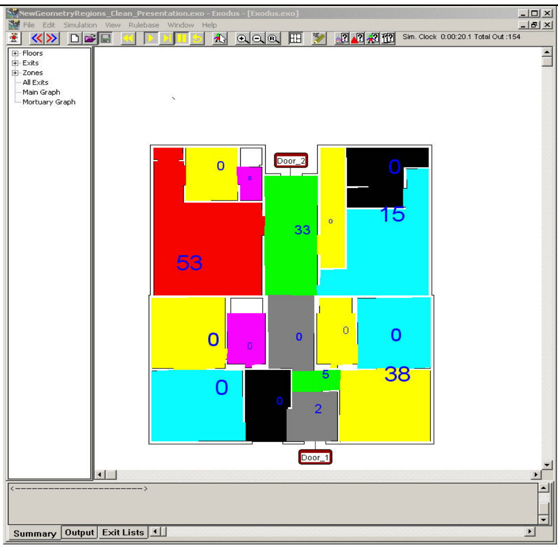
From the results presented in **Figure 8-4** and **Table 8-2**, Hybrid-1 and Hybrid-2 appear to produce very similar results. However, through investigation of **Figure 8-5** and **Figure 8-6**, it may be argued that Hybrid-2 appears to produce marginally better results than Hybrid-1, especially towards the end of the simulation. Here better implies that it is more closely resembles the All-Continuous case. The potential reduction in accuracy of the Hybrid-1 simulation is due to the significant presence of the coarse nodes in this small example. Indeed in reality, it is unlikely that such a case would have been modelled using coarse nodes due to the small size.



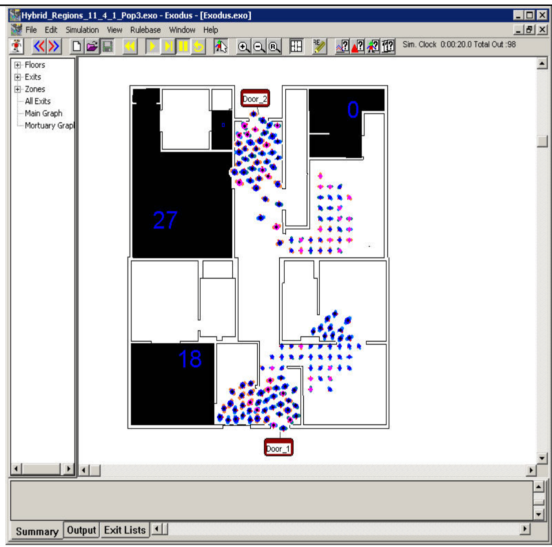
(a) All-Continuous at 20 sec



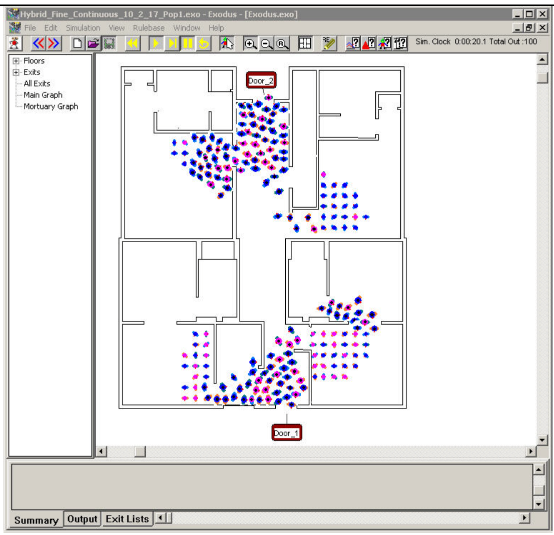
(b) All-Fine at 20 sec



(c) All-Coarse at 20 sec

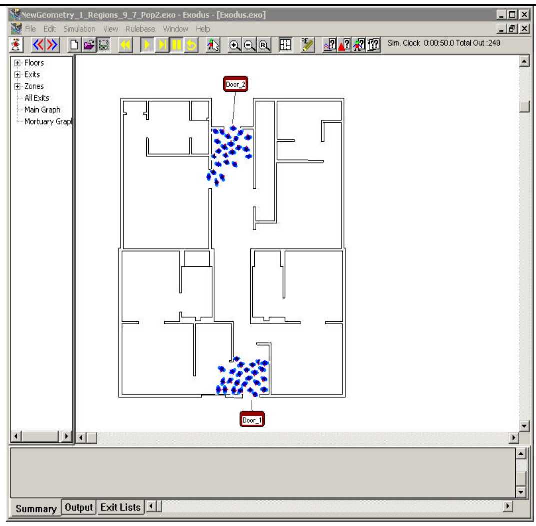


(d) Hybrid-1 at 20 sec

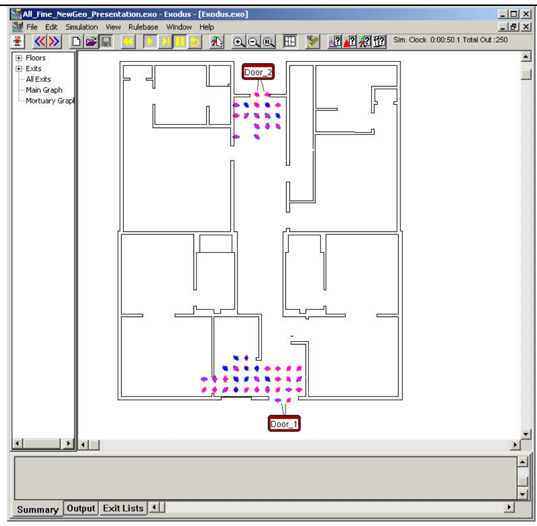


(e) Hybrid-2 at 20 sec

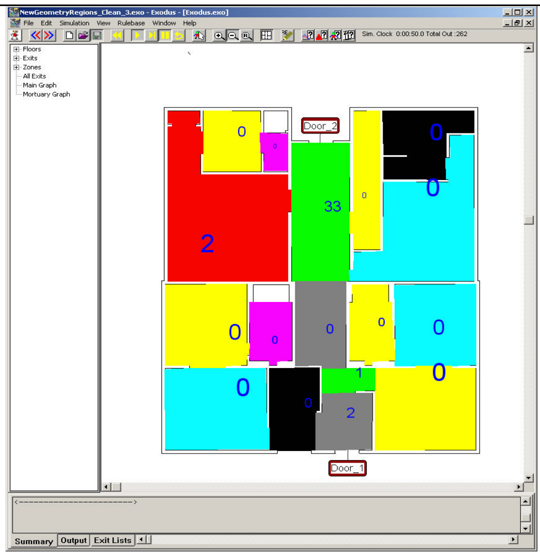
Figure 8-5 Comparison of the population distribution 20 seconds into the evacuation for all five models



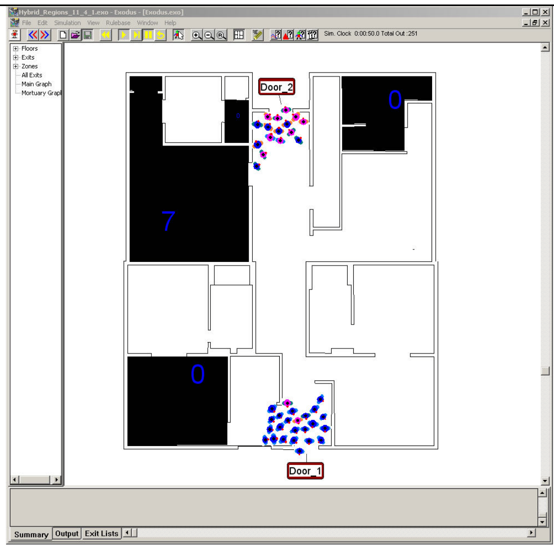
(a) All-Continuous at 50 sec



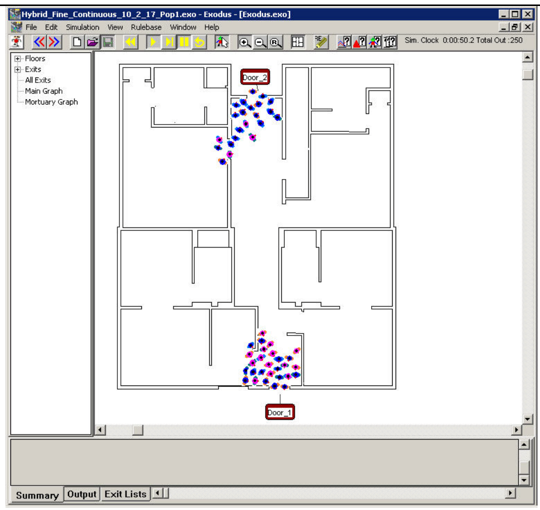
(b) All-Fine at 50 sec



(c) All-Coarse at 50 sec



(d) Hybrid-1 at 50 sec



(e) Hybrid-2 at 50 sec

Figure 8-6 Comparison of the population distribution 50 seconds into the evacuation for all five models

Also shown in **Table 8-2** are the average run times for the 10 repeat simulations for each case. The average run times were determined with the run-time visualisation graphics turned off, as the All-Coarse case produces a trivial amount of run-time graphics compared to the All-Fine and All-Continuous cases. By turning the graphics off a better estimation of the computational effort required for each approach is determined. As can be seen from **Table 8-2**, the All-Continuous case produces the longest run time of 14.6 sec, which is 10 times as long as the All-Fine case which requires some 1.5 sec of run time. As noted in Chapter 2, the superior computational speed offered by the All-Fine approach is one of its principle advantages over the All-Continuous approach however, it should be noted that the differences in run time between All-Fine and All-Continuous discretisations in this case are for a relatively small example.

Furthermore, the implementation of the All-Continuous plug-in module, in particular the manner in which it interfaces with the core code has not been fully optimised. As a result, the differences between the performance of the All-Continuous and the All-Fine are expected to decrease once the optimisation is performed. In the relatively small demonstration case presented, the All-Coarse and All-Fine cases take approximately the same amount of run time with the All-Coarse being only 7% quicker than the All-Fine. It should be noted that in larger cases the All-Coarse simulation has achieved 20% faster run times than the All-Fine simulation (see Section 6.5). Furthermore, as with the All-Continuous case, the implementation of the All-Coarse plug-in module has not been fully optimised and so the run time for the All-Coarse case is larger than would be expected.

The Hybrid-1 case requires a run time of 7.1 sec which is 4.7 times as long as that for the All-Fine case while the All-Continuous case requires over 2.0 times as much run time as the Hybrid-1 case. Thus, while the Hybrid approach is more than twice as fast as the All-Continuous approach, it is still substantially slower than the All-Fine simulation. If the run time scaled simply in proportion to the floor area represented by the various discretisations, it would be expected that the Hybrid-1 would require some 6.7 sec (based on the 40:30:30 split between discretisation approaches and the times presented in **Table 8-2**). Clearly the scaling is more complex than this. The scaling is also dependent on the amount of time the agents spend within each region, which in turn is dependent on the level of crowding within the section as well as the physical size of the region. Furthermore, it should be noted that the software implementation of bEX-H prototype has not yet been fully optimised. Computational inefficiencies currently exist within the prototype implementation of the Coarse node discretisation code and interface between the Continuous regions and other discretisation types. These computational inefficiencies are most significant for transitions from the continuous discretisation to the coarse or fine discretisation and from the coarse or fine discretisation to the continuous discretisation. This partially explains why the run time for the Hybrid-2 case is 18% longer than that of the Hybrid-1 case, even though the Hybrid-2 case contains a smaller proportion of the computationally slower continuous discretisation (32%) compared to the Hybrid-1 case (40%). Within the Hybrid-2 case there are almost twice as many (14 compared with 8) transition regions to and from the continuous regions compared to the Hybrid-1 case. In addition, Hybrid-2 contains no coarse regions – which are computationally the most efficient.

8.2 City Block Case

This section describes the application of the HSD approach to an extremely large and complex city block case depicted in **Figure 8-7**. The outer boundary of the city has overall dimensions of 2 km x 2 km connected to a 200 m x 200 m assembly area. The road network comprises of dual carriageway roads of width 7 m coupled with 1.5 m wide pavements on either side [TWUK 2010] [DFT 2010]. The total free space area occupied by the network of roads and the assembly area is 214260 m^2. Moreover, the assembly comprises of 10 compartments whereby each compartment has dimensions of 40m x 50m and is connected to a 10m wide exit. The geometry was set up so that 71% of the free space area was modelled using the Coarse Mesh approach and 29% of the free space area was modelled using the Fine Mesh approach. The locations of the different discretisation regions are as shown in **Figure 8-7**. A population of 10 000 agents is initially distributed so that all the roads, with the exception of the 200 m road connecting the city and the assembly area, are occupied. 10% of the agent population is assigned to exit 1 as target, 10 % to exit 2 and same proportions up to exit 10. Furthermore, the agents were assigned instantaneous response times and their remaining attributes were defined as shown in **Table 8-1**. In addition, there were no flow rate limitations imposed on the exits.

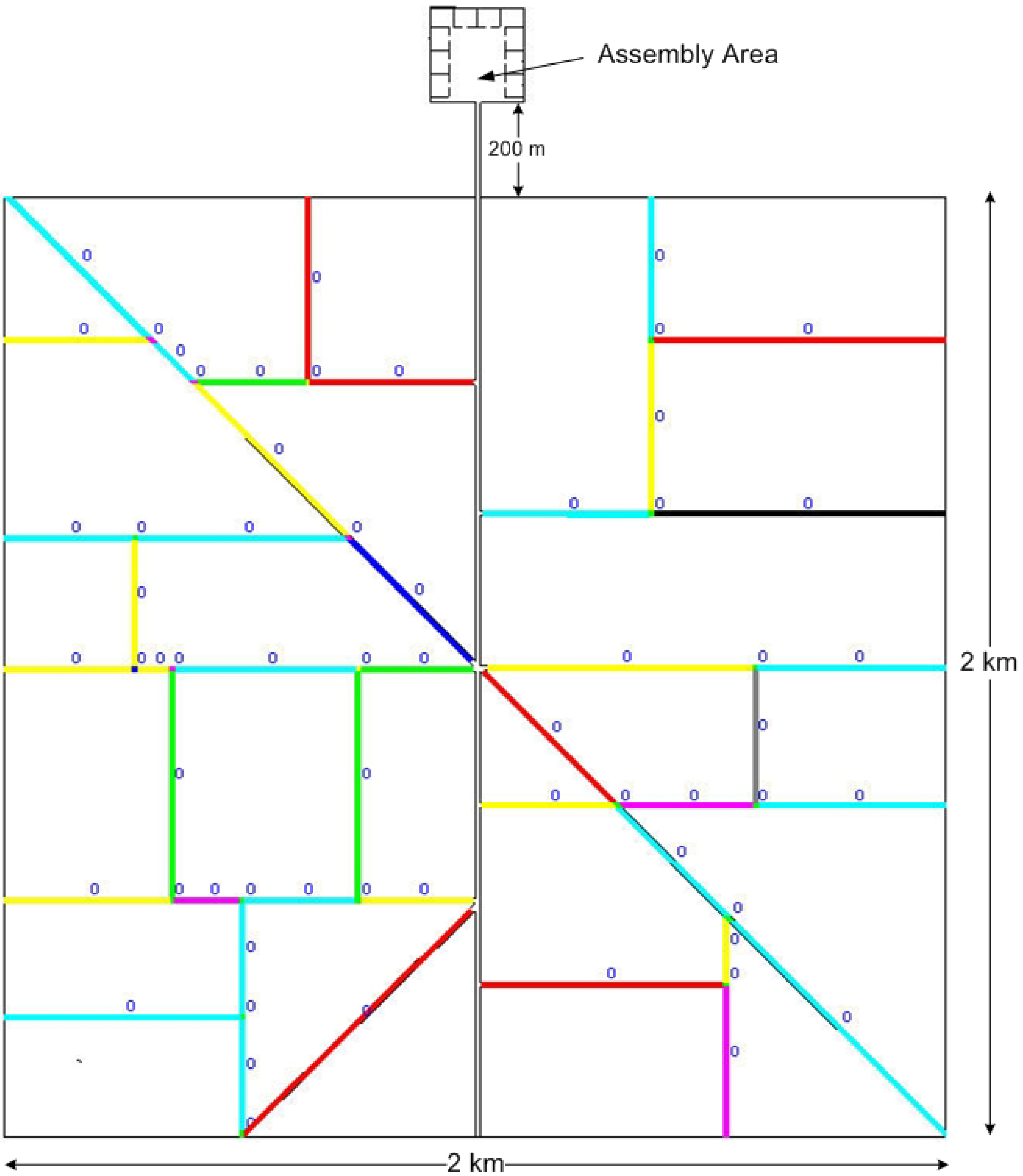


Figure 8-7 Large City Block geometry; empty spaces represent blocks of buildings surrounded by networks of dual carriageway roads; shaded roads represent coarse node approach, white road in the middle of the city leading to assembly area represents fine node approach.

In order to demonstrate the similarities and differences between the HSD approach and the Fine Mesh approach, the city block case is repeated using all Fine Node. 10 simulations were conducted for each case. Each repeat simulation involved the same agents located within the same road segments; however, their location within each road segment was randomised for each repeat.

8.2.1 Results and Discussion

The evacuation time curves for the All-Fine and Hybrid cases are shown in **Figure 8-8**. These curves represent an average for the 10 repeat simulations. As can be seen both curves demonstrate similar trends. Both cases have been set up precisely in an identical manner and the core software for each case is identical. Nevertheless, differences in the overall evacuation times do exist (See **Table 8-4**).

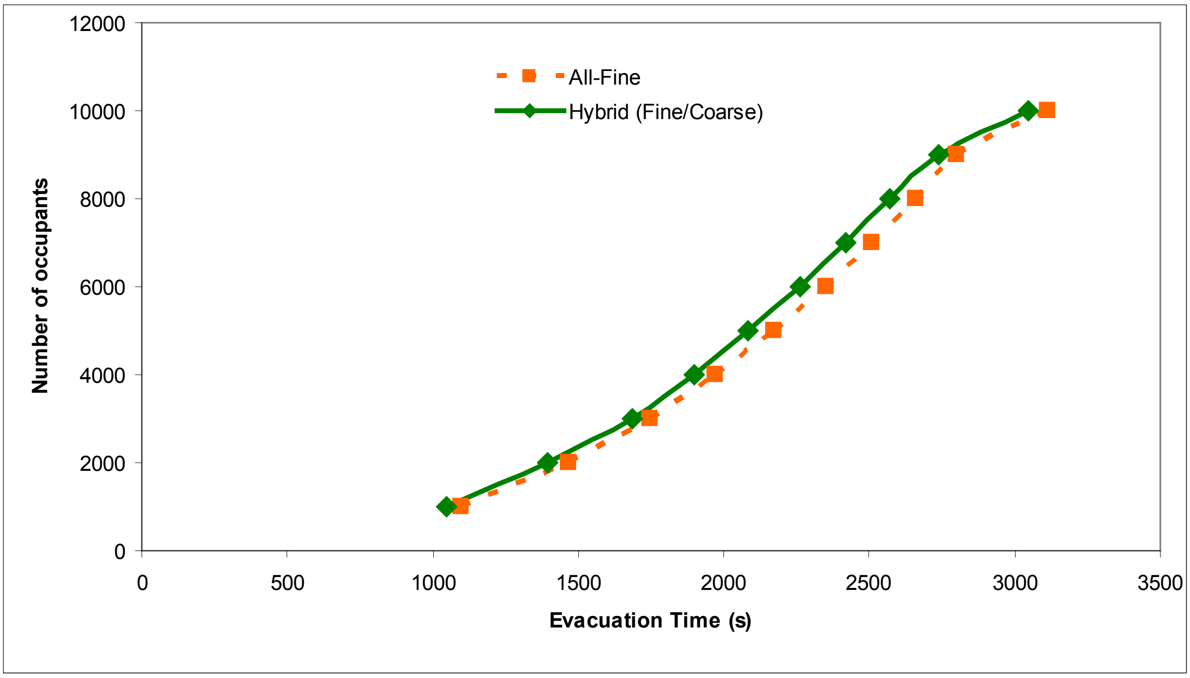


Figure 8-8 Time taken for a specific percentage of people to evacuate using All-Fine and Hybrid spatial representation

Compared to the average total time predicted by the All-Fine model, the Hybrid model predicts an egress time which is 2.1% faster. The exit curves and the total evacuation times suggest that the Hybrid is generating similar evacuation dynamics to the All-Fine model. Because of the inability of Coarse nodes to correctly represent the nature of agent-agent interactions, the flow rates at the entrance of the main artery road (road in the middle of the city) is faster in the Hybrid configuration than in the All-Fine model. This results in the Hybrid model producing overall faster evacuation times than the All-Fine model.

Table 8-4 Summary of results averaged over 10 simulations

Spatial Representation	Average Total Evacuation Time (s)	Average Run Time (s)
All-Fine	3112.7	1225.6
Hybrid	3048.8	818.0

Also shown in **Table 8-4** are the average run times for the 10 repeat simulations for each case. The Hybrid model produces the fastest run time of 818 s which is 33.3% faster than the All-Fine model. This significant speed up is due to the superior computational speed offered by the large proportion of Coarse nodes in the Hybrid configuration.

In the light of the results shown in **Figure 8-8** and **Table 8-4**, the HSD approach is seen to provide a substantial improvement in the run times of the simulation whilst maintaining the accuracy of the fine nodal approach. The computational benefits offered by this approach make it suitable for the simulation of egress in extremely large environments. In addition, the results demonstrate that the algorithmic procedures employed in modelling the transitions of the agents across the interface regions operate appropriately and consistently.

8.3 Large Tunnel Station Complex Case

This section describes the application of the HSD approach using a large tunnel station complex depicted in Figure 8-9. The enclosure comprises of a long 700m long and 2m wide tunnel connected to a platform and has a total free floor space area of $1900m^2$. The segment L represents a 4m wide corridor with a centrally placed 2m wide internal exit on either side of the corridor. The end of the platform is connected to two external exits each of width 2m. The geometry presented is modelled using all three spatial representation types. The spatial discretisation strategy for this geometry is chosen such that region in the far field is represented using Coarse Nodes, pinch points and exit locations are represented using Continuous Regions and remainder of the geometry is modelled using Fine Nodes.

The location of the various discretisation regions is as shown in **Figure 8-9**. Region A is modelled using the Coarse Mesh approach. Continuous Regions were used in different areas of the geometry as follows; C1: Region connecting the tunnel and the platform, C2 and C3:

Regions surrounding internal exits and C4: Region surrounding the exit locations. The remainder of the free space was represented using the Fine Mesh approach. The percentage occupied by the various spatial types was as follows; 73% of the area was modelled using the Coarse Mesh approach, 23% of the area was modelled using the Fine Mesh approach and 4% using the Continuous Region. A building population of 2000 agents is initially set up distributed in a 250m long segment of the tunnel as shown in **Figure 8-9**. With this distribution of discretisation regions, three interface transition regions namely; Coarse Node → Continuous, Continuous→ Fine Node, Fine Node → Continuous, are utilised during the simulation. The population characteristics were the same as described in Section 8.2.

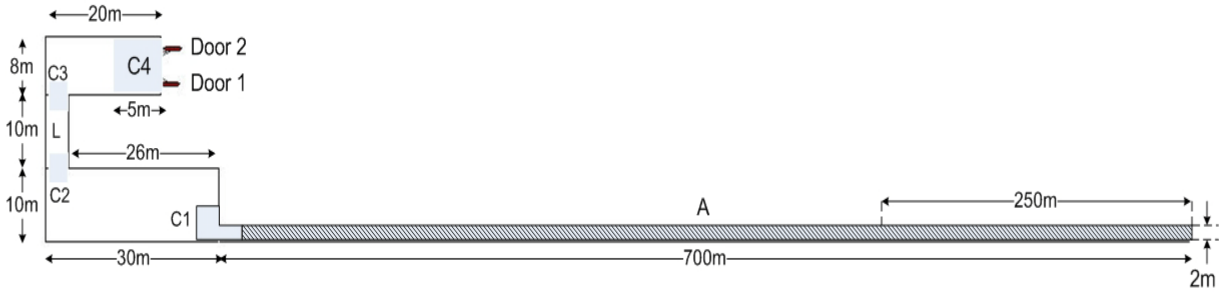


Figure 8-9 Large Tunnel Station Complex modelled using all spatial representation types; Coarse node approach (A), Continuous Region (C1,C2,C3,C4), Fine node approach (Remainder of free space)

With a view to demonstrating the differences and similarities between the HSD approach and the Fine Mesh approach, the Large Tunnel Station Complex case is repeated all Fine Nodes such that ten simulation runs were conducted for each case. Each repeat simulation involved the same agents located in the same starting segment of the tunnel; however, their location within that segment was randomised for each repeat.

8.3.1 Results and Discussion

The evacuation time curves for each case are as illustrated in Figure 8-10. These curves represent an average for 10 repeat simulations. In this comparison, the same core software and identical set up was used for both cases, however, differences in the overall times (see **Table 8-5**) do exist. The Hybrid model predicts an evacuation time which is 3.5% slower than that predicted by the All-Fine model.

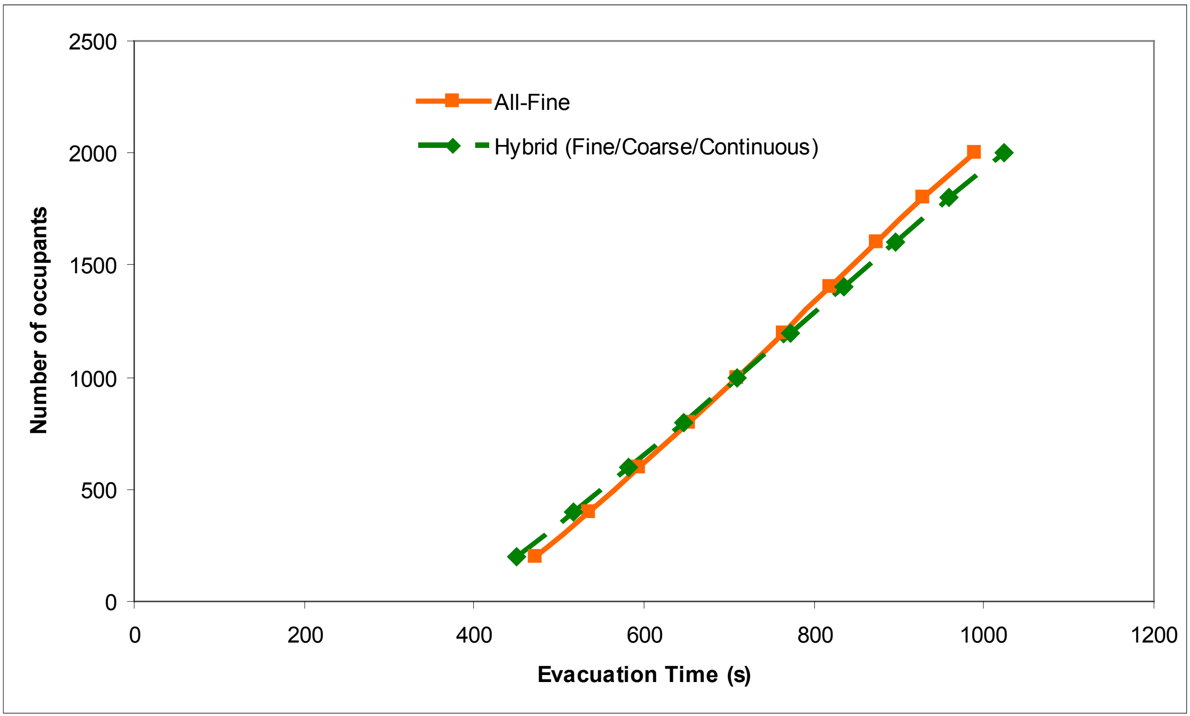


Figure 8-10 Time taken for a specific percentage of people to evacuate using All-Fine and Hybrid spatial representations

Table 8-5 Summary of results averaged over 10 simulations

Spatial Representation	Average Total Evacuation Time (s)	Average Run Time (s)
All-Fine	988.8	80.6
Hybrid	1023.8	42.0

As seen from Figure 8-10, the Hybrid model is predicting faster evacuation times than the All-Fine model up to around 700 seconds into the simulation, after which the Hybrid model predicts slower evacuation times. The agents are initially distributed in a segment of the 2m wide tunnel such that the resultant population density is 4 persons/m^2. The narrow configuration of the tunnel limits the amount of lateral space available to the agents during their movement and therefore reduces the amount of overtaking that can take place. However, there are some significant differences in the representation of the behaviour and movement of the agents through the tunnel in the Hybrid model (Coarse nodal section) and the All-Fine model.

In the All-Fine model, during the initial stage of the simulation, the front part of the stream of agents start to move, while the agents at the back remain stationary until there is enough space available for them to move. In this respect, the population density of the front part of the stream of agents is rather sparse. This results in the flow rate at the exit of the tunnel to be low in the beginning of the simulation and the flow rate gradually increases as the density of the people moving towards the exit of the tunnel builds up. However, in the Hybrid model, due to the lack of agent-agent interactions in the Coarse Nodal approach, the flow rate of agents at the exit of the tunnel is high even at the start of the simulation. This explains the faster evacuation times predicted by the Hybrid approach up to 700 seconds into the simulation. In the Hybrid model, flow rate at exit of the tunnel is almost constant throughout the simulation and in Fine Node model, the flow rate increases gradually increases to a peak value which exceeds the peak value in the Hybrid case. After about 700 seconds into the simulation, the flow rate at the tunnel exit in the fine mesh approach starts to exceed the flow rate in the hybrid approach. Moreover in the hybrid approach, the continuous regions predict a slower flow rate at the entrance of the internal exits C2 and C3 and near the external exit location C4. This results in the Hybrid model predicting slower evacuation times towards the end of the simulation.

Also shown in **Table 8-5** are the average run times for the 10 repeat simulations for each case. It can be seen that the Hybrid model is computationally 47.9% faster than the All-Fine model. Although the Hybrid model includes the computationally least efficient Continuous approach, however, the speed up gained by the considerably larger proportion of the computationally most efficient Coarse Node approach (73%) compensates for the lower efficiency of Continuous approach. This test case has demonstrated that the HSD approach can be used in an optimal way for performing egress analyses on large complex domains while maintaining the accuracy of results.

8.4 Investigating the impact of various discretisation schemes on computational performance of HSD

The HSD approach provides the capability of representing enclosures using all three spatial representations, each having different numerical efficiencies. In this respect, the different combinations of the various spatial types will have an impact on the computational performance that can be achieved. The verification tests in Section 8.1 showed that the run time of the hybrid configuration does not simply scale in proportion to the floor area represented by the various spatial types. The run time also depends on other complex factors such as the amount of time the agents spend within each region which in turn depends on the level of crowding and the physical size of the regions. Moreover, although the different spatial types offer different performance benefits, these benefits can be harnessed only if the agents are passing through the area of floor space covered by these spatial types. This is demonstrated in some tests performed on the geometry as shown in **Figure 8-11**.

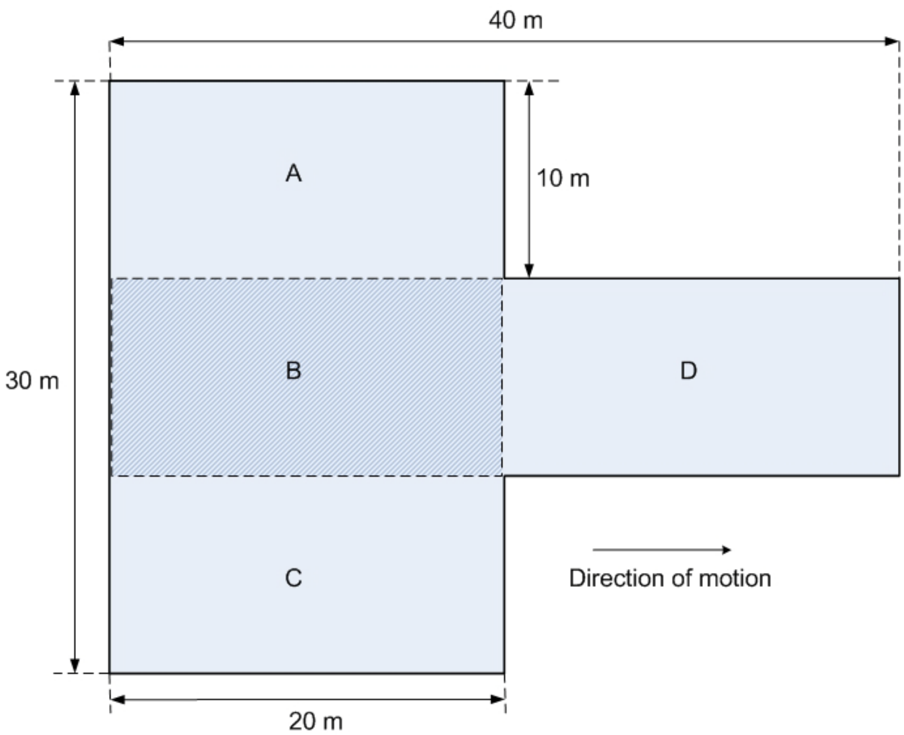


Figure 8-11 Geometry for demonstrating the impact of time of passage of agents

The floor space area was divided into 4 regions (A, B, C and D), each 20m long and 10 m wide. Region B was modelled using a coarse node whereas regions A, C and D were all modelled using the fine node network. The set up for the simulations was as follows:

- 500 agents in region B
- No flow rate limitations were imposed at the exit of the corridor (region D)
- Fast walk (Unimpeded walking speed) : 1.5 m/s
- Response time: 0 s

In order to compare the differences in the run times, the demonstration case was repeated using an all-Fine configuration. The results are as shown in **Table 8-6** and represent an average for 5 repeat simulations.

Table 8-6 Summary of run time results averaged over 5 simulations

Spatial Representation	Average Run Time (s)
Hybrid-1	1.40
All-Fine	1.97

The results indicate that the Hybrid configuration provides a speed up of 28.2% over the all-Fine. The discretisation scheme in Figure 8-11 was then modified so that region A was represented using a coarse node and region B using a fine node network (Hybrid-2). The agents were generated in region B. The average run time for Hybrid-2 was 1.97 s which is the exactly the same run time as the all-Fine configuration. This demonstrates that although 25% of the entire floor space in Hybrid-2 was represented with coarse nodes, there was no improvement in the run times. This is because the agents did not pass over the region of space represented by the coarse node during the entire simulation and as a result, the computational benefits of the coarse node were not gained. This suggests that when using the HSD approach, it is the time of passage of the agents on the various spatial types which impacts the run time rather than the proportion of the floor space represented by the different types. Furthermore, these results illustrate the difference in the computational costs involved in evacuation models as compared to fire field models [Grandison 2004]. In fire field models, the available space is discretised into a network of cells whereby all the cells are actively utilised for computation throughout the entire simulation. This is in contrast with evacuation models in which, only the regions of space which have agents passing over them, are involved in the computation.

In order to further investigate the relationship between the time of passage of occupants over the various spatial types and the associated computational costs incurred, additional tests were conducted on the geometry as shown in **Figure 8-12**. The corridor was divided in half into two regions A and B. The initial set of tests on this geometry entailed generating an agent at the furthest end of the corridor in region A and recording the simulation time and corresponding run time. The aim of these tests was to calculate the computational cost incurred (run time) of moving an agent by one second (time of passage) in the simulation, when using the all-Fine, all-Coarse, all-Continuous and Hybrid configurations. For the Hybrid configuration, all the 6 possible transition types were considered. Furthermore, the results of these tests would be used for deriving equations which would allow estimates of the run time to be made when using different discretisation schemes (prior to running the simulations using the HSD approach). These estimates would be conducive towards identifying discretisation schemes that could maximise the computational efficiency.

The set up for the simulations were as follows:

- 1 agent in region A placed furthest from the exit of the corridor (region B)
- No flow rate limitations were imposed at the exit of the corridor (region B)
- Fast walk (Unimpeded walking speed) : 1 m/s
- Response time: 0 s

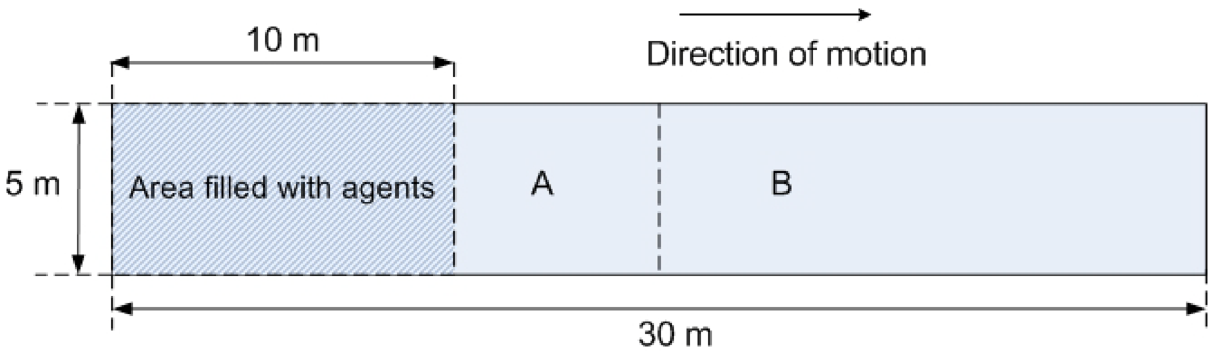


Figure 8-12 Geometry used for investigating the impact time of passage of agents on run time

The computational cost required to move an agent 1 second in the simulation is calculated using Equation 8-1.

$$C = \frac{R}{T}$$ **Equation 8-1**

Where

C : Cost of moving an agent per second of passage

R : Run time of simulation

T : Time of passage of agent in spatial type

Since there was only one agent in each simulation, the simulation time (which is equal to the time of passage in the entire corridor) is a constant of 30 s. Presented in **Table 8-7** are the run times of the simulations and the calculated computational cost for each type. From **Table 8-7**, it can be seen that there is not a significant difference between the computational cost for the Coarse and Fine spatial types. This is because in such models, the computational cost is incurred mostly when computing agent-agent interactions. Computational costs are also incurred in computing person-structure interactions. However, the wall and obstacle locations are static and agents are dynamic, therefore the computational cost is more pronounced for person-person interactions. The results for the Coarse and Fine types in **Table 8-7** correspond to only one agent navigating in the structure without encountering collisions with the walls. Therefore, the computational costs due to person-person and person-structure interactions are not incurred which explains the similarity of the results for the Coarse and Fine cases. The Continuous case shows a higher computational cost as compared both the Coarse and Fine cases. This is because the Continuous model is required to keep track of a larger number of attributes at every time step of the simulation as compared to the Coarse and Fine models.

Table 8-7 Run time results and computational cost for moving an agent by 1 second in different spatial types

Spatial Type	Run Time (s)	Cost
Coarse	0.0150	0.00050
Fine	0.0156	0.00052
Continuous	0.0468	0.00156
Continuous → Fine	0.0310	0.00103
Fine → Continuous	0.0310	0.00103
Continuous → Coarse	0.0310	0.00103
Coarse → Continuous	0.0310	0.00103
Coarse → Fine	0.0156	0.00052
Fine → Coarse	0.0156	0.00052

The section follows with a description of the comparison of the similarities and differences of the actual and estimated computational costs.

The run time for a Hybrid configuration (for instance, Continuous → Fine) can be calculated as follows:

$$R_{hybrid} = R_{continuous} + R_{fine}$$

Using the definition of R from Equation 8-1,

$$R_{hybrid} = (C_{continuous} \times T_{continuous}) + (C_{fine} \times T_{fine}) \qquad \textbf{Equation 8-2}$$

Where:

R_{hybrid} : Run time of entire simulation in hybrid configuration

$R_{continuous}$: Run time of simulation in the continuous segment of the geometry

R_{fine} : Run time of simulation in the fine node segment of the geometry

$C_{continuous}$: Cost of moving an agent per second of passage in continuous model

C_{fine} : Cost of moving an agent per second of passage in fine node model

$T_{continuous}$: Time of passage of agent in continuous space

T_{fine} : Time of passage of agent in fine node network

Since the time taken by the agent to cover the entire distance is 30 s, therefore the time of passage of the agent is equal (15 s) in both the continuous region and the fine node network. Using **Equation 8-2** and the cost values in **Table 8-7**, the value of R_{hybrid} is found to be 0.031 s.

This estimated value of R_{hybrid} is exactly the same as the actual run time result for the Continuous → Fine configuration shown in **Table 8-7**. The same estimates were computed for the other configurations and were found to be the same as the actual run time results. This illustrates that the cost equations (**Equation 8-1** and **Equation 8-2**) provide an adequate estimation of the run time costs incurred by the various hybrid configurations, under the conditions examined.

Further tests were then conducted by gradually increasing the number of agents in the simulation and then recording the simulation times and corresponding run times. The increase in the number of agents results in an increase in the number of person-person interactions thereby increasing the run times of the simulations. Simulations for each spatial configuration were run 5 times in order to obtain an average of the run times. The run time curves for each case are shown in **Figure 8-13**.

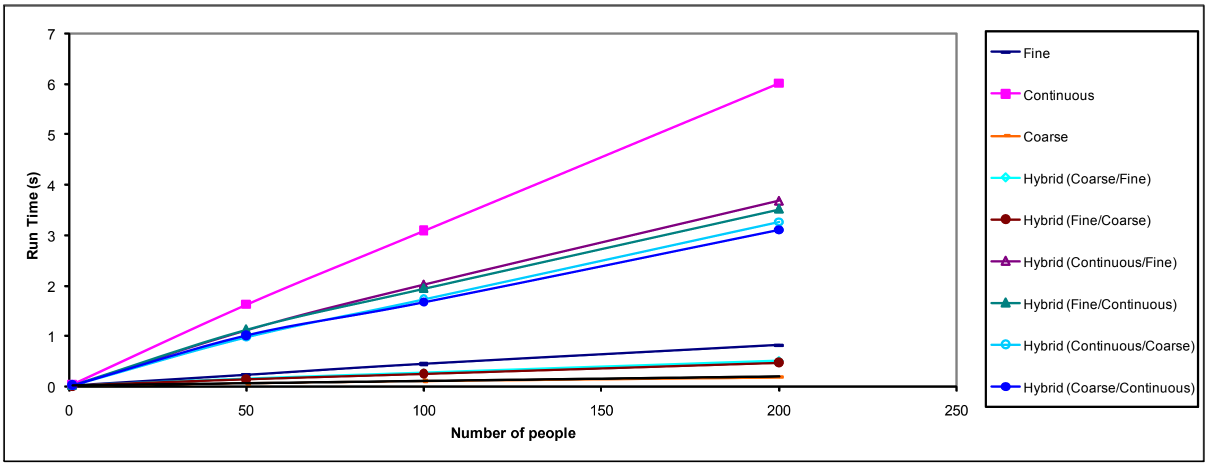


Figure 8-13 Variation of run time with increasing number of agents

The results show a linear increase of run time with increasing number of agents for each case, with the coarse showing the fastest run times and the continuous the slowest as expected. The curves showing the variation of the computational cost (run time per second of passage) with increasing numbers of agents are as shown in **Figure 8-14**. Although the Continuous model incurs the highest computational costs, however, the linearity of the cost curve shows the promising scalability of the model. This scalability is achieved due to the implementation of the Grid Method for optimising the spatial search of agents in the Continuous Model (Chapter 5).

From **Figure 8-14**, it can be seen that the Continuous→ Fine configuration incurs slightly higher computational costs than the Fine→ Continuous configuration with increasing number of agents in the simulation. This is due to the differences in the computational costs of computing the person-person interactions in the Continuous and Fine Node models. In the Continuous → Fine configuration, the agents initially start in the continuous space. Due to the close packing of the agents at the beginning of the simulation, there are an increased number of person-person interactions in the continuous space (in the form of collision detection and avoidance). This is in contrast with the Fine→ Continuous configuration, in which the agents are initially packed in the fine nodal section of the corridor. At the start of the simulation, although the density of agents in the fine nodal section is high, however, the person-person interactions are less computationally expensive as compared to the person-person interactions in the continuous space in the Continuous→ Fine configuration. In this respect, the Fine→ Continu ous configuration demonstrates lower computational costs than the Continuous → Fine configuration. The same explanation accounts for the difference in the Continuous → Coarse and Coarse → Continuous curves in **Figure 8-14**.

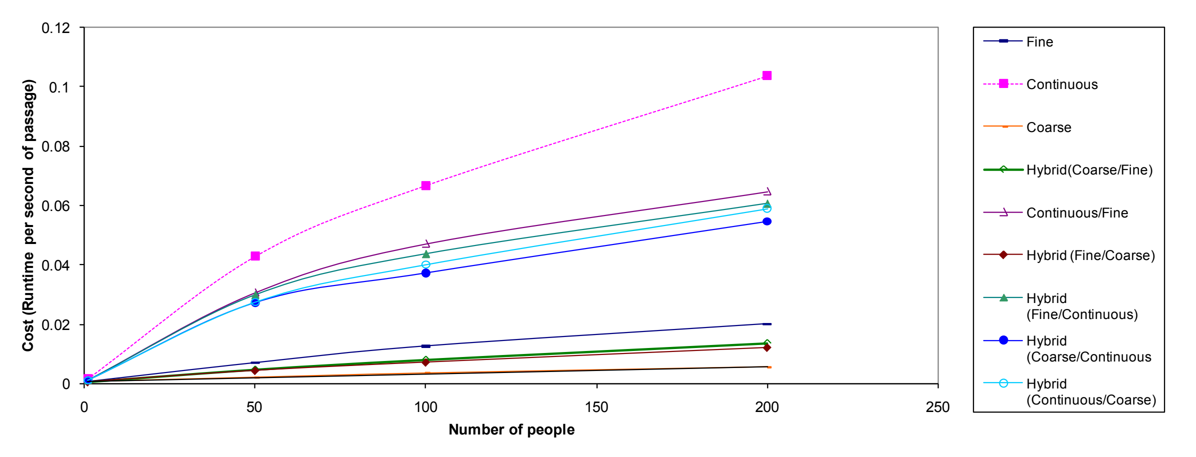


Figure 8-14 Variation of cost with increasing number of agents

8.4.1 Run Time Estimation – Tunnel Station Case

In order to demonstrate an example of cost estimation for a large geometry modelled using all the spatial types and involving multiple occupants, the methodology described in Section 8.4 was used to determine the run time of the Large Tunnel Station case simulations (Section 8.3). The run time estimate is then compared against actual run time values. This comparison analysis aims at providing a more consolidated understanding on the factors which impact the numerical efficiency of the HSD approach. The hybrid configuration of the tunnel station geometry was modelled using a combination of all the spatial representation types (Section 8.3). From **Figure 8-14**, the linear equations pertaining to the variation of the computational cost of the Coarse, Fine and Continuous models with increasing number of agents was obtained. The equations are as follows:

$$C_{coarse} = 0.00003n + 0.0006$$ **Equation 8-3**

$$C_{fine} = 0.0001n + 0.0015$$ **Equation 8-4**

$$C_{continuous} = 0.0005n + 0.0104$$ **Equation 8-5**

Where:

C_{coarse}	: Cost of moving agents per second of passage in coarse network
C_{fine}	: Cost of moving agents per second of passage in fine network
$C_{continuous}$	: Cost of moving agents per second of passage in continuous space
n	: Number of agents in spatial type

An estimate of the run time of the hybrid configuration can be obtained using the sum of the run times of simulation in the coarse, fine and continuous segments as follows:

$$R_{hybrid} = R_{coarse} + R_{fine} + R_{continuous}$$

Using the definition of R from Equation 8-1,

$$R_{hybrid} = C_{coarse}\,T_{coarse} + C_{fine}\,T_{fine} + C_{continuous}\,T_{continuous} \qquad \textbf{Equation 8-6}$$

Where:

R_{hybrid}	: Run time of entire simulation in hybrid configuration
$R_{continuous}$	: Run time of simulation in the continuous segment of the geometry
R_{fine}	: Run time of simulation in the fine node segment of the geometry
R_{coarse}	: Run time of simulation in the coarse node segment of the geometry
$C_{continuous}$	: Cost of moving agents per second of passage in continuous model
C_{fine}	: Cost of moving agents per second of passage in fine node model
C_{coarse}	: Cost of moving agents per second of passage in coarse node model
$T_{continuous}$	: Time of passage of agents in continuous space
T_{fine}	: Time of passage of agents in fine node network
T_{coarse}	: Time of passage of agents in coarse node network

Substituting C_{coarse}, C_{fine} and $C_{continuous}$ in Equation 8-6 with the right hand side of the terms in equations **Equation 8-3**, **Equation 8-4** and Equation 8-5 respectively results in a cost equation which considers the number of agents passing through the different spatial types. In the hybrid configuration of the Tunnel Station case, there are 2000 agents passing through each spatial type.

$$R_{hybrid} = [(0.00003N_{coarse} + 0.0006) \text{ x } T_{coarse}\ ] + [(0.0001N_{fine} + 0.0015) \text{ x } T_{fine}\ ] + [(0.0005N_{continuous} + 0.0104) \text{ x } T_{continuous}\ ]$$

Equation 8-7

Where:

N_{coarse} : Number of agents passing in coarse network

N_{fine} : Number of agents passing in fine network

$N_{continuous}$: Number of agents passing in continuous regions

The time of passage of the agents in a spatial type (e.g. coarse network) is calculated as follows:

$$T_{coarse} = \frac{D_{coarse}}{D_{total}} \times T_{total}$$

Equation 8-8

Where:

T_{coarse} : Time of passage of agents in coarse network

T_{total} : Total time of passage of agents in entire geometry

D_{coarse} : Distance travelled in coarse network

D_{total} : Total distance travelled in entire geometry

The calculation of the time of passage is based on the following assumptions:

- The total distance covered by all the agents is equal to the sum of the longest paths in the various segments of the geometry. For instance, the total distance in the Tunnel Station case is calculated as the sum of the lengths of the various segments which is equal to 778m (Figure 8-9).

- The total time of passage of all the agents is obtained by dividing the total distance travelled with the average travel speed of the agents (assumed to be 1.35 m/s). This travel speed is the average of the minimum (1.2 m/s) and maximum (1.5 m/s) travel speed values used for the agent population.

- All the agents passing over a segment (represented by a particular spatial type) cover the entire distance given by the length of segment. For instance, in the Tunnel Station case, the agents are generated in a section of the tunnel of length 700 m which is represented by a coarse node. It is assumed that all the agents will cover the entire 700 m, although in principle, many agents (located far from the end of the tunnel), will be covering a smaller distance.

Using **Equation 8-7**, the run time estimates for the simulations in the Hybrid and all-Fine configurations are calculated and compared against the actual run times, as shown in Table 8-8.

Table 8-8 Comparing actual and estimated run times

Spatial Representation	Actual Run Time (s)	Estimated Run Time (s)
Hybrid	42.0	53.8
All-Fine	80.6	116.1

The results in Table 8-8 show that the methodology used for the run time estimations are consistent with the actual run times in the sense that the Hybrid configuration demonstrates a faster run time estimate than the All-Fine configuration. However, the methodology used provides an over-estimation of the run times in both the Hybrid and all-Fine cases. Differences between the actual run times and estimated run times for each spatial representation type are to be expected, especially considering the assumptions upon which the calculations have been based. **Equation 8-7** shows that the time of passage of agents over the coarse nodes, fine nodes and continuous regions has a significant impact on the run time calculation. Therefore, any under-estimation or over-estimation of the times of passage of the agents will be reflected in the final run time calculation.

The investigation conducted in this section have demonstrated that the run time of the hybrid model does not simply scale in proportion to the floor area represented by the various discretisations, but is in fact dependent on the time of passage of agents over the discretisations.

8.5 Summary

This chapter described the verification of the HSD approach through multiple test cases. The test cases have been particularly chosen with a view to assess and demonstrate the various capabilities of this novel approach. The first test case consisted of a complex multi-compartment geometry used to compare the results of the HSD approach against the All-Fine, All-Coarse and All-Continuous configurations. In the light of the results, it was found that the HSD approach showed results of similar accuracies to the other modelling approaches. Also, since the multiple compartment geometry was modelled in such a way that it represented all the possible transition types (Coarse Node ↔ Fine Node, Fine Node ↔ Continuous and Coarse Node ↔ Continuous), this test case was appropriate to validate the behavioural and movement algorithms used in the transition regions. In addition, the second test case comprised of an extremely large and complex city block environment, with the aim of demonstrating the capability of the Hybrid model to handle extremely large environments. The second test case results showed that the Hybrid configuration comprising of Coarse and Fine nodes provides a substantial speed up of 33.3% over the All-Fine model. The third test case involved the application of the HSD on a large tunnel station complex using all the spatial representation types. The results of this test case showed that the HSD approach provided a considerable speed up of 47.9% over the All-Fine model. These test cases have shown that the HSD approach is able to encompass the benefits of the three spatial representation methods and maximise computational efficiency while providing an optimal environment to represent the movement and interaction of agents. In addition, validation is an ongoing and integral part of the software development lifecycle of a computational model, and therefore further validation studies will improve the confidence in the HSD approach. This chapter has also provided a methodology which allows the run times of the various spatial types as well as their combinations to be estimated. Such estimates provide an indication of the numerical efficiencies which can be achieved by the various elements of the HSD approach.

Chapter 9 – Conclusion

This chapter evaluates the different aspects of this research and summarises the achievements of this thesis in the field of evacuation modelling.

The main result of this research is a novel approach, known as the Hybrid Spatial Discretisation or HSD to represent the discretisation of space in circulation and evacuation simulation models. The HSD approach, implemented within the buildingEXODUS software allows enclosures to be modelled using a mixture of the three techniques for space discretisation namely coarse networks, fine networks and continuous regions. The aim of the HSD approach is to encompass the benefits of the three spatial representation methods and maximise computational efficiency while providing an optimal environment to represent the movement and interaction of agents. The present study has addressed the research questions raised in Chapter 1 which were as follows:

Can we derive a new approach that has the benefits and accuracy of the continuous model but without the computational overheads?

- *How can we represent agent navigation in continuous space?*

This research has led to development of a Continuous Model (Chapter 5) which is capable of simulating human behaviour and movement for egress analyses. The Continuous Model utilises a novel concept of a Navigational Graph which is comprised of a network of waypoints and path segments. The waypoints in the graph act as sub-goals guiding the agents from their starting locations to their destinations. An agent navigating in the continuous space can have up to three goal locations namely short term goal, medium term goal and long term goal. The long term goal represents target locations such as external exits, the medium term goal controls an agent's route to the long term goal such as internal exit and the short term goal is used for navigating around obstacles within enclosures.

- *How can we represent agent perception and behaviours in continuous space?*

As described in Chapter 5, the evacuees are modelled from an individual perspective in the form of autonomous agents which demonstrate some forms of adaptive behaviour. The agents are equipped with a vision system coupled with a series of sensors which allows them to sense and perceive changes in their immediate physical and social environment, make decisions and act accordingly. The following key advanced features were developed:

  - Crowd avoidance behaviours allowing occupants to maintain a desired separation from each other and enabling them to avoid collisions with occupants moving in unaligned directions.
  - Behavioural anticipation – the ability to look ahead by a constant time. The distance projected ahead is proportional to travel speed.
  - The agents possess some abilities which allow them to be robust and to recover from failure. For example, the agents can re-navigate if they have lost sight of their goals, as a result of being pushed by other neighbouring agents.
  - The Continuous Model can demonstrate behaviours such as queuing and competitive behaviours as well as demonstrating emergence of crowd phenomena such as lane formation.

- *Can we derive a framework to allow dynamic integration of behaviours into the system?*

A very important consideration in the design of the behavioural system of the agents was its ability to accommodate new behaviours dynamically. This was achieved by using a component oriented approach. In other words, the behaviours were modelled as simple components which can serve as building blocks for simulating more complex behavioural routines as described in Chapter 5. This approach allowed behaviours to be implemented and tested individually, followed by an incremental integration.

- *Can high densities of agents in continuous spaces be simulated without occurrence of deadlocks?*

The nature of the continuous spatial representation makes it suitable for demonstrating close packing of agents. However, such models can be prone to locking where agents are unable to move, especially in crowded situations. This is because the collision detection and response amongst large numbers of interacting agents can often lead to the agents not being able to move past each other. However, as described in Chapter 5, there has been no occurrence of deadlocks in all the scenarios in which Continuous Model has been applied. This is due to the combination of the Seek and Agent Avoidance behaviours which allow the agents to move past each other even at very high population densities. This makes the model appropriate for high density crowd simulation involving uni-directional flows as well as contra-flows.

- *What optimisation strategies can we use for improving the numerical performance of continuous models?*

There are three key strategies which have been used for optimising the numerical performance of the Continuous Model (Chapter 5). These are summarised as follows:

  - Navigational Graph – This graph is automatically generated in the pre-processing phase. The innovative characteristic of this graph is that the waypoints are generated only at locations where the internal angles are concave. This approach greatly simplifies the Navigational Graph associated with the continuous regions thus improving the efficiency of the route finding algorithms of the agents. Also, the reason for placing the waypoints only at concaved corners is that they are sufficient to direct the agents away from the location of the walls of the enclosure and around corners. The use of the sub-goal evaluation algorithms in conjunction with the Navigational Graph is an improvement over other existing approaches such as the visibility graph.

  - Grid Method – The Continuous Model utilises a spatial hashing strategy, called the Grid Method for optimising the spatial search of agents in the continuous environment. The Grid Method makes use of an underlying grid of cells which is automatically generated for each continuous region during the pre-processing phase. The use of this method brings the computation complexity from $O(N^2)$ down to $O(NM)$, whereby N is the total number of agents in the simulation, and M is the total number of agents within an agent's current cell and neighbouring cells of the underlying grid.

- Wall Search optimisation - The Continuous Model incorporates a Wall Search optimisation mechanism which reuses the cells generated by the Grid Method. This mechanism operates by assigning the wall boundaries to the underlying cells during the pre-processing phase. In this respect, the agents need not search the entire continuous region for the location of wall boundaries, but instead limit their search only to the wall boundaries assigned to their current and neighbouring cells.

- *How can we represent agent navigation in a coarse network model?*

This research has lead to the development of a model capable of running simulations using a coarse spatial representation as described in Chapter 6. Using this approach, the available physical space can be segmented into partitions, such that each partition can represent a section of the geometry such as a room. However, the nature of this approach restricts the modelling of a region with simple shapes for instance rectangles, which might not be sufficient to cater for the complexities of enclosures. The Coarse Node model in bEX-H is capable of representing non-convex regions unlike other coarse node models.

This model uses a flow to density equation in conjunction with the novel concept of a navigational grid, which allows the representation of non-convex coarse nodes. The underlying navigational grid is based on the same principle as the Navigational Graph used in the Continuous Model. The waypoints and arcs within the mesh provide the information required to model the flow of occupants in the coarse nodes.

- *How can we harness the benefits of the continuous and nodal approaches?*

All evacuation and pedestrian dynamics models typically use one of three basic approaches to represent physical space namely; a continuous representation of space, a fine network of nodes or a coarse network of nodes. Therefore, this would limit the modeller only to the features and advantages offered by the particular spatial representation approach employed in the model. However, by integrating the three spatial representation approaches and interoperating them within the same software environment, it is possible to utilise the advantages offered by each approach. Chapter 7 describes the design and implementation of the Hybrid Spatial Discretisation (HSD) approach, which encompasses the benefits of each spatial type within a single integrated computational model, bEX-Hybrid.

- *What benchmark tests are available for validating evacuation models designed for the building environment?*

A key objective of this research entailed the simulation of realistic agent behaviours navigating in continuous space. Therefore, a lot of emphasis was placed in the need for validating and calibrating the Continuous and HSD models. However, there appeared to be a lack of a standard for the purposes of model validation and calibration within the building evacuation field community. There was a lack of standard measures and procedures that could be used towards validating even the basic underlying principles of the simulation model e.g. travel speeds. The International Maritime Organisation (IMO) [IMO 2002] had devised a series of guidelines for performance based evacuation analyses for new and existing passenger ships. Given that the test cases presented as part of these guidelines were designed for flat surfaces, they were equally applicable to the validation of evacuation tools for the building environment as well. These test cases also catered for some essential elements of the validation process: Component testing, Functional validation and Qualitative validation. They were thus found to be suitable for validating the Continuous and HSD models. In the light of the validation tests that were carried out, it can be rationally stated that the IMO test cases provide an appropriate framework for validating some essential aspects of evacuation models. However, as stated in the IMO guidelines, those test cases cannot cater for the quantitative validation of egress tools.

Can a hybrid representation which utilises the 3 fundamental spatial representation methods be used in such a way as to optimise accuracy and performance?

- *What type of software architecture is appropriate for a hybrid model?*

The bEX-H makes use of the core architecture of the buildingEXODUS software. However, because the HSD approach integrates three approaches which have different underlying mechanics, it was crucial to choose an architecture which would allow the spatial representations to be used and tested individually as well as when combined. In this respect, the buildingEXODUS software was modified to allow plug-in modules to be included into the core software using a component oriented engineering approach. This architecture provides a platform whereby new functionalities can be independently developed and incorporated into the model as required. The Coarse Network and Continuous Region are examples of plug-in modules developed for the bEX-H model.

- *How can the transition of agents be represented in a hybrid model comprising of coarse networks, fine node networks and continuous regions?*

The HSD approach involved the mixing of macroscopic (coarse node) and microscopic (continuous and fine) modelling methodologies which in itself presented several challenges. In the continuous and fine node models, the agents are modelled from an individual perspective whereby their movements, exact locations and behaviours can be tracked. However, in the coarse network approach, the agents are treated as a homogenous ensemble, such that the locations and physical space occupancy of the agents are not represented. The connectivity in between various spatial types is achieved via arcs such that the combination of these interconnecting arcs forms transition regions. As described in Chapter 7, bEX-H incorporates some behavioural and movement mechanisms to facilitate the transition of agents across all the six possible transition regions namely: Coarse Node ↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous.

- *Can agents maintain their physical and experiential attributes when traversing the different spatial representation types?*

When the agents transit from one spatial representation type to another, the agent attributes (e.g. age, gender, body width) are maintained while the experiential attributes (e.g. distance travelled, cumulative wait time) are updated accordingly during the simulation.

Moreover, although the graphical output of coarse nodes in bEX-H only illustrates the total number of agents within the node at any point during the simulation, the underlying algorithmic procedures keep track of the individual identities of each agent. Furthermore, the locations of the agents along the arcs of the coarse node's navigational mesh are also recorded. Using these methodologies, the HSD model is able to keep track of the agents which have reached their exit points within the coarse node, and re-position those agents into the adjacent segment of physical space represented by the fine network or continuous region.

- *How do we map the different spatial representation types to different areas of an enclosure in a hybrid model?*

The geometrical construction tools in the bEX-Hybrid allow the flexibility of manually specifying the hybrid discretisation schemes for enclosures. However, the choice of a suitable hybrid discretisation scheme is essential in order to optimise computational efficiency and accuracy. In the HSD model, the continuous modelling approach is highest on the scale of granularity. The Continuous Model consists of complex behavioural and movement mechanisms which are triggered at every $(1/12)^{th}$ of a second. This enables the Continuous Model to provide highly detailed representations of the crowd phenomena. Consequently, this model incurs the highest computational overheads as compared to the other spatial types. On the other hand, the coarse node approach requires the least amount of computational resources. Therefore, by utilising continuous regions only in areas which are central to the analysis and using coarse nodes for representing areas in the far field and fine nodes for the remainder bulk of the enclosure, the use of available computational resources is optimised.

How suitable are each of the spatial representation approaches for modelling different terrain types and scenarios?

When using the HSD approach, the choice of modelling different segments of an enclosure using the three spatial types is user-based thereby giving modellers the flexibility in mixing the three techniques. However, since each of the spatial representation approaches has its relative benefits and limitations, the use of different discretisation schemes could have an impact on the accuracy of model predictions. In this respect, this research has lead to the investigation of the potential benefits and limitations of each spatial representation technique. This investigation was instrumental in identifying suitability of using the coarse network, fine network or continuous approach for modelling different types of scenarios and terrains. As a result, this study has generated a list of recommendations of where the various spatial representations are appropriate for use. These guidelines have been presented in Chapter 2.

What are the benefits of a hybrid approach in terms of size, accuracy and performance of the problem that can be tackled?

- *What are the advantages of the hybrid approach over the other spatial representation approaches?*

The hybrid approach presents several benefits over the other spatial representation approaches namely:

  - It has the capability of solving problems which were not possible before due to their size and complexity. Such problems include urban environments such as town and cities. Solving problems on such an extremely large scale with fine node models or continuous regions is not computationally feasible due to the extremely large overheads that would be incurred during the people-people, people-structure and people-environment interactions. Moreover, solving such problems solely using the coarse nodes would lead to loss of accuracy due to the inability of this approach to represent the nature of people-people and people-environment interactions throughout the entire enclosure.

  - It has the capability of simulating egress on large complex buildings with improved accuracy but without large overheads on performance. Utilising an all-Continuous configuration for representing such geometries is computationally least efficient whereas an all-Fine configuration will not capture the evacuation dynamics at the same level of detail as the all-Continuous. However, a hybrid configuration involving a combination of the Fine Node approach and Continuous Regions provides improved accuracy over the all-Fine approach whilst providing higher computational performance than the all-Continuous approach.

  - It has the capability of solving moderately large problems with reasonable accuracy and performance. A hybrid configuration involving all three spatial representations will provide improved accuracy over an all-Coarse approach but will not be as computationally fast. Moreover, such hybrid configuration enables the accuracy advantages of the Continuous approach to be harnessed but without incurring the large computational overheads of the all-Continuous model. Furthermore, the hybrid configuration allows a more accurate representation of the evacuation dynamics than the all-Fine model while providing approximately same performance or even improved performance over the all-Fine model.

- *How would such an approach be utilised to address for:*

  a) *Very large complex scale problems such as urban environment* – This has been demonstrated in the application of the HSD approach to an extremely large complex urban environment comprising of city blocks (see Chapter 8 Section 8.2). In this geometry, the areas in the far field are represented by a network of coarse nodes, which constitute 71% of the entire free space area. All the coarse nodes converge into a middle artery road which leads to the assembly area. The middle artery and the assembly area are modelled entirely using a network of fine nodes. A population involving 10 000 agents was distributed in the coarse nodal regions as well as in the middle artery. In this configuration, the Hybrid model is seen to provide a substantial speed up of

33.3% over the all-Fine approach whilst maintaining the accuracy of the all-Fine. This depicts the applicability of the HSD towards solving extremely large problems which were not feasible before.

b) *Moderately large complex structure such as tunnel station* – This has been demonstrated in the application of the HSD approach to a moderately large station tunnel complex (see Chapter 8 Section 8.3). In this geometry, a long tunnel of dimensions 700m x 2m is modelled using the Coarse Mesh approach (73% of total area). The entrance of the tunnel into the platforms and the regions surrounding the internal exits and external exits represented pinch points and were therefore modelled using Continuous Regions (4% of total area). The remainder of the structure was modelled using the Fine Mesh approach (23% of total area). The population comprised of 2000 agents and was generated in a segment of the tunnel. In this configuration, the HSD approach is seen to provide a very promising speed up of 47.9% over the all-Fine approach whilst providing improved accuracy.

c) *Complex buildings with multiple compartments* – This has been demonstrated in the application of the HSD approach to a large complex geometry comprising of 17 irregular shaped compartments and a population of 300 agents spread throughout the entire building (see Chapter 8 Section 8.1). The HSD approach was applied using two modes namely Hybrid-1 which comprised of all three spatial representation types and Hybrid-2 which consisted only of fine nodes and continuous regions. The discretisation scheme in Hybrid-1 was chosen in such a way that it allowed the representation of all six possible transition regions namely; Coarse Node ↔ Fine Node; Fine Node↔ Continuous and Coarse Node ↔ Continuous. The Hybrid-1 configuration was seen to produce similar evacuation dynamics as compared to the all-Continuous and all-Fine approaches which illustrated that the algorithmic procedures used in the implementation of the various transition regions were operating appropriately. Similarly, the Hybrid-2 configuration was seen to produce similar evacuation dynamics to the other approaches. Moreover, the Hybrid-2 was seen to produce results which closely resembled the all-Continuous approach whilst providing a significant speed up of 42.4% over the all-Continuous model.

Chapter 10 - Further Work

This chapter highlights some potential areas for further investigation towards this research work. The outcome of these further explorations could be used as input in the development of additional algorithmic procedures for enhancing the behavioural and movement models in bEX-H.

10.1 Enhancing the perception capabilities of the agents

Another area for future research may be the enhancement of the perception of the agents within the Continuous and Coarse Models of the bEX-H software so that they could detect other essential cues within their environment, for instance signage, and respond accordingly. The fine network version of buildingEXODUS incorporates a method of incorporating the visibility of particular objects through the application of a concept called Visibility Catchment Area (VCA) [Xie 2005]. This feature could be extended to the Continuous and Coarse Network Models. The improvement of the agents' sensing ability would imply that the agents will possess the capabilities of demonstrating more complex human behaviours within the Hybrid Spatial Discretisation (HSD) model.

10.2 Optimising computational performance of the Continuous and HSD models

Both the Continuous and HSD models rely on the operation of a multitude of complex algorithms for instance, spatial searches (goals, neighbouring agents, and obstacles) within the continuous space, agent behavioural routines and the transition of agents across different spatial types. Although the development of the continuous model entailed the use of optimisation strategies such as domain decomposition (using multiple continuous regions) and the Grid Method, there are yet certain aspects of the code which could be further optimised. For instance, further work could be done on improving the time complexities of the various algorithms coupled with the identification of the best-case, average-case and worst-case performances. Moreover, the system architecture of the bEX-H software is as such that it entails a lot of parameter passing from the core software to the plug-ins and vice versa which could lead to performance overheads, especially for large geometries comprising of large population sizes. In this respect, the Interface Module of the bEX-H system architecture could be optimised by minimising redundant copying of parameters between the plug-ins and the core system. These enhancements would allow the numerical efficiencies offered by using a mix of discretisation schemes using the HSD model to be fully exploited.

10.3 Incorporating objects in the Continuous Model and Hybrid Model in bEX-H

In more recent studies, it has been found that a considerable proportion of people within crowds tend to carry their personal belongings during evacuation. For example, in the pre-evacuation data analysis of the World Trade Centre, 11% of the actions reported during pre-movement time involved occupants collecting belongings (e.g. bags, laptops, clothing) and these actions were carried out by 26.5% of surviving population [Galea 2004b]. Moreover, hospital evacuation procedures must address the needs of non-ambulant patients for instance by using stretchers and wheelchairs. In this respect, the incorporation of a facility to represent objects in circulation and evacuation models is essential as it would enable the investigation of the effects of such objects on crowd phenomena.

The current implementation of the Continuous Model within bEX-H includes a basic Objects Model in which individual agents can be assigned objects that they can push. In this model, an object can be defined as a collection of attributes. Although an object can have a multitude of attributes, only the ones which could have an impact on the locomotion and behaviour of the agents are identified and modelled. The chosen attributes are mass (kg), length (cm), width (cm) and height (cm). While these attributes would be sufficient to model objects having relatively simple shapes, however they might not be adequate to represent objects having spherical shapes or made up of non-convex polygons. In effect, rather than the actual object, the smallest bounding box enclosing the object is considered. For instance the bounding box for a spherical object would be as shown in **Figure 10-1**.

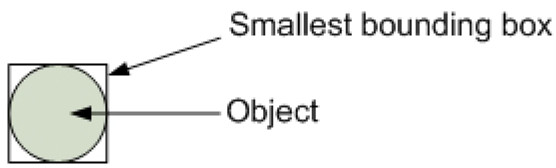


Figure 10-1 Small bounding box

The interaction of agents with objects is likely to affect some of the characteristics of the agents for instance travel speeds, turning rates and individual personal spaces. Moreover, the behaviour of the agents would vary depending on the type of interactions in which the agents are engaged, for instance, pushing, pulling or carrying objects. For instance, in an airport terminal, passengers queuing up with their handheld luggage and baggage trolleys prior to checking in, have to incorporate the space occupied by the object within their personal space requirements. Therefore, further experiments would be required in order to derive relationships between the different categories of objects and the resultant effect (e.g. travel speed, mobility, turning rate) on the agents. Moreover, the functionality allowing the interaction of the agents with objects could be extended to include agents pulling and carrying objects.

Furthermore, the interaction of occupants with objects could be extended to the HSD model. Due to the different levels of granularity in each of the spatial representation types, the objects would be represented using different approaches. For instance, in the continuous space, the objects would be represented by a bounding box which most closely represents the physical space occupied as shown in **Figure 10-1**. In the coarse node model, interaction with objects could be simulated by adjusting the travel speeds and mobility of the occupants. In the fine node model, the physical characteristics of the objects could be represented using multiple unoccupied fine nodes. The incorporation of these new features should be facilitated due the modularity of the bEX-H software.

10.4 Adaptive Behaviour for different environments

The agents within the bEX-H Continuous Model have been designed as autonomous characters which have the ability to sense changes in their environment and act accordingly. The behavioural and movement mechanisms developed in this dissertation have been applied primarily to the building environment. Further work could be carried out to adapt the current agent capabilities to operate in other environments such as marine, aircraft and railway. For marine environments, the agents could be adapted so that they are sensitive to the different movements of the ship. In the case of aircraft environments, the behavioural system could be enhanced to include more advanced route choice capabilities coupled with behaviours such as seat jumping [Galea 2002]. For railway rolling stock environments, the behavioural adaptation could include the ability of the agents to navigate in overturned railway carriages. The inclusion of the new capabilities should be facilitated due to the component oriented nature of the agent's perception and behavioural system. Moreover, other forms of behaviour such as group and leader-following behaviours could be incorporated into the model.

10.5 Incorporating Hazard and Toxicity in Continuous and HSD models

Further work could be carried out towards extending the existing Hazard and Toxicity sub-models in the fine-network version of buildingEXODUS [Galea 1994], to operate within the bEX-H Continuous and HSD models as well. The inclusion of these features would enable agents in the Continuous and HSD models to be sensitive to environment effects such as smoke, high temperature, toxic gases thereby enhancing the behavioural capabilities of the agents.

10.6 Enhancing the visualisation capabilities of the bEX-H software

The EXODUS suite of software includes a 3-D post-processing visualisation tool named vrEXODUS [Galea 2004a] which currently operates with the output generated from the buildingEXODUS fine network model. The bEX-H software and vrEXODUS could be adapted such that the navigation of the agents in the bEX-H Continuous and HSD models could be represented in 3-D. These new enhancements would allow the graphical representation of agent navigation irrespective of the underlying method of spatial representation. The current version of the bEX-H software includes a simple integrated 3-D visualisation tool which allows the 3-D representation of enclosures modelled using the mixed approach, as shown in **Figure 10-2**. This 3-D visualisation capability was used as a debugging tool for investigating the transition of agents to and from the coarse nodes. The features of this integrated visualisation tool could be ported to the vrEXODUS software.

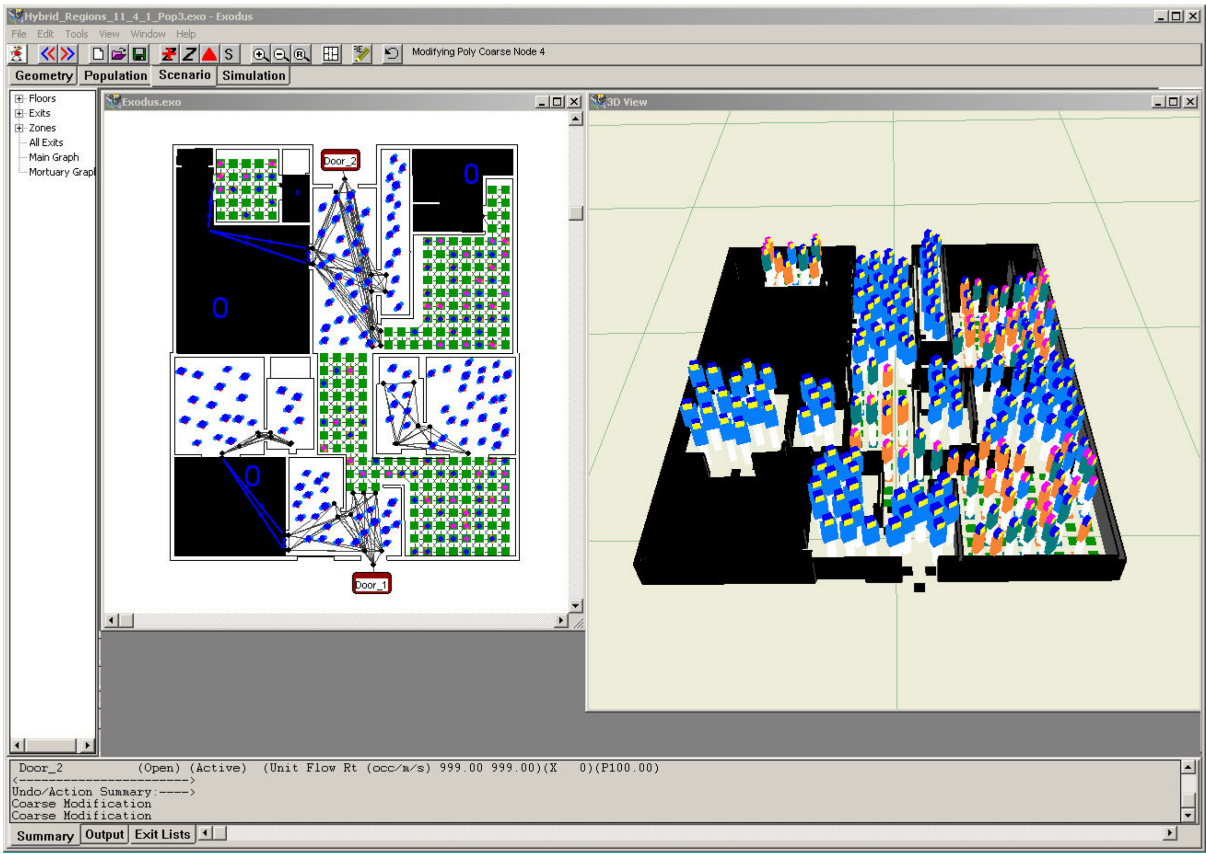


Figure 10-2 Two dimensional and three dimensional representation of a geometry modelled using a combination of fine network, coarse network and continuous regions in the BEX-H software.

10.7 Saving and loading geometries and populations in XML format

During the progression of this research, a simple XML parser was developed which allowed the serialisation and de-serialisation of the geometrical and population constructions in the bEX-H Continuous Model using XML. Using the XML language offers multiple benefits namely data reusability, flexibility, accessibility and portability [W3Schools 2010]. Further work could be conducted towards testing the XML handling capabilities of the bEX-H software.

10.8 Parallelising the HSD approach

The inherent features of the HSD approach allows the mixing of all three different spatial representation methods whilst maximising computational efficiency. However, there is scope for parallelising the HSD approach in order to achieve even higher computational performance. The HSD approach could be extended so that it works efficiently within the parallel computing environment currently offered by buildingEXODUS [Grandison 2007]. In the parallelised HSD approach, the various segments of the enclosures, each modelled using different spatial types, could be allocated to different processors. This approach would enable the computational load to be shared across the different processors during the processing phase of the simulations.

References

[Ando 1988] Ando, K., Ota, H., Oki, T., 1988. Forecasting the Flow of People. *RailwayResearch Review*, (45), 8-14.

[Antsaklis 1997] Antsaklis, P.J., Koutsoukos, X.D., 1997. On Hybrid Control of Complex Systems: A Survey. Technical Report of the Interdisciplinary Studies of Intelligent Systems (ISIS), ISIS-97-017, University of Notre Dame.

[Arup 2001] ARUP, 2001. 2nd Further Note on PEDROUTE, 2-4-04 Draft C

[Bateman 2002] Bateman, J.M., Edwards, B., 2002. Gender and Evacuation: A Closer Look Why Women are more likely to Evacuate for Hurricanes. *Natural Hazards Review,* 3(3), 107-117.

[Best 1977] Best, R.L., 1977. Reconstruction of a Tragedy: The Beverly Hills Supper Club Fire, Southgate, Kentucky, May 28, NFPA

[Berrou 2005] Berrou, J.L., Beecham, J., Quaglia, P,. Kagarlis, M.A., Gerodimos, A., 2005. Calibration and validation of the Legion simulation model using empirical data. *In: Pedestrian and Evacuation Dynamics 2005*, 27-30 September 2005 Vienna. Springer, 167-181.

[Borrmann 2010] Borrmann,A., Kneidl, A., Koster, G., Ruzika, S., Thiemann, M., 2010. Bidirectional Coupling of Macroscopic and Microscopic Pedestrian Evacuation Models. *In: 5th International Conference on Pedestrian and Evacuation Dynamics,* 8-10 March 2010 Gaithersburg, USA.

[Boxall 2005] Boxall, P., Gwynne, S., Filippidis, L., Galea, E.R., Cooney, D., 2005. Advanced Evacuation Simulation Software and its use in Warships. *In: RINA International Conference on Human Factors in Ship Design, Safety and Operation,* Feb 2005.

[Bre 2008] BRE Fire Research and Consultancy. *Human behaviour in fire and emergency evacuation design.* [online]. Available from: http://www.bre.co.uk/fire/page.jsp?id=269
[Accessed 6 May 2008]

[Bryan 1996] Bryan, J., 1996. Behavioural Response To Fire And Smoke. Dilenno, P.J., Beyer, C.L.,Custer, R.L.P.,Walton, W.D.,Watts, J.M.W.,Drysdale, D., Hall, J.R, ed. *The SFPE Handbook Of Fire Protection Engineering, 2nd ed.,* 1-241-1-262. National Fire Protection Association, Quincy, MA.

[Buckmann 1994] Buckmann, L.T. and Leather, J., 1994. Modelling station congestion the PEDROUTE way. *Traffic Engineering and Control*, 35, 373-377.

[Castle 2004] Castle, J.E., 2004. Agent Based Modelling of Pedestrian Evacuation: A study of King's Cross Underground Station. *Association For Computing Machinery(ACM).*

[Cisek 2010] Cisek, M., Kapalka, M., 2010. The use of fine – coarse network model for simulating building evacuation with information system. *In: 5th International Conference on Pedestrian and Evacuation Dynamics,* 8-10 March 2010 Gaithersburg, USA.

[Cunningham 1993] Cunningham, P.P., Cullen, D.A., 1993. Pedestrian flow data collection and analysis. *Proceedings Institution of Civil Engineers*, 59-69

[Dash 2007] Dash,N., Gladwin,H., 2007. Evacuation Decision Making and Behavioural Responses: Individual and Household. *Natural Hazards Review,* 8(3), 69-77.

[Dearden] Dearden, J. Pedestrian Evacuation Model in Second Life, Technical Details. Centre for Advanced Spatial Analysis, University College London.

[Deitel 2005] Deitel, H.M. and Deitel, P.J., 2005. *C++ How To Program*. 5th ed. New Jersey: Prentice Hall.

[DFT 2010] Department for Transport, *Inclusive Mobility* [online] Available from: http://www.dft.gov.uk/transportforyou/access/peti/inclusivemobility?page=3 [Accessed 29 November 2010].

[Doheny 1994] Doheny, J., Fraser, J., 1994. Modelling Decisions in Emergency Situations. *In: Proceedings of the European Petroleum Computer Conference*, 15-17 March, 1994, Aberdeen.

[Doheny 1995] Doheny, J., Fraser, J., 1995. MOBEDIC – A Decision Modelling Tool for Emergency Situations. *Expert Systems With Applications*, 10 (1), 17-21.

[Doyle 1995] Doyle, A. B., 1995. *Algorithms and Computational Techniques for Robot Path Planning.* Thesis(PhD). University of Wales, UK.

[EVI 2001] Vassalos, D., Kim, H., Christiansen, G., Majumder, J. 2001. A Mesoscopic Model for passenger evacuation in a virtual ship-sea environment and performance-based evaluation. *In: Pedestrian and Evacuation Dynamics* 4-6 April, 2001, Duisburg.

[EVI 2008a] EVI. *Modelling the ship evacuation process*, *technical information* [online]. Available from: http://www.safety-at-sea.co.uk/evi/mesoscopicmodelling.htm [Accessed 19 January 2008]

[EVI 2008b] EVI. *Modelling the ship evacuation process, features* [online]. Available from: http://www.safety-at-sea.co.uk/evi/features.htm [Accessed 19 January 2008]

[Fang 2003] Fang, Z., Lo, S.M., Lu, J.A., 2003. On the Relationship between Crowd Density and Movement Velocity. *Fire Safety Journal,* 38, 271-283.

[Feurtey 2000] Feurtey, F., 2000. *Simulating the Collision Avoidance Behaviour of Pedestrians*. Thesis (Masters). The University of Tokyo, School of Engineering, Department of Electronic Engineering.

[Finlay 1996] Finlay, J., 1996. An introduction to artificial intelligence .UCL Press, London, ISBN No 1857283996.

[Fiorini 1998] Fiorini, P., Shiller, Z. 1998. Motion Planning in Dynamic Environments Using Velocity Obstacles. *Int. Journal of Robotics Research,* 17(7), 760–772.

[Foliente 2000] Foliente, G.C., 2000. Developments in Performance-Based Building Codes and Standards. *Forest Products Journal.* 50 (7/8) 13-19.

[Fraser-Mitchell 1994] Fraser-Mitchell,J.N., 1994. An Object-Oriented Simulation (CRISPII) for Fire Risk Assessment. *In: Proceedings of the Fourth International Symposium of Fire Safety Science*, 793-804.

[Friedman 1992] Friedman, R., 1992. *Combustion Science and Engineering Inc. International Survey of Computer Models for fire and smoke* [online] Available from: http://www.firemodelsurvey.com/EgressModels.html [Accessed 20 January 2008].

[Fruin 1971] Fruin, J.J., 1971. *Pedestrian planning and design*. Metropolitan Association of Urban Designers and Environmental Planners Inc, New York.

[FunctionX 2006] *Microsoft Foundation Classes* [online] FunctionX, Inc. Available from: http://functionx.com/visualc/index.htm. [Accessed 12 February 2006]

[Galea 1994] Galea E.R., Galparsoro J.M.P., 1994. A Computer Based Simulation Model for the Prediction of Evacuation from Mass Transport Vehicles. *Fire Safety Journal,* 22, 341 - 366.

[Galea 1997a] Galea, E.R., 1997. Validation of Evacuation Models. *In: CMS Press*, Paper No 97/IM/21, ISBN No 1 899991 22 0.

[Galea 1997b] Galea, E.R. , Gwynne, S., Lawrence, P.J., Owen, M., 1997. *Escape as a social response*. Fire Safety Engineering Group, University of Greenwich ,UK: CMS Press, (97/IM/26).

[Galea 1997c] Galea, E.R., Owen, M., Lawrence, P., Filippidis, L., 1997. The airEXODUS Evacuation Model and its Application to Evacuation Certification, Crew Training and Accident Investigation. *In: Proceedings of the Aircraft Cabin Safety Conference*, 20-21 March 1997, pp. 15.1-15.14. Published by the Royal Aeronautical Society. ISBN 1 85768 038 3.

[Galea 2000] Galea, E.R., Gwynne, S., Lawrence, P.J., Filippidis, L., Blackshields, D,. 2000 *.buildingEXODUS V3.0 User Guide and Technical Manual*. Fire Safety Engineering Group, University of Greenwich ,UK.

[Galea 2001] Galea, E. R., 2001. Simulating evacuation and circulation in planes, trains, buildings and ships using the EXODUS software. *In: Proceedings of Conference on Pedestrian and Evacuation Dynamics*, 203–226.

[Galea 2002] Galea E.R., Blake S.J., Lawrence P.J., and Gwynne S., 2002. The airEXODUS Evacuation Model and its Application to Aircraft Safety. *In: Proceedings Fire Safety and Survivability, Applied Vehicle Technology (AVT) Panel Symp,* NATO Research and Technology Organisation, Aalborg Denmark, 23-26 Sept 2002.

[Galea 2003] Gwynne S., Galea E.R., Lyster C., Glen I., 2003. Analysing the Evacuation Procedures Employed on a Thames Passenger Boat using the maritimeEXODUS Evacuation Model. *Fire Technology,* 39(3) 225-246.

[Galea 2004a] Galea, E.R. , Gwynne, S., Lawrence, P.J., Filippidis, L., Blackshields, D,. 2004 .b*uildingEXODUS V4.0 User Guide and Technical Manual*. Fire Safety Engineering Group, University of Greenwich, UK.

[Galea 2004b] Galea, E.R. , Gwynne, S., 2004. Principles and Practice of Evacuation Modelling 6th edition. *CMS Press,* ISBN No. 1904521185

[Galea 2004c] Galea, E.R., Lawrence, P., Gwynne, S., Sharp, G., Hurst, N., Wang, Z., Ewer, J., 2004. Integrated fire and evacuation in maritime environments. *In: Proceedings of the 2nd International Maritime Safety Conference on Design for Safety*, Sakai Japan, Publisher Ship and Ocean Foundation, 27-30 Oct 2004.

[Galea 2010] Galea, E.R., Adams, T., 2010. An Experimental Evaluation of Movement Devices used to assist People with Reduced Mobility in High-Rise Building Evacuations *In: 5th International Conference on Pedestrian and Evacuation Dynamics,* 8-10 March 2010 Gaithersburg, USA.

[Grandison 2004] Grandison A.J., Galea E.R., Patel M.K., Ewer J., 2004. The Development of Parallel Implementation for a CFD based fire field model utilising conventional office based PCs. *Journal of Applied Fire Science*, 12(2) 137-157.

[Grandison 2007] Grandison A; Muthu Y; Lawrence P; Galea E.R., 2007. Simulating the Evacuation of Very Large Populations in Large Domains using a Parallel Implementation of the buildingEXODUS Evacuation Model. *In: Proceedings of the 11th International Fire Science & Engineering Conference, Interflam* 2007, 3-5th September 2007, Royal Holloway College, University of London, UK, Volume 1, pp. 259-270. ISBN 978 0 9541216-8-6.

[Green] Green, R. Steering Behaviours. Bullfrog Productions Ltd.

[Guanquan 2006a] Guanquan, C., Jinhua, S., 2006. The Effect of Pre-movement Time and Occupant Density on Evacuation Time. *Journal of Fire Sciences,* 24, 237-259.

[Guanquan 2006b] Guanquan, C., Jinhua, S., Qingsong, W., Sining, C., 2006. Simulation Study on the Effect of Pre-evacuation Time and Exit Width on Evacuation. *Chinese Science Bulletin 2006,* 51(11), 1381-1388.

[Gwynne 1999] Gwynne, S., Galea, E.R., Lawrence P.J., Owen, M., and Filippidis, L., 1999. A Review of the Methodologies used in the Computer Simulation of Evacuation from the Built Environment. *Building and Environment* 34, 741-749.

[Gwynne 2000] Gwynne, S., 2000. *The Introduction of Adaptive Social Decision-Making in the Mathematical Modelling of Egress Behaviour.* Thesis (PhD). University of Greenwich.

[Gwynne 2008] Gwynne, S., Rosenbaum E.R., 2008. Employing the Hydraulic Model in Assessing Emergency Movement. P.J.DiNenno et al., ed. *The SFPE Handbook of Fire Protection Engineering,* 4th ed., 3-355-3-372, National Fire Protection Association, Quincy, MA.

[Gwynne 2009] Gwynne, S., Kuligowski., E.D., Kratchman, J., Milke, J.A., 2009. Questioning the Linear Relationship between Doorway Width and Achievable Flow Rate. *Fire Safety Journal,* 44, 80-87.

[Hakonen 2003] Hakonen, H., 2003. *Simulation of Building Traffic and Evacuation by Elevators*. Licentiate Thesis. Department of Engineering Physics and Mathematics, Helsinki University of Technology, Finland.

[Halcrow 1987] Halcrow, 1987. *Planning, Design and Management Services for Infrastructure Development Worldwide* [online]. Available from: http://www.halcrow.com/html/our_markets/fire/fire_evacuation.htm?zoom_highlight=PEDROUTE [Accessed 21 July 2009].

[Hall 1966] Hall., E., 1966. *The Hidden Dimentions*. Anchor Books. ISBN 0-385-

08476-5.

[Hankin 1958] Hankin, B.D., Wright, R.A., 1958. Passenger Flow in Subways. *Operational Research Quarterly,* 9, 81-88.

[Helbing 1992] Helbing, D., 1992. A Fluid-dynamic Model for the Movement of Pedestrians. *Complex Systems* , 6, 391-415.

[Helbing 1997] Helbing, D., 1997. *Self-organisation Phenomena in Pedestrian Crowds. Self-Organization of Complex Structures: From Individual to Collective Dynamics,* F. Schweitzer, ed. Gordon and Breach, London. 569-577.

[Helbing 2000] Helbing, D.,2000. Simulating Dynamical Features of Escape Panic. *Nature,* 407, 487-490.

[Helbing 2001] Helbing, D., Molnar, P., Farkas, I.J., Bolay, K,. 2001. Self-organising Pedestrian Movement. *Planning and Design 2001,* 28, 361-383.

[Hostikka 2007] Hostikka, S., Korhonen, T., Paloposki, T., Rinne, T., Matikainen, K., Heliovaara, S., 2007. Development and Validation of FDS+Evac for Evacuation Simulations. VTT Technical Research Centre of Finland.

[HMSO 2010] Her Majesty's Stationary Office (HMSO) [online]. Available from: http://www.opsi.gov.uk/ [Accessed 15 June 2010].

[IMO 2002] *Interim Guidelines for Evacuation Analyses for new and existing passenger ships. International Maritime Organisation*. MSC/Circ. 1033, 2002. Available from: http://www.imo.org/Home.asp?topic_id=1367 [Accessed 13 May 2010]

[Ketchell 1994] Ketchell, N., Cole, S.S., Webber, D.M., 1994. The Egress Code for Human Movement and Behaviour in Emergency Evacuation, Engineering for Crowd Safety, Smith,R.A. and Dickie,J.F.,Elsevier, ed. 361-370.

[Kisko 1985] Kisko,T.M. And Francis,R.L., 1985. Evacnet+: A Computer Program To Determine Optimal Evacuation Plans. *Fire Safety Journal*, 9, 211-220.

[Kisko 1998] Kisko, T.M., Francis, R.L., Nobel, C.R., 1998. *Evacnet4 Users Guide* [online]. University of Florida. Available from: http://www.ise.ufl.edu/kisko/files/evacnet/EVAC4UG.HTM#sec_2_0. [Accessed 10 January 2008]

[Klawonn 2008] Klawonn, F. 2008. *Introduction to Computer Graphics*, Springerlink.

[Klingsch 2008] Klingsch, W.F., Rogsch, C., Schadschneider, A., Schreckenberg, M., 2008. Comparison of Various Methods for the Calculation of the

Distance Potential Field. *In: Pedestrian and Evacuation Dynamics,* 27-29 February 2008, Wuppertal, Germany.

[Klupfel 2001] Klupfel, H., Meyer-K¨onig, T., Schreckenberg, M., 2001. Microscopic Modelling of Pedestrian Motion – Comparison of Simulation Results with an Evacuation
Exercise in a Primary School. *In: Traffic and Granular Flow,* 2001, Berlin, Germany.

[Klupfel 2003] Klupfel, H., Meyer-K¨onig, T., 2003. Simulation of the Evacuation of a Football Stadium using the CA Model Pedgo. *In: Traffic and Granular Flow,* 2003, Delft, The Netherlands.

[Ko 2006] Ko, S., Spearpoint, M., Teo, A., 2006. Trial evacuation of an industrial premises and evacuation model comparison. *Fire Safety Journal.* 42, 91-105.

[Kretz 2006] Kretz, T., Grunebohm, A., Schreckenberg, M., 2006. Experimental study of pedestrian flow through a bottleneck. *Journal of Statistical Mechanics:* Theory and Experiment P10014.

[Kuligowski 2003] Kuligowski, E.D., 2003. *The Evaluation of a Performance-Based Design Process For a Hotel Building: The Comparison of Two Egress Models.* Thesis (Masters). University of Maryland.

[Kuligowski 2005] Kuligowski, E.D., Peacock, R.D., 2005. A Review of Building Evacuation Models. Technical Note 1471. National Institute of Standards and Technology (NIST), U.S Department of Commerce.

[Lakoba 2005] Lakoba, T.L., Kaup, D.J., Finkelstein, N.M., 2005. Modifications of the Helbing-Molnar-Farkas-Vicsek Social Force Model for Pedestrian Evolution. *SIMULATION, The Society for Modeling and Simulation International.* 81(5), 339 – 352.

[Lahey 2000] Lahey, D.M. 2000. *A Software Framework for Developing High Quality Control Systems for Autonomous Robots.* Thesis (Msc). The University of British Columbia.

[Legion 2008a] Legion. *What we do* [online]. Available from: http://www.legion.com/about-us/what-we-do.php [Accessed 20 January 2008].

[Legion 2008b] Legion. News [online]. Available from: http://www.legion.com/news/patent.php [Accessed 20 January 2008].

[Levin 1989] Levin,B., 1989. Exitt, A Simulation Model of Occupant Decisions and Actions in Residential Fires. *In: Proceedings Of The Second International Symposium On FireSafety Science*, 561-570.

[Lin 2006] Lin, J.J., Feng, C.M., Hu, Y.Y., 2006. Shifts in Activity Centers along the Corrifor of the Blue Subway Line in Taipei. *Journal of Urban Planning and Development.* DOI: 10.1061/(ASCE) 0733-9488(2006)132:1(22).

[Lo 2000] Lo, S.M., Fang, Z., 2000. A Spatial-Grid Evacuation Model for Buildings. *Journal of Fire Sciences*, 18, 376-394.

[Lo 2002] Lo, S.M., Fang, Z., Zhi, G.S., Yuen, K.K., 2002. A Computer Simulation Model of Emergency Egress for Space Planners. *Facilities.* 20(7/8), 262-270.

[Lo 2004] Lo, S.M., Fang, Z., Lin, P., Zhi, G.S., 2004. An evacuation model: the SGEM package. *Fire Safety Journal*, 39, 169-190.

[Loscos 2003] Loscos, C., Marchal, D., Meyer, A., 2003. Intuitive Crowd Behaviour In Dense Urban Environments using Local Laws. *Proceedings of the Theory and Practice of Computer Graphics*, 0-7695-1942-3/03.

[MAIB 2003] Marine Accident Investigation Branch 2003, Report on the Investigation of a Fatal Accident during a Vertical Chute Evacuation Drill from the UK Registered ro-ro Ferry P&OSL Aquitaine in Dover Harbour on 9 October 2002, Report No 18/2003.

[Milinskii 1979] Milinskii, A.I., Predtechenskii, V.M., 1979. *Planning for Foot Traffic Flow in Buildings.* American Publishing Co. Pvt Ltd.

[Moulin 2004] Moulin, B., Chalker, W., Gancet, J., 2004. PADI-Simul: An agent based geosimulation software supporting the design of geographic spaces. *Computers, Environment and Urban Systems,* 28, 387-420.

[Muncaster 1990] Muncaster., R., 1990. Physics. 3rd edition. ISBN 0748700501.

[Nagai 2006] Nagai, R., Fukamachi, M., Nagatan, T., 2006. Evacuation of crawlers and walkers from corridor through an exit. Physica A 367, 449-460

[Nelson 2002] Nelson, H. E., Mowrer, F. W., 2002. Emergency Movement. P.J.DiNenno and W. D. Walton, ed. *The SFPE Handbook of Fire Protection Engineering,* 3rd ed., 3-367-3-380. Society of Fire Protection Engineers, Bethesda, MD.

[NFPA 1995] National Fire Protection Association., 1995. *The SFPE Handbook of Fire Protection Engineering*. 2nd ed. USA: NFPA.

[Nober 1981] Nober, E. H., Pierce, H., Well, A. D., Johnson, C. C., Clifton, C., 1981. Waking Effectiveness of Household Smoke and Fire Detection Devices. *Fire Journal*, 75, 86-91.

[Okasaki 1993] Okasaki,S. And Matsushita,S., 1993. A Study of Simulation Model For Pedestrian Movement with Evacuation and Queuing, *In: Engineering*

For Crowd Safety, 271-280.

[ONeil 2008] O'Neil, J., Steele, G.K., Huisingh, C., Smith, G.A., 2008. Escalator-related injuries among older adults in the United States. *Accident Analysis and Prevention*, 40(2), 527-533.

[Pan 2006a] Pan, X., 2006. *Computational Modelling of Human and Social Behaviours for Emergency Egress Analysis*. Thesis (PhD). Stanford University.

[Pan 2006b] Pan, X., Han, C.S., Dauber, K., Law, K.H., 2006. Human and Social Behavior in Computational Modeling and Analysis of Egress. Automation in Construction.

[Patterson 2005] Patterson, J.A. 2005. *Implementing Autonomous Crowds in a Computer Generated Feature Film.* Thesis (Msc) . Texas A & M University.

[Pathfinder 2009] Pathfinder, 2009. *Agent-Based Emergency Egress Simulation Technical Reference.* Thunderhead Engineering. Available From: http://www.thunderheadeng.com/pathfinder/documentation.html [Accessed 17 June 2010].

[Pelechano 2007] Pelechano, N., Allbeck, J.M., Badler N.I., 2007. Controlling Individual Agents in High-Density Crowd Simulation. *In: D.Metaxas, J. Popovic. ACM SIGGRAPH Symposium on Computer Animation*, 3-4 August 2007, San Diego, CA.

[Pelechano 2008a] Pelechano, N., Stocker, C., Allbeck, J.M., Badler N.I., 2008. Being a Part of the Crowd: Towards Validating VR Crowds Using Presence. *In: Proceedings of 7th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons ,ed., 136-142.

[Pelechano 2008b] Pelechano, N., Malkawi, A., 2008. Evacuation Simulation Models: Challenges in Modelling High Rise Building Evacuation with Cellular Automata approaches. *Automation in Construction,* 17, 377-385.

[Predtechenkii 1978] Predtechenskii, V.M., Milinskii, A.I., 1978. *Planning for Foot Traffic Flow in Buildings*. Amerind, New Delhi.

[Proulx 1994] Proulx, G., 1994. Time Delay to Start Evacuating Upon Hearing the Fire Alarm. *In: Proceedings of Human Factors and Ergonomics Society 38 th Annual Meeting*, 811-815.

[Proulx 1996] Proulx, G., 1996. The Impact of Age on Occupants' Behaviour during a Residential Fire. *In: Human Factors and Ergonomics Society 40th Annual Meeting,* Philadelphia, Pennsylvania

[Proulx 1999] Proulx, G., 1999. How to Initiate Evacuation Movement in Public Buildings. *Facilities*, 17(9/10), 331-335.

[Proulx 2001] Proulx, G., 2001. Occupant Behaviour and Evacuation. *In: Proceedings of the 9th International Fire Protection Symposium*, 25-26 May 2001, Munich, 219-232.

[Purser 2009] Purser, D., 2009. Implications of Modelling and Experimental Studies of Evacuation Behaviour on Stairs for Multi-storey Building Design. *In: Proceedings of Human Behaviour in Fire Conference,* 147-160.

[Reynolds 1987] Reynolds, C.W., 1987. Flocks, Herds and Schools: A Distributed Behavioural Model. *ACM, Computer Graphics*, 21(4), 25-34.

[Reynolds 1999] Reynolds, C. W., 1999. *Steering* Behaviors for Autonomous Characters. *In: the proceedings of Game Developers Conference 1999*, San Jose, Miller Freeman Game Group, San Francisco, California, 763-782.

[Reynolds 2000] Reynolds, C.W., 2000. Interaction with Groups of Autonomous Characters. *In: Game Developers Conference*, 2000.

[Reynolds 2006] Reynolds, C., 2006. Crowd Simulation on PS3. *In: Game Developers Conference, 2006.* Sony Computer Entertainment US Research and Development.

[Rogsch 2005] Rogsch C, Seyfried A, Klingsch W., 2005. Comparative Investigation of the Dynamic Simulation of Foot Traffic Flow. *In: Pedestrian and Evacuation Dynamics 2005*, 27-30 September 2005 Vienna. Springer, 357-362.

[Rogsch 2007] Rogsch, C., Klingsch, W., Seyfried, W., Weigel, H., 2007. How reliable are commercial software-tools for evacuation calculation? In: Interflam Conference, 3-5 September 2007 London.

[Sakuma 2005] Sakuma, T., Mukai, T., Kuriyama, S., 2005. Psychological Model for Animating Crowded Pedestrians. Visual Agent Laboratory – Toyohashi University of Technology.

[Schadschneider 2009] Schadschneider, A., Seyfried, A.(2009). Validation of CA models of Pedestrian Dynamics with Fundamental Diagrams. *Cybernetics and Systems,* 40(5), 367 — 389.

[Schneider] Schneider, V. ASERI (Advanced Simulation of Evacuation of Real Individuals). A model to simulate evacuation and egress movement based on individual behavioural response. I.S.T. Integrierte Sicherheits-Technik GmbH, Frankfurt / M., Germany

[Schneider 2001] Schneider, V., Konnecke, R., 2001. Simulating Evacuation Processes with ASERI. *In: Pedestrian and Evacuation Dynamics,* 4-6 April 2001, Duisburg.

[Seyfried 2009] Seyfried, A., Passon, O., Steffen, B., Boltes, M., Rupprecht, T., Klingsch, W., 2009. New insights into pedestrian flow through bottlenecks, *Transportation Science*, 43, 395-406.

[Shestopal 1994] Shestopal,V.O., Grubits,S.J., 1994. Evacuation Model For Merging Traffic Flows In Multi-Room And Multi-Storey Buildings. *In: Proceedings Of The Fourth International Symposium On Fire Safety Science*, Kashiwagi,T., ed., 625-632.

[Shields 1997] Shields, J. Boyce K.E., Silcock, G.W., Dunne, B., 1997-1998. The Impact of a Wheelchair Bound Evacuee on the Speed and Flow of Evacuees in a Stairway during an Uncontrolled Unannounced Evacuation. *Applied Fire Science,* 7(1), 29-39.

[Shields 1999] Shields, J., 1999. The Science of Human Behaviour, Past Research Endeavours, Current Developments and Fashioning a Research Agenda. *In: IAFSS Conference,* Poitiers, France.

[Shields 2000] Shields, J., Boyce, K.E., 2000. A Study of Evacuation from Large Retail Stores. *Fire Safety Journal,* 35, 25-49.

[Sime 1989] Sime, J.D., 1989. Crowd facilities, management and communications in disasters. *Facilities,* 17 (9/10), 313 -324.

[Sime 1985] Sime, J., 1985. The Fire 'Game': Future Directions for Research and Development, *In*: *Fire Protection*.

[Sime 1983] Sime, J., 1983. Affiliative Behaviour during Escape to Building Exits. *In: Journal of Environmental Environmental Psychology*, 3(1), pp21-24, 1983.

[Sime 1993] Sime, J., 1993. Crowd Psychology and Engineering: Designing for People or Ballbearings? *In: International Conference on Engineering for Crowd Safety*, 17-18 March, London, UK.

[Smith 1994] Smith, M.A., 1994. *Object-oriented software in C++*. 1st ed. London: Chapman and Hall.

[Smith 1995] Smith, R.A., 1995. Density, velocity and flow relationships for closely packed crowds. *Safety Science,* 18, 321-327.

[Stahl 1982] Stahl,F., 1982. BFIRES-II: A Behaviour Based Computer Simulation of Emergency Egress during Fires. *Fire Technology,* 18, pp 49-65.

[Still 2000] Still, G.K., 2000. *Crowd dynamics*. Thesis (PhD). University of Warwick

[Takahashi 1989] Takahashi,K., Tanaka,T., Kose,S., 1989. An Evacuation Model for use in Fire Safety Design of Buildings. *In: Proceedings Of The Second International Symposium On Fire Safety Science,* 551-560.

[Thompson 1994] Thompson, P.A., 1994. *Developing new techniques for modelling crowd movement*. Thesis (PhD). University of Edinburgh.

[Thompson 1995] Thompson, P. and Marchant, E., 1995. A Computer Model for the Evacuation of Large Building Populations. *Fire Safety Journal*, 24, 131-148.

[Thompson 1996] Thompson, P., Wu, J., Marchant, E. Modelling Evacuation in multi-storey buildings with Simulex. *Fire Engineers Journal*, 56(185), 6-11.

[Tomlinson 2004] Tomlinson, S.L., 2004. The long and short of Steering in Computer Games. *International Journal of Simulation*, 1 -2 (5), 33-46.

[Tu 1994] Tu, X., Terzopoulos, D., 1994. Artificial Fishes: Physics, Locomotion, Perception, Behavior. *In: Computer Graphics, SIGGRAPH 94 Conference Proceedings*, 43-50.

[TWUK 2010] Transport Watch UK, *Facts sheet 3: Widths and headroom* [online]. Available from: http://www.transport-watch.co.uk/transport-fact-sheet-3.htm [Accessed 29 November 2010].

[Van den Berg 2008] Van den Berg, J., Patil, S., Sewall, J., Manocha, D., Ming Lin., 2008. Interactive Navigation of Multiple Agents in Crowded Environments. *In: Proceedings Of The 2008 Symposium On Interactive 3D graphics and games.* 15-17 February 2008, California.

[Ward 2006] Ward, J., 2006. *Pedestrian Dynamics: Models of Pedestrian Behaviour* [online]. Available from: http://www.pedestriandynamics.com [Accessed 5 December 2007].

[Weidmann 1993] Weidmann, U. 1993. Transporttechnik der Fussgänger, Transporttechnische Eigenschaften des Fussgängerverkehrs (Literturauswertung) Schriftenreihe des IVT Nr. 90, Zweite, ergänzte Auflage, Zürich, März 1993 (109 Seiten)

[Weidmann 2010] Weidmann, U. 2010. Pedestrian Data, Weidmann [online] Available from: http://www.ped-net.org/index.php?id=21 [Accessed 15 July 2010]

[Wood 1990] Wood PG, 1990. Survey of Behaviour in Fires. *Fires and Human Behaviour, 2nd ed.* Fulton, 1990.

[W3Schools 2010] W3Schools. *How XML can be used?* [online] Available from: http://www.w3schools.com/xml/xml_usedfor.asp [Accessed 15 July 2010].

[Xie 2005] Xie, H., Filippidis, L., Galea, E.R., Gwynne, G., BlackShields, D., Lawrence, P.J., 2005. Experimental Study and Theoretical Analysis of Signage Legibility Distances as a Function of Observation Angle. *In: Pedestrian and Evacuation Dynamics 2005*, 27-30 September 2005 Vienna. Springer, 131-143.

Appendices

1. Publication produced during the course of the research project

This paper was presented at the Fifth International Conference on Pedestrian and Evacuation Dynamics, National Institute of Standards and Technology, Gaithersburg, MD USA. March 8-10, 2010

1.1 Implementing a Hybrid Space Discretisation Within An Agent Based Evacuation Model

N. Chooramun, P.J. Lawrence and E.R.Galea

Fire Safety Engineering Group
University of Greenwich, London SE10 9LS, UK

Abstract Egress models typically use one of three methods to represent the physical space in which the agents move, namely: coarse network, fine network or continuous. In this work, we present a novel approach to represent space, which we call the 'Hybrid Spatial Discretisation' (HSD), in which all three spatial representations can be utilised to represent the physical space of the geometry within a single integrated software tool. The aim of the HSD approach is to encompass the benefits of the three spatial representation methods and maximise computational efficiency while providing an optimal environment to represent the movement and interaction of agents.

Introduction

Within all evacuation and pedestrian dynamics models, the physical space in which the agents move and interact is discretised in some way [1,2]. Models typically use one of three basic approaches to represent space [1], a continuous representation of space e.g. SIMULEX [3], a fine network of nodes e.g. buildingEXODUS V4.07 [4] or a coarse network of nodes e.g. PEDROUTE [5]. Each approach has its benefits and limitations, the continuous approach allows for an accurate representation of the building space and the movement and interaction of individual agents but suffers from relative poor computational performance, the coarse nodal approach allows for very rapid computation but suffers from an inability to accurately represent the interaction of individual agents with each other and with the structure. The fine nodal approach represents a compromise between the two extremes providing an ability to represent the interaction of agents while providing good computational performance.

In this paper, we present a novel approach to represent space, which we call the 'Hybrid Spatial Discretisation' or HSD. In the HSD approach all three spatial representations can be utilised to represent the physical space of the geometry within a single integrated software tool. The aim of the HSD approach is to encompass the benefits of the three spatial representation methods and maximise computational efficiency while providing an optimal environment to represent the movement and interaction of agents. Using the HSD approach, the fine nodal approach is used to map the majority of the geometry, providing reasonable speed and an ability to model agent interaction. In parts of the geometry where greater precision is required to model detailed interaction between agents, the continuous approach is used and in regions where knowledge of detailed agent interaction is not required the coarse nodal approach can be used, providing improvements in speed and computational efficiency. This approach is particularly useful in modelling very large complex spaces and urban environments.

In this paper, we examine where the various spatial representations are appropriate for use and we briefly describe the key algorithms developed for the bEX-H implementation, in particular those required to model the continuous and coarse nodal regions. In addition, we provide a demonstration example of how the technique can be applied and discuss related accuracy and performance issues. In the work presented here, the HSD approach is implemented within the buildingEXODUS V4.07 software and is identified as the buildingEXODUS-Hybrid prototype or bEX-H. The buildingEXODUS (bEX) model has been frequently described in other publications [4, 6] and will therefore not be described here.

Software Architecture

The bEX-H makes use of the core architecture of the buildingEXODUS software. The buildingEXODUS software has been modified to allow plug-in modules to be included into the core software using a component oriented engineering approach. This architecture provides a platform whereby new functionalities can be independently developed and incorporated into the model as required. The coarse network and continuous region are examples of two components which have been developed as plug-in modules for the bEX-H model.

Continuous Region Component

When using a continuous approach for the discretisation of space, it is possible to take into consideration a larger number of agent attributes allowing for a wider range of agent behaviours to be modelled. In this section, we describe the planning approach of the agents and some of their advanced behaviours. bEX is based on a multi-agent system whereby each agent is modelled as an autonomous agent which exhibits some forms of adaptive behaviour. In other words, the agent has the ability to navigate in a life-like manner and react to stimuli in its environment. The agent will at some point in its trajectory encounter obstacles. The obstacles can be static such as walls or tables or dynamic, for instance, other agents navigating in the same environment. In this respect, the agent is able to detect these obstructions and react to them accordingly. Some of the additional attributes of the continuous agents are as shown in Table 1 below.

Table 1. Additional agent attributes within bEX-H

Attributes	Description	Quantity
2-D Position	The location of the person in continuous space	Vector
Velocity	Rate of change of displacement	Vector
Acceleration	Rate of change of velocity	Vector
Max Acceleration	Maximum acceleration of a continuous person	Scalar
Orientation	The possible movement directions (headings)	N basis vectors
Body Frame Width	The width of the person excluding the size of	Scalar

	the shoulders	
Body Shoulder Width	The width of the person's shoulders	Scalar

The agent navigates around its environment using two levels of navigation comprising of local and global strategies each influencing different aspects of the individual's movements. Local navigation relates to low level reactive behaviours which are required for collision avoidance. Whereas the global strategy relates to navigation and high level decision making processes for example an agent deciding which route to adopt from their current location to their target.

The path of an agent within continuous space from a start location to an end location can be described as a continuous map [7]. However, the complexity of this map can increase significantly in large geometries for example, buildings with multiple internal rooms and floors. Moreover, the presence of static obstacles within the enclosure makes the path planning process of the agents even more complex. In order to reduce the complexity of the continuous map, the continuous region in bEX-H uses a Navigational Graph. This is a network of waypoints and path segments which is automatically generated in the pre-processing phase. Each waypoint is assigned a potential value which represents the shortest visible arc distance from the external door. Illustrated in Figure 1 is a geometry with multiple compartments and its corresponding navigational graph. However, unlike other roadmap methods such as the visibility graph [7] where the links (path segments) are connected to each and every visible vertex, in the navigational graph, the waypoints are generated only at locations where the internal angles are concave.

The behaviour of the agents is modelled as simple components which can serve as building blocks for simulating more complex behavioural routines. Moreover, this approach allows behaviours to be implemented and tested individually, followed by an incremental integration. Examples of behaviours in bEX-H include:

- Seek - used to steer a person towards a specific goal, which takes into account the agents speed and turning rate,
- Wall Avoidance – ability to detect collisions which could happen in the future given the current trajectory,

- Agent Avoidance and Lane Formation - enables the agents to maintain a desired interpersonal space from each other which is proportional to the agents velocity and the body width of neighbouring agents.

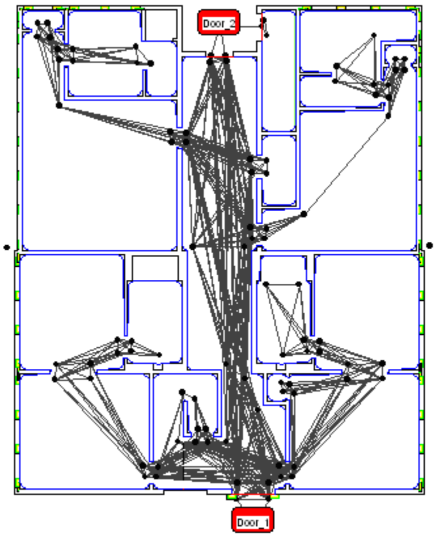


Fig. 1. The Underlying Navigational Graph

The Continuous Region component includes both an adaptive and prescriptive setting for agent travel speeds. In the adaptive setting, the agent considers each neighbouring agent within its perception box (see Agent Avoidance) in turn, and computes a repulsive force. This force is normalised and scaled to be inversely proportional to the squared of the inter-person distance so that distant neighbours have a negligible effect on the assessing agent. The sum of the vectors has a decelerating effect on the agents thereby reducing the travel speed of the agent. In other words, the agents adapt their travel speeds according to population density in their surroundings. In the prescriptive setting, the travel speed of each agent is dictated by a speed density relationship such as, S = k – akD [8], where S is the speed, D is the population density and a and k are constants. As the population density within a room or compartment may be non-uniform, bEX-H uses a localised density calculation approach.

Coarse Network Component

In this section we describe the coarse network approach for representation of physical space, and its implementation within bEX-H. Using this approach, the available physical space can be segmented into partitions whereby each partition can represent a section of the geometry such as a room or corridor. Each partition is a node which is connected via arcs or links to represent doorways or other forms of connectivity in the structure. Each node has a limit on the number of agents it can contain (maximum capacity) while the arcs have a maximum flow capacity, that is, the maximum number of agents that can traverse the arc per time period. There are seven types of coarse node implemented within bEX-H, these are; compartment (general region in which agents exit or enter), intersection (complex region in which flows from different directions merge), interchange (larger version of intersection), gates (narrow metered passage such as turnstiles), stairs, escalators and travelators. The agents within coarse networks transit from one segment to another while the physical movement within the segment itself is not represented. The flow rate through a coarse node in bEX-H is a function of travel speed, travel distance and population density in the region. The travel distance is defined as the distance the agent has to travel, within the node, from their entry point to their exit point. This is implemented using a "Flow to Density Equation" also known as the F-D Model **[5]**. The rate of flow, in a coarse node, is also subject to two other limitations, the maximum capacity of the coarse node and the connecting arcs. When an agent enters a coarse node, their path, travel distance and speed is fixed based on the Flow to Density Equation. The only dynamic events which can affect their dwell time in the node are the capacity of the arcs out and congestion in adjacent nodes.

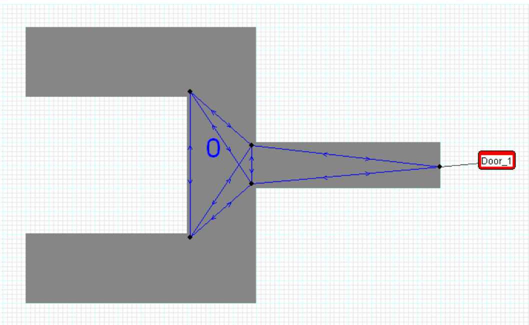


Fig. 2. Coarse Node Implementation of non-convex geometry in bEX-H

The nature of the coarse network approach restricts the modelling of a region with simple geometrical shapes such as rectangles. However, these simple shapes might not be sufficient to accurately model all the complex segments within an enclosure. Unlike other coarse node models, the coarse node implementation in bEX-H allows the creation of non-convex regions as shown in Figure 2. The key data structure behind a coarse node is a mesh which is based on the same principle as the Navigational Graph.

The coarse node model in bEX-H features a behavioural mechanism which enables agents entering a coarse node to adjust their paths depending on the evolving conditions of population density within the coarse node. This feature is implemented through a load balancing algorithm. This algorithm dynamically adjusts the potential value of the waypoints to account for the number of agents heading towards them.

The HSD approach involves the mixing of macroscopic (coarse node) and microscopic (continuous and fine) modelling methodologies which in itself presents several challenges. In the continuous and fine node models, the agents are modelled from an individual perspective whereby their movements, exact locations and behaviours can be tracked. However, in the coarse network approach, the agents are modelled from a global perspective whereby the population is treated as a homogenous ensemble, such that the locations and physical space occupancy of the agents are not represented. bEX-H incorporates some behavioural and movement mechanisms to facilitate the transition of agents across the transition regions. The current implementation of bEX-H features the representation of all six possible interface transition regions namely: Coarse Node ↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous. The mechanisms for some of the key transitions are briefly described below.

Coarse Node/Continuous Region Transition: This transition represents the two extremes on the scale of granularity. When an agent traverses from the coarse node to the continuous region, its starting location in the continuous region is set equal to the coordinates of the waypoints from which it emerges. This may result in a narrow stream of agents emerging from the coarse node, which may appear unrealistic. bEX-H incorporates a behaviour called Separation which is invoked temporarily by the agents upon entering the continuous region. This generates a repulsive force between agents which is inversely proportional to their inter-person distance. This allows the agents to spread out and make a more realistic use of the available continuous space.

Coarse Node/Fine Node Transition: When agents traverse from coarse nodes to fine nodes, they are re-positioned on the available fine nodes which are connected by arcs to the coarse node region.

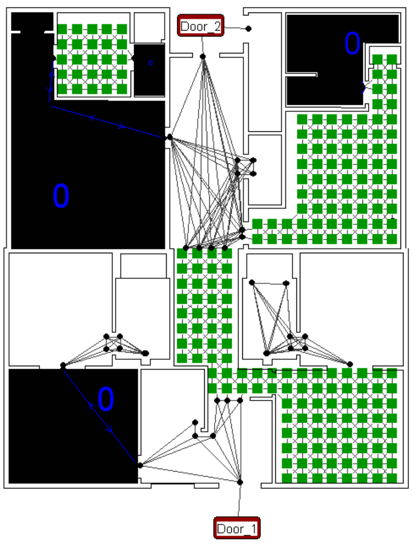
Fine Node/Continuous Region Transition: The transition of agents from the fine nodes to the continuous regions is based on the same approach as for the Coarse Node/Continuous Region.

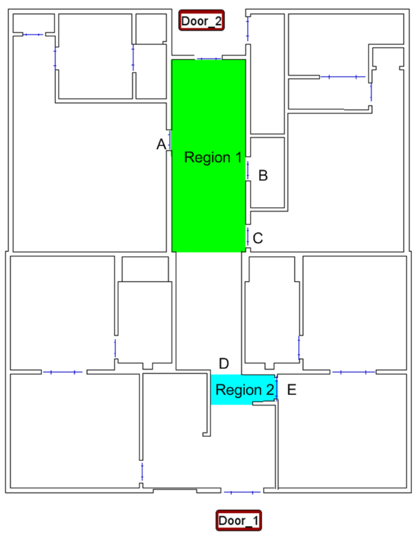
Demonstration Case using HSD approach

In this section we demonstrate the application of the HSD approach to the complex geometry depicted in Figure 3a. In this example a multi-compartment geometry is modelled using all three spatial representations. The building population initially occupies every compartment, thus the scenario investigated demonstrates all six possible interface transition regions: Coarse Node ↔ Fine Node; Fine Node ↔ Continuous and Coarse Node ↔ Continuous. The simulations were performed using a PC with Intel E8600 Core 2 Duo CPU, 3.3 GHz and 8GB of RAM with a 512 MB GeForce GTS 250 graphics card. The experimental set up in bEX-H was as follows:

- 300 agents in total with average density of 2 persons/m^2 in each compartment.
- Free-Flow conditions were imposed on the exits i.e. no flow rate limitations were imposed.
- Fast Walk (Unimpeded walking speed) : 1.5 m/s
- Response Time: 0 s

Both external exits available and all internal exits are 1.0 m wide.





(a) Geometry modelled using bEX-H involving 40% Continuous (white regions), 30% Coarse (dark regions) and 30% Fine node regions (grid of nodes).

(b) Location of Intersection Nodes (shaded regions 1 and 2) in the geometry. All the other areas modelled using Compartment Nodes.

Fig. 3. Complex geometry used in demonstration example

In order to demonstrate the differences and similarities between the various approaches, the demonstration case is repeated using; all Coarse Nodes, all Fine Nodes and all Continuous Regions. Ten simulation runs were conducted for each case. In the all Coarse Nodes case, the geometry was modelled by using a combination of compartment and intersection blocks as shown in Figure 3b.

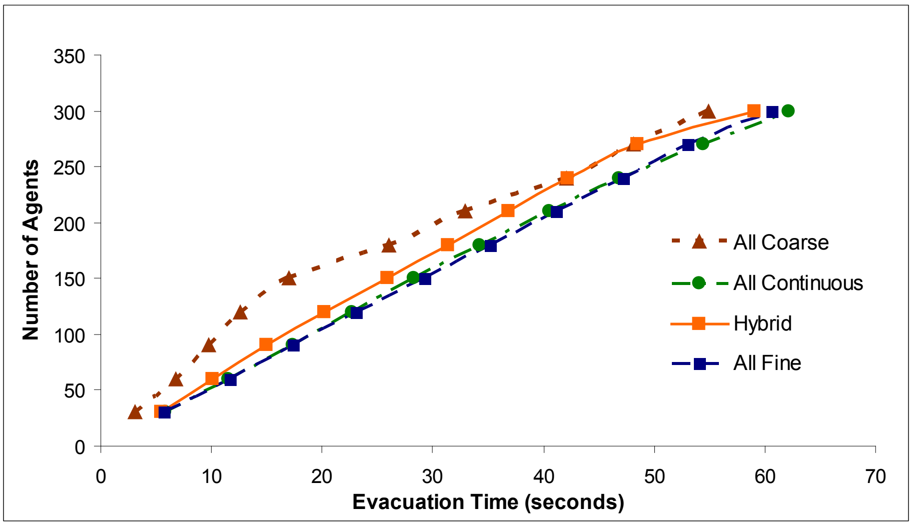


Fig. 4. Time taken for a specific percentage of people to evacuate using All-Fine, All-Continuous, All-Coarse and Hybrid spatial representations

Table 2. Summary of results averaged over 10 simulations

Spatial Representation	Average Total Evacuation Time (sec)
All-Fine	60.6
Continuous	62.2
All-Coarse	54.9
Hybrid	59.1

The evacuation time curves for each case are shown in Figure 4. As can be seen, the evacuation curves for the All-Coarse, All-Fine and Continuous cases are similar, with the All-Coarse case consistently underestimating the egress times throughout the evacuation. The All-Fine and Continuous simulations produce virtually identical evacuation histories up to the final part of the evacuation. During the last 10% of the evacuation, the exit flow rate in the Continuous model tails off at a slightly greater rate than in the All-Fine model. This produces a slightly longer total evacuation time for the Continuous model as shown in Table 2. The average total evacuation time for the Continuous model is some 2.6% slower than the

All-Fine model while the average total evacuation time for the All-Coarse model is some 9.4% faster than the All-Fine model.

As can be seen the Hybrid simulation curve falls between the curves for the All-Coarse simulation and the All-Continuous simulation (see Figure 4). The average total evacuation time falls between the two extremes produced by the All-Coarse and the All-Continuous models and is marginally (2.5%) smaller than that produced by the All-Fine model (see Table 2).

More effort is required to determine the impact of using different combinations of the three discretisation approaches on both the accuracy of predictions and the speed of performance. These factors are also expected to be influenced by the type of discretisation that is used to represent specific regions of the geometry and the size and location of the simulated population. However, two primary applications are anticipated for bEX-H, the first are large complex structures such as airport terminals and high-rise buildings or systems of structures such as long tunnels with complex interchanges. In these types of applications, the majority of the structure would primarily consist of a combination of Fine and Coarse nodes. The Fine node structure would be used throughout the bulk of the structure where detailed analysis was required while the Coarse nodes would be used in the far field to represent parts of the structure which are not central to the analysis. The Continuous approach would only be utilised in special areas to represent regions such as pinch points or exits.

The second type of application involves urban environments such as a town or city. In applications involving urban scale geometries, the bulk of the geometry would be represented using the Coarse node approach with key areas such as assembly points or interchanges represented using the Fine node approach. It is unlikely that the Continuous approach would be utilised in such large scale applications.

In addition to the performance enhancements offered by the hybrid version of buildingEXODUS, the software can also be run in parallel using multiple computers [9]. This capability will also be expanded to include the hybrid version of the software.

Conclusions

In this paper we have presented a novel approach, known as the HSD to represent the discretisation of space in circulation and evacuation models. The HSD approach allows enclosures to be modelled using a mixture of the three basic techniques for space discretisation, coarse networks, fine networks and continuous. In the example presented, in which 30% of the domain was represented by the coarse discretisation, 30% by the fine discretisation and 40% by the continuous discretisation, the HSD approach was shown to produce results of similar accuracy to that produced by the All-Fine and All-Continuous approach. While further testing is required, the HSD approach appears to provide flexibility in defining the mix of approaches used in discretising the circulation space within the geometry. Further work is also required to optimise the numerical efficiency of the various plug-in components so that numerical efficiencies offered by using a mix of discretisation schemes can be fully exploited. In addition, the HSD approach will be extended so that it works efficiently within the parallel computing environment currently offered by buildingEXODUS.

References

1. Gwynne, S., Galea, E.R., Lawrence P.J., Owen, M., and Filippidis, L.: A Review of the Methodologies used in the Computer Simulation of Evacuation from the Built Environment, Building and Environment Vol. 34, pp 741-749, (1999).
2. Kuligowski, E.D., Peacock, R.D.: A Review of Building Evacuation Models. *Technical Note 1471.* National Institute of Standards and Technology (NIST), U.S Department of Commerce, (2005).
3. Thompson, P. and Marchant, E.: A Computer Model for the Evacuation of Large Building Populations, Fire Safety Journal ,**24**, pp 131-148, (1995).
4. Galea E.R. and Galparsoro J.M.P.: A Computer Based Simulation Model for the Prediction of Evacuation from Mass Transport Vehicles. Fire Safety Journal Vol 22 pp 341 - 366, (1994).
5. Buckmann, L.T. and Leather, J.: Modelling station congestion the PEDROUTE way, Traffic Enigineering and Control, 35, pp373-377, (1994).
6. Gwynne,S., Galea, E.R., Lawrence, P.J., and Filippidis, L.: Modelling Occupant Interaction with Fire Conditions Using the buildingEXODUS Evacuation Model, Fire Safety Journal, 36, pp. 327-357, (2001).

7. Doyle, A. B.: Algorithms and Computational Techniques for Robot Path Planning. Thesis (PhD). University of Wales, UK, (1995).
8. Nelson, H. E. and Mowrer, F. W.: Emergency Movement. In: P.J.Denno & W.D. Walton (Eds.), The SFPE Handbook of Fire Protection Engineering (Third ed., pp. 3-367-3-380). SFPE, Bethesda, MD, (2002).
9. Grandison A; Muthu Y; Lawrence P; Galea E.R.: Simulating the Evacuation of Very Large Populations in Large Domains using a Parallel Implementation of the buildingEXODUS Evacuation Model.", Proceedings of the 11th International Fire Science & Engineering Conference, Interflam 2007, 3-5th September 2007, Royal Holloway College, University of London, UK, Volume 1, pp. 259-270. ISBN 978 0 9541216-8-6, (2007).

2. Flow to Density Equation (F-D Model)

This section describes the speed/flow parameters used in the Coarse Node model within buildingEXODUS-Hybrid.

t=t0+A*(V/C) ^n

t travel time (seconds) at flow level V

t0 free-flow travel speed (seconds)

t1 travel time at Max Capacity

V occupants demand (pass/m/hr) or density (pass/m^2)

C Max Capacity (pass/m/hr or pass/m^2)

A and n Constants where A=t1-t0

Compartment

Parameter	**Value**
Free Flow Speed	1.50 m/s
Speed at Capacity	0.60 m/s
n	9.5
Population Density at Max Capacity	2.39 occ/m^2
t0	0.67 s
t1	1.67 s

Travelators

Parameter	**Value**
Free Flow Speed	1.92 m/s
Speed at Capacity	0.99 m/s
n	9.5
Population Density at Max Capacity	1.45 occ/m^2
t0	0.52 s
t1	1.01 s

Stairs Up

Parameter	Value
Free Flow Speed	0.59 m/s
Speed at Capacity	0.36 m/s
n	1.8
Population Density at Max Capacity	2.87 occ/m^2
t0	1.70 s
t1	2.78 s

Stairs Down

Parameter	Value
Free Flow Speed	0.67 m/s
Speed at Capacity	0.56 m/s
n	2.7
Population Density at Max Capacity	2.03 occ/m^2
t0	1.49 s
t1	1.79 s

Escalator Up

Parameter	Value
Free Flow Speed	0.84 m/s
Speed at Capacity	0.51 m/s
n	2.3
Population Density at Max Capacity	3.92 occ/m/hr
t0	1.19 s
t1	1.96 s

Escalator Down

Parameter	Value
Free Flow Speed	1.00 m/s
Speed at Capacity	0.640 m/s
n	1.4
Population Density at Max Capacity	3.13 occ/m^2
t0	1.00 s
t1	1.56 s

Gates

Parameter	Value
Free Flow Speed	1.530 m/s
Speed at Capacity	1.530 m/s
n	0.0
Population Density at Max Capacity	0.27 occ/m^2
t0	0.654 s
t1	0.654 s

Intersection

Parameter	Value
Free Flow Speed	1.56 m/s
Speed at Capacity	0.34 m/s
n	1.1
Population Density at Max Capacity	2.15 occ/m^2
t0	0.64 s
t1	2.94 s

Interchange

Parameter	Value
Free Flow Speed	1.57 m/s
Speed at Capacity	0.41 m/s
n	10.9
Population Density at Max Capacity	2.15 occ/m^2
t0	0.64 s
t1	2.44 s