# THE DEVELOPMENT OF A SCENARIO INDEPENDENT METHOD FOR EVALUATING THE EVACUATION COMPLEXITY OF A BUILDING

**HONGJUN JIANG**

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

**March 2012**

The University of Greenwich,
School of Computing and Mathematical Science,
30 Park Row, Greenwich, SE10 9LS

# Declaration

I certify that this work has not been accepted in substance for any degree, and is not currently submitted for any degree other than that of Doctor of Philosophy (PhD) of the University of Greenwich. I also declare that this work is the result of my own investigations except where other stated.

X------------------------------------
Hongjun Jiang

X------------------------------------
Dr. Peter J Lawrence
(Supervisor)

X------------------------------------
Prof. Edwin Galea
(Supervisor)

# ACKNOWLEDGEMENTS

I am so grateful for having had the support of so many people and without it this research would not have been possible

Firstly, I would like to acknowledge my deep gratitude to my supervisors, Dr. Peter J Lawerence and Prof. Ed Galea, who gave me the opportunity to study in a friendly environment. I also owe them deeply to their great interesting, enthusiasm, guidance and attention to detail. They provided me with many helpful comments and suggestions during my studies and on the writing of this thesis.

I am also heartily thankful to Dr S. Gwynne, for both his guidance at the beginning stage of this study and helpful comments during my writing up.

The financial support provided by FSEG of the University of Greenwich is gratefully acknowledged. I would like to thank many others members in CMS school for providing support and help, ZhaoZhi Wang, FuChen Jia, XiangDong Xue, Arun Mahalingam, Yanbo Wang, Anand Veeraswamy and many more. I would like also thank Mrs Froncoise Barkshire and Mr David Cooney for their help over several years.

Finally, I would like thank my mother, my brother and my sister for their encouragement and support. Extra special thanks go to my wife, Limin Chen for her understanding and believing in me during many years of this study.

# ABSTRACT

Over the past two decades, more than 30 evacuation models have been developed to reproduce people’s movement patterns in evacuation. However, evacuation models cannot assess whether one building is better than another in regards to evacuation wayfinding.

There exist techniques that attempt to compare different buildings for evacuation complexity. However, these graph measures are primarily used to measure the relative accessibility of different locations in a spatial system and were not generated for the purpose of comparing the complexity of different buildings. Currently only one method exists, Donegan’s method [DT98] [PD96] [DT99], which can be applied to compare building for evacuation ability. However, this technique is severely limited to specific building layouts and only considers connectivity.

Taking the Donegan’s method as a first step, this thesis extends this algorithm to obtain a new *Distance Graph Method*, which considers travel distance as well as being able to be applied to graphs with circuits. Then a further building complexity measures is presented, the *Global Complexity (PAT)* method. This is shown to be a valid measure which considers additional important factors such as wayfinding time, travel distance and the areas of compartments.

The *Distance Graph Method* and *Global Complexity (PAT)* methods are based on a room graph representation which does not have the descriptive power to describe the actual routes taken during the wayfinding process. To resolve this drawback a further method is presented which utilises a ‘route-based graph’ that has the ability to represent the real route that an evacuee will take during the wayfinding process.

Furthermore the *Distance Graph Method* and *Global Complexity (PAT)* methods assume a “worst state” calculation for the nodal information. This means for buildings with more than one exit these methods calculate a global building complexity according to a mathematical formula, which considers all exits separately. To address these problems, the final method, *Complexity Time Measure*, is presented, which is based around a number of wayfinding behaviour rules over a ‘route-based graph’ representation. This addresses the question: *‘If an occupant is positioned at a random location within a building, on average how long does the occupant need to spend to find an available exit?’* Hence, provides a means to compare complex buildings, with circuits, in relation to evacuation capability.

## TABLE OF CONTENTS

# Chapter 1 Introduction

## 1.1 Background

### 1.1.1 The life loss in the fire

Many people are killed in fires every year around the world. If we just examine the data for the UK alone, the number of deaths due to fire has consistently fallen between 400 and 600 per annum in the past decade [FS10]. In 2008, there were 451 fire related deaths in the UK, an increase of 2% on the 2007 figure of 443. The 2008 total compares favourably with figures prior to 2007 (491 in 2006, 491 in 2005, 508 in 2004, 593 in 2003) [FS08] [FS10] even though there has been a general steady decline in fire deaths, this figure still remains relatively high.

The majority of fire-related deaths (around three-quarters) occur in dwelling fires. In 2008, 353 deaths were recorded. This compares with 331 in 2007, 363 in 2006, 376 in 2005, 374 in 2004 and 446 in 2003 [FS08] [FS10]. As in previous years, dwellings also had more fire related deaths than any other location. However, the press tend to only focus on the more prominent and newsworthy cases such as large-scale high-profile fires, for example in high density dwellings or entertainment complexes.

### 1.1.2 Evacuation modelling for buildings

Evacuation analysis is becoming an important part of performance-based analysis to assess the level of life safety provided in buildings. In early analyses, the engineers used hand calculations to assess life safety. Engineers usually employed the equations given in the emergency movement chapter of the Society of Fire Protection Engineers Handbook to calculate mass flow evacuation from any height of building [NM02]. In general models employing this method, the occupants are assumed to be standing at the doorway to the stair on each floor when the evacuation starts. Hence, the calculation focuses mainly on points of constriction throughout the building and calculates the time for the occupants to flow past each point on their route to an external exit. To solve these limitations and to achieve a more realistic evacuation calculation computer based evacuation models are employed. These offer the potential of overcoming the shortfalls of experimental or formulaic means of determining evacuation behaviour.

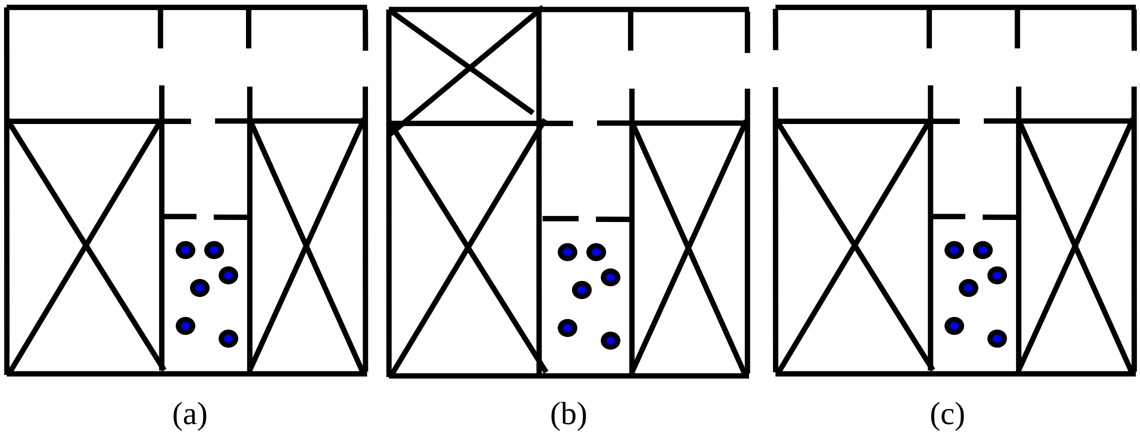
Over the past two decades, more than 30 evacuation models [Ku08] for buildings have been developed and corresponding simulation tools have been implemented to reproduce people's movement patterns during emergency and non-emergency conditions, e.g. buildingEXODUS, Pathfinder, FDS+EVAC, Simulex etc.

These models are intended to provide an estimate of the time required to evacuate a building layout for a prescribed scenario. The scenario includes factors such as the number, nature and distribution of the population, response times of the population, exit availability, etc.

By comparing predicted evacuation time for various building design options, one can compare and assess different building performances for a given scenario. Furthermore, the wayfinding capability of the evacuation model must also be considered. Some models simply direct people to the nearest exit. Others will direct people to nearest known exit while some more sophisticated models may also include factors such as signage to direct people to an exit. However, all these options are usually specified as part of the scenario specification. Hence, the evacuation time derived from such models is strongly dependent on the scenario specification.

Despite the power of evacuation models none of them have addressed the more fundamental notion of building complexity in regard to a habitable building. Any multi-compartment building may be regarded as a complex object or structure, which an occupant must navigate around. None of the existing models provide an understanding of why the configuration (and subsequent complexity) of one building is better than another, particularly relating to evacuation performance.

It would be desirable to have a measure of the building complexity which provided some insight into how difficult a building layout was from an evacuation point of view. Ideally, this measure would be scenario independent and readily calculable. Such a measure would be a useful quantity to assess building evacuation capabilities in addition to the evacuation time determined by evacuation models. The evacuation time derived from evacuation models provides a measure of how long it would take a population to evacuate from a particular building layout for a given scenario or scenarios; the complexity measure would provide some insight into how easy it would be to evacuate from that building. These two measures taken together provide a more complete insight into the suitability of the building or structure from an evacuation point of view.



(a) (b) (c)

Figure 1.1: Three sample building layouts

For example consider the three building layouts shown in Figure 1.1. The evacuation modelling scenario considers a group of people located in the lower room, as shown in Figure 1.1. Using the shortest path approach all three buildings would produce almost identical evacuation times, with everyone using the exit on the top right. However, from a building complexity point of view the building are quite different.

Lets assume for simplicity that we consider a single person starting from the same room as the group of people in the evacuation simulation. The building in Figure 1.1(a) is the most complex of the three as there is a T intersection which provide the occupant with a decision point, do they go left or right. If they go right they enter a dead end room and must return and proceed to the appropriate room and exit. The building in Figure 1.1 (b) is less complex than Figure 1.1 (a) as it does not present this decision point or the existence of a dead-end room. However, the building in Figure 1.1(c) is the least complex of the three as it provides an additional exit and no matter what choices the person makes it always leads to an external exit. In this case which ever route the person selects, at a decision point, it will always lead to an exit.

If we now consider that the individual could be positioned initially anywhere in the structure, we still can conclude that the building in Figure 1.1(c) is the least complex for evacuation wayfinding.

It is important to note that this concept of evacuation complexity does not take into consideration congestion, does not take into account population characteristics and does not generate an expected evacuation time for the building. For this an evacuation model is required. Combining the concept of building complexity with likely evacuation times provides a more complete analysis of a building evacuation capability. This thesis will focus on the design of a new building complexity measure, which can be included in the evacuation efficiency assessment of a building, and will then go to discuss the relative benefits of this.

## 1.2 Current graph measures used to assess building layout.

For the purpose of the architectural analysis, the representation of the building is the starting point. The popular graph representation methods (see Chapter 3) for a building mainly include the following kinds of graphs: room-based graph, axial map, and the visibility graph, a full definition of which are given in Chapter 3.

These methods are primarily used to measure the relative accessibility of different spaces in a spatial system. Most of these measures were generated to analyse axial maps and visibility graphs of a buildings (see Chapter 3). These measures mainly include the followings:

- Clustering coefficient [WS98][ST05][HP93]: The ‘clustering coefficient’ measure has its origin in the analysis of small-world networks, it is useful for the detection of junction points in environments. The clustering coefficient was first introduced by Watts and Strogatz for understanding the structural properties of small-world graphs [WS98], it indicates the average interconnectedness or ‘cliquishness’ of all neighbourhoods in a graph

- Mean depth [Wi47] [HP93]: The mean path length from a vertex is the average number of edge steps to reach any other vertex in the graph using the shortest number of steps possible in each case. It was first advanced by Wiener [Wi47]. It is used in visibility graph analysis, due to the parallels with the use of integration in space syntax theory [HP93], to show how visually connected a vertex is to all other vertices in the system.

There are also some other important graph measures, E.g. Point depth entropy [HH87], Step depth, Metric step depth etc [Tu04].

These graph measures have been used to analyse building structures. However, all these graph measures can only be used to measure the relative accessibility of different locations in a spatial system. These measures only provide local information relating to accessibility and do not provide a global measure of the evacuation capabilities of a building. They were not generated for the purpose of comparing the evacuation complexity of different buildings.

Donegan et al. [DP94] brought forward a graph specific measure technique which is distinct from scenario based measures of evacuation time modelling. For comparing different building evacuation ability, Donegan et al. proposed complexity theory to compare the evacuation capability of buildings. They call their method ‘egress complexity’. Egress Complexity is a scenario independent, non-metric methodology that assesses the egress capability of a compartmentalized floor plan [DP94].

For calculating the egress complexity for a given building, Donegan et al. employed the room-based graph representation for the building. Based on the room graph representation of the building plan, Donegan et al. applied ‘Shannon Entropy’ [Sh48] as a measure Of Structural Complexity [DP94] [DT98]. Such complexity could be equated to the summative uncertainty associated with a naive occupant in exhaustive pursuit of an exit without the benefit of signage [DP94]. However there are many limitations to this method which will be discussed in the next section.

## 1.3 Research Objectives and questions

Evacuation models do not consider the evacuation ability of a structure but simply estimate the time required to evacuate the structure. There is a lack of an efficient approach for comparing the evacuation wayfinding ability of different buildings. Ideally, we want to view a building plan and be able to suggest that building layout A will be better than building layout B for evacuation efficiency. Hence, the current study concentrates on the development of ‘building complexity’ models which can be applied to compare different buildings for their wayfinding or evacuation ability. When using building complexity to compare two or more structures with respect to evacuation ability, the building with the smallest value of complexity is the simplest to navigate. The building complexity measure is a standalone model, which can be used in conjunction with an evacuation model, to help the engineer assess whether a given structure is better for evacuation than another. Hence, this study will attempt to answer the following question:

**How can we define some measures of building complexity with regard to evacuation efficiency?**

Before generating the building complexity measures, the following questions should be answered first:

**Question 1: Are there any existing graph measures which can be used to compare different buildings in relationship to building evacuation efficiency?**

- **Question 1.1: What are the current techniques to represent a building as a graph?**
- **Question 1.2: What are the different types of measures that can be used in conjunction with the graphs representation of a building?**
- **Question 1.3: Can these measures be used to compare buildings in regards to evacuation capability?**
- **Question 1.4: Which measure is best suited for comparing building in relationship to evacuation capability?**

Based on the room graph representation of the building plan, Donegan et al. have attempted to compare different buildings [DP94] [DT98] [DP96]. A limitation of their method is that it can only be applied to calculate the complexity of a tree graph. However, normally the building graph is not just a tree, it often contains graphs with many circuits. In the situation where a graph contains circuits, they try to calculate the complexities for all the spanning trees of the graph. Then, the maximum value of the set of all spanning trees is used to indicate the complexity of the building. However, the problem of enumerating all the spanning trees of a graph is NP-complete [ME84].

Furthermore, Donegan’s method does not take into consideration the important factor of ‘travel distance’, which clearly is an important consideration with regard to complexity.

**Question 2: How to extend Donegan's method to obtain a new algorithm to address the following two sub-questions?**

- **Question 2.1: How can we introduce the concept of circuits into building graphs to calculate building complexity measures?**
- **Question 2.2: How can we introduce travel distance into building complexity to generate a new measure?**

Donegan's method mainly focuses on the amount of information possessed by an occupant of a building. During the wayfinding process, the information is gained gradually, and therefore reduces the uncertainty. However, as introduced in most evacuation models and the traditional safety codes, for example [BD82] [TH94] [Ta91], the maximum travel distance (or the travel time) is a very important factor which influences the evacuation ability. These vital factors for an occupant on finding an available exit were not considered by Donegan. As the travel distance and time are important factors in the wayfinding process, the following questions need to be answered: *How far and how long will occupants positioned at any location within the building need to travel to find an available exit?*

Minimizing the travel distance or time during the wayfinding process is the basic objective of the occupant. So the following question also needs attention: *What is the maximum distance the occupant needs to travel before successful egress from the building*? In other words, what is the maximum time the occupant needs to spend to find an available exit? Therefore, for buildings where the occupants has no knowledge of the structure, a building with smaller travel distances or times to find an exit is better for evacuation than buildings with a larger travel distances or times.

This research will also try to answer the following questions:

**Question 3: Can we develop a new graph measure, based on a room graph, which answers the following question:**

> ***"What is the maximum distance the occupant needs to travel before successful egress from the building*?"**

**Then for any new measures found:**

- **Question 3.1: Is there a relationship between these new measures and Donegan's method.**

All of the graph measures are based on a room graph representation of a building to determine the building complexity. This may be a natural choice but it is limited because a room-based graph does not have the ability to describe the actual routes taken during the wayfinding process. In most situations, the room-based graph cannot express the real environment of the building exactly, also the edges in the room-based graph cannot represent the real route the evacuee will travel to their destination. So such a simple outline of the structure by just specifying the connectivity between compartments is not sufficient for the required level of complexity of simulating the wayfinding process. So this research will try to address the following questions:

**Question 4: Is there another way to represent the building layout as a graph which overcomes this problem? In particular,**

- **Question 4.1: How can we represent the routes within a building?**

- **Question 4.2: How can we link routes within the building geometries with room entities?**

The methods based on room graphs have been developed based on the tree graph representation of the environment. This research will seek a suitable method to break circuits within graphs. However, the new route-based graph will include many circuits for expressing the exact routes within the building environment. Therefore, much of the information provided by the arcs will be lost after breaking the circuits, and the new tree graph will no longer denote the real routes within the environment. We also consider the important factor of travel distance in the wayfinding process. However, we will consider the average distance instead of the maximum travel distance to find an available exit. Hence it will be necessary to answer the question: *If an occupant is positioned at a random location within a building, on average how far the occupant needs to travel to find an available exit.*

Therefore, the following questions based on a new route graph representation will need to be addressed.

**Question 5: How can we apply a suitable technique to generate a measure of building complexity which can be used to answer the question *"If an occupant is positioned at a random location within a building, on average how far does the occupant needs to travel to find an available exit?"***

This raises the following sub-questions.

- **Question 5.1: How can we work from any graph which does not require the breaking of circuits?**
- **Question 5.2: What is the best way to implement a new route based algorithm?**

In this study, substantial effort has been directed towards developing the measures of building complexity to compare different building for wayfinding and evacuation efficiency. Attention has been focused on all the problems and issues presented above.

## 1.4 Structure of the thesis

***Chapter1:*** This chapter provided an overview of the research problem, enumerated the research questions posed and indicated how the objective of this research could be achieved in a manageable way.

***Chapter 2:*** This chapter provides a background to evacuation models techniques which are currently available, and highlights that none of these models have addressed building complexity measures for comparing different buildings in relationship to evacuation ability. This chapter also introduces buildingEXODUS modelling since the algorithms developed in this thesis will be compared with the results generated by this model for validating the building complexity measures.

***Chapter 3:*** This chapter reviews the applicable literature concerning graph measures related to building research. It covers the various techniques available to construct a graph representation for a building and the general measures which have been developed to analyse a spatial structure. The Chapter then highlight the vital factors which affect the evacuation ability, and it emphasizes the available techniques employed to compare different buildings for the evacuation ability.

***Chapter 4:*** In this chapter, attention is focused on egress complexity model generated by Donegan. Given the limitations of this model, an effective extension to Donegan method has been developed to deal with circuits within graphs.  This includes a complete revaluation of both the theories and the applications, behind this technique, which allows for the inclusion a distance factor.

***Chapter 5:*** This chapter describes the development of new building complexity measures using the room graph representation. Due to the limitations of the Donegan method and theories analysis, a new node complexity measure is produced. By extending the new measure, some new complexity calculating methods are developed.

These methods meet other important factors which influence the building evacuation ability, maximum travel distance and time.

***Chapter 6:*** In this chapter, a new kind of graph representation for a building structure is outlined which will be employed to generate a new building complexity model. In most situations, the room-based graph representation does not have the ability to express the real environment of the building exactly. The new route-based graph has the ability to describe an exact travel route of an occupant when way finding within a building.

***Chapter 7:*** In this chapter, new building complexity models are developed applying the route-based graph representation of a building plan. These models are generated based on answering the following question: if an occupant is positioned at a random location within a building, on average how far does the occupant need to travel to find an available exit. The new building complexity models are mainly based on the random walk technique.

***Chapter 8:*** This chapter provides a summary of the main conclusions of this thesis and a discussion on how this research can be taken forward with regards to ideas for further work.

# Chapter 2 Background to Evacuation Modelling

## 2.1 Introduction

This chapter will firstly outline the techniques which are currently applied to computer evacuation models for buildings, and try to answer *the question "do these modelling techniques provide any measures of building complexity?"* By analysing computer evacuation models for buildings, the following question will also be highlighted "*Would it be beneficial to include building complexity measures in an evacuation model?"*

Secondly, after generating a building complexity measure, we need to answer the question *"how do we validate these measures?"* i.e. show that one building is better or poorer for evacuation than another. The rest of the thesis will make use of the buildingEXODUS software as a validation tool.

## 2.2 Review of Evacuation modelling for buildings

### 2.2.1 Overview

Evacuation calculations are increasingly becoming an important part of performance-based analyses to assess the level of life safety provided in buildings [CM97]. There are two broad categories of evacuation analysis: hand calculations and evacuation models. In general, engineers utilise the hand calculations which are defined in the Society of Fire Protection Engineers (SFPE) Handbook to calculate mass flow for any building [NM02]. However, to produce more sophisticated analysis, engineers have been looking to evacuation computer models to assess a building's life safety [KP05]. These models have the potential to include more factors and represent their interaction. Currently, there is a number of building evacuation models available, which will be reviewed in the rest of this chapter.

Over the past three decades, more than 30 building evacuation models have been developed and in some cases corresponding simulation tools have been implemented to reproduce people's movement patterns [Ku08]. Some of the models have never been fully implemented and some are not longer in use [KP05] [Ku08]. This chapter will mainly look at the publically available models. Evacuation applications adopt various modelling approaches to simulate the egress of occupants from a building. There are 18 evacuation models (see Table 2.2.1) which will be considered in this section. To assess and compare building evacuation models, there are several reviews of building evacuation models available [KP05] [Ca07] [Ku08]. In this section, the models will be described according to the techniques that are employed which relate to the objectives of this thesis.

Evacuation models are then categorized according to the technique used to represent the enclosure. There exist three key methods that can be used to represent the enclosure: fine network, coarse network, and continues network. The detail of the enclosure representations will be described in Section 2.2.2.

Table 2.2.1: the evacuation models categorized by the enclosure representation

| | |
|---|---|
| **MODELS with coarse network representation** | EVACNET4[KF85][TR96][KF98], WAYOUT[SG94], TIMTEX[Ha96], EXITT[Le89][Le87], E-SCAPE[Re96], ALLSAFE[HM98], EESCAPE[Ke95] |
| **MODELS with fine network representation** | buildingEXODUS, STEPS[HH97][HH98][Ma03], PathFinder[Ca00], EGRESS2002[KC94][Ke95][AE02], CRISP[BF98][Fr01][Fr03], PedGo[KM03] [MK01] |
| **MODELS with continuous network representation** | Simulex[TM94][TM95][TW96], Legion[BB07], FDS+Evac[KH07][KH09], GridFlow[BP02], ASERI[Sc01] [SK01] |

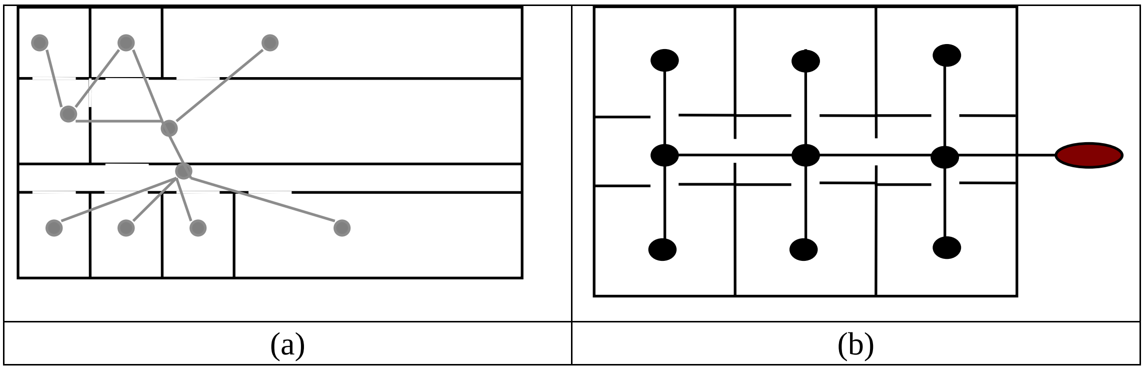
## 2.2.2 Enclosure Representations

The representation of the enclosure is a very important facet of a computer evacuation model, as it directly influences the methods and algorithms utilised in the simulation of the population. To complete the evacuation simulation process, the user must first configure the building representation within the model. Currently, there are three kinds of representation of the enclosure: fine network, coarse network and continues network models (see Table 2.2.1). Coarse network models divide the floor plan into rooms, corridors, stair sections, etc. Fine network models divide a floor plan into a number of (typically uniform) small grid cells that the occupants move to and from. A continuous network applies a continuous (co-ordinate) space to the floor plans of the structure, allowing the occupants to walk from one point in space to another throughout the building. This section will give the detail of these methods.

### 2.2.2.1 Coarse network

The first kind of representation method of the enclosure to be considered is the coarse network. A coarse network segments a structure into a set of compartments which form a graph connected by arcs, so that people travel from node to node; for example, EVACNET4, WAYOUT, and TIMTEX (see Table 2.2.1). These nodes represent rooms, workplaces, hallway, stairwell, lobby, refuge areas etc, irrespective of their physical properties. In a coarse network the nodal mesh is often non-uniform in relation to capacity. Nodes are connected by arcs that represent the actual connectivity within the structure. In a coarse network model occupants move from segment to segment, and their precise position is less defined than in the fine network models. An occupant might therefore move from room to room instead of from one area inside a room, to another.

The Figure 2.2.1(a) and 2.2.1 (b) give two simple coarse graph representations of simple structures. The Figure 2.2.1(a) is an E-SCAPE [Re96] representation of simple structure. Figure 2.2.1 (b) is a buildingEXODUS coarse node model representation of simple structure [VL09].

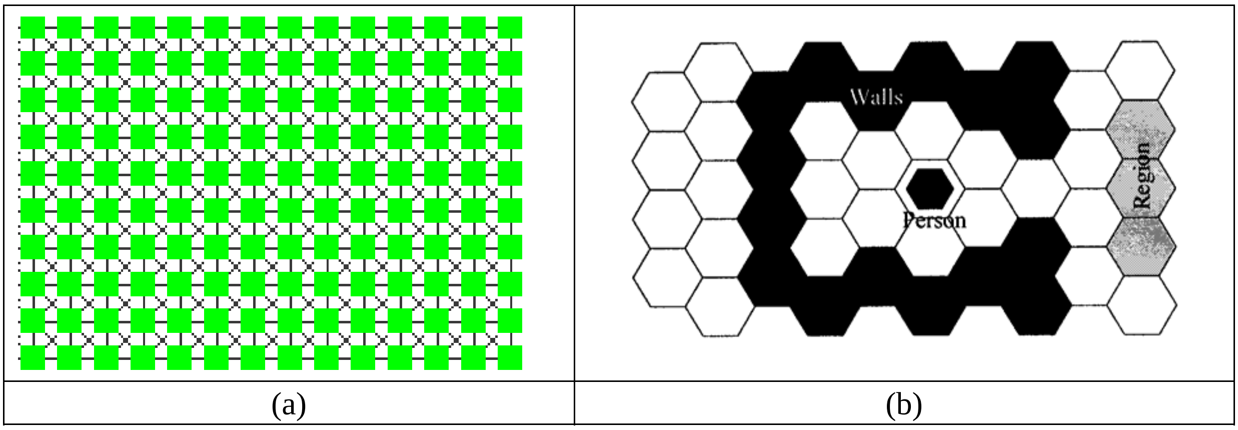


(a) (b)

Figure 2.2.1: (a) E-Scape representation of simple structure [Re96] (b) buildingEXODUS coarse graph representation of simple structure.

### 2.2.2.2 Fine network

The next kind of representation of an enclosure to be considered is the fine network. In the fine network approaches, the entire floor space of the enclosure is usually covered in a collection of nodes. The fine network will include a larger number of nodes to mesh the enclosure comparing to the coarse network. The size and shape of a node varies from model to model, however is typically uniform within a model. For example buildingEXODUS typically uses 0.5m x 0.5m square nodes (see Figure 2.2.2(a)), while EGRESS [AE02] uses hexagonal nodes (see Figure 2.2.2(b)), of sufficient size to cater for a single occupant. A large geometry may be made up of thousands of nodes and each compartment within the geometry, may be made up of many nodes. In this way, it is possible to accurately represent the geometry, and its internal obstacles, and accurately locate each individual at any time during the evacuation.



(a) (b)

Figure 2.2.2: (a) buildingEXODUS fine network representation showing nodes and arcs (b) EGRESS fine network representation [AE02]

#### 2.2.2.3 Continues network

Finally, the last kind of representation of the enclosure is a continuous network. In fact, the continuous network approach may also include a node mesh. However，these nodes are only used for navigations [TW96] [DP02]. The distinction between the fine network and continuous network approaches is that the continuous network has a more precise representation of the building layout with people movement charted at a much more refined level instead of being locked to a nodal grid. Continuous networks also have the ability of more accurately simulating the movement of pedestrians within the enclosure; e.g. navigating around obstacles, etc. However, the major limitation of continuous network approach suffers from relative poor computational performance，i.e. they can be very computationally expensive. Another limitation of a continuous network method is that the occupants can get ‘stuck’ at pinch points, due to congestion. If this occurs, further adjustments may be required to the model’s geometry [KM04]. Thus, few computer evacuation models have adopted the continuous space approach (see Table 2.2.1).

### 2.2.3 Occupant Representations

In an evacuation model people can be represented, within the structure, either at individual or global level (see Table2.1.2). These two types of representation will be discussed in the next two sections.

Table 2.2.2: the evacuation models categorized by the occupant representation

| | |
|---|---|
| **MODELS which represent occupants at a Global level** | EVACNET4[KF85][TR96][KF98], WAYOUT[SG94], TIMTEX[Ha96] , ALLSAFE[HM98], EESCAPE[Ke95] |
| **MODELS which represent occupants at an individual level** | buildingEXODUS, STEPS[HH97][HH98][Ma03], FDS+Evac[KH07] [KH09], PathFinder[Ca00], EGRESS2002 [KC94][Ke95][AE02], CRISP[BF98][Fr01][Fr03], Legion[BB07], PedGo[KM03][MK01], Simulex[TM94][TM95][TW96], GridFlow[BP02], ASERI[Sc01] [SK01], EXITT [Le89] [Le87], E-SCAPE [Re96] |

**Individual level to represent the occupants**:

When the evacuation model has an individual view of the occupants, the characteristics of each occupant or group of occupants can be assigned by the user. These attributes are the basis for the movement and decision-making processes of each occupant. In addition, the model can track the movement of individuals throughout the simulation and can give information about those individuals current status [KP05] [Ca07].

**Global level to represent the occupants:**

If the evacuation model considers occupants at the global level, the model sees its occupants in terms of homogeneous groups of people moving towards the exits. The characteristics of each group/sub-population of occupants can be assigned by the user. However, the average attributes are applied to a group/sub-population as a whole. The output from a global level representation is often just the number of occupants that have exited the enclosure after a certain time. These models normally do not provide information on where and when specific occupants exit an enclosure, since they do not track individual movement [Ca07], [Ku08].

## 2.2.4 Behavioural Perspective of Occupants

The behaviour of the occupants is represented in many different ways by building evacuation models. In general, the various approaches of simulating occupant behaviour can be separated into the following four categories [KP05] [Ca07] (see Table 2.2.3).

Table 2.2.3: the evacuation models categorized by the occupant representation

| | |
|---|---|
| **No behaviour** | EVACNET4[KF85][TR96][KF98], WAYOUT[SG94], TIMTEX[Ha96], PathFinder[Ca00], EESCAPE[Ke95] |
| **Implicit behaviour** | ALLSAFE[HM98], GridFlow[BP02], PedGo[KM03][MK01], Simulex[TM94][TM95][TW96] |
| **Rule-based behaviour** | buildingEXODUS, EGRESS2002[KC94][Ke95][AE02], CRISP[BF98][Fr01][Fr03], EXITT[Le89][Le87], E-SCAPE[Re96], ASERI[Sc01] [SK01], STEPS[HH97][HH98][Ma03] |
| **Artificial Intelligence behaviour** | Legion [BB07], FDS+Evac[KH07] [KH09] |

**No behaviour:**

This type of evacuation models does not attempt to simulate the behavioural of occupants, with most of them relying completely upon equations [NM02] to represent time/flow rate between and over compartments to simulating occupant movement in order to simulate the evacuation, for example, EVACNET4, WAYOUT, TIMTEX and EESCAPE. These *no behaviour* evacuation models consider occupant's speed reductions based on the density of the space and the capacity of the doors and stairways, for example, PathFinder.

**Implicit behaviour:**

A building evacuation model of this type does not explicitly specify the behaviour of occupants. They attempt to model behaviour implicitly by assigning certain response delays or occupant characteristics that affect movement throughout the evacuation [KP05] (see Table 2.2.3). For example, ALLSAFE includes many data inputs, such as, background noise, social and economic barriers among the occupants, language, and fire scenarios. These input data affect the evacuation time by adding or subtracting the time intervals extracted from a database within the model [HM98].

**Rule-based behaviour:**

This type of evacuation models explicitly considers the behavioural traits of individual occupants, attempting to simulate occupant decision making according to predefined rules or responses. The evacuation process of an occupant will typically be influenced by many factors, for example: obstacles, conflicts, alarm signals, smoke, prior knowledge of the structure, etc. these factors will influence the occupant’s route choice and walking speed, etc [KP05] [Ca07].

**Artificial Intelligence behaviour:**

This type of models implement artificial intelligence in an attempt simulates human behaviour or an approximation of human-behaviour during an evacuation [KP05].

## 2.2.5 Wayfinding

Human wayfinding research investigates how people find their way in the physical world [MM97]. Within the building environment, Veeraswamy et al. [VL09] described wayfinding as the process by which an individual located within a complex enclosure decides on a path or route in order to reach a goal location. Within the building evacuation context, Veeraswamy et al. describes wayfinding as the process in which an individual attempts to find a path which leads them to relative safety, usually the exterior of the enclosure. In Veeraswamy et al. [VL09], the criteria used by occupants in wayfinding are described as; total distance, total time, total number of turns, longest leg first, angle of turns and total number of decision points. They call this collection of criteria the ‘building wayfinding criteria’. According to these ‘building wayfinding criteria’ A cost function has been calculated which is associated with each path. The cost function is determined by taking a weighted sum of the normalised route preference criteria associated with an occupant. Then an occupant will choose the minimum cost path to evacuate from a building.

Golledge defined wayfinding as people’s cognitive and behavioural abilities to find a way from an origin to a destination; wayfinding is a purposive, directed and motivated activity [Go99]. Humans use different wayfinding strategies depending both on their own individual spatial awareness, and also on their knowledge of the environment they are travelling through [Wi05]. Wayfinding models include global movement patterns and local movement patterns. A global movement pattern is where agents make a decision between various routes within the structure. Where as a local movement pattern only considers local information, such as which exit to select in a given room or compartment.

Garbrecht [Ga71] [Ga73] studied the differences between local *(random walk)* and global movement patterns (*random path)* selection strategies. Random walk describes a movement in a labyrinth where a person makes a choice-randomly at each intersection. Hence, *random walk* is local movement behaviour. However, *random path* refers to an initial choice of a complete path from origin to destination randomly. *Random path* is a global movement pattern. Garbrecht also showed that these two ways of route selection lead to different paths taken by the occupants.

Wayfinding in buildings can be an extremely complex task, which can be confused or assisted by numerous factors, such as the visual access of the structure, any signage and its positions and messages, the presence of structural makers, the connectivity of the spaces within the structure itself, etc[We81] [RB01][MM98].

Normally, a complex building does not supply occupants with all the information required to perform an optimal global route choice. The occupants do not have enough time to establish a complete cognitive map of the building. In such situation, Gunnar [Gu99] presented some local movement pattern models of wayfinding behaviour, ranging from the simplest random walk models to more complex shortest path rules. Gunnar utilised the EvacSim [Po94] simulation program to study the impact of various movement patterns. However, the findings of this research cannot be verified since EvacSim is not generally available.

The SGEM [LF00] [LF04] evacuation model simulated individual behaviour using a wayfinding function. The wayfinding function is affected by occupant's characteristics (age, gender, patience, etc) and the environmental stimuli (obstacles, conflicts, smoke, alarm signals, etc.). This wayfinding function ultimately adjusts the movement direction of occupant's depending on obstacles and conflicts and the individual's speed. In 2006, Lo et al. [SH06] presented a game theory based exit choice model for evacuation. It has also been integrated in The SGEM evacuation model. However, SGEM is also not publicly available.

In most of the building evacuation modelling tools mentioned in this section, the wayfinding process has been ignored. The two evacuation models, EvacSim and SGEM, which do considered the wayfinding process, are not publicly available. On the whole, most building evacuation models assume that the occupants have complete knowledge of the building structure and simply direct people to the nearest exit. Some models may assume that a proportion of the occupants have partial knowledge of the structure and direct people to nearest known exit. Some more sophisticated models may model occupants which are not completely familiar with the structure and include feature, such as signage, to direct people to an exit.

### 2.2.6 Conclusions

Building evacuation models can provide an estimate of the time required to evacuate a building layout for a prescribed scenario. The scenario includes factors such as the number, nature and distribution of the population, response times of the population, exit availability, etc. By comparing evacuation times for a given scenario, various building design options can be compared and assessed by building designer. The wayfinding capability of occupants is often ignored in most building evacuation models. In general these models simply direct people to the nearest exit. Others may direct people to nearest known exit, while some more sophisticated models may also include factors such as signage to direct people to an exit. The evacuation modelling approach does not address the more fundamental notion of building complexity for a given building. Any multi-compartment building may be regarded as a complex object or structure. However, evacuation models do not provide an understanding of why one building is better than another for evacuation, they only provide an overall evacuation time. Therefore, it would be desirable to have a measure of building complexity which provides some insight into how difficult a building layout was from an evacuee’s point of view. Ideally, this measure would be scenario independent and readily calculable. Such a measure would provide some insight into how easy it would be to evacuate from the building. Taking the evacuation time, as simulated by an evacuation models, in conjunction with a building complexity measure would provide a more complete insight into the evacuation suitability of buildings.

## 2.3 buildingEXODUS

The buildingEXODUS evacuation model has been chosen as a platform to compare the building complexity measures developed in this thesis to evacuation simulation results generated by an evacuation model. It is widely used model which is familiar to the author. In addition the buildingEXODUS model can also generate a coarse network graph, which can be used for building complexity analysis, from the fine mesh it utilizes for evacuation analysis.

## 2.3.1 EXODUS Overview

EXODUS [EM96, ME95, EM94, ME96, EP99, ER95, EJ93, GG98, OG98, GG00, GG99] is a suite of software tools designed to simulate the evacuation of large numbers of people from a variety of complex enclosures. It was developed by the Fire Safety Engineering Group of the University of Greenwich. The EXODUS family of evacuation models consists of buildingEXODUS, airEXODUS and maritimeEXODUS.

BuildingEXODUS is an evacuation modelling package used to simulate the evacuation of large numbers of people from complex structures. This software is designed for applications in the built environment and is suitable for application to supermarkets, hospitals, cinemas, rail stations, airport terminals, high rise buildings, schools etc. buildingEXODUS can be used to demonstrate compliance with building codes, evaluate the evacuation capabilities of all types of structures and investigate population movement efficiencies within structures.

The buildingEXODUS model comprises five core interacting sub-models: the OCCUPANT, MOVEMENT, BEHAVIOUR, TOXICITY and HAZARD sub-models (see Figure 2.3.1). These sub-models operate on a region of space defined by the GEOMETRY of the enclosure. The buildingEXODUS software has been written in C++ using Object Orientated techniques and rule-base concepts to control the simulation. The software is rule-based, with the progressive motion and behaviour of each individual being determined by a set of heuristics or rules. Architectural plans can be loaded straight in to the simulation suite to represent the structure or the user may avail themselves of a number of interactive design tools. On the basis of an individual's personal attributes, the Behaviour Sub-model determines the occupant’s response to the current situation, and passes its decision on to the Movement Sub-model.

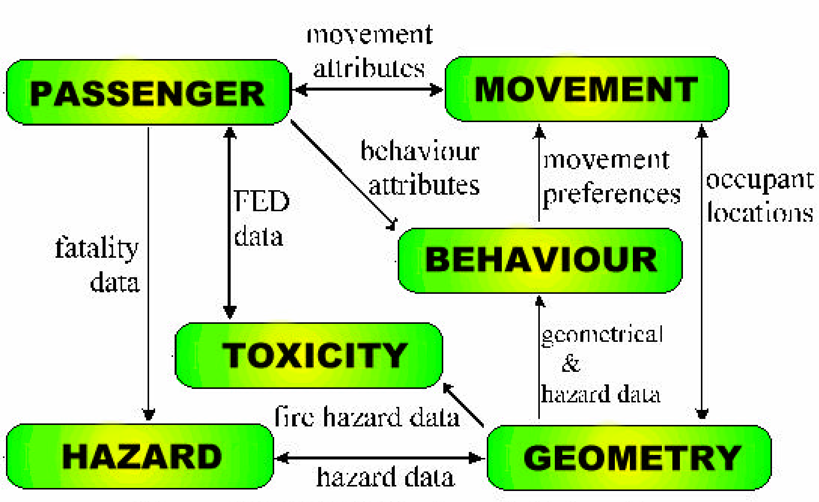


Figure 2.3.1: buildingEXODUS sub-model interaction

In the currently version of buildingEXODUS, The GEOMETRY of the enclosure can be defined in three ways. Firstly, it can be read from a geometry library. Secondly, it can be constructed interactively using the tools provided. Lastly, it also can read from a Computer-Aided Design (CAD) drawing using the Drawing Exchange Format (DXF) format [HtEn]. Internally the entire space of the geometry is covered in a mesh of nodes that are typically spaced at 0.5m intervals. The nodes are then linked by a system of arcs. Each node represents a region of space typically occupied by a single occupant.

The OCCUPANT SUB-MODEL describes an individual as a collection of defining attributes and variables which broadly fall into four categories: physical (such as gender, age, weight, agility etc), psychological (such as response time, patience, drive etc), positional (such as travel distance, Personal Evacuation Time (PET) etc) and hazard effects (such as an individual cumulative exposure to radiative and convective heat (FIN) etc). Some of the attributes are fixed throughout the simulation while others are dynamic, changing as a result of inputs from the other sub-models.

The MOVEMENT SUB-MODEL is concerned with the physical movement of individual occupants from their current position to the most suitable neighbouring location, or supervises the waiting period if one does not exist. The movement may involve such behaviour as overtaking, sidestepping, or other evasive actions.

The BEHAVIOUR SUB-MODEL determines an individual's response to the current prevailing situation on the basis of his/her personal attributes, and passes its decision on to the movement sub-model. The behaviour sub-model functions on two levels: global and local. The local behaviour determines an individual’s response to his/her local situation while the global behaviour represents the overall strategy employed by the individual. This may include such behaviour as, exit via the nearest serviceable exit or exit via most familiar exit.

The HAZARD SUB-MODEL controls the atmospheric and physical environment and allows the user to specify the specific simulation scenario. It distributes pre-determined fire hazards such as heat, smoke and toxic products throughout the atmosphere and controls the opening and closing of exits and the availability of exits.

The TOXICITY SUB-MODEL functions only when the fire hazards are present. It determines the effects on an individual exposed to toxic products distributed by the hazard sub-model. These effects are communicated to the behaviour sub-model which, in turn, feeds through to the movement of the individual.
More details about these SUB-MODELS can be found [EM96] [EP99] [ER95] [GG98] [GG00].

## 2.3.2 Enclosure Description

Within buildingEXODUS, the enclosure GEOMETRY is represented as two-dimensional grids. As described in section 2.3.1, the GEOMETRY can be defined in three ways. It can be (i) read from a geometry library, (ii) constructed interactively using the tools provided or (iii) read from a CAD drawing using the DXF format.

Each location on a grid is called a *node*. Internally the entire space of the geometry is covered in a mesh of nodes that are typically spaced at 0.5m intervals. Each node represents a region of space typically occupied by a single occupant. Nodes are linked to its nearest neighbours by a number of *arcs* (see Figure 2.3.3). There is no limit to the number of arcs emanating from a node and all nodes need not possess the same number of arcs. Typically, a node will possess four or eight arcs. Occupants travel from node to node along the arcs.

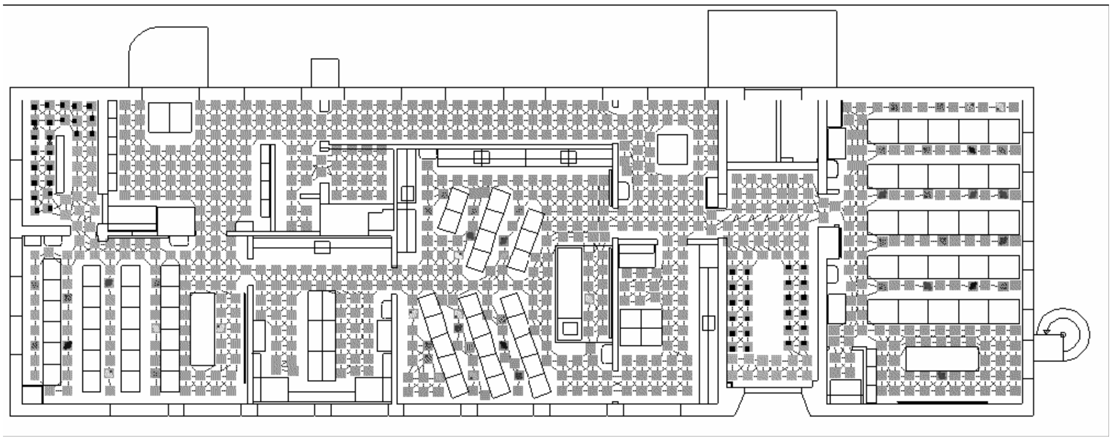


Figure 2.3.3: Outline of a building showing nodes and arcs

Associated with each node is a set of attributes that are used to define the node's terrain type, environmental state and location. There are 12 terrain types in buildingEXODUS. These are STAIRS, LANDING, SEATS, EXTERNAL EXITS, INTERNAL EXITS, FREE-SPACE, CENSUS REGIONS, BOUNDARY, ATTRACTOR, REDIRECTION, SOURCE, DIRECTION and DISCHARGE. The nature of the terrain type will influence the behaviour and maximum travel speed of the occupant passing over the node.

The environmental states include: concentration of HCN (ppm), CO (ppm), $CO_2$ (%), oxygen depletion (%), smoke (l/m), temperature (degree C), HCL (ppm), HBr (ppm), HF (ppm), SO2 (ppm), NO2 (ppm), Acrolein (ppm), Formaldehyde (ppm) and

Radiative Flux (kW/m$_2$). With the exception of Radiative Flux, for each of these variables, two values are stored, representing the value at head height and near floor level.

Each node possesses an attribute known as the *Potential*. The *Potential* is a measure of the node's distance from the nearest exit. *Potential*s are *grown* from each exit and increase with each step from the seed exit. Once the geometry has been constructed and the exits defined, buildingEXODUS will automatically create the potential map.

Associated with each arc are two attributes. The first attribute is the length attribute. This represents the actual physical distance between nodes. In most cases this distance is typically around 0.5m. The second attribute is known as the *Obstacle*. The *Obstacle* attribute is an integer measure of the degree of difficulty in passing over the node. Nodes representing open space are linked with arcs which have an obstacle value of 0, while nodes littered with debris may have a higher obstacle value of 1 or 2.

## 2.3.3 Escape strategy

The global behaviour includes two aspects within buildingEXODUS. The first involves occupants implementing an escape strategy that leads to their direct escape. The second aspect involves occupants completing a set of tasks prior to their evacuation [EP99] [GG98]. In the first aspect of global behaviour, the occupants implement an escape strategy that leads them to exit via their nearest available exit, an assigned exit, or an exit based on their knowledge of the structure. There are several methods to implement these strategies [EP99] [GG98]. This default method is used to direct occupants to their nearest serviceable exit or most familiar exit. This strategy is achieved via the use of the node potentials and biased exit-potentials. The node potential is a measure of the distance from the node to the most attractive exit point. This method can be implemented by using a potential map, which is grown from available exits, such that, in an unbiased system, people will tend to evacuate towards the nearest available exit. The second aspect of *global* behaviour enables the occupants to be attributed with short-term tasks that are completed prior to their exit. This is based on the fulfilment of procedural tasks that are often required of occupants during an evacuation, or events that may occur prior to the commencement of an evacuation that may influence the outcome of an evacuation.

### 2.3.4 Advanced features for graph generating

Within buildingEXODUS, the enclosure geometry is normally represented as two-dimensional grids as introduced in Section 2.3.2. This model can also generate a coarse graph representation in addition to the fine node mesh as applied during the evacuation simulation process. Currently, the buildingEXODUS model can generate two types of graph representation for a building layout. The first type of graph is the room-based graph (see Chapter 3). This model defines a suit of compartments for a building as the nodes of the graph. These compartments represent rooms, workplaces, hallway, stairwell, lobby, refuge area etc. Normally, the arcs represent the connection between the nodes (see Figure 2.2.1(b)). The second type of graph representation is called fine node route-based graph (see Chapter 6.5). Basically, the fine node route-based graph generated in buildingEXODUS is based on the fine mesh geometry. Currently this kind of graph includes four types of nodes which are internal door, external door, waypoint node, and a room node (see Figure 6.5.3 of Chapter 6**)**. These two types of graph output in buildingEXODUS will be utilized in the building complexity analysis developed in this thesis.

### 2.3.5 Limitations of buildingEXODUS and Conclusions

This section has given an overview of the buildingEXODUS evacuation model, and mainly focuses on the following aspects:

(1) The descriptions of the fine mesh representation for building layout within buildingEXODUS.

(2) Two types of graph representation for a building layout which can be generated by the buildingEXODUS model.

(3) The escape strategy that leads to occupant’s direct escape

The buildingEXODUS evacuation modelling can provide an estimate of the time required to evacuate a building layout for a prescribed scenario. As is the case for all evacuation models, buildingEXODUS does not address the concept of building complexity. buildingEXODUS cannot provide a measure for why one building is better than another for evacuation, whereas a building complexity measure can provide some insight into how easy it would be to evacuate from that structure. Hence taking these two methods in conjunction, building complexity measure and an evacuation time, can provide a more complete insight into the suitability of a structure in regards to an evacuation viewpoint.

# Chapter 3 Review of graph techniques

## 3.1 Introduction

This chapter will present the current state of the art with respect to different graph representations for a building and various graph measures and techniques. By investigating the current graph measures and techniques for the graph representation of a building, the following questions should be answered:

1. What are the current techniques to represent a building as a graph?
2. What are the different types of measures for the graph representations of a building?
3. Can these measures be used to compare buildings in regard to evacuation capability?
4. What is the best measure which can be used to compare buildings for complexity?

For comparing different buildings with respect to their evacuation ability, this research will focus on generating graph measures based on the graph representation of buildings. Then these measures can be used to compare different building for the purpose of evacuation ability. When generating such a complexity measure for a building layout, there are two factors that need to be considered. The first is the representation of the building structure. This means how we choose a kind of graph which is suitable to describe the building layout. Second, a set of measures need to be generated to represent the building complexity based on the suitable graph representation of the building. In this chapter, the graph representation techniques of the building architecture and the graph measures with respect to these graphs will be introduced.

## 3.2 Graph Representations for buildings

Graph-based techniques have been applied to architectural analysis for over 30 years. March et al. [MS71] and Tabor [Ta76a] [Ta76b] have analysed the floor plan designs, the former in the terms of an electrical network analogy in order to generate systematically floor plans described as mosaics of rectangles, they examined the relationship between rooms in buildings; the later analysing communication and route patterns in terms of circulation cost based on an Euclidean metric or time dimension [MS71] [Ta76a] [Ta76b]. Steadman [St73] demonstrated how a graph may be constructed to describe an architectural arrangement that considers relationships between architectural units (such as rooms or corridors), although the original concept dates back further [MS71]. Kruger [Kr79] shows how similar graphs of the relationships between urban units may be constructed. Hillier and Hanson [HH83] [HH84] extend these findings to a new form of description, they introduced the visibility relationships into graph analysis of buildings, and constructed the set of axial lines for a building, which are the fewest longest lines of sight and access in the building which traverse all the convex spaces within the building system.

For the purpose of evacuation simulation and the architectural analysis, most coarse graph representations of buildings have been used in most evacuation models and the egress complexity analysed ( [DP94] [Fa91] [BC95] [Le87] [PB94]). These graph representations were mainly based on the following ideas: they defined a suit of compartments for the building as the nodes of the graph. These compartments represented rooms such as workplaces, hallways, stairwells, lobbies, refuge areas, etc. Normally, the edges are the distances between the nodes. We call this kind of graph a ‘room-based’ graph. This graph can be used for architectural analysis of a building.

Another graph representation of buildings is the visibility graph. The term visibility graph was introduced to landscape analysis by De Floriani et al. [DM94], it is also used in computational geometry and artificial intelligence [DK97]. Both of these forms of visibility graph are more sparsely connected, as they include only key locations in the environment. For example in landscapes, points are selected from a Triangulated Irregular Network (TIN), whereas in computational geometry points are selected from corners of two-dimensional polygons. However, the architectural visibility graph also uses selected locations as vertices and a mutual visibility relationship to form edges. Turner et al. [TD01] used a set of isovists to generate a graph of mutual visibility between locations. And they demonstrated that this graph could also be constructed without reference to isovists and that they are in fact invoking the more general concept of a visibility graph. Using the visibility graph, they generated a new methodology for the investigation of configurationally relationships of the architectural space.

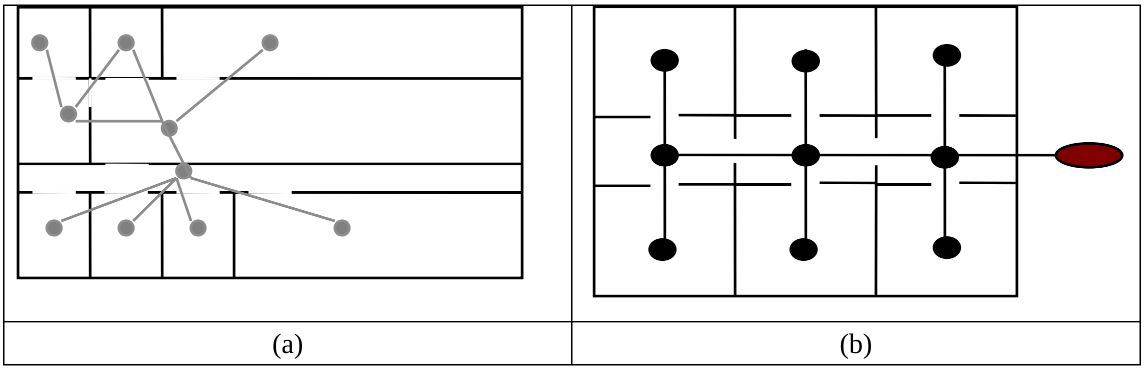
The popular graph representation methods for a building mainly include the following kinds of graphs described as above: room-based graph, axial map, and the visibility graph.

In the evacuation simulation model, all the coarse graph representations of the building are based on a room graph. This kind of graph usually cannot describe fully the routes taken by evacuees. So another kind of graph, called a ‘route-based’ graph, is another important representation of the architectural space. The concept of a route is often defined in the literature [SB00] [HW99] [Ku00] as a sequence of decision points that are connected by segments. Depending on the problem domain, this general definition can be instantiated for different real world scenarios, such as railroad connections between large cities, paths in a park, or corridors in an office building. The concept of ‘route graphs’ has been introduced by Werner [WB00], used for navigation with respect to several scenarios such as humans using railway, underground, road or street networks. However, there is no mention of this method being applied to buildings. The route-based graph for a given building should describe all the routes available for evacuees; see Chapter 6 for more detail.

## 3.2.1 Room Graph

### 3.2.1.1 Room Graph Representation

For a 2D space, a kind of coarse graph representation of a given building has been broadly applied to the evacuation models areas. As shown in Chapter 2, some evacuation models have applied the coarse graph as a kind graph representation method for the building [DP94] [Fa91] [BC95] [Le87] [PB94]. Basically, they defined a suit of compartments for the building as the nodes of the graph. These compartments represent rooms, workplaces, hallways, stairwells, lobbies, refuge areas, etc. Normally, the edges are the distances between the nodes. The Figure 3.2.1.1(a) and 3.2.1.1(b) give two simple room-based graph representations of simple structures. The Figure 3.2.1.1(a) is an E-Scape [Re96] representation of simple structure. Figure 3.2.1.1(b) is a buildingEXODUS coarse node model representation of simple structure [VL09].



(a) (b)

Figure 3.2.1.1: (a) E-Scape representation of simple structure [Re96], (b) buildingEXODUS representation of simple structure.

### 3.2.1.2 Room Graph Generation

In order to take advantage of the topological structure of a room-based environment, the first step is to discern the topology of the environment. One way to represent this topology is in a graph with weighted edges, where the nodes represent the structure compartments of the buildings. These compartments represent compartments such as rooms, workplaces, hallway, stairwell, lobby, refuge area. The edges connect two compartments which are adjacent, and the weights on the edges represent distances between adjacent nodes.  While more complicated techniques may be employed to approximate path cost between two adjacent compartments. Normally, Euclidean distance is applied to represent the path cost. In most situations, the room-based graph can just specify the connectivity between compartments with in the building. It cannot express the real environment of the building exactly, also the edges in the room-based graph cannot point out the real route the evacuee needs to travel. The detail of the limitations will be introduced in Chapter 6.

## 3.2.2 Visibility graph

### 3.2.2.1 Visibility graph Representation

The normal concept of the visibility graph is the set of non crossing line segments in the plane. This can be traced back to 1979 [LW79], Lozano-Perez and Wesley used the visibility graphs as an approach to shortest path problems in the plane. This kind of graph was described as follows: Let S define the set of n non-intersecting line segments in the plane. The visibility graph G(s) of S is the graph that has the endpoints of the segments in S as nodes and in which two nodes are adjacent wherever they can see each other [ME88].

The important thing in the visibility graph is how to choose the vertices. In the visibility graph used for landscape analysis [DM94], the vertices of the graph were selected from a triangulated irregular network. In computational geometry and artificial intelligence [DK97], the vertices of the visibility graph were selected from the corners of a set of simple polygonals, the visibility graph was defined as follows: Let S be the set of simple polygonal obstacles in the plane, then the nodes of the visibility graph of S are the vertices of S, and there is visibility edge between vertices

V and W if these vertices are mutually visible. These forms of visibility graphs are more sparsely connected compared to the visibility graph described in next paragraph, as they include only key locations in the environment.

A methodology for the analysis of architectural space was generated by Turner et al. [TD01]. They applied a set of isovists to generate a graph of mutual visibility between locations. The concept of an isovist has been applied in both architecture and geography, as well as mathematics for over 40 years. As one of the originators of the term 'isovist', Tandy presents isovists as a method of 'taking away from the [architectural or landscape] site a permanent record of what would otherwise be dependent on either memory or upon an unwieldy number of annotated photographs' [Ta67]. In the area of architectural space analysis, Turner et al. described an isovist as the area in a spatial environment directly visible from a location within the space [TD01]. Currently, isovist have been mainly used to construct visibility graphs or the axial maps. Turner et al. generated a visibility graph by constructing an isovist graph and also demonstrated that this graph could also be constructed without using isovists. In fact they defined the general concept of a visibility graph. According to this visibility graph, they generated a new methodology for the investigation of configurational relationships. The technique applied by Turner et al. is similar to those used in landscape analysis by De Floriani et al. [DM94], or in computation geometry [DK97]. They just choose the key locations as the nodes of the visibility graph. In this visibility graph, they take a grid of many of points across the space rather than selecting a few key locations. Certainly, the architectural visibility graph also applied the selected locations as vertices and a mutual visibility relationship to form edges, thus all the graphs are of identical form.

### 3.2.2.2 Visibility graph Generation

For generating a visibility graph of a set of n non-intersecting line segments in the plane, a lot of methods can be used. For example, the first efficient algorithm were generated by Lee [Le79] and Sharir and Schorr [SS84]. Welzl and Asano et al. designed algorithms to generate visibility graphs [We85] [AG86]. In 1987, Ghosh and Mount proposed an 'optimal' algorithm [GM87]. Overmars and Welzl introduce two other methods [ME88] and they described their methods as being simple to implement.

However, for the purpose of analysing the architectural space, the architectural visibility graph is the main visibility graph representation introduced here. This kind of graph can be generated by constructing an isovist graph. Isovists have particular relevance to architectural analysis, as noted by Turner et al. [TD01]:

*The appeal of the concept is that isovists are an intuitively attractive way of thinking about a spatial environment, because they provide a description of the space `from inside', from the point of view of individuals, as they perceive it, interact with it, and move through it.*

Early in 1979, Benedikt introduced a set of measurements of isovist properties which can be applied to achieve quantitative descriptions of spatial environments. He started by considering the volume visible from a location and then simplified this representation by taking a horizontal slice through the `isovist polyhedron'. The isovists are always single polygons without holes, as shown in Figure 3.2.2.1. He defined an 'isovist field' of his measurements. Isovist fields record a single isovist property for all locations in a configuration by applying contours to point out the way those features vary through space.

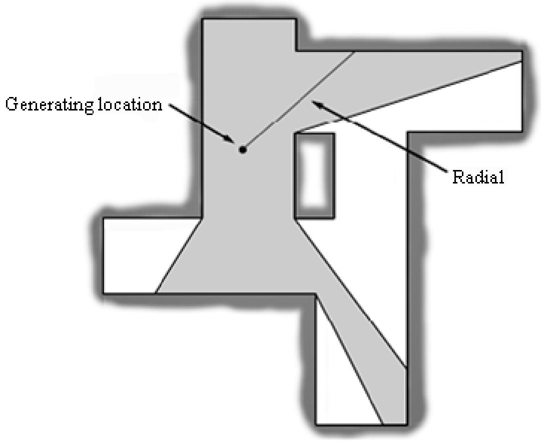
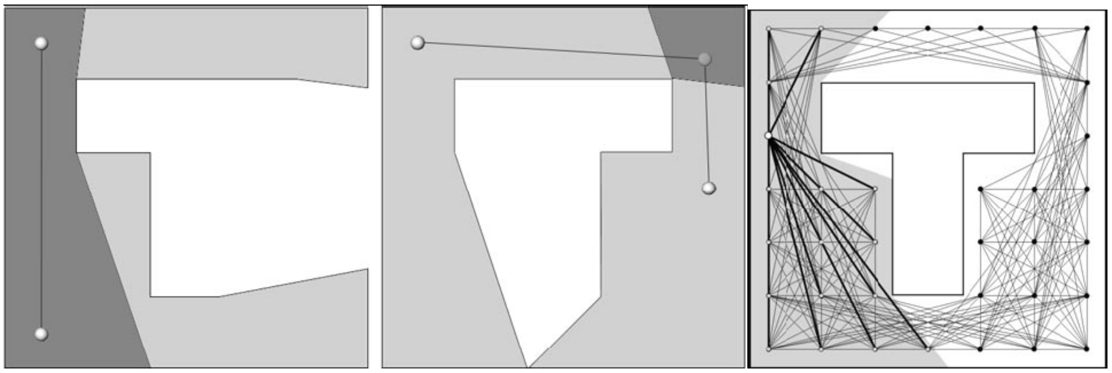


Figure3.2.2.1: An isovist polygon, incorporating the visible area from a generating location [Be79] [TD01]

However, in the Benedikt method, the applications of the isovist in architectural analysis are limited to a small number of studies. Benedikt just indicated local properties of space, and ignored the visual relationship between the current location and the whole spatial environment. He also did not give any propositions about how to interpret usefully the result of his isovist measures.

In 2001, Turner et al. introduced a methodology which included how visual characteristics at locations are related. Firstly, they aimed at constructing an isovist graph for a spatial environment. For constructing such a graph, firstly they needed to choose two different sets. The first is to select a set of isovists. In fact they only need to select an appropriate set of generating locations of isovosts according to some suitable rule. The selected generating locations then are used to form the vertices of the isovists graph. The other is to determine which relations between these isovists are important, then applying them to form edges in the graph. Ideally, they would like to select some set of isovists that `fully describes' the spatial system. Considering the computational complexity, in practice they must compromise and try to select a set of generating locations that provides an acceptable ‘near-full’ description of the space. In fact, Turner et al. applied regularly spaced intervals (one metre) to generate isovists throughout a spatial system. Once a set of generating locations of isovists have been selected, the following step is to consider which relationships between different isovists in an environment should be included in an isovist graph. They defined two kinds of relations between two isovists. Two isovists are intersected and their generating locations are mutually visible. They referred to this as a first-order relationship (Figure 3.2.2.1(a)). They also defined another relationship, called a second-order visibility relationship (3.2.2.1 (b)). In fact, for determining only the first-order relationship, isovists are not required at all. A graph can be made with physical locations as vertices, and form edge connections between pairs of locations if they are mutually visible. If an isovists graph is generated by just including first-order relationship. This produced a visibility graph of the system (Figure 3.2.2.1 (c)).



| (a) | (b) | (c) |
|---|---|---|

Figure 3.2.2.2 [TD01]: (a) First-order and (b) Second-order visibility relationships between isovists (c) An example of a first-order visibility graph, showing the pattern of connections for a simple configuration

The visibility graph of a spatial system is generated for the analysis of architectural space. In a visibility graph, some locations will be selected as the vertices of the visibility graph that can fully describe the spatial system. In practice a set of locations will be selected which provides an acceptable near-full description of the space. So the key factor for generating a visibility graph is how to select the vertices of the graph. Different users will generate different visibility graphs for the same architectural space.

## 3.2.3 Axial map

### 3.2.3.1 Axial map Representation

The concept of an axial map was first introduced by Hillier and Hanson in 1984 [HH84]. The axial map is one of the primary tools of space syntax, which provides a method for partitioning a spatial system into relatively independent but connected subspaces. They defined an axial map of the open space structure of a settlement as the least set of axial lines which pass through each convex space and make all axial links [HH84]. In their definition, an axial line refers to the longest line that can be drawn through an arbitrary point in the spatial configuration and convex space is a ‘fully fat’ convex polygon around a point (see Figure 3.2.3.1). In use, the axial map is connected into a graph so that the lines are represented as nodes, and the intersections of lines as connections between the nodes. From the axial map, we can get information of whether nodes are important in terms of their relative nearness or accessibility. The axial map is mainly used to analysis large-scale architectural space, i.e. town centres and cities.

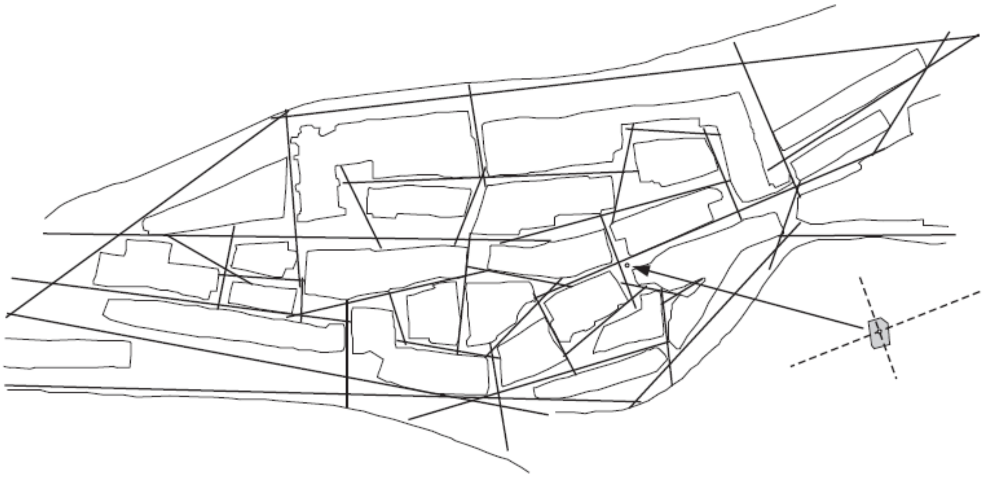


Figure 3.2.3.1: The original hand-drawn axial map of Gassin, France, with detail of 'stringy' (axial) and 'beady' (convex) extensions of a point [HH84]

In 2005, Turner et al. [TP05] redefined the concept of the axial map. They defined an axial map as the minimal set of axial lines such that these axial lines taken together fully surveils the system, and that every axial line that may connect two otherwise unconnected lines is included. The axial line applied here is based on the definition given by Penn et al. [PC97]. In this definition, an axial line was described as any line that joins two intervisible vertices within the system in one of the following ways: (1) both intervisible vertices are convex; (2) one is a convex and one is a reflex vertex, or joining the vertices can be and is extended through open space past the reflex vertex; or (3) both are reflex, and the line joining the vertices can be extended through open space past both vertices. Currently, the axial map is one of the primary tools of space syntax. A typical approach in space syntax is to construct an axial map for public space based on a city map by drawing a set of axial lines, which represent the minimum number of visible lines that cover all the space in question.

### 3.2.3.2 Axial map Generation

According to the original definition described by Hillier and Hanson in 1984, the main problem in applying axial maps is the definition of axial lines. There is no unique method for generating axial lines. Different users will generate different sets of axial lines for the same space in question. Hence generating canonical axial maps is impossible [TP05] [BR04].

The first algorithm to generate axial maps was introduced by Peponis et al. [PW98]. Before generating an algorithm, they first simplified the problem: rather than trying to find a minimal set of axial lines from the set of all possible axial lines, they started with a subset of the lines first published by Penn et al. [PC97]. Penn et al. defined all possible axial lines (three kind of axial lines) as described in section 3.2.3.1,The set of all possible such lines is called an ‘all-line map’. Based on the all-line map, Peponis et al. minimized this number of axial lines to a fewest-line set. In order to reduce the number of axial lines to a minimal set, they employed a well known greedy algorithm. Peponis et al.'s algorithm is guaranteed to find a set of lines that surveil a system. However, the algorithms did not reproduce the same graph what might be drawn by hand (Figure 3.2.3.1). Some lines will be missed applying this method. In order to solve this problem, they improved their algorithm. Figure 3.2.3.2 gives an example generated by using the improved method. When the improved algorithm was applied to the hand drawn examples (Figure 3.2.3.1), the axial map is almost identical to the hand drawn map.

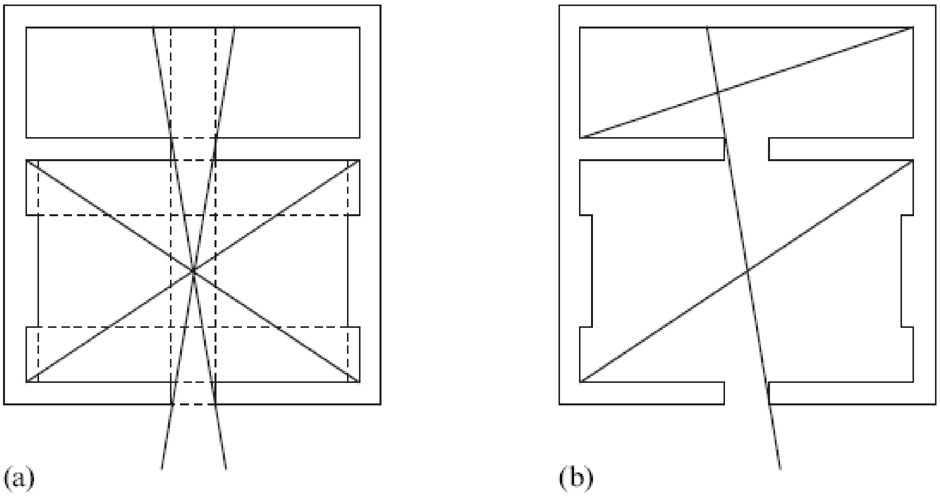


Figure 3.2.3.2: (a) Details on wall surfaces form a pathological case using Peponis et al.'s algorithm. (b) An axial map of the same system. [TP05]

In 2004, Batty and Rana [BR04] generated a similar solution to that of Peponis et al.. They first simplify the problem by generating a set of axial lines and then reduced this set using a greedy algorithm. The method they suggested for generating axial lines depended on defining isovists (see section 3.2.2). They considered a map of the open space of the system, and then made a simplification akin to the selection of an all-line map: for every pixel in their map they constructed an isovist for the location, and then generated a set of axial lines from the sampled system. They reduced the axial lines set by using a greedy algorithm. Figure 3.2.3.3 gives an axial map example using this method.

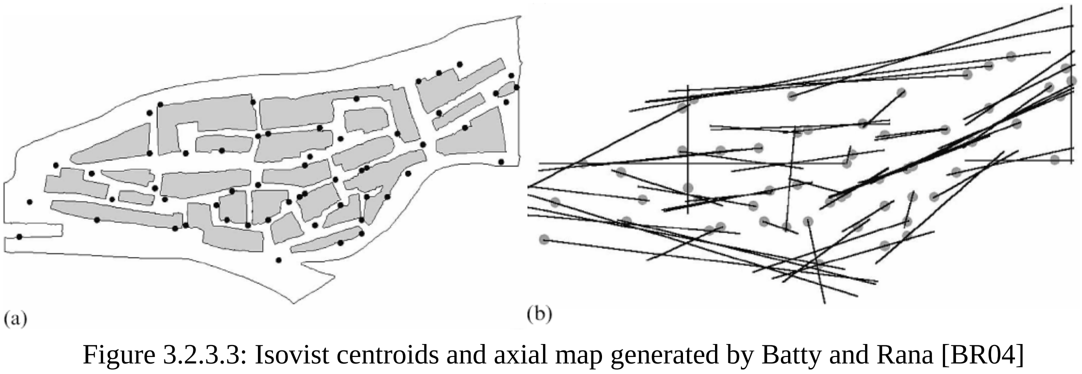


Figure 3.2.3.3: Isovist centroids and axial map generated by Batty and Rana [BR04]

In 2005 [TP05], Turner et al. generated another simple formal minimisation technique to produce an axial map which was better connected. They started by analysing the original definition of the axial map. They then found that there were two aspects to the computational problem with the original definition as outlined by them as stated here [TP05]:

*One, there is indeed a problem, and it stems from a famous problem posed by Klee in 1973: how many guards does it take to fully surveil an art gallery? (O'Rourke, 1987). Two, there is a confusion which pervades the critique of the axial map. The confusion appears to stem from the proximity in Hillier and Hanson's text of the definition of a convex map.*

By analysing the original definition and methods, Turner et al. gave a new definition of the axial map which was described in section 3.2.3.1. Comparing with the previous methods, Turner et al. used a more restrictive definition of an axial line. They choose the all-line map on the basis that it does not require a decision on resolution of pixelation. Figure 3.2.3.4 show an example of using this method.

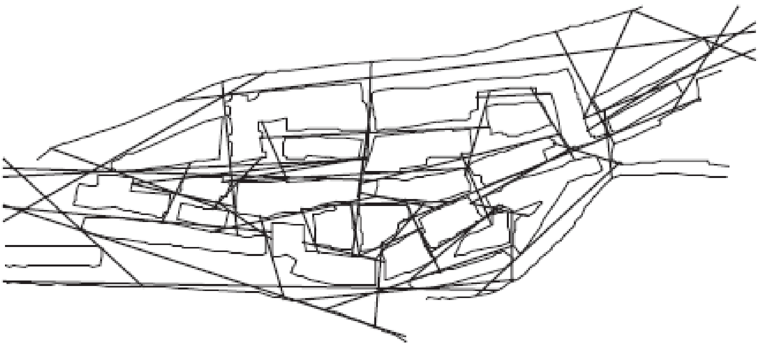


Figure 3.2.3.4: The axial map generated using Turner's algorithm for the map of Gassin [TP05]

The original definition of the axial map was generated for the open space structure of a settlement but not for the analysis of a building. Several algorithms have been mentioned which can be used to generate an Axial map. However, each method will produce a different map.

## 3.3 General measures used with graph representations of buildings

To analyse a building structure using the graph representations methods described in section 3.2, many graph measures have been used to measure the relative accessibility of different locations. In this section, a detail description will be of these graph measures.

## 3.3.1 Basic graph measures

The measures introduced here are some basic graph measures which can be applied to any form of graph analysis.

***Neighbour size of a Node*** [HtFA] [HtPe]

Neighbour size of a Node, also refer to as the Order (degree) of a Node, is used to measure the number of attached links or the number of the connected nodes for a given node "v" .This is a very simple method, but an effective measure of nodal importance. The higher its value, the more important the node as many links converge to it.

***Diameter*** **[**HtPe**]**

The measure of the Diameter is defined as the length of the shortest path between the two most distant nodes of a graph. Diameter measures the extent of a graph and the topological length between two nodes

The diameter can be used to measure the development of a network in time. The higher the diameter, the less linked a network tends to be. In the case of a complex graph, the diameter can be found with a topological distance matrix, which computes for each node pair its minimal topological distance. Graphs whose extent remains constant, but with a higher connectivity, tend to have lower diameter values.

The number of edges between the furthest nodes (2 and 7) of the graph (Figure 3.3.1.1) is 4. Consequently, the diameter of this graph is 4. The matrix in the Table 3.3.1.1 refers to the topological distances between different vertices of the Figure 3.3.1.1.

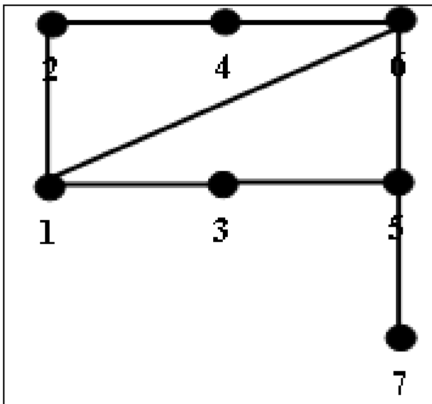


Figure 3.3.1.1: A sample graph

| V | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| **1** | 0 | 1 | 1 | 2 | 2 | 1 | 3 |
| **2** | 1 | 0 | 2 | 1 | 3 | 2 | **4** |
| **3** | 1 | 2 | 0 | 3 | 1 | 2 | 2 |
| **4** | 2 | 1 | 3 | 0 | 2 | 1 | 3 |
| **5** | 2 | 3 | 1 | 2 | 0 | 1 | 1 |
| **6** | 1 | 2 | 2 | 1 | 1 | 0 | 2 |
| **7** | 3 | **4** | 2 | 3 | 1 | 2 | 0 |

Table 3.3.1.1: Topological Distance Matrix, V refer to vertex

The highest value of the distance in this Topological Distance matrix (Table3.3.1.1) is the diameter of the graph.

***Characteristic path length*** [VM02]

The characteristic path length measures the typical separation between two generic nodes of a graph. It is a measure of the average of the shortest link path lengths between all pairs of nodes.

***Number of Circuits*** [HtPe]

Number of Circuits (NoC) is defined as the maximum number of independent circuits in a graph. This number can be calculated in terms of the number of nodes (v), links (e) and non-connected sub-graphs (p). NoC= e-v+p. for a connected graph, this measure can be calculated using the formula: NoC=e+1-v.

Trees and simple networks have a value of 0 since they have no circuits. The more complex a graph is, the higher the value of NoC, so it can be used as an indicator of the level of development and complexity of a transport system. Figure 3.3.1.2 gives an example of how to calculate the NoC for connected graphs.

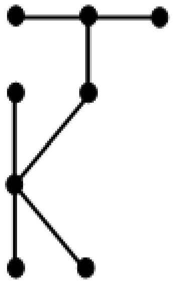
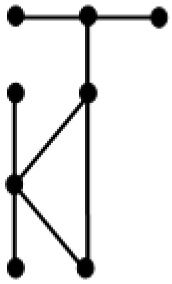
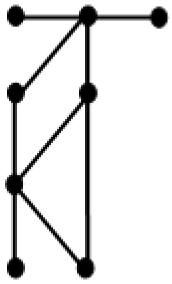
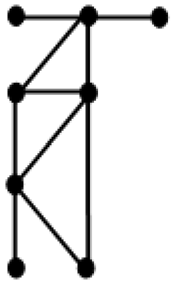
<table>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td></td><td>e</td><td>v</td><td>NoC</td></tr>
<tr><td>A</td><td>7</td><td>8</td><td>0</td></tr>
<tr><td>B</td><td>8</td><td>8</td><td>1</td></tr>
<tr><td>C</td><td>9</td><td>8</td><td>2</td></tr>
<tr><td>D</td><td>10</td><td>8</td><td>3</td></tr>
</table>
</td>
</tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>Values of NoC</td></tr>
</table>

Figure 3.3.1.2: Example to calculate the NoC for connected graphs, ‘e’ refers to the number of links, and ‘v’ refers to the number of nodes

### 3.3.2 Some Indexes [Ka63] [HtFA] [HtPe][Je06]

Indexes are more complex methods to represent the structural properties of a graph since they involve the comparison of one measure with another. However, these measures can also be applied to any graph.

***Detour Index***

Detour Index (DI) is a measure of the efficiency of a transport network. It can be calculated using the Formula 3.3.1:

$$DI = \frac{DD}{TD}$$ Formula 3.3.1

In Formula 3.3.1, DD is the straight distance between two nodes in a graph, TD is the real travel distance between these two nodes.

The closer the detour index gets to 1, the more the network is spatially efficient. Networks having a detour index of 1 are rarely, if ever, seen and most networks would fit on an asymptotic curve getting close to 1, but never reaching it.

For instance, the straight distance (*DD*) between two nodes may be 6 km but the transport distance (*TD*; real distance) is 10 km. The detour index is thus 0.6 (6 / 10). The complexity of the graph is often a good indicator of the level of detour.

### *Network Density*

Network Density (ND) shown in Formula 3.3.2 is used to measure the territorial handhold of a transport network in terms of distance of links (*L*) per square meter of surface (*S*). The higher it is, the more a network is developed.

$$ND = \frac{L}{S} \qquad \text{Formula 3.3.2}$$

In Formula 3.3.2, L is the sum of distance of edges in the graph, and S is the area of the surface of the graph. This method is mainly used to analyse of a huge transport network.

### *Pi Index*

Pi Index ($\Pi$) refers to the relationship between the total length of the graph *L(G)* and the distance along its diameter *D(d)*. It is labelled as Pi because of its similarity with the real number Pi (3.14), which expresses the ratio between the circumference and the diameter of a circle. A high index shows a developed network. It is a measure of distance per units of diameter as shown as Formula 3.3.3.

$$\Pi = \frac{L(G)}{D(d)} \qquad \text{Formula 3.3.3}$$

### *Eta Index*

Eta Index ($\eta$) is the measure of average length per edge. It is calculated using the Formula 3.3.4:

$$\eta = \frac{L(G)}{e} \qquad \text{Formula 3.3.4}$$

In Formula 3.3.4, *L(G)* is the sum of length of all edges in a graph and e is the number of the edges in the graph.

The value of Eta will decrease when adding new nodes as the average length per link declines.

***Theta Index***

The Theta Index ($\theta$) is used to measure the function of a node, which is the average amount of traffic per intersection. It can be described by Formula 3.3.5:

$$\theta = \frac{Q(G)}{v} \qquad \text{Formula 3.3.5}$$

Where $Q(G)$ is the total amount of traffic in all nodes, and *v* is the number of nodes. The higher theta is, the greater the load of the network, it is mainly used in a traffic network

***Beta Index***

Beta Index ($\beta$) is used to measure the level of connectivity in a graph. It is expressed by the relationship between the numbers of edges (e) over the numbers of nodes (v). Beta Index can be calculated using Formula 3.3.6:

$$\beta = \frac{e}{v} \qquad \text{Formula 3.3.6}$$

Trees and simple networks have Beta value of less than one. A connected graph with one circuit has a value of 1. A more complex graph with more than one circuits have a value greater than 1. In a graph with a fixed number of nodes, the higher the number of edges, the higher the number of paths possible in the graph. Basically, complex graph have a high value of Beta.

***Alpha Index***

Alpha Index ($\alpha$) is a measure of connectivity which evaluates the number of circuits in a graph in comparison with the maximum number of circuits. It is calculated using the Formula 3.3.7:

$$\alpha = \frac{u}{v(v-1)/2+1-v} \qquad \text{Formula 3.3.7}$$

In Formula 3.3.7, $u$ is the number of circuits in the graph, and $v$ is the number of nodes in the graph.

The higher the alpha index, the more a graph is connected. Trees and simple graph will have a value of 0. A value of 1 indicates a completely connected graph. Alpha Index is used to measure the level of connectivity independently of the number of nodes. It is very rare that a graph will have an alpha value of 1 in a building graph.

***Gamma Index***

Gamma Index ($\gamma$), calculated shown as Formula 3.3.7, is a measure of connectivity that considers the relationship between the number of observed edges and the number of possible edges.

$$\gamma = \frac{e}{v(v-1)/2} \qquad \text{Formula 3.3.8}$$

The value of gamma is between 0 and 1 where a value of 1 indicates a completely connected network which would be extremely unlikely in reality.

## 3.3.3 Measures used in evacuation model analysis

In this thesis, the representation of a building is a coarse graph, the following graph measures have been used with respect to any generated evacuation models [BD82].

***Local path number***

Local path number is the number of paths from a particular space. It describes the Order (degree) of a Node (neighbour size) in the basic graphs.

***Remote path number***

Remote path number is the number of separate paths to an exit.

***Travel distance***

"Travel distance" is the length of the path from a room to the closest exit. This path length is restricted in order to limit the maximum possible occupant exposure to hazardous combustion products, thus providing a higher egress system.

***Misdirected travel distance***

"Misdirected travel distance" is also called "dead-end" distance, it is associated with a path that does not lead to an exit. During the evacuation, an evacuee may enter a corridor and travel toward the dead-end since the person is unfamiliar with the floor layout. In any simulation case, travel toward the dead-end will waste time. So the measure characterizes a design according to the potential effects of misdirected travel. For each node, the length of travel along a path that does not lead to an exit is calculated. This measure considers only the longest of these paths if there is more than one.

## 3.3.4 Measures used for visibility graphs and axial map analysis

***Clustering coefficient*** [WS98] [ST05] [HP93]

The graph measure 'clustering coefficient' has its origin in the analysis of small-world networks, it is useful for the detection of junction points in environments. The clustering coefficient was introduced by Watts and Strogatz for understanding the structural properties of small-world graphs [WS98], it depicts the average interconnectedness or 'cliquishness' of all neighbourhoods in the graph.

As described in section 3.3.1, a neighbourhood of a network node *v* is the sub-graph consisting of *v* and all nodes directly connected with *v*, and all edges connecting these nodes. Two network nodes that are directly connected by an edge are said to be neighbours. In order to compute the clustering coefficient, edges in the graph are treated as undirected edges. If the number of neighbours of a network node *v* is k (not including node *v*), then the maximum number of possible undirected edges in the neighbourhood is k(k – 1)/2. If T denotes the number of connections between the neighbours of node *v*, the *clustering coefficient C*, can be calculated by the equation [ST05]: $C_v$ =2T/k (k-1).

From this formula, the clustering coefficient gives a measure of the proportion of vertices which are actually connected within the neighbourhood of the vertex *v*, compared to the number that could possibly be connected.

The value of this measure may vary between 0 (disconnected node with no neighbours) and 1 (all neighbours are interlinked with each other). The clustering coefficient reflects the probability that nodes connected with a node *v* also are connected with each other.

***Control*** [HH84] [Tu01]

Control for a location in a graph, is defined by Hillier and Hanson [HH84]. It is calculated by summing the reciprocals of the neighbourhood sizes adjoining the vertex. It is can be calculated by the Formula 3.3.9:

$$c_i = \sum_{v_j \in V(\Gamma_i)} \frac{1}{k_j} \qquad \text{Formula 3.3.9}$$

In Formula 3.3.9, $V(\Gamma_i)$ is the set of nodes in the neighbourhood of the current node.

And $k_j$ is the neighbours' size of $v_j$.

This measure has been made some modification for use with visibility graph analysis [Tu01]. In a visibility graph many of the immediately adjoining neighbourhoods will overlap, thus the control of a graph control is calculated as the area of the current neighbourhood with respect to the total *area* of the immediately adjoining neighbourhood, thus instead of using the sum of all the adjoining neighbourhoods, the size of the union of those adjoining neighbourhoods is used.

***Controllability*** [Tu01]

Controllability is a measure proposed in [Tu01], it is used in visibility graph analysis. In visibility graph analysis, the measure 'control' picks out visually dominant areas, whereas controllability picks out areas that may be easily visually dominated.

Controllability is described as follows: for a location it is simply the ratio of the total number of nodes up to radius 2 to the connectivity (i.e. the total number of nodes at radius 1).

***Integration*** [HH84] [HP93]

Integration is a measure which has seen much mention in the space syntax literature; it is defined by Hillier and Hanson [HH84]. The measure is essentially a normalized version of the mean depth, and it is important because it has been found to correlate well with pedestrian movement 'gate' counts, as remarked in the introduction [HP93].

***Mean depth* [Wi47] [HP93]**

The mean depth from a vertex ($MD_i$) is the average number of edge steps to reach any other vertex in the graph using the shortest number of steps possible in each case. This graph measure has a long history stretching back as far as 1947. It was first advanced by Wiener [Wi47]. The mean depth can be calculated by Formula 3.3.10:

$$MD_i = \frac{\sum_{j=1, j\neq i}^{N} n_{i,j}}{N-1} \qquad \text{Formula 3.3.10}$$

In Formula 3.3.10, $n_{i,j}$ is the number of edge steps from node (i) to node (j) in the graph using the shortest number of steps possible in each case. N is the number of nodes in the graph.

The mean depth is used in the visibility graph analysis due to parallels with the use of integration in space syntax theory [HP93], showing how visually connected a vertex is to all other vertices in the system.

Mean depth is calculated for each node from the shortest path, through the graph is calculated to each other node within the graph. These are summed and divided through by the number of nodes in the graph (minus the node we are considering). If the graph is divided into two separate sections which cannot connect each other, then the number of nodes is simply the number in correlative section of the graph.

***Point depth entropy*** [HH87]

Point depth entropy was first used for space syntax by Hiller and Hanson [HH87], which was defined using the Formula 3.3.11 for a given location, $s_i$:

$$s_i = \sum_{d=1}^{d_{\max}} -p_d \log p_d \qquad \text{Formula 3.3.11}$$

Where $d_{\max}$ is the maximum depth from vertex $v_i$ and $p_d$ is the frequency of point depth d from the vertex.

Point depth entropy allows us to explore measures based on the frequency distribution of depths. It can give an insight into how ordered the system is from a location, and it is the least number of edges that need to be traversed to get from one vertex to the other. Point depth entropy for a vertex is simply the average of the shortest path lengths from that vertex to every other vertex in the system, and so represents an average of the number of turns required for any journey within the system.

***Step depth*** [Tu04]

Step depth is defined for visibility graph and axial map analysis [Tu04]. It is viewed as the number of turns (plus one) that it takes to get from the current node to any other nodes within a graph. The nodes directly connected from the current node at depth one, every other nodes from that at depth two, and so on throughout the graph. In visibility graph analysis, the step depth is called the *visual depth* of each point. In reality, the visual depth is a measure of the `shortest path' through the graph. You could turn umpteen times to get to any node within the graph, but on the shortest path from one node to any other, you make as few turns as possible.

***Metric step depth*** [Tu04]

Comparing step depth, the metric step depth uses a weighting scheme for the edges of the graph, so it is not just one step to move from one node to another. However, it gives the distance measure for the movement. Therefore, Metric step depth is a measure of the shortest path for the current node to each other node in the graph. In visibility graph analysis, it is also called ‘metric step shortest-path length’.

***Metric step shortest-path angle*** [Tu04]

Metric step shortest-path angle is used to calculate the angle through which you have to turn to get to target node by metric step depth route, it is also used to analysis the visibility graph.

## 3.4 Measures used to compare different buildings for evacuation complexity

### 3.4.1 The important factors of buildings that influence evacuation

The protection of the health and safety of citizens was an early motive underlying legislation to safeguard the quality of the built environment [LH03]. A number of factors are known to influence evacuation efficiency. These can be collected into four broad categories: *Configuration* of the Enclosure, the *Procedures* implemented within the Enclosure, *Environment* inside the Structure, and the *Behaviour* of the Occupants. The relationship between these influences is shown in Figure 3.4.1.1.

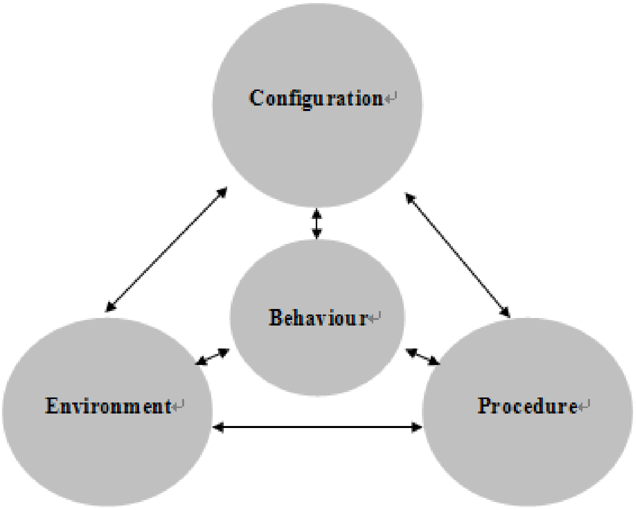


Figure 3.4.1.1: Inter-relationship of the factors influencing evacuation.

Recently, some building designers have started to move towards performance-based regulations due to restrictions of the fixed criteria of the traditional prescriptive methods. However, the enclosure design is forever vital to the success of the evacuation. For the purpose of this thesis, we mainly considered the traditional building codes to find out the important configurational factors which will affect the evacuation results.

The enclosure defines the configurational influence upon the emergency evacuation. It establishes the arena in which the event occurs, and provides many of the relationships that the population develops with each other, and the fire event, by determining the purpose behind an individual’s presence at the event. Configurational considerations are those generally covered by the traditional building codes and involve building layouts, the number of exits, exit width, travel distance, location of exits etc.

The physical configuration of the enclosure is governed by regulations that are country specific. The successful enclosure design is the precondition to the success of the evacuation, prescriptive regulations are in place to ensure basic levels of design and safety [TH94][Ta91], Tanaka described regulations used in Australia, France, Japan, The UK, and The USA. These regulations indicate the important factors which influence the ease of evacuation within a building. These configuration factors include the following:

1. The *minimum number of exits*, which in most countries is two.
2. The *maximum travel distance***,** which depends upon the number of directions of means of escape, the layout of the floor area, the building features, the occupancy type, the physical and mental capabilities of the occupants, and the hazard control equipment available.
3. The *common path of travel* of the population, which is considered one of the most important factors, and is restricted in all countries. Tanaka explains that,
   *"Two means of escape must be located as remotely from each other as practical to avoid them both being blocked by the same fire"* [Ta91]
   I.e. they do not share a common path of travel.
4. The *exit capacity*. The number of occupants which can pass through an exit at any one time.
5. The number of *dead end corridors***.** The number and length of these is limited, to prevent people being trapped by smoke, and prevent time wastage.
6. The *occupant load*. The expected number of occupants within the structure.

At the time of several infamous tragedies, the structures did not conform to the regulations. In the Beverly Hills Supper [Be77] and Gothenburg Club tragedies [CD00] the enclosures had too few exits for their eventual occupancy levels. Hence neither of them had the required exit capacity to evacuate the large number of occupants safely.

A general set of recommendations has been compiled by the National Bureau of Standards (NBS) [VS80], that suggest targets at which designers should aim to encourage a more distributed use of the enclosure. These concentrate upon,

1. The simplicity of access and movement routes.
2. The replacement of stairs with ramps where possible

3. The reversibility of egress systems.

### 3.4.2 The measures to compare different buildings for evacuation ability

A variety of egress modelling strategies have been produced and some of which include sophisticated behavioural and tracking characteristics. Despite the various distinct aims of these models, these strategies have not addressed the more fundamental and potentially comparable notion of egress complexity for habitable buildings. The current egress complexity algorithms are typically based on route complexity, for example Donegan's method.

All the graph measures described in section 3.2 have been used to analysis building structure. However, all these graph measures can only be used to measure the relative accessibility of different locations in a spatial system. Based on these measures, we can obtain a quick insight into which location is better, and which location is worse for evacuation within a building. But these measures can not answer whether one structure is better than another.

Donegan et al. [DP94] proposed a graph specific measure which is distinct from scenario based measures of evacuation time modelling. For comparing different building evacuation ability, Donegan et al. proposed the use of complexity theory. They call their method 'egress complexity'. Egress Complexity is a scenario independent, non-metric methodology that assesses the egress capability of a compartmentalized floor plan [DP94]. It is based on the room graph representation of a building.

For calculating the egress complexity for a given building, Donegan et al. employed the room-based graph building representation. Based on the room graph representation of the building plan, Donegan et al. applied the 'Shannon Entropy'

[Sh48] as a measure Of Structural Complexity [DP94] [DT98]. Such complexity could be equated to the summative uncertainty associated with a naive occupant in exhaustive pursuit of an exit without the benefit of signage [DP94].

This method was developed from information theory based on the concept of Shannon entropy. It is formulated as being ‘equivalent’ to the amount of information required to describe a total egress system. For a naive occupant positioned at a random location within a building it is fundamental to ask “how much information would that occupant need to accumulate in order to egress successfully from the building”. The information step includes both positive and negative information steps. When an edge between two nodes is initially traversed, a positive information step is taken, and information is acquired. If the same path is backtracked, a negative information step is taken, meaning that no new information is acquired.

The Donegan’s method can only be applied to a tree graph, where in general this is not the case. Most buildings are far more complex with many circulation regions that form graphs with circuits. For a graph with circuits Donegan’s method tries to calculate the complexities for all spanning trees of the graph.

The problem of enumerating all the spanning trees of a graph is NP-complete [ME84]. Currently, there are a number of algorithms [RT75, GM78, To93, KR95, ST97] that can be applied to find all the spanning trees for a graph. In 1975, Read and Tarjan presented an algorithm by using a technique called backtracking [RT75], their algorithm requires O(V+E+ME) time, where V is vertices, E is edges and M the spanning trees. Gobow and Myers [GM78],in 1978, refined the backtracking approach and obtained an algorithm with O(V+E+MV) time. This was followed by Tomomi [To93] who presented an algorithm which has the same complexity time of the Gobow’s algorithm. In 1995, Kapoor and Ramesh proposed an algorithms for scanning all of the spanning tree [KR95], which require O(V+E+M) time. Then in 1997, Shioura and Tamura [ST97] presented an algorithm with the same time complexity as Shioura’s algorithm. In all of these algorithms to enumerate all spanning trees, the number of the spanning trees M has been included in the algorithms time complexity. In the worst case, for a complete graph, M can be very large and can be calculated by this formula $M = \mathrm{V}^{\mathrm{V}-2}$, which is known as the Cayley's formula [HtMa]. Hence, the enumerating of all spanning trees of a given graph is extremely computationally time consuming to calculate.

Once all the spanning trees for a given graph have been obtained, the Donegan’s method then calculates the building complexity value for each of the spanning trees. Then the maximum complexity value from the set of all spanning trees is chosen as the complexity measure for the original graph. In addition, the Donegan method fails to consider other important factor, such as ‘travel distance’, obstacles and travel time.

## 3.5 Conclusions

A number of graph measures which have been applied to analyse building structures have been described in Section 3.3. However, these graph measures are primarily used to measure the relative accessibility of different locations in a spatial system. Based on these measures, one can obtain a quick insight into which location is better or worse for evacuation within a building. The measures were not generated for the purpose of comparing the complexity of different buildings. The only method that is currently available for comparing different building with respect to evacuation ability is the Donegan’s method which is described in Section 3.4. However, this method can only be applied to a tree graph, where in general this is not the case, most building are far more complex with many circulation regions which form graph with circuits. For a graph with circuits Donegan’s method tries to calculate the complexities for all spanning trees of the graph. However, the problem of enumerating all the spanning trees of a graph is NP-complete [ME84], which is often extremely complex to calculate. In addition the Donegan method fails to consider other important factor, such as ‘travel distance’, obstacles and travel time. In Chapter 4 a new method is presented which addresses the limitations of the Donegan's method and presents a new algorithm which includes these important factors. In Chapter 4, I will also expand some of the local measures described in Section 3.3 to formulate a basic global measure which is used, in conjunction with buildingEXODUS, to aid in validating the global measures presented in this thesis.

# Chapter 4: Development of a Distance Graph method for application to evacuation analysis using room graphs

## 4.1 Introduction

This chapter will initially review Donegan's method and its application to comparing building in relationship to their evacuation capabilities. By analysing the strength and weakness of Donegan's method, an improved method will be generated, which is called the 'Distance Graph method'. For generating such an improved method, there are several issues that first need to be addressed: Firstly, some necessary complementary proofs are derived from Donegan's theories. Secondly, Donegan's method is extended so that it can deal with complexity graphs which include circuits. Thirdly, a distance measure will be incorporated into the complexity calculation, to form a new method which will be referred to as the 'Distance Graph method'. For verifying the validation of this new method some test cases will be presented which include graphs that contain circuits. However, since they contain circuits these test cases cannot be evaluated using Donegan's method. Hence, the results obtained from the Distance Graph method will be validated against simulated results generated by buildingEXODUS.

When a building is designed, one of the prime aims is to ensure that the building complies with the fire regulations. Such regulations are written with three major objectives: life safety, property protection and the prevention of conflagration [DP96]. Within these objectives, the most important objective is life safety as it relates specifically to the vital system of evacuation [Kl85].

Some of fire safety literature contain egress models addressing the established survival inequality [SE08] [Gu09]:

$$\boldsymbol{RSET \le ASET}$$

Where *RSET* is the time needed to evacuate a fully occupied building under alarm conditions, *ASET* is the time from beginning of event until conditions of untenability prevail. Traditionally, such models are scenario based and none address the fundamental issue of egress complexity. As described in Chapter 1, the building complexity is another important research tool for comparing different buildings for the purpose of evacuation capability. To achieve such an objective, the first task is to generate the graph representation for a building, then analyse the evacuation system of the building by applying a graph technique. For the purpose of this discussion, in this dissertation, the term 'building' will refer to an enclosed environment with a least one floor level comprising habitable compartments that connect to a system of egress, also the room-based graph representation of such an enclosed environment must satisfy the following:

1. The room-based graph has at least one non-exit node
2. The room-based graph has at least one external exit node
3. The room-based graph must be a connected graph
4. The room-based graph must be a simple graph that contains no loops.

As described in Chapter 3, the graph representation methods for a building structure mainly include three kinds of graphs: room-based graph, axial map, and the visibility graph. These graphs have been applied to architectural analysis for more than 30 years. The methods of architectural analysis are mainly based on all sorts of graph measure techniques as described in Chapter 3.2. Applying these measures to a graph representation of a floor plan provides a quick insight into the properties of different locations within one building. However, all of these measures cannot calculate a global value for a building structure, hence cannot be employed for comparing different building structures for the purpose of evacuation ability.

For comparing the evacuation ability of different buildings, Donegan et al. proposed the egress complexity theory so that complexity could be equated to the summative uncertainty associated with a naive occupant in exhaustive pursuit of an exit without the benefit of signage [DP94]. Egress Complexity is a scenario independent, non-metric methodology that assesses the egress capability of a compartmentalized floor plan [DP94]. The complexity measure method is based on the room graph representation of a building. We will refer to this complexity measure method as 'Donegan's method'. The discussion in this chapter will be based on Donegan's method.

## 4.2 Important factors Influencing Evacuation Feature

As the discussion of evacuation measures will be the main purpose of this section, the important factors influencing evacuation need to be emphasized again.

As described in Chapter 3.3, the physical configuration of the enclosure is governed by regulations that are country specific. The successful enclosure design is the precondition to the success of the evacuation, prescriptive regulations are in place to ensure basic levels of design and safety, Tanaka [TH94][Ta91] described regulations used in Australia, France, Japan, UK, and USA. These regulations indicate the important factors which influence the evacuation results within a building. These important configuration factors mainly include the following sections:

*The number of exits, the location of exits, the travel distance, exit capacity* and *dead end corridors*.

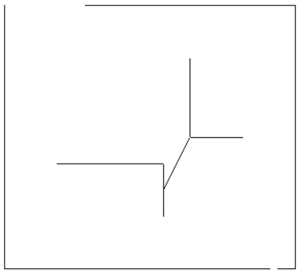
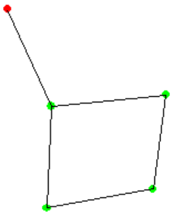
An ideal complexity measures method should be based on all of these factors. In reality, we can only consider most of these factors.

## 4.3 Test cases

In this section, five different test cases are given which will be used with respect to this chapter, Chapter 5 and Chapter 7 to show the performance of different methods of calculating building complexity. Figures from 4.3.1 to 4.3.5 are the plans of some example building structures, which associated Graphs from 4.3.1 to 4.3.5 representing the room-based graph of these buildings separately. The graphs are ordered from the simplest to the most complex. All of the graph representations of these test cases include circuits, which cannot be calculated by Donegan’s method directly.

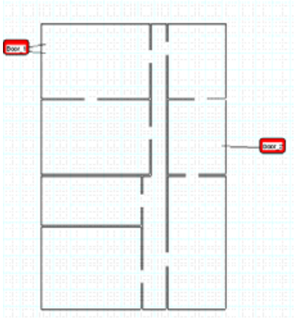
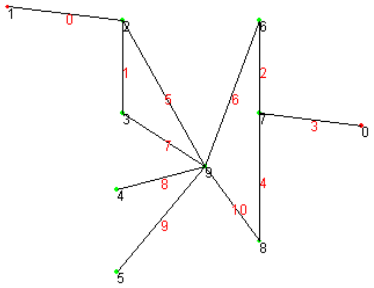
### *Test case 1:*

The building in the first case has just one single external exit and four rooms (Figure 4.3.1). The room-based graph representation of this building has a single circuit (Graph 4.3.1).

| | |
|---|---|
|  |  |
| Figure 4.3.1: Building 1 | Graph 4.3.1: Room-based graph of building 1 |

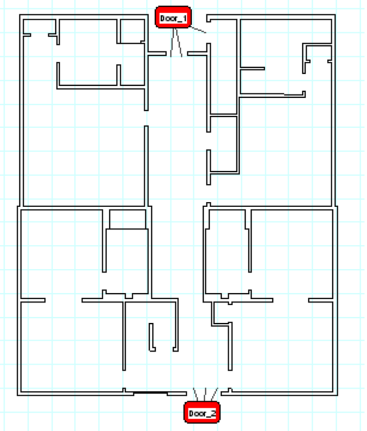
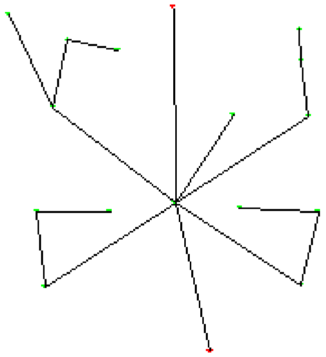
### *Test case 2:*

Test case 2 has two external exits, and includes seven rooms and one corridor (Figure 4.3.2). So the room-based graph representation in buildingEXODUS contains eight non-exit nodes as shown in Graph 4.3.2. As we can see from Graph 4.3.2, there are two circuits in the room-based graph.

| | |
|---|---|
|  |  1<br>0<br>2<br>6<br>1<br>2<br>5<br>6<br>3<br>7<br>3<br>0<br>7<br>9<br>8<br>4<br>4<br>10<br>9<br>8<br>5 |
| Figure 4.3.2: Building 2 | Graph 4.3.2: Room-based graph of building 2 |

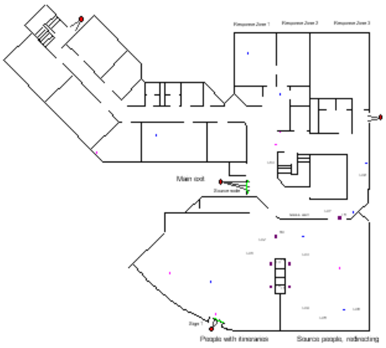
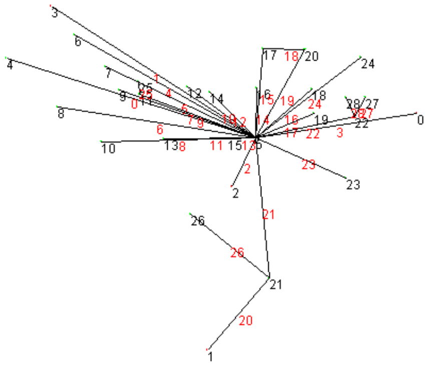
## *Test case 3:*

For test case 3 the floorplan is as shown in Figure 4.5.2, it also includes two external exits. However, the rooms in this building are not the same simple style as the last two cases. Applying the graph generation method in buildingEXODUS, this building contains 14 compartments. So the relevant room-based graph (Graph 4.5.3) has 14 non-exit nodes and two exit nodes. This graph representation does not include any circuits, hence forms a tree graph.

| | |
|---|---|
|  |  |
| Figure 4.3.3: Building 3 | Graph 4.3.3: Room-based graph of building 3 |

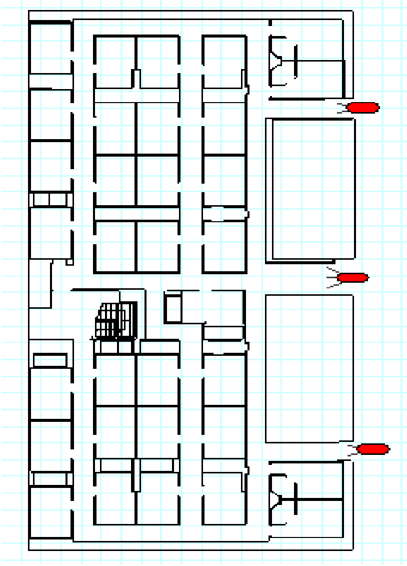
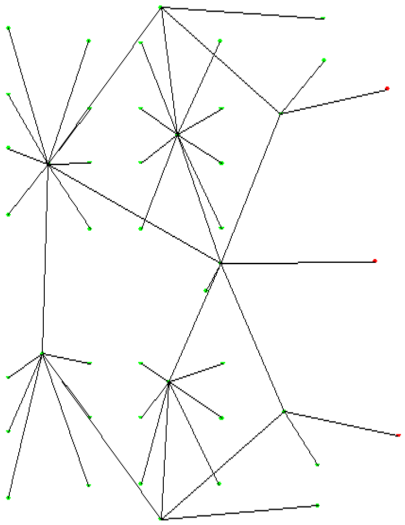
## *Test case 4:*

Test case 4 has a more complex floorplan than the last three cases. Figure 4.5.4 is the floorplan which contains 24 compartments and 4 external exits. So the corresponding room-based graph includes 24 non-exit nodes and 4 exit nodes.

| | |
|---|---|
|  |   3<br>6<br>4<br>7<br>9<br>8<br>10<br>13<br>14<br>17<br>18<br>20<br>24<br>19<br>22<br>0<br>23<br>2<br>26<br>21<br>20<br>1 |
| Figure 4.3.4: Building 4 | Graph 4.3.4: Room-based graph of building 4 |

### *Test case 5:*

When comparing with all of other test cases, the building considered in test case 5 would be expected to be the most complex one. The building (Figure 4.3.5) contains 42 compartments and only three exits. The relevant room-based graph representation is as shown in Graph 4.3.5

|  |  |
|---|---|
| Figure 4.3.5: Building 5 | Graph 4.3.5: Room-based graph of building 5 |

The set of test cases thus includes 5 different buildings with different complexity degrees. In this chapter, Chapter 5 and Chapter 7, these test cases will be employed to verify the different complexity measures algorithms.

## 4.4 Donegan’s method

### 4.4.1 Introduction

Wayfinding is a basic activity that people do through their lives as they navigate from one place to another. People navigate relying on their understanding of their environment, i.e. their knowledge of the environment. Hence, it is very important for the people to understand the evacuation environment.

For an occupant positioned at a random location within a building it is fundamental to ask “how much information would that occupant need to accumulate in order to egress successfully from the building”. Also it is can be asked how difficult it is to understand the whole building. Donegan’s method was generated base on such consideration. A building which is simple to cognize is easier to evacuate from. Hence it is important to compare the evacuation complexity for different buildings to provide an understanding of how an evacuation will be affected by building layout.

Building complexity is used to compare different buildings for the evacuation ability. It includes a lot of aspects, for example: the complexity of the process of recognizing the building structure, the traveling distance or time to recognizing the building, and the evacuation time of a building. In Donegan’s method, they used the concept of ‘egress complexity’, and mainly considered the first aspect of complexity, i.e. recognition. For assessing the evacuation time of a building, most evacuation models have been developed as described in Chapter 2. In scenario-independent modelling, we mainly focus on the first two aspects of the complexity.

For considering building complexity, Donegan employed the room-based graph representation of a building in his method. This kind of room graph of a building can be generated automatically in buildingEXODUS.

### 4.4.2 Implementation of Donegan’s method

Every building has a latent measure of route complexity that may be computed algorithmically from the graph representation of the building’s floorplan. The value of this complexity is a ‘cold’ or inert measure of the given graph of a building. Therefore a graph specific measure is distinct from scenario based measures of evacuation time modelling [DP94].

Donegan et al. applied the ‘Shannon Entropy’ [Sh48] as a measure of Structural

Complexity [DP94] [DT98] [PD96] [DT99] [DL07]. Based on the Entropy, they developed a measure for a room-based graph representation of a building to compare the complexity of wayfinding tasks. They call their measure ‘Egress complexity’. The Egress Complexity is a scenario independent, non-metric measure that assesses the egress capability of a compartmentalized floor plan [DP94]. The measure of capability is effectively the egress route complexity – initially developed to measure the uncertainty experienced by a naive occupant in pursuit of an exit.

The model generates a statistic that is a measure of the relative complexity of the structure from an evacuation point of view. As such it is not an evacuation model; however it can be used to investigate the relative complexity of buildings during an evacuation. As they commented:

*“The strength of challenge is related to the information which an occupant has in respect of available egress routes.”[DP94]*

This complexity is due mainly to the number of options available to individuals who wish to leave via a recognised exit. The routes open to them provide the layout with an entropy (complexity) measure. The method is designed to encompass egress uncertainty about the structure.

In Donegan’s method, the mathematical formulation is based on information theory and entropy [PD96]. Building plans are interpreted as room-based graphs. According to the description in Chapter 3, the compartments of the structure are represented by nodes. And the links between the nodes are identified by the edges which are referred to as information steps. Knowledge is gained from moving between nodes for the first time. Thereafter on the subsequent passes of information step makes no contribution is made to the accumulated knowledge. Specifically, knowledge is not gained if an edge is backtracked. This is because all of the information gained is assumed to have been acquired during the first traversal. Indeed, by the time an occupant has evacuated, he is assumed to have traversed the entire network and to have calculated the most efficient exit route. The probabilities of acquiring and of not acquiring egress information in a sweep from any general compartment to a predetermined exit node are calculated using algorithms produced by Donegan et al. [DP96]. Where the term ‘sweep’ was defined as followed [DL07]:

> ‘*A sweep from a vertex v is a trajectory through the graph beginning at v and ending in the external exit node, traversing each edge at most twice while visiting each terminal node exactly once.*’

The egress complexity is formulated as being ‘equivalent’ to the amount of information required to describe a total egress system. For a naive occupant positioned at a random location within a building it is fundamental to ask ‘how much information would that occupant need to accumulate in order to egress successfully from the building’ [DT98] [PD96].

Based on the information step described above, the mutually exclusive events of acquiring and not acquiring information about evacuating from any non-exit node to a predetermined exit node via all other non-exit nodes in the room graph are given, respectively:

$$p^{+} = \frac{n^{+}}{n^{+} + n^{-}}, p^{-} = \frac{n^{-}}{n^{+} + n^{-}} \qquad \text{Formula 4.4.1}$$

In Formula 4.4.1, $n^{+}$ is used to indicate the number of positive instance of information, $n^{-}$ is the number of negative information step. According to Shannon [Sh48], the corresponding average information involved in this process is given by the entropy function:

$$H(v) = p^{+} \log_{2}(\frac{1}{p^{+}}) + p^{-} \log_{2}(\frac{1}{p^{-}}) \qquad \text{Formula 4.4.2}$$

Factoring the last expression by ( $n^{+} + n^{-}$ ) generates a measure:

$$I(v) = n^{+} \log_2 \frac{n^{+} + n^{-}}{n^{+}} + n^{-} \log_2 \frac{n^{+} + n^{-}}{n^{-}} \qquad \text{Formula 4.4.3}$$

Formula 4.4.3 is used to calculate the unbounded nodal information needed to egress from the specified non-exit node to the stated exit. The egress complexity of the given room graph relative to a fixed exit e is defined using

$$Egress.Complexity = \sum_i n_i^{+} \log_2 \frac{n_i^{+} + n_i^{-}}{n_i^{+}} + n_i^{-} \log_2 \frac{n_i^{+} + n_i^{-}}{n_i^{-}} \qquad \text{Formula 4.4.4}$$

*The degree of exit complexity* was obtained by dividing the egress complexity by the number of non-exit nodes. When more than one exit is used, the ***global complexity*** is calculated using the Formula 4.4.5:

$$Global.Complexity = \frac{1}{\sum_i \left[ \frac{1}{\sum_j I_j} \right]_i} \qquad \text{Formula 4.4.5}$$

Where $I_j$ represent the node information of the $j^{th}$ non-exit node and $i$ is the number of exit nodes.

All of the formulas can only be employed for a building with a single storey which has a tree graph representation. For a building with more than one storey, the stairwells on any upper floor are considered as exit nodes. Donegan et al. have tried to consider various approaches to describe the complexity of a multi-storey building. However, they indicate that a vector method [DT98] is the most effective and is described in Chapter 5.

## 4.4.3 Demonstration of Donegan's method

In this section, some simple cases are given to calculate the egress complexity of buildings using Donegan's method. In each case, the building floorplan is illustrated using a room-based graph. The first two cases were introduced by Donegan in 1999 [DT99]. As shown in these cases, the room graphs are simple trees. Donegan's method can only be used to calculate the complexity for a tree graph representation.

***Example 1:***

This example introduced the simplest kind of floorplan which just contains one external exit, and the graph representation of the building structure is just a tree. Table 4.4.3.1 give the calculated results of the node information and total complexity. As shown in Table 4.4.3.1, the complexity value is 83.50.

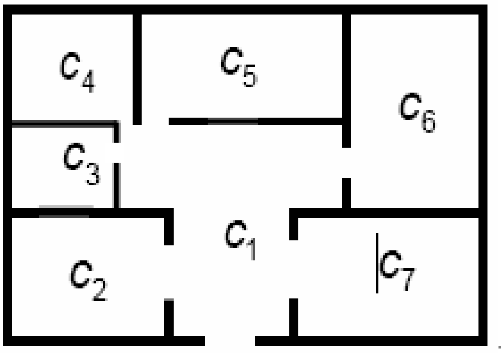
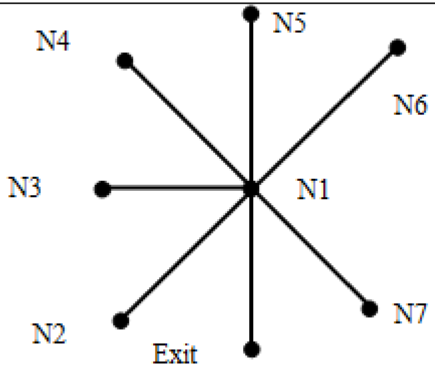
|  |  |
|---|---|
| Figure 4.3.3.1: Building A | Graph 4.3.3.1: Room-based graph of Building A |

Table 4.4.3.1: The egress complexity values applying Donegan's method

| **Node** | $n^+$ | $N^-$ | Nodal Information |
|---|---|---|---|
| N1 | 7 | 5 | 12.94 |
| N2 | 7 | 5 | 11.76 |
| N3 | 7 | 5 | 11.76 |
| N4 | 7 | 5 | 11.76 |
| N5 | 7 | 5 | 11.76 |
| N6 | 7 | 5 | 11.76 |
| N7 | 7 | 5 | 11.76 |
| Total | Egress Complexity=83.50 | | |

***Example2:***

Comparing with Example 1, the floorplan employed in example 2 contains two external exits, and contains one more compartment. The room-based graph representation is also a simple tree. The egress complexity for Exit1 is 111.48 and 103.74 for Exit2. Hence the global complexity is 53.74 (see Table 4.4.3.2).

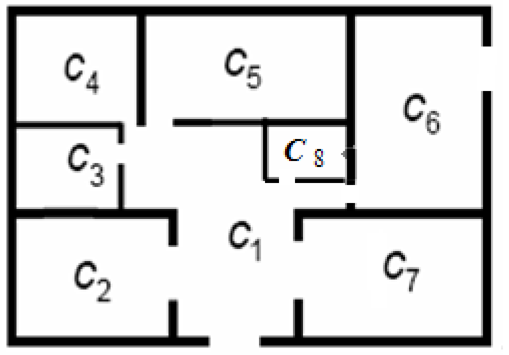
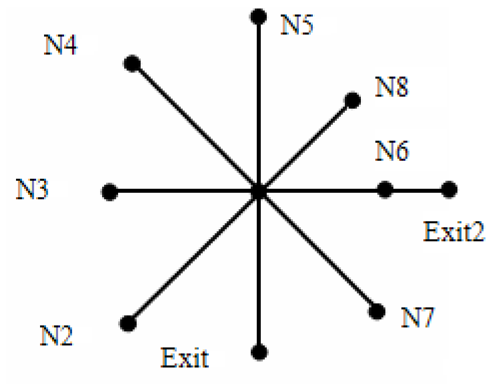
| | |
|---|---|
|  |  |
| Figure 4.3.3.2: Building B | Graph 4.3.3.2: Room-based graph of Building B |

Table 4.4.3.2: The egress complexity values applying Donegan's method

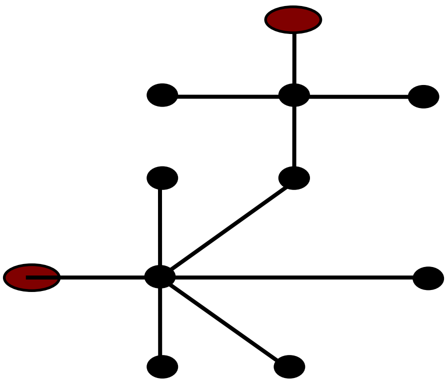
| **Node** | $n^+$ | $n^-$ | Nodal Info(Exit1) | $n^+$ | $n^-$ | Nodal Info.(Exit2) |
|---|---|---|---|---|---|---|
| N1 | 8 | 7 | 14.95 | 8 | 6 | 13.79 |
| N2 | 8 | 6 | 13.79 | 8 | 5 | 12.50 |
| N3 | 8 | 6 | 13.79 | 8 | 5 | 12.50 |
| N4 | 8 | 6 | 13.79 | 8 | 5 | 12.50 |
| N5 | 8 | 6 | 13.79 | 8 | 5 | 12.50 |
| N6 | 8 | 6 | 13.79 | 8 | 7 | 14.95 |
| N7 | 8 | 6 | 13.79 | 8 | 5 | 12.50 |
| N8 | 8 | 6 | 13.79 | 8 | 5 | 12.50 |
| Total | Egress Complexity=111.48 | | | Egress Complexity=103.74 | | |
| Global | The global complexity=53.74 | | | | | |

Comparing the complexity of the last two building plan (Example1 and Example2), the second case obtained a smaller value than the first example. This seems to be a logical conclusion.

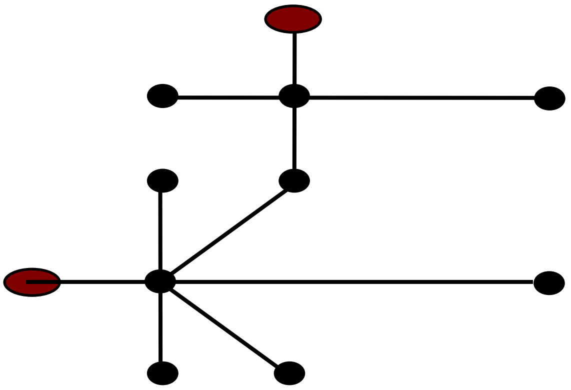
## 4.4.4 Strength and Weakness

The graph specific measure technique that Donegan et al. proposed is distinct from the scenario-based measures of evacuation time modelling. It is scenario independent, and assesses the egress capability of a compartmentalized floor plan. Such a measure, based on a room-based graph representation of a building, can be applied to compare the complexity of wayfinding tasks. The measure of capability is effectively the egress route complexity.

When using egress complexity to compare two or more of structures with respect to evacuation ability. The building with the smallest value of complexity is the simplest to navigate. However, currently this method can only be used to calculate the complexity of a tree graph, but normally a building graph is not just a tree. For a graph with circuits, Donegan et al. try to calculate the complexities for all the spanning trees of the graph. However, the problem of enumerating all the spanning trees of a graph is NP-complete [ME84]. Donegan's method also does not take into consideration an important factor 'travel distance'. For example, the following two graphs (Graph 4.4.4.1 and Graph4.4.4.2) represent two building layout respectively, and they are both simple tree structures. These two graphs look similar, but two edges in the Graph4.4.4.2 are longer than in Graph 4.4.4.1. It is obviously that the second graph is more difficult to evacuate than the first one due to the extra travel distance involved. If we apply Donegan's method here it obtains the same complexity value of 67.119 for both structures**.** .



Graph 4.4.4.1: Sample Graph A



Graph 4.4.4.2: Sample Graph B

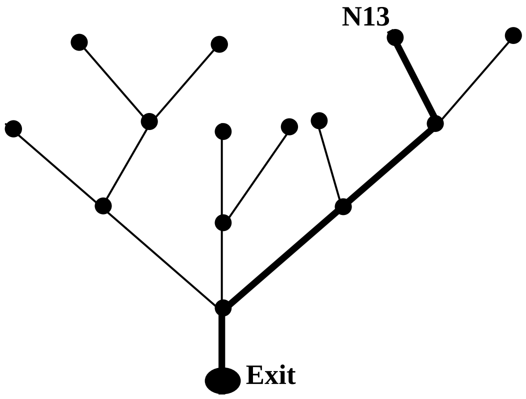
## 4.5 Distance Graph method

### 4.5.1 Theory and implementation of the Distance Graph method

This section considers three issues which require addressing: Firstly, some complementary proofs to Donegan's method's theories are generated. Secondly, Donegan's method is extended by calculating complexity measures for graphs with circuits. Thirdly, the distance measure will be incorporated into the complexity calculation.

### 4.5.1.1 Complementary theories to Donegan's method

The theory of Donegan's method is based on the tree graph representation of a building, and the calculating formulas were based on the concept of Shannon Entropy. In the node information calculation formula, the vital task is how to account two important variable $n^+$ and $n^-$. For explaining the method of accounting these two variables, a simple tree graph will be employed here (Graph 4.5.1.1).



Graph 4.5.1.1: Sample Graph C

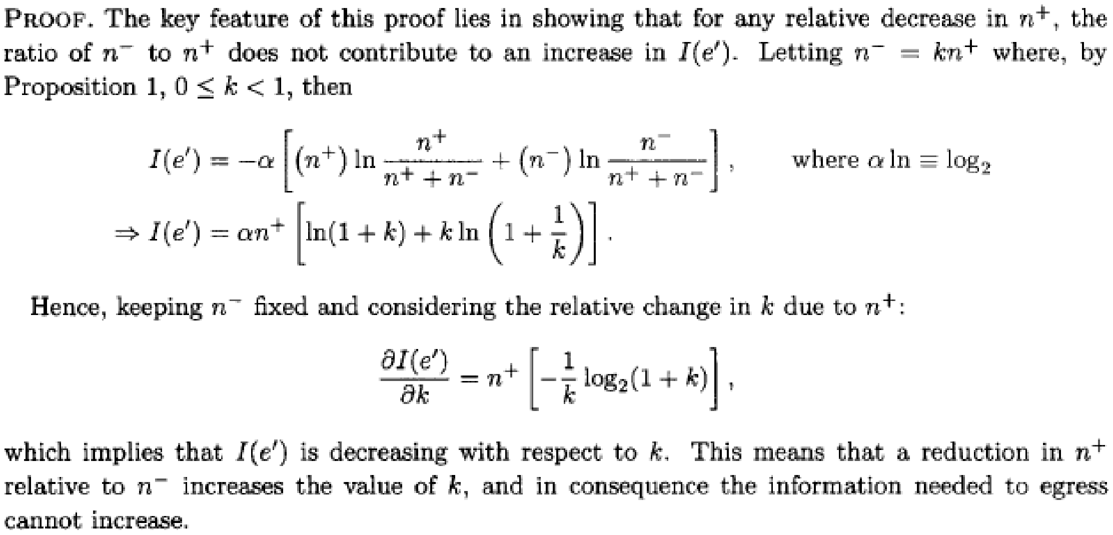
In Graph 4.5.1.1, there are 14 non-exit node and one exit node, so the number of edges is 14. Here $n^+$ is used to indicate the number of positive instance of information, i.e. the maximum number of positive information steps executed by an occupant from a given node in pursuit of an exit. Obviously, $n^+$ should be 14 for node $N^{13}$. In reality, when an occupant sweeps the whole graph, he has to travel on every edge in the graph due to there being no circuits in the graph. So $n^+$ equal the number of edges of the graph for any node. The node $N^{13}$ has the potential value of 4 (the route has been emphasized by the bold edges in Graph 4.5.1.1). $n^-$ is the number of negative information step. When an occupant sweeps the whole graph, he has to travel all other edges two times except the edges generated the potential steps. So if d is used to denote the potential value of the node (the minimum steps to the exit). Then $n^{-} = n^{+} - d$ Hence $n^{-}$ has a value of 10 for node $N^{13}$.

There is an important theorem of complexity in Donegan et al.'s research works, as described by them:

*'The theorem is important for subsequent work on subfloor networks' [DP96]*

Then they give the following theorem [DP96]:

**THEOREM 4.5.1:** The information needed to egress from any non-exit node gc within the total egress system to exit node e' does not increase if the number of positive instances $n^{+}$ is reduced. Then the proof of this theorem is given as [DP96]:



PROOF. The key feature of this proof lies in showing that for any relative decrease in $n^+$, the ratio of $n^-$ to $n^+$ does not contribute to an increase in $I(e')$. Letting $n^- = kn^+$ where, by Proposition 1, $0 \leq k < 1$, then

$$I(e') = -\alpha\left[(n^+)\ln\frac{n^+}{n^+ + n^-} + (n^-)\ln\frac{n^-}{n^+ + n^-}\right], \qquad \text{where } \alpha \ln \equiv \log_2$$

$$\Rightarrow I(e') = \alpha n^+\left[\ln(1+k) + k\ln\left(1+\frac{1}{k}\right)\right].$$

Hence, keeping $n^-$ fixed and considering the relative change in $k$ due to $n^+$:

$$\frac{\partial I(e')}{\partial k} = n^+\left[-\frac{1}{k}\log_2(1+k)\right],$$

which implies that $I(e')$ is decreasing with respect to $k$. This means that a reduction in $n^+$ relative to $n^-$ increases the value of $k$, and in consequence the information needed to egress cannot increase.

However, there are some issues in this proof, which will be addressed here. Firstly, let us consider the condition in the theorem, *the number of positive instances n + is reduced.* According to the discussion above, this theorem is based on the tree graph representation of a building structure. So it must satisfy $n^{-} = n^{+}$*-d.* For a given tree graph, the value of $n^{+}$ is a constant and is equal to the number of edges in the graph. To meet the condition in the theorem, for $n^{+}$ to be reduced this must be achieved by removing nodes in the tree graph, and keeping the connected sub-graph (also a tree) of G which includes the exit node e'. Let $G_1$ express this sub-graph. So the conclusion in the theorem is based on comparing $G_1$ to G.

After analysing the theorem, let us consider the process of proof. First, they make an assumption in their proof: keeping $n^-$ fixed. In the graph $G_1$, the potential are all the same value with that in Graph G. According $n^- = n^+ \text{-} d$, $n^-$ will reduce with respect to the $n^+$ for any node in $G_1$ comparing with the corresponding node in G. Obviously, the assumption of keeping $n^-$ fixed is not correct. Secondly, applying the method in the proof to calculate $\frac{\partial I(e^{'})}{\partial k}$ to analyse the variation of $I(e^{'})$ with respect to k, an assumption we need to guarantee that $n^+$ and k is independent. However, they are not. Thirdly, the result of $\frac{\partial I(e^{'})}{\partial k}$ is not correct in the calculation. The correct result is $\frac{\partial I(e^{'})}{\partial k} = n^+ \log_2(1+\frac{1}{k})$.

Hence a new proof of theorem 4.5.1 is outline bellow.

## Proof.

To prove the THEOREM 4.5.1, we just need to prove the following conclusion: $\forall G_t \subseteq G$, where $G_t$ is connected and contain the exit node e'. $\forall N_i \in G_t$, then there is $N_i \in G$, we only need to prove $I_i^{G_t}(e) \geq I_i^{G_t}(e)$.

Considering the formula for calculating node information:

$$I_i\,(e) = n^+ \log_2 \frac{n^+ + n^-}{n^+} + n^- \log_2 \frac{n^+ + n^-}{n^-}$$

Then $n^+ = n^- \text{-} d$ for node $N_i$ *in* $G_t$ and *G*. Furthermore, *d* is the same value in $G_t$

and *G*.

Applying $n^{-} = n^{+}$-$d$ replace $n^{-}$ in $I_i\ (e)$, then

$$I_i\ (e) = n^{+} \log_2 \frac{2n^{+} - d}{n^{+}} + (n^{+} - d)\log_2 \frac{2n^{+} - d}{(n^{+} - d)}$$

Considering the derivative of the corresponding real function:

$$I_i\ (e) = x \log_2 \frac{2x - d}{x} + (x - d)\log_2 \frac{2x - d}{x - d}$$

According to the formula $\log_a b = \frac{\ln a}{\ln b}$, $I_i\ (e)$ can be expressed as the following function:

$$I_i\ (e) = \log_2 e\left[x \ln \frac{2x - d}{x} + (x - d)\ln \frac{2x - d}{x - d}\right]$$

Then

$$\frac{d(I)}{d(x)} = \log_2 e\left[(\ln \frac{2x - d}{x} + x \times \frac{x}{2x - d} \times \frac{d}{x^2}) + (\ln \frac{2x - d}{x - d} + (x - d) \times \frac{x - d}{2x - d} \times \frac{(-d)}{(x - d)^2})\right]$$

$$= \log_2 e\left[\ln \frac{2x - d}{x} + \ln \frac{2x - d}{x - d}\right]$$

$$= \log_2 \frac{2x - d}{x} + \log_2 \frac{2x - d}{x - d}$$

Obviously, $\frac{2x - d}{x} > 1,\ \frac{2x - d}{x - d} > 1,$ when $x > d$ .

So $\frac{d(I)}{d(x)} > 0$ Which implies that $I(x)$ will increase with respect to $x$ .

Since $n^{+}$ in $G_t$ is less than $n^{+}$ in ***G***. So $I_i^{G_t}(e) \geq I_i^{G_t}(e)$ .

**Finish.**

As described in [DP96], '*the theorem is important for subsequent work on subfloor networks'*. So they give Proposition 4.5.1 and the proof based on the theorem. The proposition is described as followed.

**Proposition 4.5.1:** If a floor network G' is obtained from a given floor network G by the removal of a gc node, then E(G') < E(G) [DP96].

In this proposition, gc node represent a non-exit node in the graph. Based on THEORM 4.5.1 and the proof, the conclusion is obviously. Then the authors give a conjecture which was described by them as following:

> *REMARK. The authors conjecture that p(*G'*) < p(G) also holds. No proof or counterexample has been produced to date* [*DP96]***.**

For this conjecture, they did not give the proof and counterexample. In the following proof, a method will be introduced to show the conjecture to be true. First a new THEOREM will be employed to describe the conjecture completely. Then give the proof process.

**THEOREM 4.5.2:** if *G* is a tree graph representation of a building plan, $G_1$ is obtained by the removal of a non-exit node from *G, then* $p\ (G_1) < p\ (G)$.

Where $p\ (G_1)$ *and* $p\ (G)$ denote the egress complexity degree with respect to $G_1$ and *G* respectively.

## Proof:

To prove the theorem, we first need to consider the location of the non-exit node which is removed from *G*. There are two cases.

***Case 1:*** the removed node N is in the top of a branch in G, which means there is just one edge to connect with N.

***Case 2:*** the removed node N is not on the top of a branch.

In reality, there is a sub-branch which has been removed in Case 2, and this sub-branch does not have any direct connection with the exit node. However, for Case 2, the $G_1$ can be obtained by a number of steps applying the method in Case 1. If the nodes in this sub-branch have the number of $k$, than the process of removing the branch can be completed in $k$ steps. In each step, one node on the top of sub-branch is removed. Hence we have a serious of sub-graph of *G*. denoted by $G_k, G_{k-1} \ldots G_{2,}, G_1$.

If in Case 1 the conclusion of THEOREM is true, then

$$p\,(G_1) < p\,(G_2) <\ldots< p\,(G_k)< p\,(G_k)< p\,(G),$$

So the THEOREM for case 2 is also correct.

Base on the discussion above, only Case 1 needs to be proved in the following.

If the number of non-exit nodes is $t$ in G, then $n_G^+ = t$.

$G_1$ is obtained by the removal of a non-exit node of G and the node location meets the situation described in Case 1. For the convenience to describe the proving process, the removed node is denoted by $N_t$. Then the number of non-exit nodes in $G_1$ should be $t$-1. So $n_{G_1}^+ = t\text{-}1$.

For any node $N_i$ (i=1, 2…$t$-1) in $G_1$, it has the same potential value with the corresponding node in G. Hence it is easy to show $n_{G_1}^-(i) = n_G^+(i)$ -1(i=1, 2…$t$-1).

Considering the formula for calculating egress complexity, then

$$E(G) = \sum_{i=1}^{t} (n_G^+ \log_2 \frac{n_G^+ + n_G^-(i)}{n_G^+} + n_G^-(i)\log_2 \frac{n_G^+ + n_G^-(i)}{n_G^-(i)})$$

$$E(G_1) = \sum_{i=1}^{t-1} (n_{G_1}^+ \log_2 \frac{n_{G_1}^+ + n_{G_1}^-(i)}{n_{G_1}^+} + n_{G_1}^-(i)\log_2 \frac{n_{G_1}^+ + n_{G_1}^-(i)}{n_{G_1}^-(i)})$$

According $n_{G_1}^+ = t\text{-}1 = n_G^+\text{-}1$, $n_{G_1}^-(i) = n_G^+(i)$ -1(i=1, 2…$t$-1) then

$$E(G_1) = \sum_{i=1}^{t-1} \left[ (n_G^+ - 1)\log_2 \frac{(n_G^+ - 1) + (n_G^-(i) - 1)}{n_G^+ - 1} + (n_G^-(i) - 1)\log_2 \frac{(n_G^+ - 1) + (n_G^-(i) - 1)}{n_G^-(i) - 1} \right]$$

Considering the formula for calculating egress complexity degree and $n_G^+ = t$, then

$$p(G) = \frac{E(G)}{t} = \sum_{i=1}^{t} \left[ \log_2 \frac{n_G^+ + n_G^-(i)}{n_G^+} + \frac{n_G^-(i)}{n_G^+}\log_2 \frac{n_G^+ + n_G^-(i)}{n_G^-(i)} \right]$$

$$= \sum_{i=1}^{t-1} \left[ \log_2 \frac{n_G^+ + n_G^-(i)}{n_G^+} + \frac{n_G^-(i)}{n_G^+}\log_2 \frac{n_G^+ + n_G^-(i)}{n_G^-(i)} \right] + (\log_2 \frac{n_G^+ + n_G^-(t)}{n_G^+} + \frac{n_G^-(t)}{n_G^+}\log_2 \frac{n_G^+ + n_G^-(t)}{n_G^-(t)})$$

$$\geq \sum_{i=1}^{t-1} \left[ \log_2 \frac{n_G^+ + n_G^-(i)}{n_G^+} + \frac{n_G^-(i)}{n_G^+}\log_2 \frac{n_G^+ + n_G^-(i)}{n_G^-(i)} \right]$$

$$=\sum_{i=1}^{t-1}\left[\log_2(1+\frac{n_G^-(i)}{n_G^+})+\frac{n_G^-(i)}{n_G^+}\log_2(1+\frac{n_G^+}{n_G^-(i)})\right]$$

The last item is represented by $p^{'}(G)$, then

$$p^{'}(G)=\sum_{i=1}^{t-1}\left[\log_2(1+\frac{n_G^-(i)}{n_G^+})+\frac{n_G^-(i)}{n_G^+}\log_2(1+\frac{n_G^+}{n_G^-(i)})\right], \text{ where } p^{'}(G)\le p(G).$$

Also,

$$p(G_1)=\frac{E(G_1)}{t-1}=\sum_{i=1}^{t-1}\left[\log_2\frac{(n_G^+-1)+(n_G^-(i)-1)}{n_G^+-1}+\frac{(n_G^-(i)-1)}{(n_G^+-1)}\log_2\frac{(n_G^+-1)+(n_G^-(i)-1)}{n_G^-(i)-1}\right]$$

$$=\sum_{i=1}^{t-1}\left[\log_2(1+\frac{(n_G^-(i)-1)}{n_G^+-1})+\frac{(n_G^-(i)-1)}{(n_G^+-1)}\log_2(1+\frac{(n_G^+-1)}{n_G^-(i)-1})\right]$$

To prove the THEOREM, we just need to show $p(G_1)\le p^{'}(G)$. Using $q_i(G_1)$ denote the $i^{th}$ item of $p(G_1)$, $q_i(G)$ represent $i^{th}$ item of $p^{'}(G)$, (i=1, 2…*t*-1). Then

$$q_i(G_1)=\log_2\left[1+\frac{(n_G^-(i)-1)}{n_G^+-1}\right]+\frac{(n_G^-(i)-1)}{(n_G^+-1)}\log_2\left[1+\frac{(n_G^+-1)}{n_G^-(i)-1}\right]$$

$$q_i(G)=\log_2(1+\frac{n_G^-(i)}{n_G^+})+\frac{n_G^-(i)}{n_G^+}\log_2(1+\frac{n_G^+}{n_G^-(i)})$$

At the moment, we just need to show $q_i(G_1)\le q_i(G)$ for i=1, 2…*t*-1.

Considering the derivative of the corresponding real function:

$$f(x)=\log_2(1+x)+x\log_2(1+\frac{1}{x})$$
$$=\log_2 e(\ln(1+x)+x\ln(1+\frac{1}{x}))$$

Then

$$\frac{d(f(x))}{dx}=\log_2 e\left[(\frac{1}{1+x})+\ln(1+\frac{1}{x})+(x\times\frac{x}{1+x}\times\frac{(-1)}{x^2})\right]$$

$$=\log_2(1+\frac{1}{x})$$

Obviously, $\log_2(1+\frac{1}{x})>0$ which implies $f(x)$ will increase with respect to *x*.

Assuming $x_1(i) = \frac{(n_G^-(i) - 1)}{n_G^+ - 1}$, $x_2(i) = \frac{n_G^-(i)}{n_G^+}$ ,

Then $x_1(i) \le x_2(i)$ for *i=1, 2…t-1.* and $q_i(G_1) = f(x_1(i)), q_i(G) = f(x_2(i))$ .

So $q_i(G_1) \le q_i(G)$ (*i=1, 2…t-1*).

**Finished.**

### 4.5.1.2 Applying Donegan’s method to structures with circuits

The main task of extending Donegan’s method is the ability to calculate complexity measures for graphs with circuits. Over the past twelve years, Donegan et al. have tried several kinds of methods to solving this question [DT99] [PD96]. The first idea was based on the following axiom. As described by them:

*AXIOM 1. Where there is a choice of particular complexity magnitudes for any egress system, the worst case will apply; i.e., the complexity value with the greatest magnitude will be deemed [DP96].*

According to this axiom, they still have not found a feasible method which solves such graph with circuits. In other word, the spanning tree that meets this axiom has not been found. Therefore, they try to solve structures with circuits by generating all spanning trees for a graph [DT99] [PD96], then calculating the complexity for each of the spanning trees. From this set of spanning tree graphs they choose the maximum value as the complexity measure of the original graph. However, the problem of enumerating all the spanning trees of a graph is NP-complete [ME84]. Therefore, this is often extremely complex to calculate (see Section 3.4).

In this section, a special spanning tree will be introduced which have been proved to be the maximum complexity value. In other word, this spanning tree meets all the demands to calculate complexity for any graph with circuits.

In the following, a definition will be introduced to describe such spanning tree.

**DEFINITION 4.5.1**: A spanning tree $G_1$ of Graph ***G*** is defined as a ***Potential Spanning Tree*** of graph G about the ***Exit E*** if the $G_1$ is generated by the following process:

Calculating a potential map (see Chapter 2) over the domain of the graph for ***Exit E,*** then the minimum travel steps of each node from the exit can be calculated. The edges that generated the minimum travel steps to the exit are kept for each node.

The next theorem is important for extending the Donegan's method to be applied in any building.

**THEOREM 4.5.3:** If $G_1$ is a *Potential Spanning Tree* of graph G about the *Exit E***,** then $G_1$ will have the maximum complexity value for *Exit E* compared to all other spanning trees.

.

**Proof:**

Assuming $G_1$ denotes the *Potential Spanning Tree* of graph G about *Exit E,* and then the corresponding egress complexity is expressed by $E_e(G_1)$ for *Exit E*. And $G_2$ is one of random spanning tree of G. Then we apply $E_e(G_2)$ to represent the building complexity of $G_2$ for *Exit E*. Hence we just need to show the following inequation: $E_e(G_1) \geq E_e(G_2)$.

According the definitions of $E_e(G_1)$ and $E_e(G_2)$, the formulas of them are expressed separately:

$$E_e(G_1) = \sum_k I_i^{G_1}(e)\text{, and}\quad E_e(G_2) = \sum_k I_i^{G_2}(e)$$

Where k is the number of non-exit nodes of these two spanning tree, $I_i^{G_1}(e)$ is the nodal information of the $i^{th}$ node in $G_1$ about *Exit E,* $I_i^{G_2}(e)$ is the $i^{th}$ nodal information of $G_2$ about *Exit E.*

All the graphs which are spanning trees of the Graph G contain the same nodes, thus have the same number of nodes and the same number of edges. So to prove $E_e(G_1) \geq E_e(G_2)$, we just need to prove that, for any node $N_i$ in G**,** $I_i^{G_1}(e) \geq I_i^{G_2}(e)$. Considering the expression of calculating node information:

$$I_i\ (e) = n^+ \log_2 \frac{n^+ + n^-}{n^+} + n^- \log_2 \frac{n^+ + n^-}{n^-}$$

For any given $N_i$**,** if applying *d* represents the potential for $N_i$ in each tree, then $d_1$ denotes the potential value for $N_i$ in $G_1$, $d_2$ denotes the potential value for $N_i$ in $G_2$**.** According to the DEFINITION 4.5.1: $N_i$ will have the minimum potential steps in $G_1$, so $d_1 \leq d_2$, where $n^+$ is the number of non-exit nodes for any given tree graph, so $n^+$ is a constant. Also $n^- = n^+ - d$ is always correct in a tree. So the node information can be express as follows:

$$I_i\ = n^+ \log_2 \frac{2n^+ - d}{n^+} + (n^+ - d)\log_2 \frac{2n^+ + d}{n^+ - d}$$

Considering to the derivative of the corresponding real function (here $n = n^+$):

$$I(x) = n\ \times \log_2 \frac{2n - x}{n} + (n - x) \times \log_2 \frac{2n - x}{n - x}$$

According the formula $\log_a b = \frac{\ln a}{\ln b}$, $I(x)$ can be expressed as the following function:

$$I(x) = \log_2 e \left[ n \times \ln \frac{2n - x}{n} + (n - x)\ln \frac{2n - x}{n - x} \right]$$

Then

$$\frac{d(I)}{d(x)} = \log_2 e \left[ n \times \frac{n}{2n - x} \times (-\frac{1}{n}) + (-\ln \frac{2n - x}{n - x}) + (n - x) \times \frac{n - x}{2n - x} \times \frac{n}{(n - x)^2} \right]$$

$$= \log_2 e \left[ \frac{-n}{2n - x} - \ln \frac{2n - x}{n - x} + \frac{n}{2n - x} \right]$$

$$= -\log_2 e \times \ln \frac{2n - x}{n - x}$$

$$= -\log_2 \frac{2n - x}{n - x}$$

$$= -\log_2(1 + \frac{n}{n-x})$$

Obviously, $-\log_2(1 + \frac{n}{n-x}) < 0$

Which implies that $I(x)$ is decreasing with respect to $x$ increasing. By reason of $d_1 \le d_2$, then $I_i^{G_1}(e) \ge I_i^{G_2}(e)$.

To generate such kind of graph, for any given exit circuits need to be broken down so that a tree can be obtained. For any given exit, the steps of each node from the exit need to be calculated. The edges that generated the minimum travel steps to the exit will be kept for each node. This is achieved by calculating a potential map over the domain of the graph for each exit. Such a spanning tree is generated according to Definition 4.5.1.

### 4.5.1.3 Incorporating a distance measure into the calculation

Another approach to extend the algorithm is to consider traveling distance, here we define two new variable $d^+$ and $d^-$, $d^+$ represents the traveling distance of gaining positive instances of information, $d^-$ represents the traveling distance of gaining negative instances of information, then using the $d^+$ and $d^-$ replace the $n^+$ and $n^-$ in Donegan's complexity calculation formula respectively. The node information can then be calculated using Formula 4.5.1:

$$I_i(e) = d^+ \log_2(\frac{1}{p^+}) + d^- \log_2(\frac{1}{p^-}) \qquad \text{Formula 4.5.1}$$

Where $p^+$ and $p^-$ were indicated by these two formulas respectively

$$p^+ = \frac{d^+}{d^+ + d^-}, p^- = \frac{d^-}{d^+ + d^-}$$

The ***complexity E*** of the given floor network S relative to a fixed ***exit e*** is defined using formula 4.5.2:

$$E(S) = \sum_i d_i^+ \log_2 (\frac{1}{p_i^+}) + \sum_i d_i^- \log_2 (\frac{1}{p_i^-}) \qquad \text{Formula 4.5.2}$$

This new equation was called the ‘Distance Graph method’. And the global complexity is represented by Formula 4.5.3:

$$Global.Complexity = \frac{1}{\sum_i \left[ \frac{1}{\sum_j I_j} \right]_i} \qquad \text{Formula 4.5.3}$$

## 4.5.2 Demonstration of Distance Graph method

When using the Distance Graph Method to calculate the building complexity, a lot of factors which influence the complexity results will be checked, for example the number of exits, number of nodes, number of edges and their length. Also, the analysis for the same graph with different edge lengths will be done.

### 4.5.2.1 First calculated results

In this section, all the test cases introduced in section 4.3 will be used to show the performance of Distance Graph method of calculating a building complexity index. Except case 3, none of the other graphs can be used with the Donegan’s method since they contain circuits. As described in section 4.3, the graphs are ordered from simplest to most complex.

The table 4.5.2.1 gives the calculated results. From these values, it is easy to identify that the first graph obtained the minimum complexity value, and the last one obtained the maximum value. Therefore the results seem reasonable using the Distance Graph method. In Table 4.5.2.1, some other measures of the graph are also shown; the number of exits, number of nodes, number of edges and the maximum potential value in the graph. These factors are all important in influencing the complexity value.

Table 4.5.2.1: the calculated results for the cases in section 4.3

| Graphs | Number of exits | Number of nodes | Number of edges | Maximum potential | Based on the graph after break circuits | Distance Graph method |
|---|---|---|---|---|---|---|
| Graph 4.3.1 | 1 | 5 | 5 | 7.7266 | 21.5258 | 54.9882 |
| Graph 4.3.2 | 2 | 10 | 11 | 23.1229 | 50.6395 | 279.852 |
| Graph 4.3.3 | 2 | 18 | 29 | 23.0982 | 236.65 | 982.711 |
| Graph 4.3.4 | 4 | 29 | 29 | 47.6026 | 297.365 | 2994.75 |
| Graph 4.3.5 | 3 | 45 | 48 | 42.8204 | 1125.58 | 5718.05 |

### 4.5.2.2 Edge Removed

In this section the examination of what happens when some edges are removed from the graph. The Graph 4.5.2.2 is derived from the Graph 4.5.2.1 by removing the two red edges, and the Graph 4.5.2.3 is derived from the Graph 4.5.2.2 by removing the two blue edges. The Table 4.5.2.2 gives the calculated results. The complexity of Graph 4.5.2.1 is the maximum value and the Graph 4.5.2.3 is a minimum value. From these results, we can see that the more edges there are the more complex the building is for evacuation, which seems to be a logical conclusion.

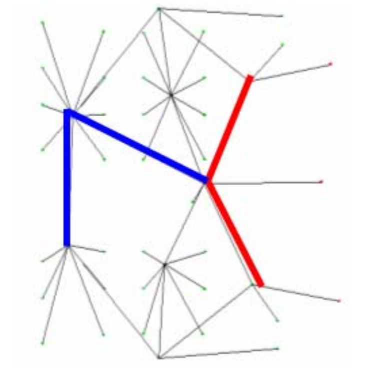
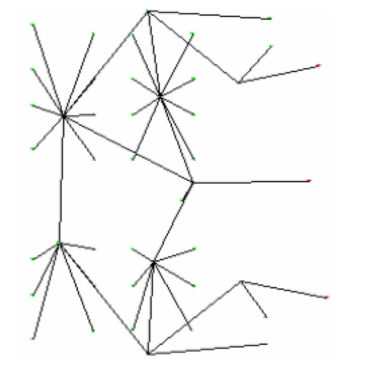
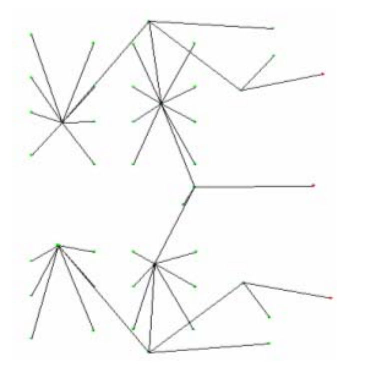
| | | |
|---|---|---|
|  |  |  |
| Graph 4.5.2.1: Room-based graph of Building 5 (Figure 4.3.5) | Graph 4.5.2.2: A graph obtained by removing red edges in Graph 4.5.2.1 | Graph 4.5.2.3: A graph obtained by removing red and blue edges in Graph 4.5.2.1 |

Table 4.5.2.2: Calculated results for Graphs 4.5.2.1 to 4.5.2.3

| **Methods**<br>**Graphs** | **Based on the graph after break circuits** | **Distance Graph method** |
|---|---|---|
| **Graph 4.5.2.1** | 1125.58 | 5718.05 |
| **Graph 4.5.2.2** | 1115.71 | 5588.02 |
| **Graph 4.5.2.3** | 1106.73 | 5520.34 |

## 4.5.2.3 Node removed

This example examines what occurs when nodes are removed from the graph. Graph 4.5.2.5 is derived from Graph 4.5.2.4 by removing two nodes and the edges connecting them. Graph 4.5.2.6 is derived from Graph 4.5.2.5 by removing the red node and its edges plus any nodes that have become disconnected from the geometry. Table 4.5.2.3 gives the calculation results. From the results, we can see that reducing the number of nodes improves the building complexity measure.

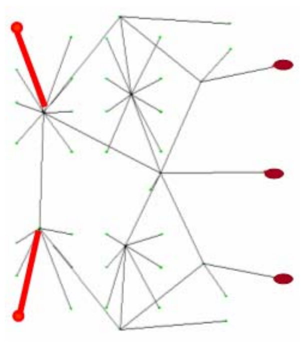
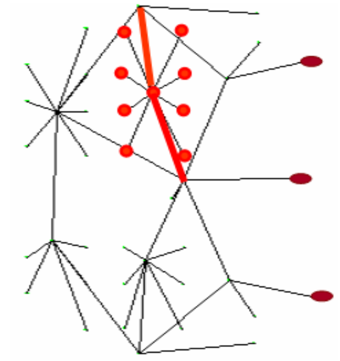
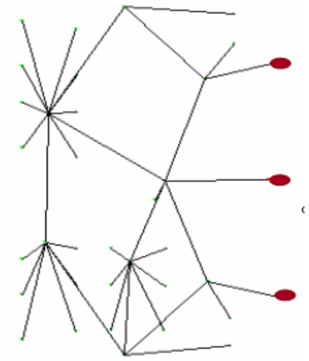
| | | |
|---|---|---|
|  |  |  |
| Graph 4.5.2.4: Room-based graph of Building 5 (Figure 4.3.5) | Graph 4.5.2.5: A graph obtained by removing red nodes and edges in Graph 4.5.2.4 | Graph 4.5.2.6: A graph obtained by removing red nodes and edges in Graph 4.5.2.5 |

Table 4.5.2.2: Calculated results for Graphs 4.5.2.4 to 4.5.2.6

| **Methods**<br>**Results** | **New methods (based on the graph after break circuits** | **Distance Graph method** |
|---|---|---|
| **Graph 4.5.2.4** | 1125.58 | 5718.05 |
| **Graph 4.5.2.5** | 1018.93 | 5003.65 |
| **Graph 4.5.2.6** | 686.229 | 3612.66 |

## 4.5.2.4 Size Variation

This section examines what happens when the graph size is varied. Presented in the Graphs 4.5.2.7 to 4.5.2.9 are groups of graph which have the same structure style but different travel distances. From the results, it can be seen that if the length of the edges is not considered then the same complexity value is obtained. However, with the Distance Graph method it can be observed from Table 4.5.2.3 that the smaller the travel distance the better the structure is for evacuation. According to Chapter 4.2, the travel distance is an important factor influencing the evacuation ability, which is not considered by Donegan’s method.

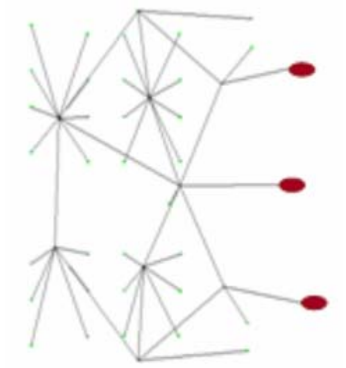
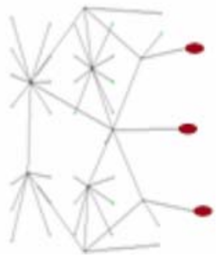
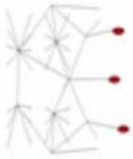
| | | |
|---|---|---|
|  |  |  |
| Graph 4.5.2.7: Room-based graph of Building 5 (Figure 4.3.5) | Graph 4.5.2.8: A graph obtained by 1/2 size of Graph 4.5.2.7 | Graph 4.5.2.9: A graph obtained by 1/3 size of Graph 4.5.2.7 |

Table 4.5.2.3: Calculated results for Graphs 4.5.2.7 to 4.5.2.9

| **Methods**<br>**Graphs** | **based on the graph after break circuits** | **Distance Graph method** |
|---|---|---|
| **Graph 4.5.2.7** | 1125.58 | 5718.05 |
| **Graph 4.5.2.8 (1/2size)** | 1125.58 | 2859.03 |
| **Graph 4.5.2.9 (1/3size)** | 1125.58 | 1906.02 |

## 4.5.3 Investigation of results

Here are presented some results for basic graph measures which are suitable for comparing different building graphs. Then for four different building described in Section 4.3(*Case2 to Case5*), these basic measures and the Distance Graph method for calculating the building complexity are generated. Currently, we cannot find any benchmark for verifying the validation of the Distance Graph method. Hence an evacuation simulation model is chosen as the validation tool. In this thesis, we will apply buildingEXODUS as the validation tool. So we will compare building complexity results using simulations data generated by buildingEXODUS.

### 4.5.3.1 Simulating the evacuation process using buildingEXODUS for 4 Cases

Evacuation simulations were performed using buildingEXODUS for four different buildings described in Section 4.3. For the first scenario presented here, 5 repeat simulations were performed using a population of 15 for each building. The populations were generated randomly in buildingEXODUS. There was almost no congestion occurring during the evacuation when setting 15 people in each building. The other scenario was designed using populations of 100 for each building. These populations were also generated randomly using buildingEXODUS. When assigning 100 people for each building, congestion should be an important factor in the evacuation. For Case 1 introduced in Section 4.3, the building structure is so simple that there is not enough space to contain 100 people, hence it is not suitable. In the following, the average evacuation results will be given for test Cases 2 to 5 described in Section 4.3 (see Table 4.5.3.1).

Table 4.5.3.1: Average evacuation statistics for the 5 repeat simulations

| | | **Simulation results** | | | |
|---|---|---|---|---|---|
| **Test Cases** | **Number of evacuees** | **TET (secs)** | **Average CWT (secs)** | **Average Distance (m)** | **Average PET (secs)** |
| ***Case 2*** | 15 | 34.75 | 0.06 | 8.49 | 24.56 |
| | 100 | 112.09 | 23.99 | 9.80 | 48.91 |
| ***Case 3*** | 15 | 45.14 | 0.06 | 8.39 | 26.43 |
| | 100 | 54.25 | 4.24 | 8.11 | 29.43 |
| ***Case 4*** | 15 | 61.79 | 0.06 | 14.98 | 32.28 |
| | 100 | 77.59 | 11.72 | 13.21 | 39.71 |
| ***Case 5*** | 15 | 47.56 | 0.21 | 14.16 | 29.95 |
| | 100 | 61.76 | 3.97 | 16.17 | 36.82 |

The simulation results shown in Table 4.5.3.1 will be used to compare with building complexity measures in Section 4.5.3.3. Hence, the analysis to these results will be described in Section 4.5.3.3.

### 4.5.3.2 Some other basic graph measures

Recalling the ‘mean depth’ graph measure introduced in Chapter 3, it can be used to analyse the graph. However it is mainly used for comparing different node within a graph.

The mean path length from a node is the average number of edge steps required to reach any other vertex in the graph using the shortest number of steps possible in each graph. The mean depth can be added together for each point as a new measure called a ‘total mean depth’. This can be used to comparing different building graphs.

The ‘total mean depth’ measure does not consider the number of exits. Hence for a building, the same measure value will be generated regardless of the number of exits.

So obviously, it is not a good method for comparing different structures for the purpose of evacuation.

For applying 'total mean depth' measure, we can make some adjustment for the mean depth. Basically, we can just focus on the following two changes:

i. Just calculate the average number of edge steps to reach any exit in the graph using the shortest number of steps possible in the graph.

ii. Calculating the average route distance to reach any exit in the graph using the shortest distance possible in the graph.

In reality, applying one of the last two kinds of adjustment, the new total mean depth measurement can be used to check some aspect of complexity for a building. In Table 4.5.3.2, some calculated results will be given based on these two points (i and ii) for the Case 2 to Case 5 described in Section 4.3.

Table 4.5.3.2: The calculating results based on mean depth

| **Cases** | **Total mean depth** | **Total mean depth to exits** | **Total mean distance to exits** |
|---|---|---|---|
| ***Case 2*** | 22.4444 | 27.5 | 152.805 |
| ***Case 3*** | 40.7059 | 38.5 | 254.731 |
| ***Case 4*** | 64.9286 | 65.75 | 787.362 |
| ***Case 5*** | 146.091 | 153 | 1037.3 |

The calculation results shown in Table 4.5.3.2 will be used to compare with building complexity measure in Section 4.5.3.3. Hence, the analysis to these results will be described in Section 4.5.3.3.

### 4.5.3.3 Comparing results and analysis

Applying the Distance Graph method, the egress complexity for C*ases 2 to 5* is presented in section 4.5.2.1. The complexity values can be used to compare with the simulation results using buildingEXODUS and the calculated results based on the mean depth (Table 4.5.3.3).

Table 4.5.3.3: Comparison of results

| Cases | Global Building complexity | Simulation Time (population of 15) | Simulation Time (population of 100) | Total mean depth | Total mean depth to exits | Total mean distance to exit |
|---|---|---|---|---|---|---|
| Case 2 | **279.852** | 34.75 | 112.09 | **22.4444** | **27.5** | **152.805** |
| Case 3 | **982.711** | 45.14 | 54.25 | **40.7059** | **38.5** | **254.731** |
| Case 4 | **2994.75** | 61.79 | 77.59 | **64.9286** | **65.75** | **787.362** |
| Case 5 | **5718.05** | 47.56 | 61.76 | **146.091** | **153** | **1037.3** |

For the simulation result, the smaller graphs demonstrated the more serious congestion when the number of people was increased. In the first building (Case 2), the congestion is a very serious factor due to the total exits width being smaller then the other buildings (Case 3 to 5). In the other buildings, the congestion is not very obvious when the number of people is increased. This is due to the total width of exits being sufficient.

The evacuation results in buildingEXODUS were obtained under normal evacuation process. There are no hazards in these buildings. In the buildingEXODUS evacuation model, it was assumed that every evacuee has complete knowledge of the building. The evacuees will always choose the nearest exit via the shortest path. If the congestion is not an important factor during the evacuation process then the maximum travelling distance will be the most important factor influencing the evacuation time.

When using the new method to calculate the building complexity, this model has assumed that the occupants have no knowledge of the building. It mainly considers the complexity of the process of recognizing the building. This means that smaller value of complexity value has the better evacuation ability.

According to the analysis, the building complexity and the simulation in buildingEXODUS check the different aspects of the building. The results of the building complexity analysis were not always consistent with the results generated in the simulation in buildingEXODUS.

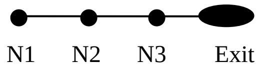
From the comparing table (Table 4.5.3.3), we can find the values are consistent using all of these graph measure. However, from the definition it is easy to see that the mean depth measure just considers the exit vertex as normal graph vertex, so it is obvious that the mean depth measure did not consider the number of exits. This measure is therefore not suitable for comparing different building for evacuation purpose. The other two improving measures (see Table 4.5.3.2) are also simple measures, hence are not good methods for comparing different building.

### 4.5.4 Limitations

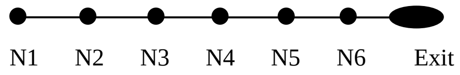
New building complexity measures have been employed to compare the evacuation ability for different buildings. The main task of the Distance Graph method is to extend Donegan's method for any graph with circuits. The THEOREM 4.5.2 ensures that the *Distance Graph Method* obtains the best results as outlined by Donegan [DP96]. Another approach has also been introduced in this chapter that extends Donegan's method by redefining the concept of the information step in which the distance to each travel step is now considered.

However, the *Distance Graph Method,* since it is derived from the Donegan's method, shares some of its limitations. Firstly, some vital factors which influence the evacuation are not accounted for in Donegan's and *Distance Graph Method*. These factors include the maximum travel distance and the maximum travel time of the wayfinding process. According to the traditional building codes, these factors are all very important and influencing the evacuation performance.

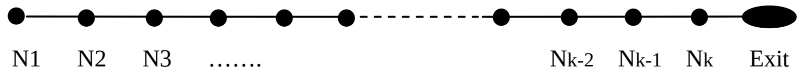
For describing the other important limitation, the following graphs are introduced here:



Graph 4.5.4.1: Sample Graph 1



Graph 4.5.4.2: Sample Graph 2



Graph 4.5.4.3: Sample Graph 3

Based on the node information formula, the node information of N1 obtained a zero value in these three tree graph (Graph 4.5.4.1 to 4.5.4.3). According to the definition of Donegan, the node information expresses the node complexity degree. It seems not suitable that N1 obtain the same values (zero) in these graphs. Obviously, N1 in the last graph should have a higher value associated with it than that in other graphs.

Based on these limitations, some new measures will be developed in Chapter 5 for comparing different building for the purpose of evacuation ability.

## 4.6 Summary

For comparing different buildings according to their evacuation ability, Donegan et al. proposed the egress complexity theory so that complexity could be equated to the uncertainty associated with a naive occupant in exhaustive pursuit of an exit without the benefit of signage. When using egress complexity to compare two or more structures (buildings) with respect to evacuation ability, the building with the smallest value of complexity is deemed to be the simplest to navigate. However, this method can only be used to calculate the complexity of a tree graph; normally a building graph is not as simple as a tree.

The author has extended Donegan’s method’s theories and applications to obtain a new *Distance Graph Method*, which considers travel distance, and the application of this measure to graphs with circuits. The *Distance Graph Method* has been used to compare different building for the purpose of evacuation. However, these methods mainly focus on the amount of information possessed by an occupant of a building during the wayfinding process. So some weaknesses are still present as described in Section 4.5.4. This method does not include the maximum travel distance and time, which are important factors influencing the evacuation ability. On the other hand, this method did not provide a reasonable node complexity value for some nodes according to Graphs 4.5.4.1 to 4.5.4.3, shown in Section 4.5.4. To address these limitations, some new building complexity measures will be presented, in Chapter 5, which can be used to compare different buildings for the purpose of their evacuation ability.

# Chapter 5: Development of New Graph measures for evacuation analysis based on room graph

## 5.1 Introduction

In this chapter, a set of building complexity measures will be developed to compare different buildings for the purpose of evacuation ability. The theory of these measures comes from the analysis of the *Distance Graph Method*. Hence resulting is a new nodal complexity measure. By extending the original nodal complexity measure described in Chapter 4, a set of new measures are obtained which meet the objectives of the travel distance and the travel time to find an available exit. The travel distance and travel time are important factors as introduced in most evacuation models and the traditional safety codes, for example [BD82] [TH94] [Ta91]. To verify the validation of these new measures the test cases presented in Chapter 4 were used to analyse them. The results obtained from the new model were validated against simulated results generated by buildingEXODUS and the calculated results obtained by the *Distance Graph Method* from Chapter 4.

In Chapter 4, Donegan’s method and *Distance Graph Method* have been used to compare different building for the purpose of evacuation. However, these methods mainly focus on the amount of information possessed by an occupant in a building. During the wayfinding process, information is gained gradually, hence reducing a person’s uncertainty of the building. However, the maximum travel distance and the travel time are very important factors which influencing evacuation ability. These vital factors for an occupant searching for an available exit were not considered by Donegan. The models described in Chapter 4 did not give an acceptable node information measure for some nodes. Due to the limitations of these measures, some improved measures which represent the node complexity are required for buildings. Hence if these new measures are applied to locations within a building, a more reasonable result will be generated. Furthermore, for comparing different buildings for their evacuation ability, an acceptable explanation and results based on these node measures will be described in this chapter.

Travel distance and time are important factors in wayfinding process, therefore the following questions need to be answered: '*how far and how long will occupants positioned at any location within the building need to travel to find an available exit?* ' Obviously, for any given building, suppose a naive occupant is positioned at a random location within the building, minimizing the travel distance or time during the wayfinding process is a basic objective of the occupant. So the following question also needs attention: '*what is the maximum distance the occupant needs to travel before successful egress from the building, in other words, what is the maximum time the occupant needs to spend to find an available exit.* ' Therefore, for buildings where the occupants has no knowledge of the structure, a building with the smaller travel distance or time to find an exit is better for evacuation than the one with a larger travel distance or time.

## 5.2 Development of the theories behind the 'new measures'

### 5.2.1 Basic theories of new measures

All the new measures generated in this chapter are derived from reviewing both Donegan's and the *Distance Graph Method*. Before outlining these new measures, we will further analyse the previous methods. A tree graph representation of the building is also the start point here.

In the following, G is assumed to be a tree graph representation of building structure. If *G* is a graph with circuits, then we break the circuits in accordance with the method described in Chapter 4 to generate a potential tree $G_1$ of *G*, which will be applied in the following discussion.

For any given node $N_i$, based on the original calculation formulas, for example [DP94] [DP96] [DT99], the node information is calculated by:

$$I_i\ (e) = n^+ \log_2 \frac{n^+ + n^-}{n^+} + n^- \log_2 \frac{n^+ + n^-}{n^-}$$

According to the concept described by Donegan [DP96] [DT99], *the degree of exit complexity* is calculated by dividing the exit complexity by the number of non-exit nodes. It expresses the average value of the node information for all non-exit nodes. In reality, the node information of a given node $N_i$ represents the complexity degree of the node. The *node complexity degree* addressed the following question: For an occupant positioned at node $N_i$ within a building, at most how much information would that occupant need to accumulate in order to find exit *e* successfully.

For comparing the *node complexity degree* for different nodes within a graph, the following conclusion can be drawn:

***Theorem 5.1:*** if *G* is a tree graph representation of building plan, $N_i$ and $N_j$ are two non-exit nodes of *G*, then $I_i\ (e) < I_j(e)$, if $n_i^- < n_j^-$.

***Proof.***

Considering the calculating formula of node information:

$$I_i\ (e) = n^+ \log_2 \frac{n^+ + n^-}{n^+} + n^- \log_2 \frac{n^+ + n^-}{n^-}$$

In this formula, $n^+$ is a constant and the value equals the number of the non exit nodes. If we replace $n^+$ by n, then considering the derivative of the corresponding real function:

$$I_i\ (e) = n\log_2 \frac{n+y}{n} + y\log_2 \frac{n+y}{y}$$

According the formula $\log_a b = \frac{\ln a}{\ln b}$, $I_i\ (e)$ can be expressed by:

$$I_i\ (e) = \ln_2 e\left[n\ln\frac{n+y}{n} + y\ln\frac{n+y}{y}\right]$$

Then

$$\frac{d(I_i\ (e))}{d(y)} = \log_2 e\left[n\times\frac{n}{n+y}\times\frac{1}{n} + \ln\frac{n+y}{y} + y\times\frac{y}{n+y}\times\frac{y-(n+y)}{y^2}\right]$$

$$= \log_2 e\left[\frac{n}{n+y} + \ln(1+\frac{n}{y}) - \frac{n}{n+y}\right]$$

$$= \log_2 e\times\ln(1+\frac{n}{y})$$

$$= \log_2 (1+\frac{n}{y})$$

Obviously $\frac{d(I_i\ (e))}{d(y)} > 0$.

Which means $I_i\ (e)$ is increasing with respect to y.

So $I_i\ (e) < I_j\ (e)$ is correct, if $n_i^- < n_j^-$. Finish.

By the theorem 5.1, $I_i\ (e)$ is increasing with respect to *n*- for any given graph. That means $I_i\ (e)$ will increase when the negative information steps are increased within a building. One question that needs to addressed is what is the range of $I_i\ (e)$. For answering this question, the following analysis process is required:

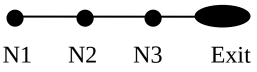
According to theorem 5.1 where 0<$n^- < n^+$, also $n^-$ is integer. So it is easy to draw the following conclusion:

$$0 \le I_i\ (e) \le \left(n^+\log_2\frac{2n^+ -1}{n^+} + (n^+ -1)\log_2\frac{2n^+ -1}{n^+ -1}\right) = \textbf{constant}$$

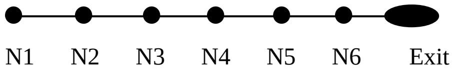
Also, $\left(n^{+}\log_2\frac{2n^{+}-1}{n^{+}}+(n^{+}-1)\log_2\frac{2n^{+}-1}{n^{+}-1}\right)<\left(n^{+}\log_2\frac{2n^{+}}{n^{+}}+n^{+}\log_2\frac{2n^{+}}{n^{+}}\right)=2n^{+}$.

So $0\le I_i\,(e)<2n^{+}$.

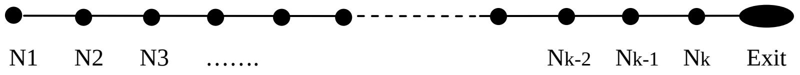
Furthermore, recalling the results in Chapter 4, the node information of N1 is all zero in these three tree graphs (see Graphs 5.2.1 to 5.2.3). Obviously, this result is not suitable because N1 in the last graph should obtain the maximum value as described in Chapter 4.



Graph 5.2.1: Sample graph 1



Graph 5.2.2: Sample graph 2



Graph 5.2.3: Sample graph 3

Based on these discussions, another formula to represent the nodal complexity needs to be found which should satisfy the following considerations:

(1) It should be consistent with Donegan node information calculating formula. If we use $S(i)$ (i=1, 2…K) denoting the nodal complexity of the new methods. The following result need to be ensured: if $I_i\,(e)<I_j\,(e)$, then $S(i)$ $<S(j)$ (i=1, 2…K)

(2) The range of values employed by the new measure should satisfy 0<$S(i)$ $<2n^{+}$ for any nodes in a given graph.

(3) The new *nodal complexity* applied by the new method can distinguish the different complexity for N1 in the three graphs shown in Graphs 5.2.1 to 5.2.3.

Considering the simple calculating formula:

$$S(i)=n_i^{+}+n_i^{-}$$

Where $n_i^{+}$ and $n_i^{-}$ have the same definition as in the Donegan's method.

For any given building, $n_i^+$ is a constant (i=1, 2…k). According to the theorem 5.1, it is easy to show that this formula satisfy (1). And 0< $n_i^+ \leq S(i) = n_i^+ + n_i^-$ <2 $n_i^+$ also ensures that $S(i) = n_i^+ + n_i^-$ satisfy (2). In addition for the three tree graphs shown in Graphs 5.2.1 to 5.2.3, the values of $S(1)$ are 3, 6, and K respectively. This can be explained as for an occupant who evacuate from N1 in these graphs, then the maximum they need to travel 3, 6 and K steps respectively to evacuate from the corresponding building.

Furthermore, the new *nodal complexity* $S(i) = n_i^+ + n_i^-$ also satisfies the following important conclusion when calculating complexity measures for graphs with circuits:

***Theorem* 5.2:** If $G_1$ is a *Potential Spanning Tree* as defined in Chapter 4 (see Definition 4.5.1 ) of a graph G with circuits about the *Exit E***,** then $G_1$ obtains the maximum *nodal complexity* for any non-exit node when comparing all other spanning trees.

**Proof:** Assuming $G_1$ denotes a *Potential Spanning Tree* of graph G about *Exit E*, and $G_2$ is a random spanning tree of G. $S_{G_1}(i)$ is the *nodal complexity* of the $i^{th}$ node in $G_1$ about *Exit E*, $S_{G_2}(i)$ is the $i^{th}$ *nodal complexity* of $G_2$ about *Exit E.* we just need to prove $S_{G_1}(i) \geq S_{G_2}(i)$.

For all the graphs which are spanning trees of the Graph G, then they contain the same nodes, also they have the same number of nodes, the same number of edges. For any given $N_i$**,** if applying $d$ represent the potential for $N_i$ in each tree, then $d_1$ denotes the potential value for $N_i$ in $G_1$, $d_2$ denotes the potential value for $N_i$ in $G_2$**.** According to the Definition 4.5.1: $N_i$ will have the minimum potential steps in $G_1$, so $d_1 \le d_2$, where $n^+$ is the number of the non-exit nodes for any given tree graph, $n^+$ is a constant. Also $n^- = n^+ - d$ is always correct in a tree as discussed in Chapter 4. So the new *nodal complexity* $S(i) = n_i^+ + n_i^-$ can be represented as $S(i) = 2n^+ - d$

Hence, $\left(S_{G_1}(i) = 2n^+ - d_1\right) \ge \left(S_{G_2}(i) = 2n^+ - d_2\right)$

**Finished.**

In reality, the formula $S(i) = n_i^+ + n_i^-$ answers the following question: *For an occupant positioned at node* $N_i$ *within a building, what is the maximum number of travel steps that the occupant would need to accumulate in order to egress successfully from the building*.

### 5.2.2 A mathematical comparison of the new nodal complexity against the Distance Graph Method

To understand the difference between $S(i)$ and $I(i)$ for any give tree graph, a mathematical explanation will be introduced in this section. Now considering the derivative of the corresponding real function of $I(i)$:

$$f(x) = n\log_2 \frac{n+x}{n} + x\log_2 \frac{n+x}{x}$$

Where n is a constant for a given tree graph.

Consider the tangent line at the point x=n, according to the discussion in Section 5.2.1, then

$$\frac{d(f(x))}{dx} = \log_2(1+\frac{n}{x})$$

When x=n, then: $\left.\frac{d(f(x))}{dx}\right|_{x=n} = \log_2(1+\frac{n}{n}) = 1$

So the tangent equation of $f(x)$ should be $g(x) = x + n$, which is the derivative function of $S(i)$.

In the following, two functional images of $f(x)$ and $g(x)$ are given for n= 5 and 10 respectively (Figure 5.2.1 and Figure 5.2.2). In these figures, for any given $n$ then $f(x) \le g(x)$, and 0≤g(x) -f(x) ≤n. Now considering the range of 0≤x≤n, then g(0)-f(0)=n, and g(n)-f(n)=0. Therefore the following conclusion can be drawn:

$f(x)$ is tending to $g(x)$ during the process of increasing x from 0 to n.

Now the measures $S(i)$ and $I(i)$ are considered again, the $n^-$ just has a variable ranging from 0 to n. the maximum value of $S(i)$ - $I(i)$ just occurs when $n^-$=0 which is the location of N1 in the graphs shown in Section 4.5.4.

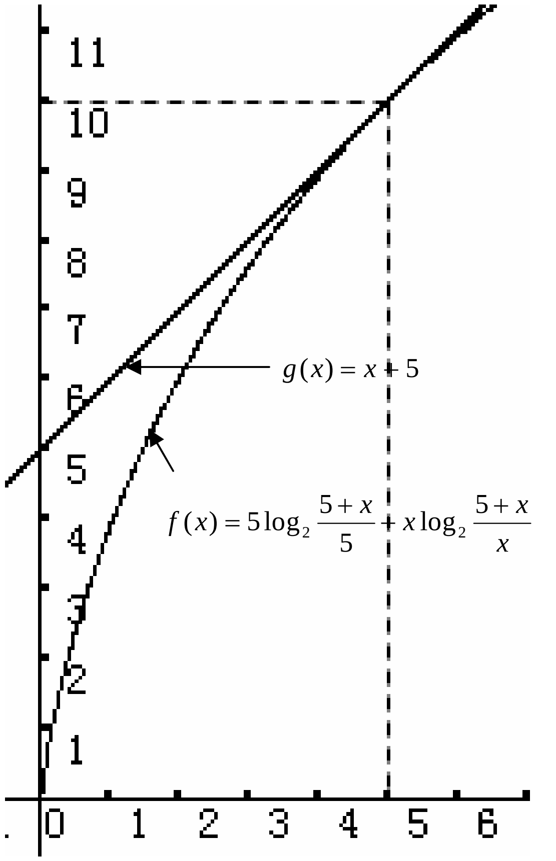


Figure 5.2.1: Function images for 5 room nodes

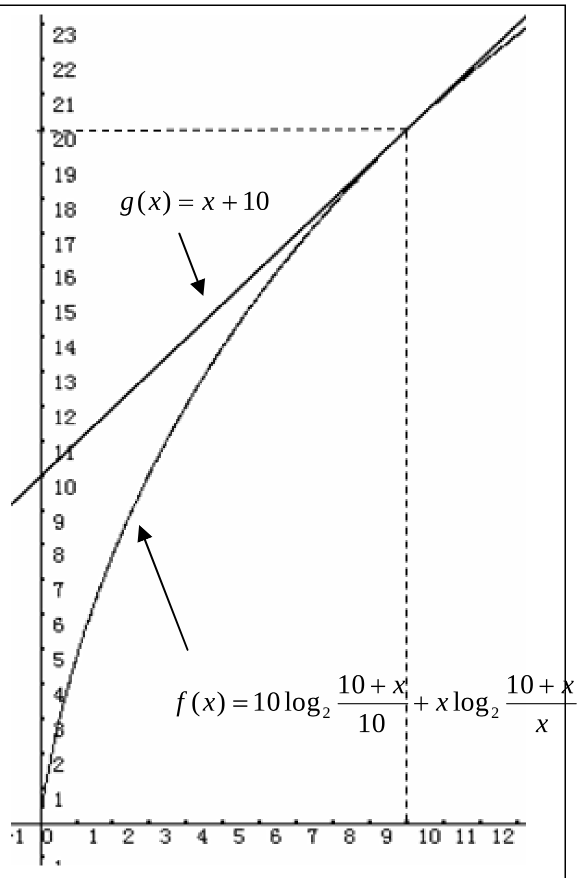


Figure 5.2.2: Function images for 10 room nodes

## 5.3 The Implementation of new measures

Based on the discussion in Section 5.2, $S(i) = n_i^+ + n_i^-$ should be used to calculate the nodal complexity, which answers the following question: for an occupant positioned at node $N_i$ within a building, what is maximum number of travel steps that the occupant would need to accumulate in order to egress successfully from the building. The extension of this idea will be employed in this chapter as a measure of nodal complexity. As described in Chapter 4, $d^+$ and $d^-$ are used to replace $n^+$ and $n^-$. Where $d^+$ represent the traveling distance of gaining positive instance of information, $d^-$ represent the traveling distance of gaining negative information. Then $S(i) = d_i^+ + d_i^-$. In reality, $S(i) = d_i^+ + d_i^-$ is the solution to the following question: '*If a naive occupant is positioned at* $N_i$ *within a building, what is the maximum distance occupant needs to travel before successful egress from the building?* '

The travel time is also an important issue that needs to be considered in the new measure. If this measure is used to compare two different building with different residents, maybe the user would like to compare the maximum travel time to find an available exit. So here we keep the discussion to the maximum travel time. However, if we assume the travel speed is uniform, for example, 1.5m/s has been used in a number of evacuation models as the normal travel speed [EP99] [EM94].

In the following, we will describe all measures based on a maximum travel distance and maximum travel time.

### 5.3.1 The building with single floor, single exit and no circuits applying a room-based graph

In this section, a simple building will be examined to explain the actual idea of the new measures. In addition the building will also be represented by a room-based graph generated in buildingEXODUS. All the buildings in the following examples have just one exit, and the corresponding room-based graphs are simple trees. A basic algorithm is counting the maximum travelling distance for a person randomly placed at any location exiting via the only external door, then calculating the relevant time according to the travel speed. The building complexity measure is the powered average value of travel distance or travel time of the occupant starting from each non-exit node to the external exit.

#### 5.3.1.1 The maximum travel distance (or time) to find the external exit for a person at a given node

For explaining the new measure of nodal complexity, two simple building plans will be considered (Figure 5.3.1 and Figure 5.3.2). By analysing these plans, the maximum travel distance can be calculated.

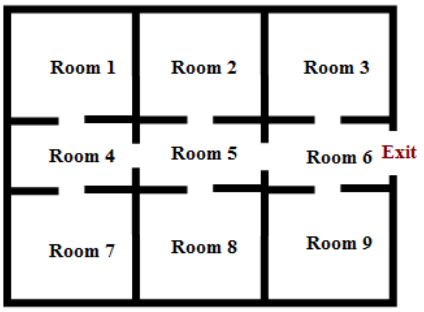
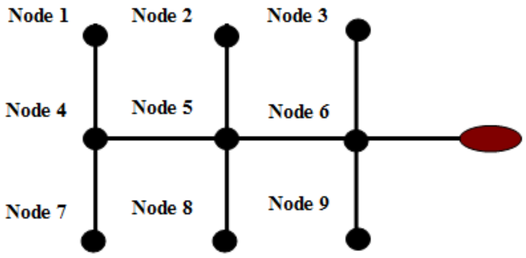


Figure 5.3.1: Building C



Graph 5.3.1: Room-based graph of Building C

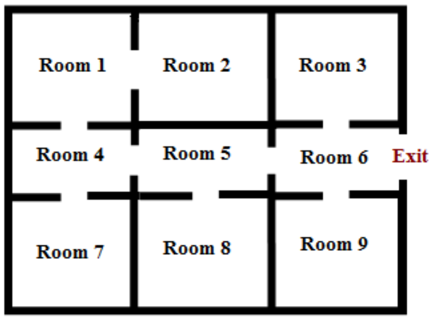
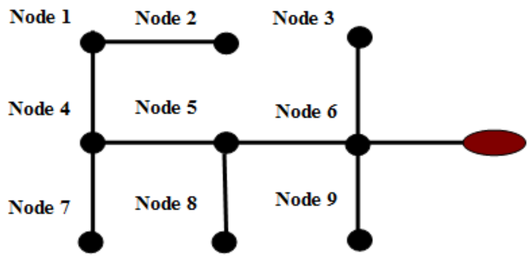


Figure 5.3.2: Building D



Graph 5.3.2: Room-based graph of Building D

The Graph 5.3.1 is a room-based graph representation of the building plan shown in Figure 5.3.1. In addition the Graph 5.3.2 is the room-based graph of the building plan shown in Figure 5.3.2. Obviously, building plan shown as Figure 5.3.2 is almost identical to Figure 5.3.1 except for the location of an internal door, which means these two buildings have the same number of rooms and the same area, the only difference being the connections. The areas of the rooms for these two buildings are shown as Table 5.3.1

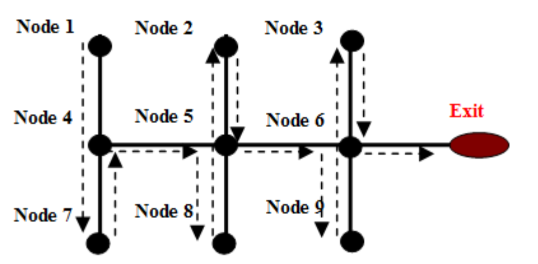
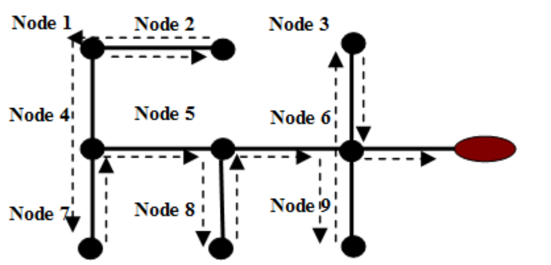
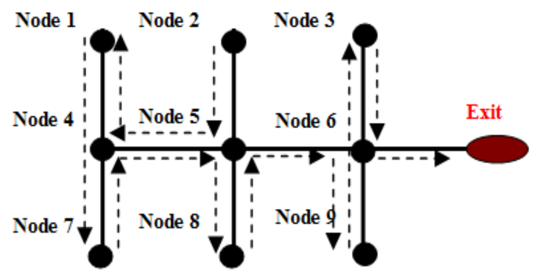
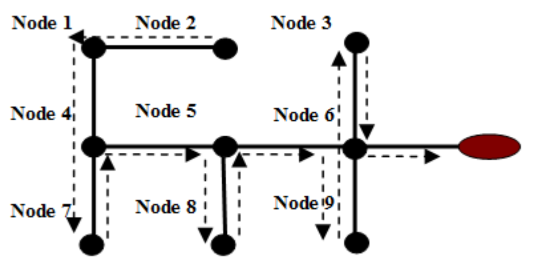
Table 5.3.1: The areas of the rooms for the Buildings A and B (see Figures 5.3.1 and 5.3.2)

| **Room number** | **Areas(m × m=m$^2$)** |
|---|---|
| 1 | 10×10=100 |
| 2 | 10×10=100 |
| 3 | 10×10=100 |
| 4 | 5×10=50 |
| 5 | 5×10=50 |
| 6 | 5×10=50 |
| 7 | 10×10=100 |
| 8 | 10×10=100 |
| 9 | 10×10=100 |

For an occupant positioned at any given location within the building, for example in room 1, i.e. on the Node 1 of the room-based graph, a basic question is what is the maximum distance that the occupant will need to travel to find the exit. Or what is the maximum time that the occupant will need to spend to find the only external exit based on the following assumption: the occupant has no previous knowledge of the structure but path memory.

The Graphs 5.3.3 to Graph 5.3.6 are the sketch maps to explain the maximum travel distance (or the maximum time) to find the external exit for an occupant who starts from room 1 or room 2. In Graph 5.3.3, the broken lines indicate the route for the occupant starting from node1 in Figure 5.3.1. For the occupant who start from node 1, the maximum distance for him to find the external exit should be the sum of all the length of these broken lines, this distance can be denoted using the symbol “D”, Then if the travel speed is denoted by “V”, the maximum travel time should be T=D/V. The distance of the occupant has travelled is the sum of the route length obtained by sweeping the tree, until the exit node is reached.

In Figure 5.3.1, it is easy to see that the distance is 107.5 meters, and if we assume the travel speed is 1.5 m/s (see Section 5.1), then the travel time is 71.7 seconds. Here if the travel speed is represented by “V”, the maximum travel distance and the maximum travel time obtained the same results when comparing structures. However, in most situations, the travel speed will be restricted by a number of factors, for example: the Occupant’s gender, age, the width of the connect edges, even the purpose of using the building, and so on. But for the purpose of explaining clearly the methods introduced here, we just assume the travel speed is 1.5m/s in this chapter which is the typical travel speed employed in most evacuation model.

| | |
|---|---|
|   Node 1<br>Node 2<br>Node 3<br>Node 4<br>Node 5<br>Node 6<br>Exit<br>Node 7<br>Node 8<br>Node 9 |   Node 1<br>Node 2<br>Node 3<br>Node 4<br>Node 5<br>Node 6<br>Node 7<br>Node 8<br>Node 9 |
| Graph 5.3.3: Route for an occupant who start from node 1 in Building C (Figure 5.3.1) | Graph 5.3.4: Route for an occupant who start from node 1 in Building D (Figure 5.3.2) |
|   Node 1<br>Node 2<br>Node 3<br>Node 4<br>Node 5<br>Node 6<br>Exit<br>Node 7<br>Node 8<br>Node 9 |   Node 1<br>Node 2<br>Node 3<br>Node 4<br>Node 5<br>Node 6<br>Node 7<br>Node 8<br>Node 9 |
| Graph 5.3.5: Route for an occupant who start from node 2 in Building C (Figure 5.3.1) | Graph 5.3.6: Route for an occupant who start from node 2 in Building D (Figure 5.3.2) |

Graph 5.3.4 shows the route for the occupant who start from node 1 in Figure 5.3.2, the maximum distance and the time are 112.5 meters and 75 seconds respectively. The Graph 5.3.5 shows the route for an occupant starting from node2 in the Figure 5.3.1. The maximum distance and the time are 117.5 meters and 78.3 second respectively. The Graph 5.3.6 shows the route for an Occupant who start from node 2. In the Figure 5.3.2, the maximum distance is 102.5 meters, and the maximum travel time is 68.3 seconds. All the calculated results are shown in Table 5.3.2 and Table 5.3.3.

Table 5.3.2: The maximum travel distance and time from node 1 and 2 for Building C (see Figure 5.3.1)

| **Nodes** | **The maximum travel distance to find the exit (metres)** | **The maximum travel time to find the exit(assuming a travel speed of 1.5m/s)(seconds)** |
|---|---|---|
| 1 | 107.5 | 71.7 |
| 2 | 117.5 | 78.3 |

Table 5.3.3: The maximum travel distance and time from node 1 and 2
for Building D (see Figure 5.3.2)

| **Nodes** | **The maximum travel distance to find the exit (metres)** | **The maximum travel time to find the exit(assuming a travel speed of 1.5m/s)(seconds)** |
|---|---|---|
| 1 | 112.5 | 75 |
| 2 | 102.5 | 68.3 |

### 5.3.1.2 Some complexity measures.

In Section 5.3.1.1, the relevant distance and travel time have been analysed for node1 and node2 for Buildings A and B (Figure 5.3.1 and Figure 5.3.2). Here all other values for all nodes have been given, the results for Building C and are shown in Table 5.3.4, and the results for Building D are shown in Table 5.3.5

Table 5.3.4: Results for Building C (see Figure 5.3.1)

| **Nodes** | **The maximum travel distance to find the exit (metres)** | **The maximum travel time to find the exit(assume a travel speed of 1.5m/s)(seconds)** |
|---|---|---|
| 1 | 107.5 | 71.6667 |
| 2 | 117.5 | 78.3333 |
| 3 | 127.5 | 85 |
| 4 | 115 | 76.6667 |
| 5 | 125 | 83.3333 |
| 6 | 135 | 90 |
| 7 | 107.5 | 71.6667 |
| 8 | 117.5 | 78.3333 |
| 9 | 127.5 | 85 |

Table 5.3.5 Results for Building D (see Figure 5.3.2)

| **Nodes** | **The maximum travel distance to find the exit (metres)** | **The maximum travel time to fine the exit(assume a travel speed of 1.5m/s)(seconds)** |
|---|---|---|
| 1 | 112.5 | 75 |
| 2 | 102.5 | 68.3333 |
| 3 | 132.5 | 88.3333 |
| 4 | 120 | 80 |
| 5 | 130 | 86.6667 |
| 6 | 140 | 93.3333 |
| 7 | 112.5 | 75 |
| 8 | 122.5 | 81.6667 |
| 9 | 132.5 | 88.3333 |

Now, for each node in the Graph 5.3.1 and Graph 5.3.2, the maximum travel distance and the maximum travel time are shown in Table 5.3.4 and Table 5.3.5. Now we ask the question which building is better for evacuation? Or which building is easier to find the exit for an occupant positioned at any random location within the two building? To answer these questions, we need to define some measures so that we can compare the buildings.

In the following section, some definitions with regard to building complexity measures will be introduced.

**Definition 5.3.1: Total Travel Distance (TD (i))** refers to the sum of maximum travel distance for each non-exit node to the Exit (i).

$$\textbf{TD (i)} = \sum_{j=1}^{N} D_{ji} \qquad \text{Formula 5.3.1}$$

In Formula 5.3.1, $D_{ji}$ refers the maximum travel distance from the j non-exit node to the Exit(i), $N$ is the total number of the non-exit nodes.

**Definition 5.3.2: Total Travel Time (TT (i))** is defined as the sum of maximum travel time for each non-exit node to the Exit (i).

$$\textbf{TT(i)} = \sum_{j=1}^{N} T_{ji} \qquad \text{Formula 5.3.2}$$

In Formula 5.3.2, $T_{ji}$ is the maximum travel time from the j non-exit node to the Exit(i), $N$ refers the total number of the non-exit nodes

According to Definition 5.3.1, the Total Travel Distance is 1080 meters in Figure 5.3.1, and the Total Travel Time is 720 seconds. In Figure 5.3.2 the Total Travel Distance is 1105 meters, and the Total Travel Time 737 seconds.

If these two measures are used to compare these two buildings, we can draw the following conclusion, the building represented in Figure 5.3.1 is better than that represented in Figure 5.3.2 for evacuation.

In these two buildings, there are the same number of rooms, so the TD(i) and the TT(i) are the effective measures for comparing these two building for evacuation ability. However, in most situations, the number of the rooms is different between buildings. For comparing the complexity of wayfinding process, the average value should be more reasonable measure. In the following, two new measures are defined to achieve this objective.

**Definition 5.3.3: Average Travel Distance (AD (i))** refers to the average value of maximum travel distance for each non-exit node to the Exit (i).

$$\textbf{AD (i)} = \frac{TD(i)}{N} \qquad \text{Formula 5.3.3}$$

In Formula 5.3.3, $TD(i)$ is the Total Travel Distance shown in Definition 5.3.1, $N$ is the total number of non-exit nodes.

**Definition 5.3.4: Average Travel Time (AT (i))** refers to the average value of maximum travel time for each non-exit node to the Exit (i).

$$\mathbf{AT\ (i)} = \frac{TT(i)}{N} \qquad \text{Formula} \quad 5.3.4$$

In Formula 5.3.4, $TT(i)$ is the Total Travel Time shown in Definition 5.3.2, $N$ is the total number of non-exit nodes.

These two equations described in Formulas 5.3.3 and 5.3.4 can be used to explain the following: if an occupant is positioned at a random location within the building, what is the maximum distance that they will travel to find the exit (i)? When assuming that the occupant has the same probability of starting from each room. Then the value of "AD(i)" will provide the solution. What is the maximum time the occupant will take to find the exit (i). The value of "AT(i)" can answer this question if we also assume that the occupants have the same probability of starting from each room.

According to Definitions 5.3.3 and 5.3.4, the Average Distance (i) is 120 meters and the Average Time (i) is 80 seconds for Building C (Figure 5.3.1). For Building D (Figure 5.3.2), the Average Distance (i) is 123 meters and the Average Time (i) is 82 seconds.

If these two measures are used to compare these two buildings, we can draw the following conclusion, Building C (Figure 5.3.1) is better than Building D (Figure 5.3.2) in the context of evacuation. This is the same conclusion as when using TD(i) and TT(i) as the measures.

### 5.3.1.3 Extending these measures to include an area factor

In Section 5.3.1.2 the measures AD (i) and AT (i) have been defined for comparing building complexity in relation to evacuation. The meanings of these two measures have been explained based on the hypothesis that an occupant has the same probability of starting from each room. In reality, the occupant would not have the same probability of being positioned in each room due to the area and the purposes of each room. However, the building complexity developed in this section is a scenario independent model, so the purpose of each room is beyond the consideration of this method.

When considering the factor of the area of the rooms, another hypothesis should be considered when describing distributing of the occupants within the building. This hypothesis is that the occupant has the same probability of starting from any location within the building. Based on this discussion, if an occupant is positioned at a random location in the building, he will have more probability of being in a larger room than a smaller room, and the probability will be a direct proportion to the area of that room.

As just discussed, two new measures will be defined here:

**Definition 5.3.5: Powered Average Distance (PAD(i))** is defined as the Powered average value of maximum travel distance for each non-exit node to the Exit (i).

$$\textbf{PAD (i)} = \sum_{k=1}^{N} \left( \frac{Area_k}{\sum_{j=1}^{N} Area_j} \times D_{ki} \right) \qquad \text{Formula} \quad 5.3.5$$

In Formula 5.3.5, $Area_k$ is the area of Room(k), $D_{ki}$ is the maximum travel distance for an occupant traveling from Room(k) to the Exit(i), $N$ is the total number of the non-exit nodes

**Definition 5.3.6: Powered Average Time (PAT(i))** is defined as the Powered average value of the maximum travel time for each non-exit node to the Exit (i).

$$\textbf{PAT (i)} = \sum_{k=1}^{N} \left( \frac{Area_k}{\sum_{j=1}^{N} Area_j} \times T_{ki} \right) \qquad \text{Formula} \quad 5.3.6$$

In Formula 5.3.6, $Area_k$ is the area of Room(k), $T_{ki}$ is the maximum travel time for an occupant traveling from room(k) to the Exit(i), $N$ is the total number of the non-exit nodes.

According to the Definitions 5.3.5 and 5.3.6, the Powered Average Distance is 119 meters and the Powered Average Time is 79 seconds for Figure 5.3.1. In Figure 5.3.2, the Powered Average Distance is 121 meters and the Powered Average Time is 81 seconds.

If these two measures are used to compare these two buildings, we can draw the following conclusion, Building C (Figure 5.3.1) is better than Building D (Figure 5.3.2) with respect to evacuation, which is the same result as using AD and AT

### 5.3.1.4 Brief summary

In last two sections, some building complexity measures have been defined based on the travel distance and the travel time. For explaining these new measures, two simple building plans have been the starting point (Figure 5.3.1 and Figure 5.3.2). By analysing and comparing two simple building structures, a serious of complexity definitions has been obtained. Here we will discuss these measures based on the measures for Building C (Figure 5.3.1) and Building D (Figure 5.3.2).

Table 5.3.6 gives a summary of the measures defined in last two sections for these two buildings.

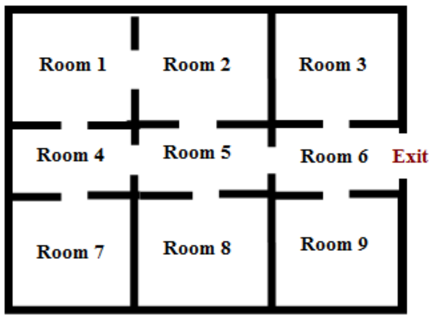
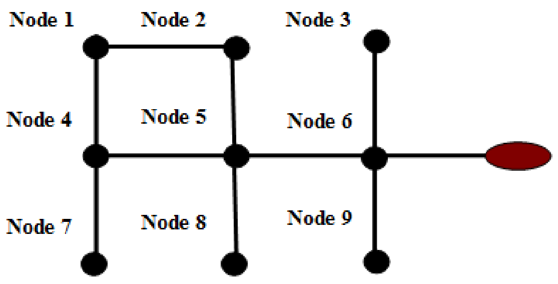
Table 5.3.6: Measures for Buildings A and B

| **Measures** | **Building C (Figure 5.3.1)** | **Building D (Figure 5.3.2)** |
|---|---|---|
| **TD (meter)** | 1080 | 1105 |
| **TT (second)** | 720 | 737 |
| **AD (meter)** | 120 | 123 |
| **AT (second)** | 80 | 82 |
| **PAD (meter)** | 119 | 121 |
| **PAT (second)** | 79 | 81 |

As described in Section 5.3.1.1, these two building plans have the same number of rooms, the same size for each room, but not the same connections. According to Table 5.3.6, Building C (Figure 5.3.1) is better for evacuation than Building D (Figure 5.3.2) in relationship to the complexity of the wayfinding process. All the measures are in the same order due to the same number of rooms and the same size of rooms. According to the discussion of these measures, in this chapter the PAD and the PAT are better measures for comparing different buildings for evacuation ability. Here we have assumed the travel speed is 1.5m/s, which the user could assign as a different value for different buildings. In this chapter it is the typical travel speed employed in most evacuation model. However, in most situations, the travel speed will be restricted by a lot of factors, for example: the occupant's gender, age, the width of the connect edges, even the purpose of using the building, and so on.

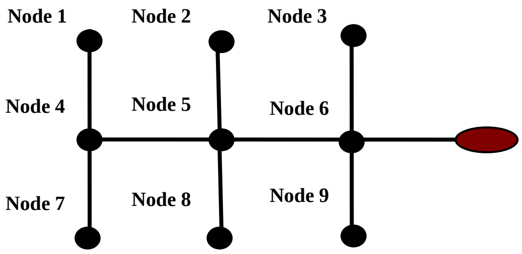
## 5.3.2 The PAD and PAT for a building with circuits and a single exit

In Section 5.3.1, we have introduced some new measures for comparing different simple buildings. The structure used in Section 5.3.2 demanded that the room-based graph representation of the building is just a simple tree, which means that there will be no circuits within the graph. In most situations, the room-based graph is not a simple tree. For example, Figure 5.3.3 shows a building with one circuit in its graph representation. However, it is impossible to calculate the maximum travel distance and time for a building with circuits to any given exit. In this section, a method for solving this problem for such graph is outlined.

|   Room 1<br>Room 2<br>Room 3<br>Room 4<br>Room 5<br>Room 6<br>Exit<br>Room 7<br>Room 8<br>Room 9 |   Node 1<br>Node 2<br>Node 3<br>Node 4<br>Node 5<br>Node 6<br>Node 7<br>Node 8<br>Node 9 |
|---|---|
| Figure 5.3.3: Building C | Graph 5.3.7: Room-based graph of Building C |

To extend the measures defined above to graphs with circuits, the same method employed in Chapter 4 will be applied here. For any given exit we need to break the circuits of the structure down so that we have a spanning tree. For any given exit, the distance of each node from the exit is calculated. The edges that generated the minimum distance route to the exit will be kept for each node. This is the *Potential Spanning Tree* as defined in Chapter 4 (see Definition 4.5.1). According to the ***Theorem 5.2,*** this is the worst case where the *nodal complexity* will obtain the greatest value.

Graph 5.3.8 give a *Potential Spanning Tree* after breaking the circuits for Graph 5.3.7, which forms the same graph as shown in Figure 5.3.1. Hence Building C (Figure 5.3.3) obtains the same PAD and PAT as Building C shown in Figure 5.3.1.



Graph 5.3.8: Potential spanning tree of Graph 5.3.7

## 5.3.3 Global complexity of a building with more than one exit

For a building with more than one exit, the method applied by Donegan will be employed to representation the global complexity here. For any given building, when additional exits are added to a building, the chance of evacuation from the building is improved. This should be reflected in a reduced building complexity value. The method of calculating the global complexity is based on the following definition as described by Donegan [DP96]:

'DEFINITION 11. *The global complexity C(S) is the exit complexity for a specified floor level defined by C(S) =* $\left[\sum(1/C_i)\right]^{-1}$*, where* $C_i$ *is the exit complexity for each exit.'*

Based on this definition, the global complexity based on PAD is defined as the Formula 5.3.7:

$$Global.Complexity_{pad} = \frac{1}{\sum_{t=1}^{m}(1/PAD_t)} = \frac{1}{\sum_{t=1}^{m}(1/\sum_{k=1}^{N}(\frac{Area_k}{\sum_{j=1}^{N} Area_j} \times D_{ki})_t)} \quad \text{Formula} \quad 5.3.7$$

If the global complexity is based on PAT, then the solution for calculating global complexity is given as the Formula 5.3.8:

$$Global.Complexity_{pat} = \frac{1}{\sum_{t=1}^{m}(1/PAT_t)} = \frac{1}{\sum_{t=1}^{m}(1/\sum_{k=1}^{N}(\frac{Area_k}{\sum_{j=1}^{N} Area_j} \times T_{ki})_t)} \quad \text{Formula} \quad 5.3.8$$

## 5.3.4 Solution of a building with multi-storey

Currently, there is not an effective method to describe a building complexity for a multi-storey building. In Donegan method, Donegan et al. applied vector quantities to represent building complexity for such kind of building. The basic idea is explained as: The stairwells on upper floor are considered as exit nodes and the stairwells on ground floor are considered as non-exit node.

Based on this method, a k-storey environment is considered, where *C(0), C(1), C(2)…C(k)* are used to represent the global complexity for each floor from ground to the $k^{th}$ floor. Then the complexity of the building is denoted by *(C (0), C (1), C (2)…C (k) )*. In reality, this method just gives a mark of the complexity for such a building. It cannot be applied for comparing of different building for evacuation capability since different buildings may have different number of storeys.

So, at the moment, an effective method has not been generated. The vector method could still be employed to represent the complexity of a multi-storey building. A more reasonable method should be developed which is left for future work.

# 5.4 Demonstration

## 5.4.1 Cases calculation results

In this section, all the test cases introduced in Chapter 4 will be used to show the performance of the new building complexity measures presented in this chapter. For all of these cases, the measures defined in Section 5.3 will be used to compare each building. In these results all the values related to time will assume a travel speed of 1.5 m/s. Hence the travel time will be applied for comparing different building from the building complexity point of view. In the following, the calculating results are described for each case respectively.

Table 5.4.1 give all the measures defined in Section 5.3 for Case 1 described in Section 4.3. In this case, there is only one external exit. The *Total Travel Time* to the only exit is 41.021s, the *Average Travel Time* for each non-exit node to the exit is 10.255s, and if we consider the areas of nodes, then the *Powered Average Time* is 10.569s, which is a little difference with the *Average Travel Time.* Due to there being only one exit for this case, so the *Global Complexity (PAT)* is equal to the *Powered Average Time.*

Table 5.4.1: Results obtained using the measures defined in Section 5.3 for Case 1

| ***Measures*** | **TD** | **AD** | **PAD** | **TT** | **AT** | **PAT** |
|---|---|---|---|---|---|---|
| Exit 0 | 61.532 | 15.383 | 15.855 | 41.021 | 10.255 | 10.569 |
| Global Complexity (PAD) | 15.855 | | | | | |
| Global Complexity (PAT) | 10.569 | | | | | |

The calculated results for Case 2 are shown as Table 5.4.2. In this case, there are two external exits. So the *Total Travel Time* to Exit0 is 371.585s and 408.075s to Exit1, the *Average Travel Time* are 46.4481s and 51.009s for Exit0 and Exit1 respectively. And the *Powered Average Time* is 45.7306s for Exit0, 51.5946s for Exit1 when considering the areas of the nodes. However, the *Global Complexity (PAT)* is only

24.2430s since there are two external exits.

Table 5.4.2: Results obtained using the measures defined in Section 5.3 for Case 2

| ***Measures*** | **TD** | **AD** | **PAD** | **TT** | **AT** | **PAT** |
|---|---|---|---|---|---|---|
| Exit 0 | 557.377 | 69.6721 | 68.5959 | 371.585 | 46.4481 | 45.7306 |
| Exit 1 | 612.113 | 76.5141 | 77.3919 | 408.075 | 51.0094 | 51.5946 |
| Global Complexity (PAD) | 36.3645 | | | | | |
| Global Complexity (PAT) | 24.2430 | | | | | |

In Case 3, there are also two external exits, same as in Case 2. The corresponding calculated results for Case 3 are shown in Table 5.4.3. In this case, the *Total Travel Time* obtained is 1118.84s for Exit0 and 1140.59s for Exit1. However, the *Average Travel Time* is only 74.5891s and 76.0394s respectively for two external exits. The *Global Complexity (PAT)* is only 38.4492s.

Table 5.4.3: Results obtained using the measures defined in Section 5.3 for Case 3

| ***Measures*** | **TD** | **AD** | **PAD** | **TT** | **AT** | **PAT** |
|---|---|---|---|---|---|---|
| Exit 0 | 1678.25 | 111.884 | 114.270 | 1118.84 | 74.5891 | 76.1801 |
| Exit 1 | 1710.89 | 114.059 | 116.446 | 1140.59 | 76.0394 | 77.6304 |
| Global Complexity (PAD) | 57.6738 | | | | | |
| Global Complexity (PAT) | 38.4492 | | | | | |

In Case 4, the number of external exits is 4. And the number of non-exit nodes is 29(see Table 4.5.2.1). All the calculated results are shown in Table 5.4.4. From Table 5.4.4, the *Total Travel Time* are all more than 8000s and the *Average Travel Time* is also more than 300s for each exit. However, the *Global Complexity (PAT)* is only 83.9908s since there are 4 external exits in this case.

Table 5.4.4: Results obtained using the measures defined in Section 5.3 for Case 4

| ***Measures*** | **TD** | **AD** | **PAD** | **TT** | **AT** | **PAT** |
|---|---|---|---|---|---|---|
| Exit 0 | 12211.1 | 488.444 | 498.051 | 8140.73 | 325.629 | 332.034 |
| Exit 1 | 12636.9 | 505.477 | 505.994 | 8424.61 | 336.984 | 337.329 |
| Exit 2 | 12433.7 | 497.348 | 497.865 | 8289.13 | 331.565 | 331.910 |
| Exit 3 | 12842.8 | 513.711 | 514.229 | 8561.85 | 342.474 | 342.819 |
| Global Complexity (PAD) | 125.986 | | | | | |
| Global Complexity (PAT) | 83.9908 | | | | | |

Table 5.4.5 give the measures for Case 5. In this case, there are 3 exits and 48 non-exit nodes (see Table 4.5.2.1). Obviously, the *Total Travel Time* has obtained a very high value which is greater than 11000s for each exit, which is generated due to the large number of nodes in this case. However, the *Average Travel Time* is around 270s for each exit which is less than the *Average Travel Time* in Case 4 (around 330s).Currently, the *Global Complexity (PAT)* is still a little higher than that in Case 4 since there is one less exit than Case 4.

Table 5.4.5: Results obtained using the measures defined in Section 5.3 for Case 5

| ***Measures*** | **TD** | **AD** | **PAD** | **TT** | **AT** | **PAT** |
|---|---|---|---|---|---|---|
| Exit 0 | 17109.9 | 407.38 | 408.693 | 11406.6 | 271.586 | 272.462 |
| Exit 1 | 17372.8 | 413.638 | 414.405 | 11581.9 | 275.759 | 276.27 |
| Exit 2 | 17119.5 | 407.607 | 408.741 | 11413 | 271.738 | 272.494 |
| Global Complexity (PAD) | 136.8653 | | | | | |
| Global Complexity (PAT) | 91.2435 | | | | | |

From the measures of Case 1 to 5 shown in Table 5.4.1 to 5.4.5, it is easy to identify that the Case 1 obtained the minimum complexity value, and the Case 5 obtained the maximum value. Therefore the results seem reasonable using *Global Complexity (PAT)*. According to the definitions of these measures and the analysis in Section 5.3, the *Total Travel Time* obtained is a high value when there are many nodes in the room graph representation of a building. So the *Total Travel Time* is not a suitable measure to compare the building complexity. And the *Total Travel Time* is just a logical increase as the number of nodes increase. So the *Global Complexity (PAT)* will be used to compare the building complexity in this chapter. The detail analysis of the

*Global Complexity (PAT)* for Case 1 to 5 will be discussed in next section, which also will be compared to the simulation results using buildingEXODUS and the calculated results based on the *Distance Graph Method*.

## 5.4.2 Comparing results and analysis

Currently, we cannot find any benchmark for verifying the validation of the measures generated in this chapter. Hence an evacuation simulation model is still chosen as the validation tool as used in Chapter 4. In this chapter, we will still apply buildingEXODUS as the validation tool. So we will compare building complexity results with simulation data generated by buildingEXODUS. We will also compare the results obtained for Case 2 to 5 (see Section 5.4.1) by the *Distance Graph Method*. As described in Section 4.5.3.1, Case 1 is not suitable for comparing simulation results using buildingEXODUS. So in this section, the comparing and analysis will just be based on *Cases 2 to 5*.

.

Applying the *Global Complexity (PAT)*, the building complexity for *Cases 2 to 5* have been given in section 5.4.1. The complexity values can be used to compare with the simulation results using buildingEXODUS and the calculated results based on the *Distance Graph Method* in Chapter 4 (Table 5.4.5).

Table 5.4.5: Comparing results

| Cases | Global Complexity (PAT) | Distance Graph Method | Simulation time(15 populations) | Simulation time(100 populations) |
|---|---|---|---|---|
| Case 2 | 24.2430 | 279.852 | 34.75 | 112.09 |
| Case 3 | 38.4492 | 982.711 | 45.14 | 54.25 |
| Case 4 | 83.9908 | 2994.75 | 61.79 | 77.59 |
| Case 5 | 91.2435 | 5718.05 | 47.56 | 61.76 |

When using *Global Complexity (PAT)* and the *Distance Graph Method* to calculate the building complexity, these models assumed that the evacuees have no knowledge of the building. It mainly considers the complexity of the process of recognizing the building. That means the smaller the value of the complexity value the better evacuation ability. From Table 5.4.5, it can be seen that *Global Complexity (PAT)* and *Distance Graph Method* obtain the same order of building complexity for Cases 2 to 5. These models all indicate that the building in Case 2 is the simplest and Case 3 is the most complex to find an available exit during the evacuation process.

When comparing simulation results generated by buildingEXODUS for Case 2 to 5, the results shown in Table 5.4.5 indicate that they did not obtain the same order. The evacuation time for Case 5 is smaller than Case 4. Actually, using buildingEXODUS, the evacuation model has assumed every evacuee has complete knowledge of the building. The evacuees will always choose the nearest exit via the shortest path. According to the analysis, the building complexity and the simulation in buildingEXODUS check different aspect of the building. Therefore, the results of the building complexity analysis were not always consistent with the result generated in buildingEXODUS.

The calculated results generated using *Distance Graph Method* also indicate that the building complexity value of Case 3 is more than 3 times than that of Case 2, and Case 5 is about 2 times than that of Case 4. However, the *Global Complexity (PAT)* indicates that the building complexity of Case 3 is less than 2 times when compared to Case 2, and Case 5 is a little bit more complex when compared to Case 4. These results are generated mainly because the *Distance Graph Method* applies the sum of *nodal information* as the complexity for a given exit, and *Global Complexity (PAT)* applies the *Powered Average Time* as the complexity for a given exit. When checking the simulation results generated by buildingEXODUS, it seems that the *Global Complexity (PAT)* obtained a more reasonable result than *Distance Graph Method.*

## 5.5 Limitations

The New *Global Complexity (PAT)* measure and the *Distance Graph Method* developed in Chapter 4 have been employed to compare the evacuation ability for different building. However, some limitations are present based on the following reasons:

Firstly, all of the graph measures are all based on a room graph representation of a building to determine the building complexity. However a room-based graph does not have the ability to describe the actual routes taken during the wayfinding process. In most situations, the room-based graph cannot express the real environment of the building exactly, also the edges in the room-based graph cannot represent the actual route the evacuee will travel to an exit. So such a simple outline of the structure by just specifying the connectivity between compartments is not sufficient for the required level of complexity needed to simulate the wayfinding process.

Secondly, these models applied the worst case scenario to calculate the nodal complexity measure. In other words, occupants need to sweep the whole building plan in completing the wayfinding tasks. As introduced in Chapter 4, the *Distance Graph Method* considers how much information an occupant would need to accumulate in order to egress successfully from the building. And the information refers to the maximum amount of information which the evacuee needs to sweep the whole room-based graph. In this chapter, the *Global Complexity (PAT)* measure considers the travel time for an evacuee to find an available exit based on a room-based graph. However, the time used in this section also refers to the worst case situation. This model calculates the maximum travel time for an occupant completing a wayfinding task.

Thirdly, these models are developed based on only one external exit. For the building with more than one exit, these methods calculate a global building complexity according to a mathematical formula. So these models do not consider all exits in a global perspective.

## 5.6 Summary

For comparing different buildings for the purpose of their evacuation ability, The *Distance Graph Method* has been generated by extending Donegan's method from Chapter 4. However, some limitations have been highlighted as described in Chapter 4. By analysing these limitations, a set of new building complexity measures have been developed in this chapter. By comparing and analysing these new measures, the *Global Complexity (PAT and PAD)* have been shown to be valid measures. These new measures have considered some important factors such as wayfinding time, travel distance and the areas of rooms. When using the *Global Complexity (PAT and PAD)* to compare two or more structures with respect to evacuation ability, the building with the smallest value of complexity is considered to be the simplest to navigate. The results generated by *Global Complexity (PAT)* for some test cases have been validated against simulated results generated by buildingEXODUS and the calculated results obtained by the *Distance Graph Method*. However, the *Global Complexity (PAT)* and the *Distance Graph Method* still have some weakness, see Section 5.5. Based on the limitations described in Section 5.5, a new graph representation needs to be generated, and this new kind of graph should have the ability to describe a more exact travel route for the wayfinding process of an occupant within the building. Also, new building complexity measures based on this new graph representation will be developed to resolve these limitations in Chapter 7.

# Chapter 6 Route-based graph for building complexity measures

## 6.1 Introduction

In this chapter, a new kind of graph representation for a building structure is outlined which will be employed to generate a new building complexity model in Chapter 7. The new graph representation is called a new ‘route-based graph’, which will have the capability to describe the routes within a building and the room entities. The initial objective is to generate such kind of graph for the employment of building complexity models as introduced in Chapter 4 and 5. However, based on the limitations of these models (see Chapter 5), a new kind of complexity model will be introduced in the next chapter based on the new graph representation described here.

In the last two chapters, the *Distance Graph Method* which was introduced in Chapter 4 and the *Global Complexity (PAT)* measure described in Chapter 5 have been described for comparing different buildings for the purpose of building complexity. These measures can be used to compare different buildings for the purpose of evacuation ability. And some reasonable results have been produced. However, these measures are all based on the room-based graph representation of a building, which in most situations cannot express the real environment of the building exactly. Also the edges in the room-based graph cannot describe the real route the evacuee needs to travel. So such a simple outline of the structure and just specifying the connectivity between compartments is not enough for the required level of complexity to simulate the wayfinding process.

The serious weakness of the *Distance Graph Method* and the *Global Complexity (PAT)* measure requires that an improved graph representation need to be used. This kind of graph should have the ability to describe a more accurate travel route during the wayfinding process for an occupant within the building.

Axial map and visibility graph representation methods which have been introduced in Chapter 3 were generated for the purpose of the analysis of architectural space.

However, they do not have the capability to describe an acceptable travel route within a building.

In this chapter, the term ‘navigation’ is applied broadly, the explanation of which is essential this thesis. It is generally accepted that people navigate around a structure mainly using two levels of navigation comprising of a local and a global strategy [Pa84]. Each level influences different aspects of the individual’s movements. Local navigation relates to the low-level navigation required to avoid an obstacle or move out of a room. This relates to short-term navigational decisions. Global navigation is involved in deciding which route an individual adopts from their current location to their target destination e.g. an exit. This in turn is based on the occupant’s knowledge of the structure. This global strategy is referred to as wayfinding [Pa84]. In this chapter ‘navigation’ mainly refers to the global strategy, i.e. wayfinding.

## 6.2 The initial purpose of generating a new ‘route-based graph’

To explain the purpose of generating the new graph representation, we will consider the building complexity models introduced in the last two chapters. The *Distance Graph Method* and *Global Complexity (PAT)* measure have been employed to compare different building for the purpose of evacuation ability, which obtained some reasonable results. These two models are all based on the process of navigation for an occupant starting at any location within a building, and calculating the maximum information or travel time during the navigation process to obtain the complexity measures. These models are based on a room graph representation of a building.

However, there are many weakness of applying a room-based graph to represent a building plan for the purpose of simulating the wayfinding process. An example is presented here to demonstrate these limitations. The building plan denoted by Figure 6.2.1 has been applied to the analysis of the building complexity models in last two chapters. The set of solid lines in this figure indicate the room-based graph representation of the building plan. A process of navigation will be considered in the following based on the room-based graph. Considering the navigation process for an occupant who starts from N1 to the Door_2 in the building, his travel routes should be described as: N1->N2->N3->N4->Door_2 (see Figure 6.2.1). The bold lines will be employed to indicate the travel route for him based on the room-based graph and the sum of length of these bold lines are used to calculate his travel distance.

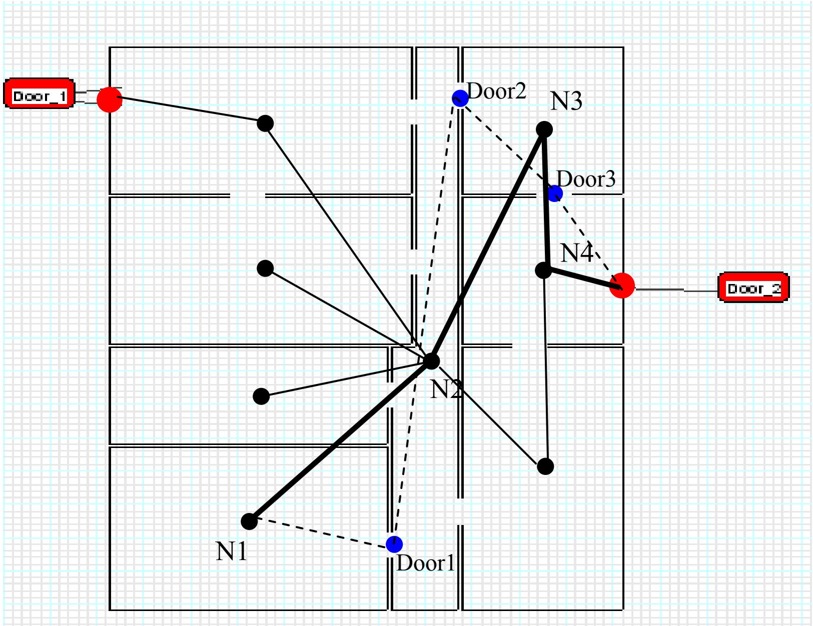


Figure 6.2.1: A scheme for a navigation from N1 to Door_2

Obviously, the room-based graph did not give the route. In reality, the travel route for the occupant would follow the path: N1->Door1->Door2-Door3->Door_2, see Figure 6.2.1. The broken line expresses the travel route for the occupant exactly.

In most situations, the room-based graph does not have the ability to express the real environment of the building exactly. Especially, the edges within the room-graph only denote the simple connection between compartments. They do not point out the real route the evacuee would need to travel. Such simple outline of the structure and just specifying the connectivity between rooms or compartments is not enough for the required level to simulate the navigation process.

Based on the analysis above, an improved graph representation will be generated. Ideally, the new graph should have the capability to describe the exact travel routes used during wayfinding process.

## 6.3 The graph representations of the buildings

Before introducing the new 'route-based graph' representation of building plan, a review of the present graph representation for a building is essential. These graph representation methods have been described in Chapter 3. They are described as: room-based graph, axial map, and the visibility graph. In the discussion of room-based graph, a detail description of the weakness has been outlined in Section 6.2. In this section, we mainly focus on the other two presentation of building plan.

Firstly, the concept of axial map was first introduced by Hillier and Hanson on 1984[HH84], they introduce the visibility relationships into graph analysis of buildings, and constructed the set of axial lines for a building, which are the fewest longest lines of sight and access in the building which traverse all the convex spaces within the building system. In use, the axial map is the connection of lines between the nodes, and the intersections of lines as connections between the nodes. According to the original definition, the most controversial problem in applying the technique involves the definition of axial lines. There is no unique method for their generation; different users generate different sets of lines for the same application. Hence generating unique axial maps is impossible. Each researcher will inevitably draw slightly different maps [TP05] [BR04]. Normally, the map with the fewest-lines is referred to as the 'axial map'. A typical approach in space syntax is to construct an axial map for public space by drawing a set of axial lines, which represent the minimum number of visible lines that cover all the space in question. The definition of an axial line was defined exactly by Penn et al. [PC97], they defined an axial line as any line that joins two intervisible vertices within the system in one of the following ways: (a) both intervisible vertices are convex; or (b) one is convex and one is reflex vertex; or joining the vertices can be and is extended through open space past the reflex vertex; or (c) both are reflex, and the line joining the vertices can be and is extended through open space past both vertices. According to this description of axial map, this kind of graph can be employed to assess the spatial structure, but not used to describe the travel routes with in the building.

Another type of graph representation of a building plan is the visibility graph. Normally, the concept of the visibility graph is the set of non crossing line segments in the plane. This can be traced back to 1979 [LW79], Lozano-Perez and Wesley used the visibility graphs as an approach to solve the shortest path problem in the plane. This kind of graph was described as: Let S be a set of n non-intersecting line segments in the plane. The visibility graph G(s) of S is then the graph that has the endpoints of the segments in S as nodes and in which two nodes are adjacent whenever they can see each other [ME88]. The important thing in the visibility graph is how to choose the vertex used in the graph. In the visibility graph used for landscape analysis [DM94], the vertexes of the graph were selected from a triangulated irregular network. In computational geometry and artificial intelligence[DK97], the vertexes of the visibility graph were selected from the corner of a set of simple polygonals, the visibility graph was defined as followings: Let S define a set of simple polygonal obstacles in the plane, then the nodes of the visibility graph of S are just the vertices of S, and there is a visibility edge between vertices V and W if these vertices are mutually visible. Another visibility graph for the analysis of architectural apace was generated by Turner et al. [TD01]. They applied a set of isovists to generate a graph of mutual visibility between locations. In Tuner's visibility graph, they take a grid of many (thousands) of points across the space rather than selecting a few key locations. Certainly, the architectural visibility graph also uses selected locations as vertices and a mutual visibility relationship to forms edges, thus all the graphs are of identical form.

According to the above descriptions of all kind of visibility graphs, the present methods of defining a visibility graph are not enough to describe the travel routes within a building. However, the visibilities relationships, as introduced into graph analysis, have made it possible to describe route graphs within a building, if the key points were choose according the positions which are vital for any occupant during wayfinding process. Even though, another important factor to describe the routes in the building also need to considered, which refer to the direction of the edges also needs to be defined.

The methods discussed here are mainly used to measure the relative accessibility of different locations in a spatial system. Both Axial map and Visibility graph are not used to analyse evacuation and navigation processes. None of these graphs have enough information to describe the travel routes within a building.

## 6.4 The Background of Route Graph

Navigation has always been an interdisciplinary topic of research, because mobile agents of different types are inevitably faced with similar navigational problems [WB00]. All kind of activities for traveling from one place to another are filled in our life, these activities mention one important event which is navigation in a spatial environment [HW99] [WB00]. However, routes are a concept commonly encountered when dealing with human spatial navigation. The concept of a *route* has been defined in the literature [SB00] [HW99] [Ku00]. A route is a sequence of decision points that are connected by segments. Depending on the problem domain, this general definition can be instantiated for different real world scenarios, such as railroad connections between large cities, paths in a park, or corridors in a building [HW99].

The concept of a Route Graph was first introduced by Werner et al. in 2000 [WB00], which have been employed for navigation by various agents in a variety of scenarios. However, the concept was introduced by Werner et al. to mediate terminology between artificial intelligence and psychology in spatial cognition. This concept is not restricted to human users. They mainly focus on expressing the concepts of route-based navigation in a common scientific language. However, this specification provides a base for generating a new 'route-based graph' for a building plan.

The main concepts around Route Graphs as defined by Werner et al. [WB00] include *Route, Place, Path along a Route, Route Segment, and Route graph.* Some of them come from a number of publications, for example, the definitions of Route and Route Segment can be found by Hunt and Waller [HW99] and Werner et al. [WB00]. These definitions can be described as followed [WB00]:

1. ***A Route*** is a concatenation of directed Route Segments from one Place to another.
2. ***A Place*** is a tactical decision point where to continue, i.e. about the choice of the next Route Segment.
3. ***A Path along a Route*** is distinct from a Route. It embodies the dynamic usage of a Route or a contiguous part of it.
4. ***A Route Segment consists*** of two Places, a Source and a Target, which are connected by a 'course'. The course contains information that allows a subject to follow the Route Segment.

**5.** ***Union of Routes into Route Graphs.*** The representation of Places and Route Segments corresponds to the mathematical notion of a directed graph with a set of nodes (Places) and edges (pairs of Source and Target in Route Segments), both enriched with their particular attributes (there may be several edges between a given pair of nodes, in both directions). The union of different Routes into Route Graphs therefore corresponds directly to the union of the corresponding sets of nodes and sets of edges, respectively. Figure 6.4.1 gives an example to explain the *Union of Routes into Route Graphs.*

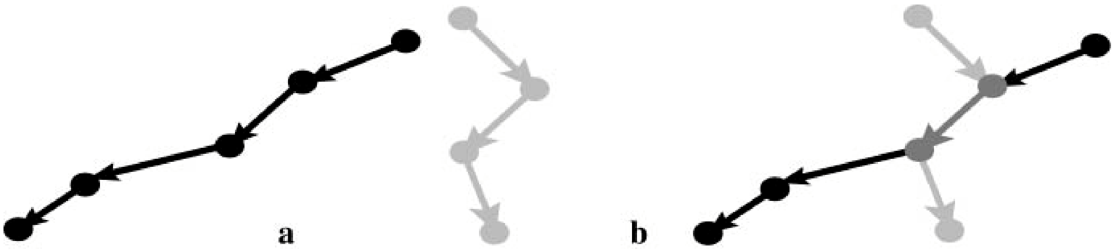


Figure 6.4.1 Union of two separate Routes (a) into one Route Graph (b) [WB00]

Based on the work of Werner et al. [WB00], some other formal definitions around route graphs are defined by Krieg-Bruckner et al. [KF05]. They gave alternative definition of the concepts of ***Segments*** **and Place** as: an edge of a Route Graph is directed from a source node to a target node. An edge (a route) is called a ***segment*** when it has three additional attributes: an *entry*, a *course* and an *exit*. Exactly what information is associated with these attributes is specifically defined for each type of Route Graph.  For example, an entry to a highway may denote a particular ramp, an exit another, while the course is just characterized by a certain length. A ***place*** is a node of a Route Graph, it has a particular position and orientation.

## 6.5 Fine node route-based graph representation in buildingEXODUS

Before the new ‘route-based graph’ representation for a building plan is introduced, another representation method which is already available in buildingEXODUS will be described briefly here.

The fine node route-based graph generated in buildingEXODUS is based on the fine mesh geometry. Currently this kind of graph includes four types of nodes which are internal door node, external Door, waypoint node, and room node. Figure 6.5.2 gives an example that shows this kind of route-based graphs. The Figure 6.5.1 is an image of a typical structure with one exit and eight internal doors.

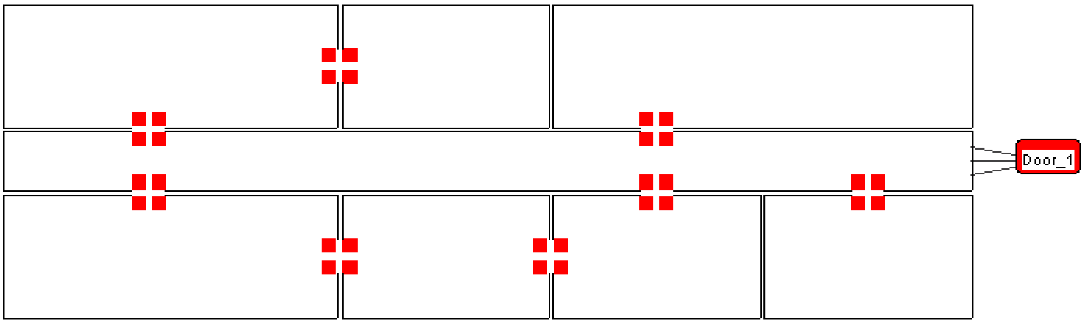


Figure 6.5.1: Example building, red squares are internal doors, Door_1 is the external exit

For the building plan shown in Figure 6.5.1, the fine node route-based graph of this building can be automatically generated in buildingEXODUS, and is shown in Figure 6.5.2.

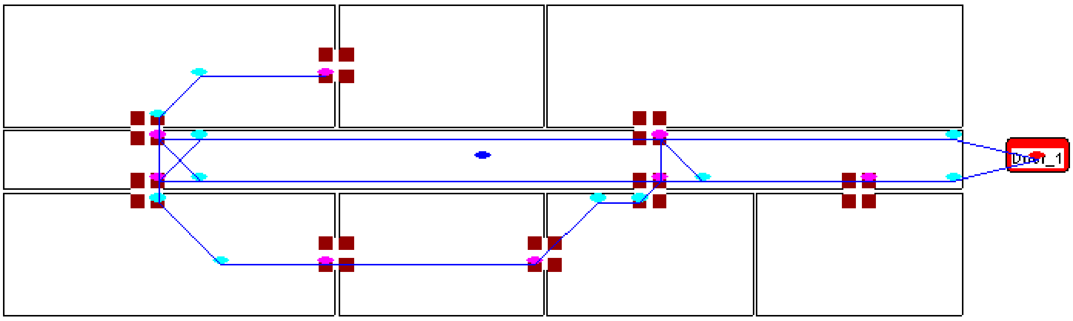


Figure 6.5.2: Route Graph of Figure 6.5.1

In Figure 6.5.2, there are three types of nodes indicated by different coloured points. The way point nodes are indicated by the light blue colour. Internal doors are indicated by magenta colour, and the external doors are indicated by red colour.

However, for the building complexity analysis applied in next chapter, on this route graph, additional information is required which relates to the position of each room and the route nodes that belong to it. Therefore, additional nodes are generated in this graph. In the following graph the blue nodes represent these room nodes. These room nodes are then connected to all route nodes associated with that room shown as green edges. The complete fine node route-based graph is shown in Figure 6.5.3.

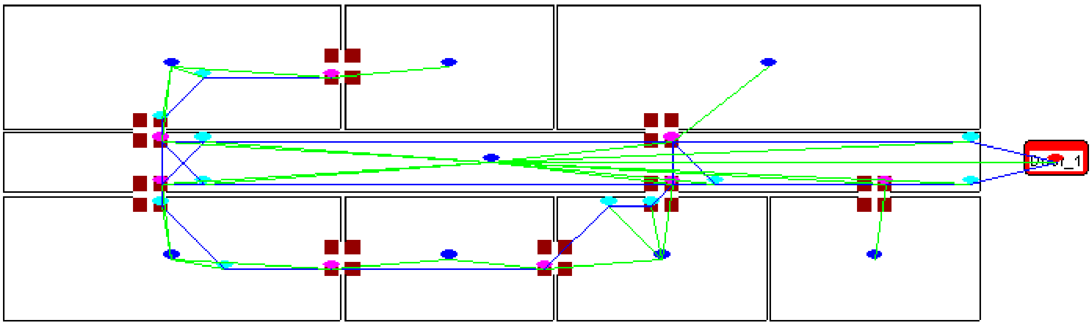


Figure 6.5.3: Full Route Based Graph

The fine node route-based graph will be utilised to generate the building complexity models presented in the next chapter. However, some limitations still exist due to the following reasons: firstly, during the process of generating the fine node route-based graph in buildingEXODUS, the waypoint node was generated based on the fine mesh. However, the locations of the waypoint nodes are uncertain. That means there are a lot of different choices for the waypoint locations. Hence, this type of graph cannot describe accurately the actual routes utilised during the wayfinding process. Secondly, for a building with a symmetrical structure, this method did not generate a symmetrical graph. The limitations of utilising this type of graph are discussed in more detail in Chapter 7. Therefore, in the next section a new 'route-based' graph representation will be outlined.

## 6.6 The new 'route-based graph' representation

### 6.6.1 The new 'route-based graph' objectives

The main reason for using the new 'route-based graph' representation is to improve the *Distance Graph Method* and the *Global Complexity (PAT)* measure developed in the last two chapters. However, except for replacing the room-based graph representation applied in the *Distance Graph Method* and the *Global Complexity (PAT)* measure, the new 'route-based graph' will also be applied to generate a new building complexity model. The reason behind the generation of a new complexity method came from the analysis of the limitations described in Chapter 5.

The models introduced in the last two chapters have used the worst case to calculate the nodal complexity. As shown in Chapter 4, the Donegan method and the Distance Graph Method considers how much information an evacuee would need to accumulate in order to egress successfully from the building. In addition the information refers to the most information which the evacuee needs to sweep the whole room-based graph. The *Global Complexity (PAT)* measure considers the travel time for an evacuee to find an available exit based on a room-based graph. However, the time used in this section also refers to the worst situation. This model calculates the most travel time for an occupant to complete the wayfinding task. Certainly, all of these models have assumed that all the evacuees have path memory ability. The *Distance Graph Method* and the *Global Complexity (PAT)* measure are developed based on only one external exit. For buildings with more than one exit, these methods calculate the global building complexity according to a mathematical formula but do not consider all exits from a global point of view. The relativity locations between all exits cannot be represented within these models.

Based on these limitations, the new 'route-based graph' representation will also be applied to find a more reasonable measure in which to calculate the complexity of building.

## 6.6.2 The definition of new 'route-based graph'

### 6.6.2.1 Basic assumptions of wayfinding within a new 'route-based graph'

To generate an acceptable route-based graph for a building, the first task is to find all the possible key locations for an occupant who starts the process of wayfinding. wayfinding in buildings can be an extremely complex task, which can be confused or assisted by numerous factors, such as the visual access of the structure, the positions and messages of any signs, the presence of structural makers, and the connectivity of the spaces within the structure itself.

However, according to the building complexity methods introduced in the last two chapters, the wayfinding process for an occupant within the building will be conducted based the following assumptions:

1. The process of wayfinding is done in the absence of signage.
2. There is no influence by others. The evacuation process will be completed

independently of any occupants within the building.

3. The occupant has no previous knowledge of the environment.
4. All exits are equally likely to be chosen.
5. The occupant has path memory ability (note that path memory ability refers to the evacuees' ability to remember all the routes and nodes which have been previously travelled).

In reality, these complexity measures are all independent of scenario, and are developed based only on the structure of building plan, hence the graph representation needs to reflect this.

### 6.6.2.2 The definitions of internal exits nodes, waypoints nodes and route segments within them

For explaining the definition of the new 'route-based graph', a simple building plan will be employed shown in Figure 6.6.1. In this building plan, there are three rooms and a hall. There is also a rectangle obstacle in the hall, see shade region Figure 6.6.1, and only one external exit.

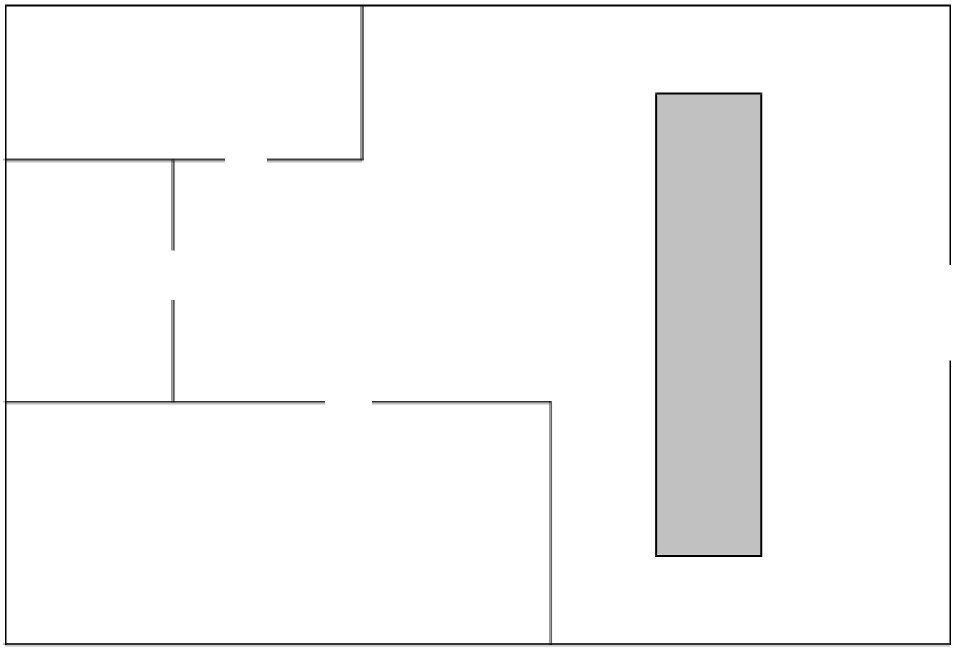


Figure 6.6.1: a simple building plan, the rectangle shadow represents an obstacle.

Based on the assumptions of the wayfinding process for occupants within this building plan, the wayfinding process will be analysed in the following. By analysing this process, two types of node will be defined according to all possible travel nodes during wayfinding process. Considering an occupant who starts from N1 (see Figure 6.6.2), his first travel step is moving toward Door1. After he gets to Door1, Figure 6.6.2 shows all the possible locations where the occupant can move to. Figure 6.6.3 shows the situation for an occupant starting from N2, and the first step for this person is to travel to Door2. Figure 6.6.3 descripts all possible locations which can be chosen from Door2.

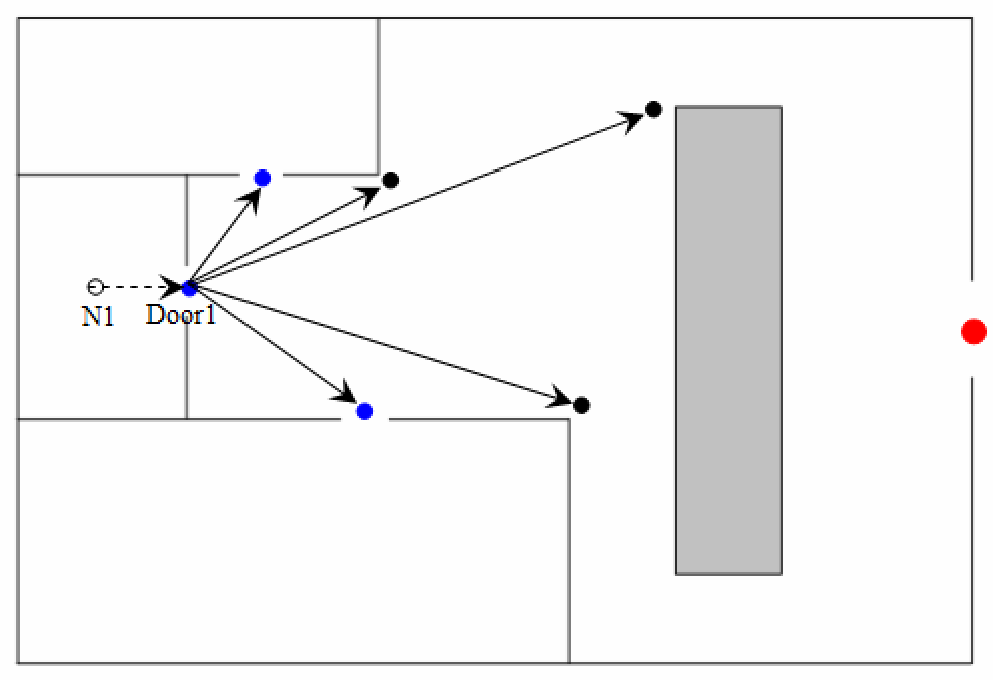


Figure 6.6.2: Outlining the wayfinding for an occupant starting from N1, the blue dots refer the internal exit nodes, black dots refer to waypoints

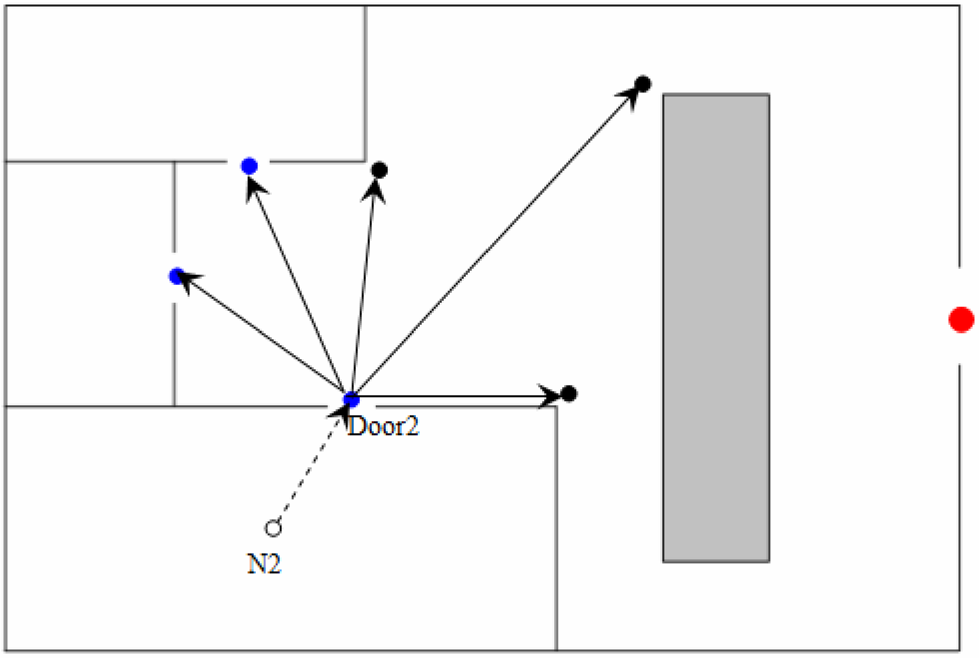


Figure 6.6.3: Representing the wayfinding for an occupant starting from N2, the blue dots refer to the internal exit nodes, black dots refer to waypoints

In Figure 6.6.2 and Figure 6.6.3, some of the possible locations for an occupant to move towards represent two types of node. The first type of node represents some possible exit location shown as the blue dots. This kind of node is defined as the internal exit node. Within a building, the internal exit nodes are easy to identify. We simply choose the middle location of each internal exit as the exit node. Certainly, these exit nodes are not an occupant's only choice. Maybe the occupant would like to move towards another type of location shown as the black dots in Figure 6.6.2 and Figure 6.6.3, this type of node is defined as the waypoint. After an occupant arrives at one of the waypoints, they will choose the next destination. The definition of the waypoints is explained as: waypoints are locations within the building plan that people use as points of reference when navigating around the geometry. Waypoints are usually placed near the location of a concave vertex, since these represent locations where people may need to change direction.

For a given building, all internal exit and waypoint nodes represent all nodes of the navigation systems. Each node is connected to all its visible neighbours. The graph generated by such nodes forms the ***navigation graph*** of the region in question. In such graph, all edges with in them are defined as route segments. However, the definition of route segment is differing from the definition by Werner and Krieg-

Bruckner [WB00] [KF05], here the route segment within the navigation graph is defined as an edge which is bidirectional and connects two visibility locations within this graph. These locations represent nodes however there is no information regarding which compartment these nodes are contained in. In reality, according to these definitions, this means all the routes within this graph are reversible. Figure 6.6.4 shows the navigation graph within this building plan. Note the external exit is not included in the navigation graph. In this graph, all the edges are bidirectional.

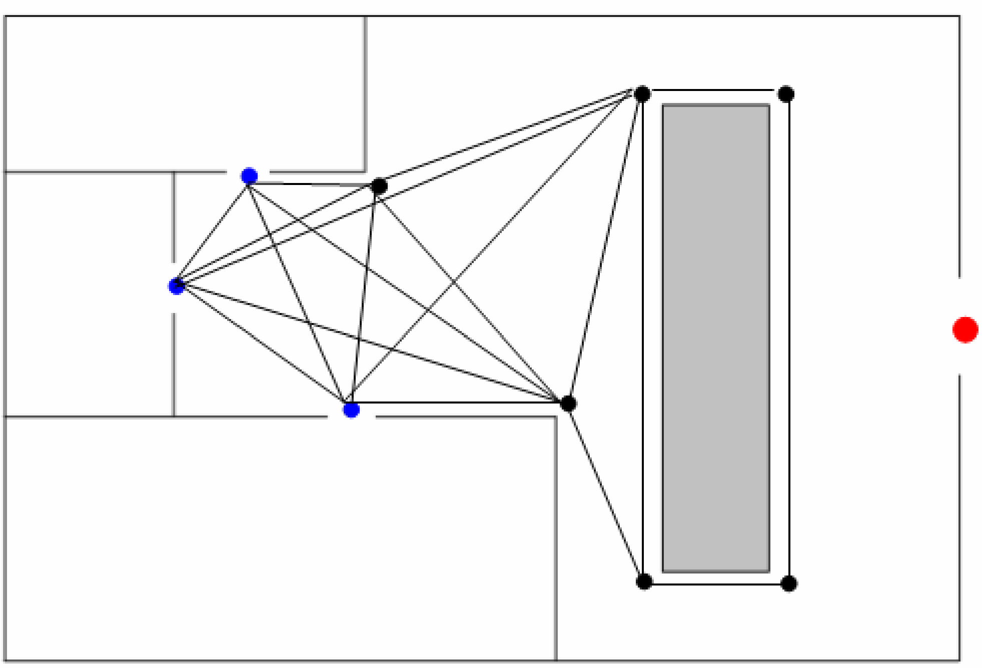


Figure 6.6.4: The navigation graph of building plan from Figure 6.6.1, the blue dots refer the internal exit nodes, black dots refer waypoints

### 6.6.2.3 The definitions of room nodes, external nodes and route segments connected with them

For completing the definitions of the new ‘route-based graph’, the nodes representing the starting locations and the final goal need to be defined next.

***Room nodes***

To define a starting location where an occupant will move from during a wayfinding process, the room node needs to be introduced. The occupant can start moving from any location within the building. Obviously, any location within the building plan must lie somewhere which is contained within a compartment. To obtain the coarse network model, the room node will be defined in the building route-based graph. This is the same process as that in room-based graph. Basically, a suit of compartments for the building will be defined as compartment nodes. These compartments represent rooms, workplaces, hallway, stairwell, lobby, refuge area etc. However, there are no direct connections between any two room nodes. Room nodes need to be connected to all the visible nodes in the navigation graph. Certainly, in some situations, the room node could also be connected to an external exit node which will be introduced in the next section. The route segments connected from a room node to any other node is defined as an edge which is directed. This can be explained as: after the 'first step' moving from a room node to any other node, an occupant will never travel to a room node during the wayfinding process.

### *External exits nodes*

The last node type will be used to describe the ultimate destinations for an occupant during the wayfinding process. This kind of node is also known as an external exit node. Exit nodes should be connected to all visible nodes in the navigation graph and to the room nodes they are contained in. The arcs connected with an external exit node are directed and point to the external exit node. These arcs are route segments.

The Figure 6.6.5 gives a new 'route-based graph' representation of the simple building plan.

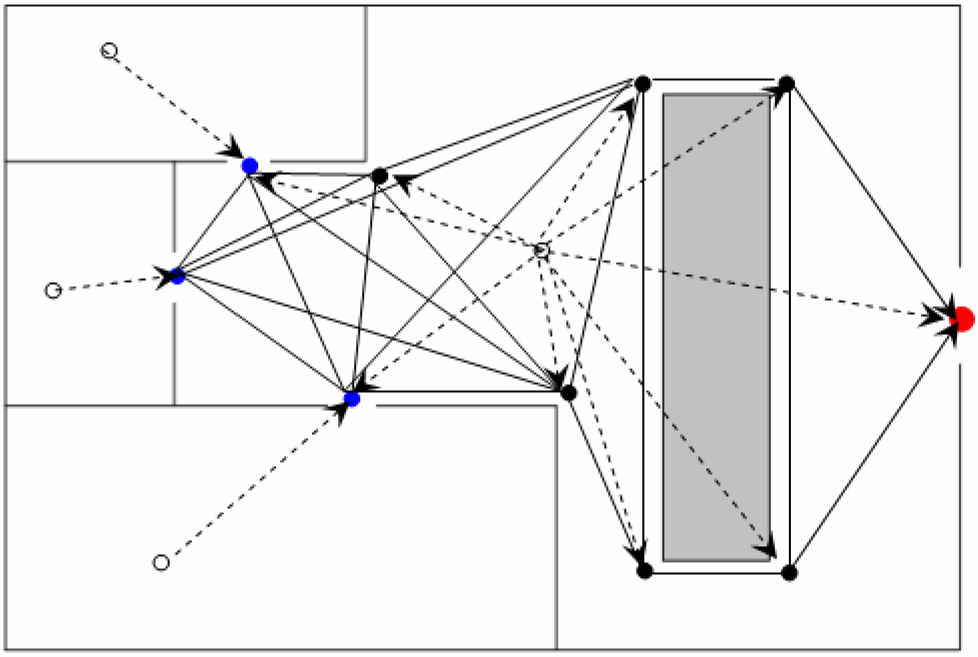


Figure 6.6.5: The new 'route-based graph' of the given building plan from Figure 6.6.1, the blue dots refer to the internal exit nodes, black dots refer to waypoints

### 6.6.2.4 Simplified wayfinding model within the new 'route-based graph'

Based on the discussion in Section 6.6.2.3, the new 'route-based graph' for a building plan has been completely generated. In such a graph, four node types have been defined which include room node, external exit node, internal exit node and waypoint node.

The node types can be divided into three different categories. The first category represents the source locations which are known as room nodes. The second category represents the destinations which are described as the External exits nodes. And the last category navigation nodes contain two node types which represent the internal exits and the waypoints.

Also, there are three kinds of route segments which have been defined within the new 'route-based graph'. The first kind of route segments is employed to connect between room nodes to navigation nodes, and they are directed edges which will point towards navigation nodes. The second kind of route segments represents the internal connections within the navigation nodes, and all of them are bidirectional. The last kinds of route segments are used to connect external exits to other kind of node, which are also directed edges pointing to the external exits.

During the wayfinding process applying the new 'route-based graph', the source nodes connects to both navigation nodes and possible directly to an external exit node. Therefore, according to the description above, the wayfinding process for an occupant starting from a room node can be described as the Figure 6.6.6, see Sources box shown in Figure 6.6.6. From the room node they enter the Navigational graph via either an internal exit or waypoint node, see Navigation oval in Figure 6.6.6. They then navigate through the Navigation graph until they reach an exit node where they are exported from the graph, see Destination box Figure 6.6.6.

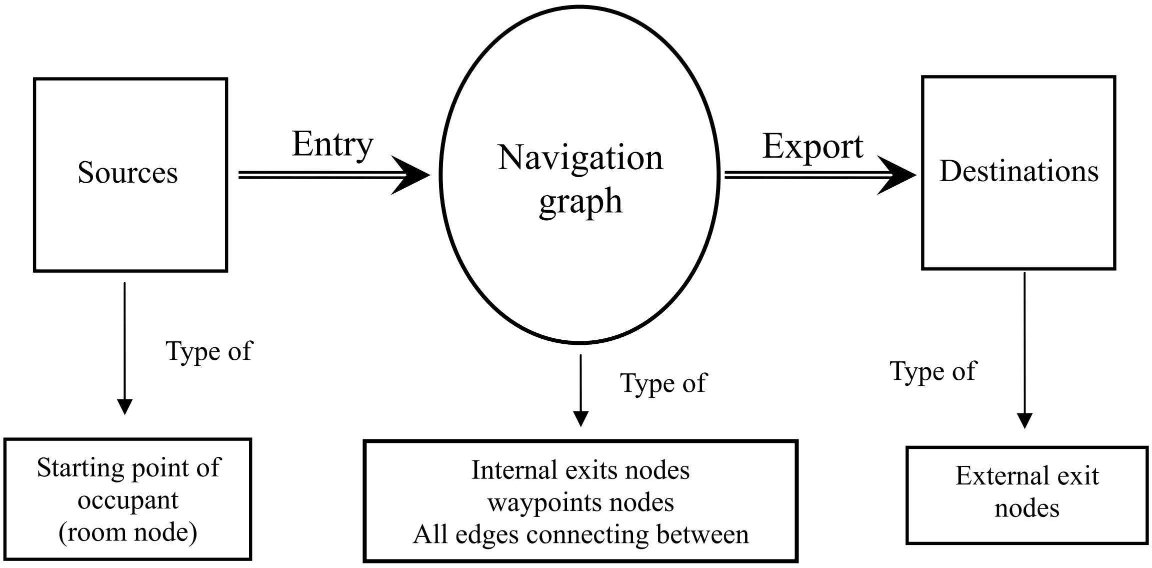


Figure 6.6.6: the wayfinding process sketch map

## 6.6.3 Implementation of new 'route-based graph'

### 6.6.3.1 Basic steps to implement new 'route-based graph'

Based on the definition of the new 'route-based graph' in Section 6.6.2, the implementation of a new 'route-based graph' mainly focuses on the following aspects:

1. Fixing the locations of all nodes type, which will contain the internal exits nodes, waypoint nodes, room nodes and external exits node.
2. Forming the navigation graph according to the visibility relationships of all

internal exits and waypoints nodes. All edges in the navigation graph are bidirectional.

3. For all external exits node, all possible connections will be generated between them and the navigation nodes according to the visibility relationships. These arcs are directed and point to a corresponding external exit node.
4. For each room node, arcs are used to connect the navigation nodes which are contained within the same compartment as the room node (waypoint nodes), or located at the boundary of the compartment (internal exit nodes). These arcs are directed and point to the navigation nodes.
5. For the room node connected to external exits which are located to the boundary of corresponding compartment, an arc is generated between them. These arcs are directed and point to the external exits.

Using these five steps, the method of generating a new 'route-based graph' is easy to achieve once the locations of all the nodes have been identified. To explain the implement process, we just need to discuss the method of generating all kind of nodes. For internal and external exits, the centre location of the exit is used to represent the nodal location. The thickness of the wall around the exits would also effect their location but this has been ignored in our analysis. The main task is to generate a new 'route-based graph' identifying the room nodes and waypoint nodes.

### 6.6.3.2 Specifying room nodes

The basic task of choosing a room node location to represent the corresponding compartment can be described as: the length of the arc connecting between the room node and a navigation node (internal exit or waypoint) can be denoted as the average distance of any locations within the compartment to the navigation node which is outlined in this section.

To specify the room nodes within a new 'route-based graph', the explanation of the potential map will be introduced first as used in buildingEXODUS. Figure 6.6.7 depicts two potential maps for a simple enclosure, the difference between the two figures being the width of the exit and the subsequent number of connections from each exit to the enclosure. Note that for the sake of ease of graphical representation and description, only horizontal and vertical connections and integer distances are used to represent this map.

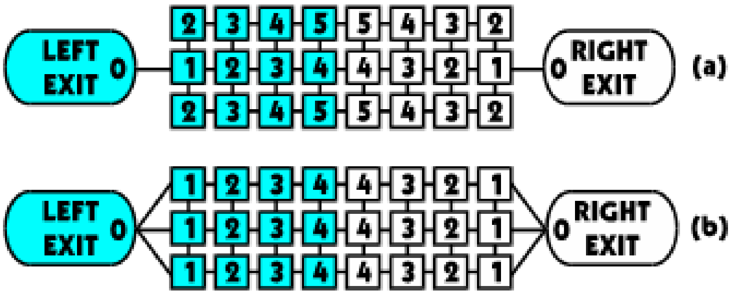
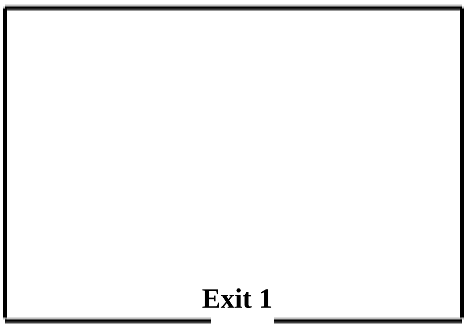
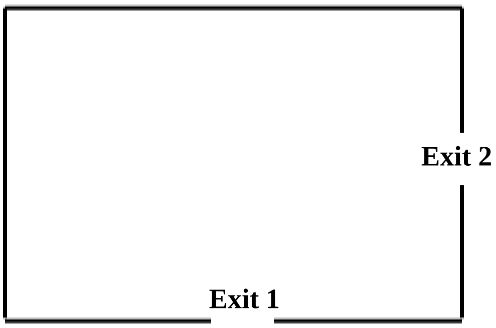
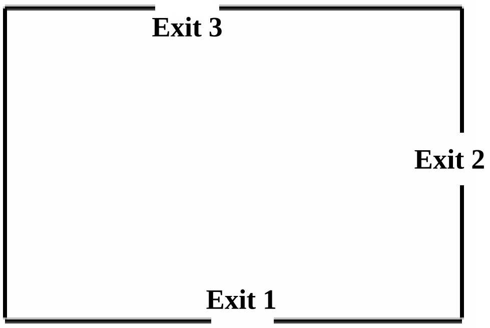
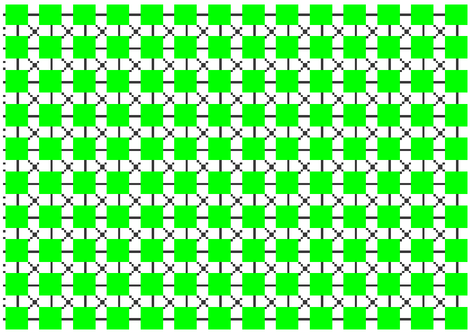


Figure 6.6.7: Example integer maps for a simple enclosure with (a) one connection from exit to enclosure, and (b) three connections from exit to enclosure

In the following, there are three cases which need be considered in specifying the location of the room node. To explain the implementation, a series of simple compartments within a building have been applied to complete the process. These rooms within a building have the same size but a different number of internal exits shown as Figure 6.6.8 to 6.6.10. There is only one internal exit for the compartment to connect with another compartment in Figure 6.6.8, two internal exits for Figure 6.6.8, and three for Figure 6.6.9. Figure 6.6.11 shows the fine mesh generated within buildingEXODUS for these three enclosures.

| | |
|---|---|
|  |  |
| Figure 6.6.8: One internal exit | Figure 6.6.9: Two internal exits |
|  |  |
| Figure 6.6.10: Three internal exits | Figure 6.6.11: Fine mesh representation |

### *Case 1:*

In this case, considering just one navigation node connected with a compartment within a building. The compartment as shown in Figure 6.6.8 will be used to describe the process of specifying a room node. First, the internal exit node is located at the middle location of Exit 1 shown as the blue dot in Figure 6.6.12. A potential map is then generated according the fine mesh within buildingEXODUS with external exits having a zero potential value. In reality, just two fine nodes nearest to the blue dot are set to a value of zero. After generating a potential map, the average value for the entire fine nodes can be calculate, and R1 (see Figure 6.6.12) used to denote this average value. If an occupant is positioned at a random location within the compartment, then the question which could be asked is how far the occupant has to travel to get to Exit 1. R1 answers this average travel distance question. Hence the room node can just be identified at a random location which is R1 from Exit 1 as shown in Figure 6.6.12.

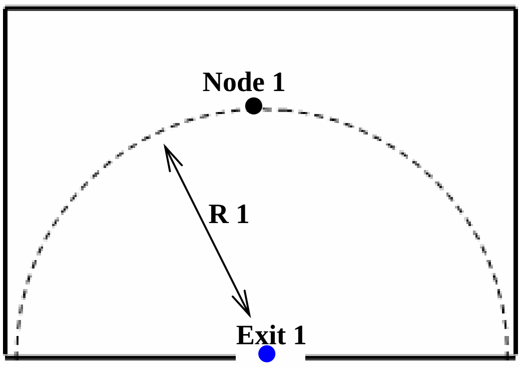


Figure 6.6.12: Location of room node (one internal exit)

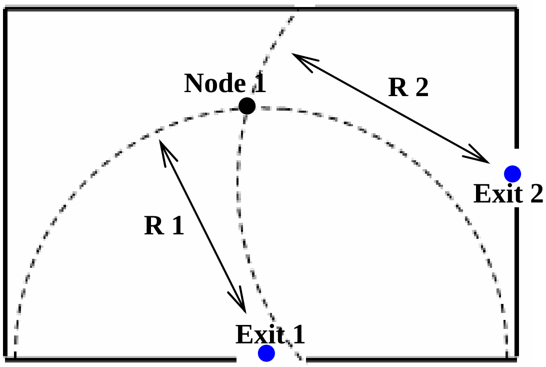


Figure 6.6.13: Location of room node (two internal exits)

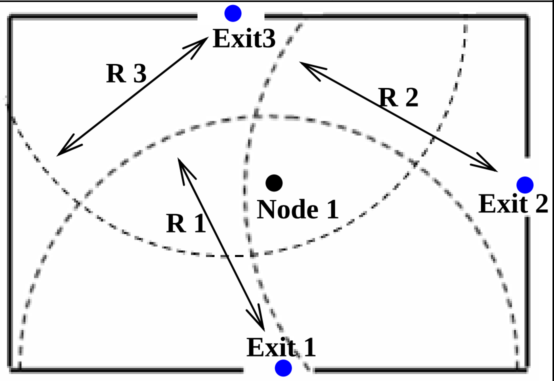


Figure 6.6.14: Location of room node (more than two internal exits)

***Case 2:***

Two navigation nodes connecting with the compartment will be considered in this case as shown in Figure 6.6.9. In this case, a room node needs to be specified to meet the following conditions: the room node can represent the average distance to Exit 1. Also it can denote the average distance to the Exit 2. If R1 is the average value of the potential values for all fine nodes to Exit 1, and R2 is the value to Exit 2. Then the Node 1 as shown in Figure 6.6.13 specifies the room node for this case.

***Case 3:***

In this case, More than two navigation nodes are considered here. According to the last two cases, an ideal location of the room node should satisfy that this node can represent the average distance of all fine mesh nodes for each navigation node within the compartment. However, this is not possible in all situations. For example, in Figure 6.6.14, there is not any node which can represent such an ideal location. So when more than two navigation nodes are in this compartment, the centre location will be employed to denote the room node as shown as in Figure 6.6.14. The centre location (*x, y*) can be calculated by the following formula: $x = \sum_{i=1}^{N} x_i \Big/ N$ and $y = \sum_{i=1}^{N} y_i \Big/ N$ , where N is the number of fine mesh nodes generated by buildingEXODUS, ( $x_i$ , $y_i$ ) representing the location of $i^{th}$ fine mesh node.

## 6.6.3.3 Implementation of waypoint node

Waypoints nodes will be placed at the locations near concaved vertices. According to the fine mesh node size, which is the 0.5x 0.5 meters, which is the average body width applied in buildingEXODUS, waypoints node are usual placed at a distance of 0.25 meters from the concave vertex. At a location which bi-sects the internal angle of a concaved vertex in half as shown in Figure 6.6.15. The round dot on the inside region represent the 'waypoint node'. In reality, waypoints represent the locations within the building that people use as points of reference when navigating around the geometry.

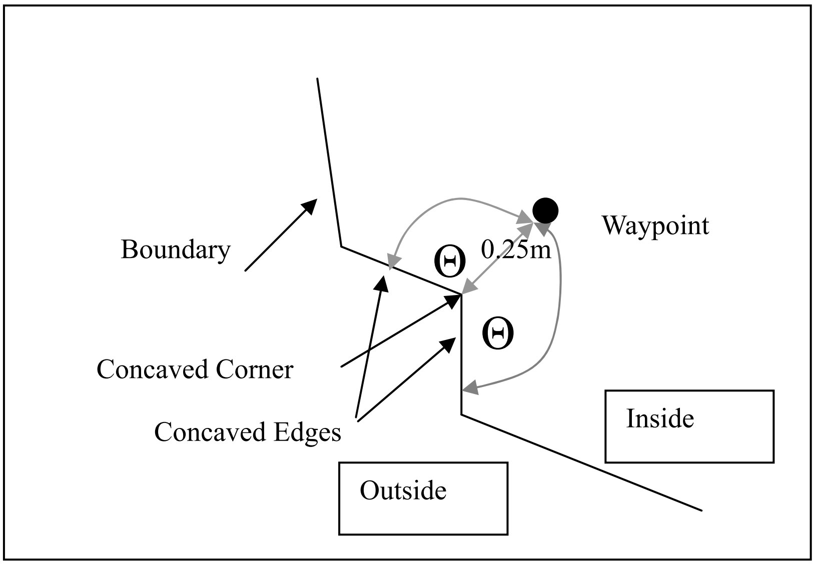


Figure 6.6.15: Location of a way point (Θ is half the angle of the internal angle between the concaved edges)

However, if there is insufficient space due to an obstruction the waypoint is then placed at the midpoint between the concaved vertex and the obstruction as shown as Figure 6.6.16.

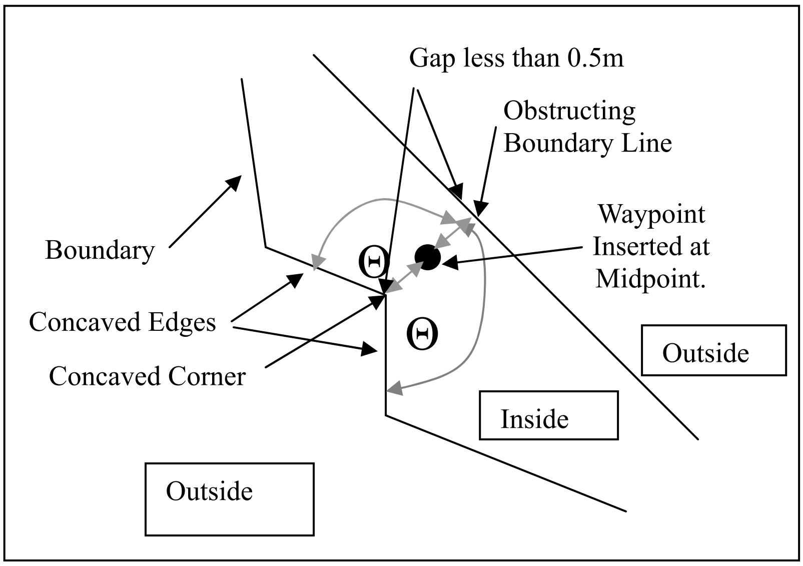


Figure 6.6.16 locating a waypoint node in confined space

In reality, the exact locations of the waypoints node are not that important since they are only used as steering points during the wayfinding process (see Chapter 7). A small deviation in the location of a waypoint has little impact on the overall travel distance for an occupant travelling to an external exit.

In the case of a concave part which is represented by a curve or arcs as shown in Figure 6.6.17. For concaved part, line segments, as shown as the broken lines in Figure 6.6.17, are applied over the curved section. Then the waypoint node is calculated using the method shown in Figure 6.6.17.

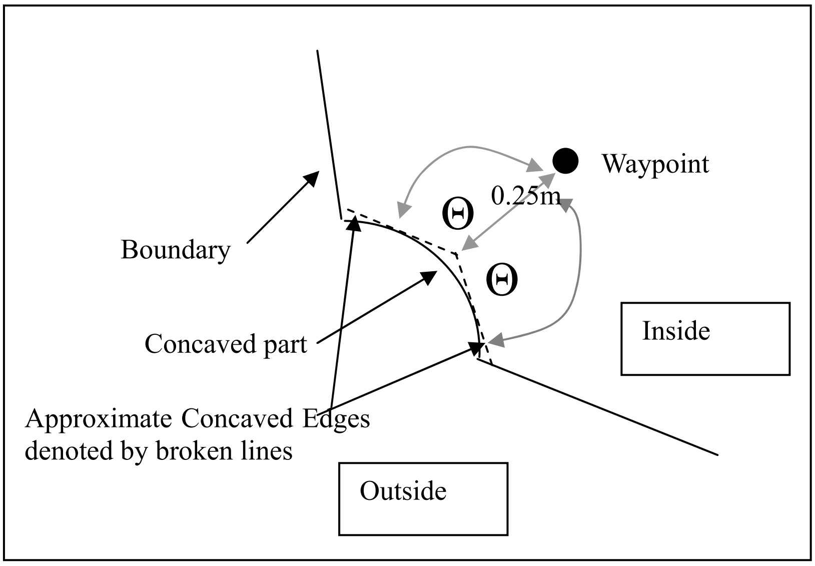


Figure 6.6.17 locating a waypoint node with no concaved vertex in concave part

## 6.7 Applying the new 'route-based graph' to building complexity

As described in Section 6.2, the original objective of generating new 'route-based graph' is for the purpose of improving the building complexity measures developed in the last two chapters. However, an explanation to achieve this objective is introduced here. This mainly covers the following two aspects:

1. The graph is generated based on a concept of route graphs, where the arcs represent the directed route segments. According to the measures used in the last two chapters, they all related to the fact that these measures need an occupant to sweep the whole building plan while completing the wayfinding tasks. Therefore to apply such a graph to these models, all route segments are bidirectional.

2. On the other hand, the node complexity degree will only need to be calculated for the room nodes. Since only room nodes represent the possible start locations for

an occupant.

However, based on the limitations as described in Chapter 5, the new 'route-based graph' will mainly be applied to another building complexity model which will be introduced in the next chapter.

## 6.8 Summary

For comparing different buildings for the purpose of their evacuation ability, The *Distance Graph Method* and *Global Complexity (PAT)* have been developed in the last two chapters. These models are all based on a room graph representation of a building structure to determine the building complexity. However, a room-based graph does not have the descriptive power to describe the actual routes taken during the wayfinding process. In most situations, the room-based graph cannot express the real environment of the building exactly. In addition the edges in the room-based graph cannot represent the real route the evacuee will travel to an exit. To resolve the drawbacks of a room-based graph representation, a new kind of graph representation for a building structure was discussed in this chapter. This new graph has the capability to represent the real route that an evacuee will take during the wayfinding process. Hence this kind of graph is called a new 'route-based graph'. Actually, the 'route-based graph' is a mixed graph. In this thesis, all connections between the nodes in a 'route-based graph' will be called arcs. The initial purpose of generating a new 'route-based graph' is to improve the *Distance Graph Method* and *Global Complexity (PAT)*, however, based on the limitations of these models (see Chapter 5), a new building complexity measure will be developed to be applied to the new 'route-based graph' in Chapter 7.

# Chapter 7 Development of new graph measures for evacuation analysis based on route-based graph

## 7.1 Introduction

This chapter introduces two kinds of methods that can be used to calculate building complexity: an analytical solution and a rule-based simulation method.

The analytical solution method is applied to a directed graph to solve a simple wayfinding case to evaluate the building complexity of a building as represented by a new ‘route-based graph’ as defined in Chapter 6, Section 6.6. In this method, the wayfinding process is considered as a simple random walk model. The searching time from any room node to the available external exit can be obtained by solving a set of equations. The complexity time then is then calculated by a formula as set out in Section 7.3. The Complexity Time addresses the following question:

> *If an occupant is positioned at a random location within a building, on average how long does the occupant need to spend to find an available exit?*

The analysis method cannot represent the occupant’s path memory ability since all values in the matrix of transition probabilities must be predefined (see Section 7.3). Hence, the rule-based simulation method is presented as the principal algorithm for calculating the Complexity time in this chapter.

The rule-based simulation method introduces four new building complexity models that have been developed based on a random walk model applied to route-based graph. The measures presented here are generated by applying a random walk [RD05] [Lo93] to simulate wayfinding behaviour according to the assumptions as set out in Chapter 6. The models presented in this chapter assume that the wayfinding process occurs while navigating the route-based graph. When an occupant reaches a certain junction they make a choice between the various available arcs. The probability of a person choosing each arc is dependent on which of these models are utilized. For each model a set of navigational behaviour rules are defined and each one is applied to different types of route-based graph representation.

The first model presented is based on simplistic rule-based wayfinding behaviour and this algorithm is applied to the fine node route-based graph as described in Chapter 6. The second model is an extension of the first model with an improved set of behaviour rules, which incorporates local route memory. The third model applies the same rules as the second model but is applied to a new 'route-based graph' as defined in Chapter 6. In the first three models, the average travel distance is used as the building complexity measure. The forth model applies the same type of route-based representation as the third model, however enhanced behaviour rules are used to generate the building complexity measure. This forth model will be referred to as the *Complexity Time Measure*.
The *Complexity Time Measure* addresses the following question:

> *If an occupant is positioned at a random location within a building, on average how long does the occupant need to spend to find an available exit?*

Certainly, this question just indicates a possible idea for comparing the wayfinding task within various building environments. For buildings where the occupants has no knowledge of the structure, a building with the smaller travel distance or time to find an exit is considered better for evacuation than the one with a larger travel distance or time [SE08] [Gu09].

## 7.2 The basic theories of the random walk

The new complexity models are based on the theory of a random walk on a finite graph. A random walk can be described as a mathematical formalization of a trajectory that consists of taking successive steps in random directions [RD05] [Lo93]. Given a graph and a starting node, we select a neighbour at random, and move to this neighbour; then we select a neighbour of this node at random and move to it etc. the random sequence of nodes selected in this way describes as a random walk over a graph [Lo93].

A graph G is a countable set V of vertices (i) connected pairwise by a set E of undirected links (i, j) = (j, i). A path in G connecting points i and j is a sequence of consecutive links {(i, k)(k, h) . . . (n,m)(m, j)} and a graph is said to be connected, if for any two points i, j $\in$ V there is always a path joining them.

A random walk will be considered based on a connected graph G. Now consider an occupant placed at vertex $v_i$. At each stage the occupant must move to an adjacent vertex. If $(v_i, v_i)$ is an edge of G, then the probability of the occupant moving to the vertex $v_j$ is $1/D(i)$ ( $D(i)$ refers to the neighbour size of $v_i$ ). Otherwise the probability is 0. Therefore if we define:

$$p_{i,j} = \begin{cases} \frac{1}{D(i)} & \text{if } (v_i, v_i) \text{ is a edge of Graph } \boldsymbol{G} \\ 0 & \text{Otherwise} \end{cases} \qquad \text{Formula 7.2.1}$$

Hence $P = (p_{i.j})$ is a Markov matrix [RD05] [Lo93] of transition probabilities. In an undirected graph, the roles of i and j are reversed hence the columns of $P = (p_{i.j})$ sum to 1. As each stage~~,~~ a sequence of adjacent vertices is produced. This sequence represents the position of the occupant at any given stage. Moreover this sequence is a walk in the graph. We call such a walk *'a random walk'* over the graph G.

A random walk on a graph is the simplest wayfinding model [Gu98]. Based on a room graph representation of a simple building, Gunnar introduced an analysis method to implement a simply wayfinding model [Gu98]. The basic random walk is normally used to analyse an undirected graph. In the next section, we will apply an analysis method to a mixed graph to analyse a simple wayfinding process, then an initial building complexity measure will be outlined.

## 7.3 Building complexity model based on new 'route-based graph'

### 7.3.1 Wayfinding based on a new 'route-based graph' representation.

The objective of this chapter is to develop building complexity measures for comparing the evacuation efficiency of different buildings. These complexity measures are all independent with respect to the scenario. And they are developed based only on the structure of the building plan. The new 'route-based graph' representation was generated based completely on the physical configuration of the building. Therefore, the wayfinding in such a building is simplified to happen in a graph representation of the building plan. Here the wayfinding refers to travelling from one of the room nodes to an external exit. Also, the wayfinding task will be progressed with some of the assumptions as outlined in Chapter 6.

The wayfinding process will be analysed by applying it to a simple case as described in Chapter 6:

Figure 7.3.1 is a sub-graph of the route-graph representation of a building plan which just contains one room node. The wayfinding process for an occupant who starts from Room 1 and travels to the Exit 1 will be explained using the following steps:

1. The occupant will choose one of the local destinations from all arcs connected with the room node. In this case Door 1 is his only choice.

2. When he gets to Door 1, he will know this is not an external exit. So to complete his wayfinding task, he need to choose the next destination from all the arcs connected with this Door 1. He will then be faced with six choices based on different probability from P1 to P6 which satisfy the equation $\sum_{i=1}^{6} p_i = 1$. However, normally he will not return to a room node. So in this example P 6 is zero.

3. After he gets to the next node, he will face the same process as described in Step 2. Until he get to Exit 1.

During the wayfinding process, $\sum_{i=1}^{k} p_i = 1$ will be satisfied in each none external exit node, where k is the neighbour size of current node. Also the value of Pi will be a very complex to confirm. In the simply case the probability assigned, Pi values, are the same for all neighbouring nodes.

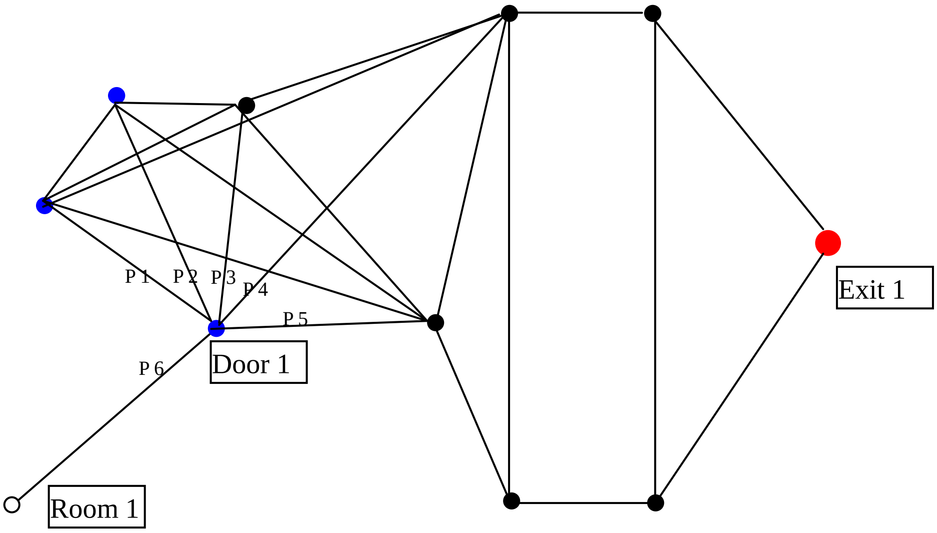


Figure 7.3.1: A sub-graph of a new 'route-based graph' representation of the building plan in Chapter 6

### 7.3.2 A building complexity measure based on the new ‘route-based graph’

A mathematical analysis will now be preformed utilising a simple building plan for demonstration purposes. In Figure 7.3.2 a new ‘route-based graph’ representation is presented of the building which was first introduced in Chapter 6. In Figure 7.3.2, there are four room nodes, nine navigation nodes, and one external exit node. Let’s reorder these fourteen nodes as shown in Figure 7.3.3. In Figure 7.3.3, R1 to R4 are room nodes. R5 to R13 are navigation nodes, and R 14 is the only external exit.

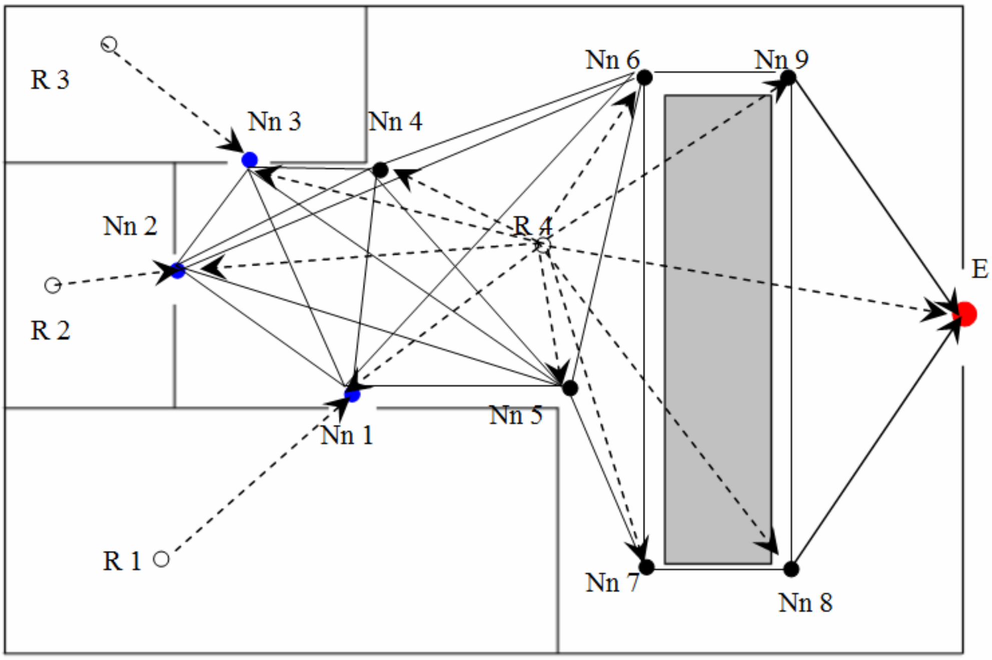


Figure 7.3.2: A new ‘route-based graph’ representation of the building plan

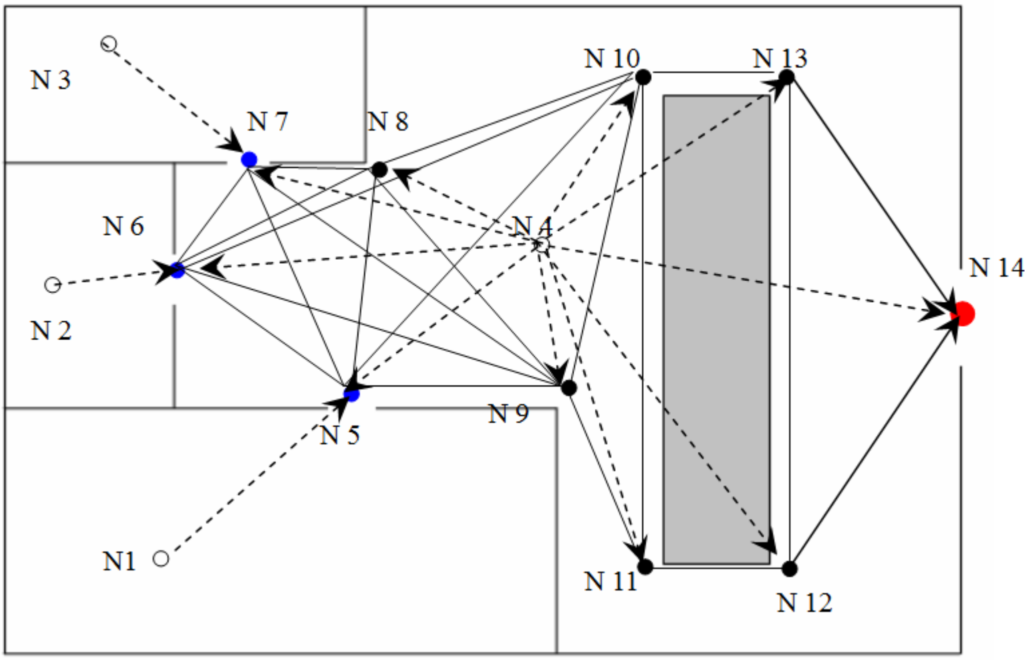


Figure 7.3.3: A new 'route-based graph' representation of the building plan by redefining nodes number

In the new 'route-based graph', each node will be considered as the decision point during the wayfinding process. Let $t_{i,j}$ be the travel time between the two decision points. The travel time can be calculated by the method as described in Chapter 5. In the simplest case an occupant will choose their next decision point randomly. This is a natural process since the occupant has no previous knowledge of the environment. In the extreme case, an occupant who has no idea of which direction to head, would need to select at random from the set of all neighbouring nodes until they reach Node 14. Hence, the wayfinding process in such case is a simple random walk model.

We use $d(i)$ to represent the order (degree) of $i^{th}$ node (i=1, 2 …14), i.e. the neighbour size of $i^{th}$ node. In the new 'route-based graph' representation of a building, the graph is a mixed graph, so the neighbours do not include the node which does not contain a directed edge from $i^{th}$ node. For example, N5 is a neighbour of N1, however, N1 is not a neighbour of N5 (see Figure 7.3.3). An occupant will perform a random selection at $i^{th}$ node with equal probability from all of the neighbours of current node. $P(i,j)_{14\times 14}$ is used to represent the matrix of transition probabilities of such a random process, $p_{i,j}$ (i=1,2 … 14, j= 1,2 …14) represents the elements of the matrix. Hence it is easy to obtain $P(i,j)_{14\times 14}$ shown as following:

$$P(i,j)_{14\times 14}=\begin{bmatrix}
0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}&\frac{1}{10}\\
0&0&0&0&0&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&0&0&0&0\\
0&0&0&0&\frac{1}{5}&0&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&0&0&0&0\\
0&0&0&0&\frac{1}{4}&\frac{1}{4}&0&\frac{1}{4}&\frac{1}{4}&0&0&0&0&0\\
0&0&0&0&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&0&\frac{1}{5}&\frac{1}{5}&0&0&0&0\\
0&0&0&0&\frac{1}{6}&\frac{1}{6}&\frac{1}{6}&\frac{1}{6}&0&\frac{1}{6}&\frac{1}{6}&0&0&0\\
0&0&0&0&\frac{1}{6}&\frac{1}{6}&0&\frac{1}{6}&\frac{1}{6}&0&\frac{1}{6}&0&\frac{1}{6}&0\\
0&0&0&0&0&0&0&0&\frac{1}{3}&\frac{1}{3}&0&\frac{1}{3}&0&0\\
0&0&0&0&0&0&0&0&0&0&\frac{1}{3}&0&\frac{1}{3}&\frac{1}{3}\\
0&0&0&0&0&0&0&0&0&\frac{1}{3}&0&\frac{1}{3}&0&\frac{1}{3}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix}$$

In matrix $P(i,j)_{14\times 14}$, the values of the first 4 columns are all zero since there is no directed connection between any two room nodes. The values of last row are also zero since the N 14 refers to the exit node and there is not a directed connection from an exit to any of the other nodes. Let $t_i$ refer to the average time for an occupant from N (i) (i=1, 2 …14) to find the exit. Here the expected wayfinding time is calculated by dividing the travel distance by a speed of 1.5m/s, which have been used previously in Chapter 5. This implies that the travel time is calculated based on the following assumptions: an occupant will start their wayfinding process immediately which means their response time is zero. All occupants walk at the same constant speed (1.5 m/s). An occupant makes their wayfinding decisions at once at each node which means the delay at the decision points has been omitted. Also, in additional all doors are assumed to be open. Hence the following equations can be easily obtained:

$$\begin{cases} t_1 = 1\times(t_{1,5}+t_5) \\ t_2 = 1\times(t_{2,6}+t_6) \\ t_3 = 1\times(t_{3,7}+t_7) \\ t_4 = \frac{1}{10}(t_{4,5}+t_5)+\frac{1}{10}(t_{4,6}+t_6)+\frac{1}{10}(t_{4,7}+t_7)...+\frac{1}{10}(t_{4,14}+t_{14}) \\ t_5 = \frac{1}{5}(t_{5,6}+t_6)+\frac{1}{5}(t_{5,7}+t_7)+\frac{1}{5}(t_{5,8}+t_8)+\frac{1}{5}(t_{5,9}+t_9)+\frac{1}{5}(t_{5,10}+t_{10}) \\ ...... \\ t_{13} = \frac{1}{3}(t_{13,10}+t_{10})+\frac{1}{3}(t_{13,12}+t_{10})+\frac{1}{3}(t_{13,14}+t_{10}) \\ t_{14} = 0 \end{cases} \qquad \text{(Formula 7.3.1)}$$

This set of equations is equivalent to the following representation (Formula 7.3.2):

$$\begin{cases} t_1 = \sum_{i=1}^{14} p_{1i}t_{1i} + \sum_{i=1}^{14} p_{1i}t_i \\ t_2 = \sum_{i=1}^{14} p_{2i}t_{2i} + \sum_{i=1}^{14} p_{2i}t_i \\ t_3 = \sum_{i=1}^{14} p_{3i}t_{3i} + \sum_{i=1}^{14} p_{3i}t_i \\ t_4 = \sum_{i=1}^{14} p_{4i}t_i + \sum_{i=1}^{14} p_{4i}t_i \\ ...... \\ t_{13} = \sum_{i=1}^{14} p_{13,i}t_{13,i} + \sum_{i=1}^{14} p_{13,i}t_i \\ t_{14} = \sum_{i=1}^{14} p_{14,i}t_{14,i} + \sum_{i=1}^{14} p_{14,i}t_i \end{cases} \qquad \text{(Formula 7.3.2)}$$

The equations in Formula 7.3.2 can be represented as following:

$$(\mathrm{I}_{14x14}\text{-}\mathrm{P(i,j)}_{14x14})\ \mathbf{t=b} \qquad \text{(Formula 7.3.3)}$$

In Formula 7.3.3, $\mathrm{I}_{14\times14}$ is a 14 X 14 identity matrix, $\mathrm{P(i,j)}_{14x14}$ is the matrix of transition probabilities, The unknown 14 x 1 vector **t** has elements $t_i$ (i = 1 ,2…14). And the 14 X 1 vector **b** has constant elements $b_i$ (i = 1, 2…14), which can be represented as $b_i = \sum_{j=1}^{14} p_{i,j}t_{i,j}$ .

Once we have obtained the solutions to the linear systems in Formula 7.3.3. We can then use the solutions for the room nodes ($t_1$ to $t_{4)}$ to calculate the building complexity. This is because the first four values represent the average wayfinding time to find an exit for an occupant starting from each of the 4 room nodes. If $Area_1$ to $Area_4$ are used to represent the area of the rooms *Room1* to *Room4* respectively, then the *Complexity time* of the building for this case can be calculated by the following formula:

$$\textit{Complexity time} = \sum_{i=1}^{4} \left( \frac{Area_i}{\sum_{j=1}^{4} Area_j} \times t_i \right) \qquad \text{Formula 7.3.4}$$

In general, let G be a new ‘route-based graph’ representation of a building plan. Then there are n room nodes, m navigation nodes and k external exit nodes within the graph G. We can then reorder the node number from 1 to n+m+k, and represent all the nodes in G by $N_i$ (i=1,2 … n+m+k) where the first n nodes represent the room nodes, the navigation nodes are represented from node n+1 to n+m, and the last k nodes represent the external exit nodes.

In graph G, each node will be considered as a decision point during the wayfinding process. Let $t_{i,j}$ be the travel time between the two neighbouring decision points. In reality, it is not necessary for an occupant to choose their next decision point randomly, i.e. the occupant will not have to perform a random choice at $i^{th}$ node with equal probability from all of the neighbours of current node. But we need to ensure that the sum of the probabilities is 1. Let $P(i,j)$ ((n+m+k)x(n+m+k)) represent the matrix of transition probabilities of a random process. $p_{i,j}$ (i=1, 2 … n+m+k, j= 1, 2 …n+m+k) be the elements of the matrix. Hence the $P(i,j)$ can be represented as the following matrix:

$$\left[\begin{array}{cccc|cccc|cccc}
p_{11} & p_{12} & \cdots & p_{1n} & p_{1,n+1} & p_{1,n+2} & \cdots & p_{1,n+m} & p_{1,n+m+1} & p_{1,n+m+2} & \cdots & p_{1,n+m+k} \\
p_{21} & p_{22} & & p_{2n} & p_{2,n+1} & p_{2,n+2} & & p_{2,n+m} & p_{2,n+m+1} & p_{2,n+m+2} & & p_{2,n+m+k} \\
\cdots\cdots & \text{ZONE 1} & & & \cdots\cdots & \text{ZONE 4} & & & \cdots\cdots & \text{ZONE 7} & & \\
p_{n1} & p_{n2} & & p_{nn} & p_{n,n+1} & p_{n,n+2} & & p_{n,n+m} & p_{n,n+m+1} & p_{n,n+m+2} & & p_{n,n+m+k} \\
\hline
p_{n+1,1} & p_{n+1,2} & \cdots & p_{n+1,n} & p_{n+1,n+1} & p_{n+1,n+2} & \cdots & p_{n+1,n+m} & p_{n+1,n+m+1} & p_{n+1,n+m+2} & \cdots & p_{n+1,n+m+k} \\
p_{n+2,1} & p_{n+2,2} & & p_{n+2,n} & p_{n+2,n+1} & p_{n+2,n+2} & & p_{n+2,n+m} & p_{n+2,n+m+1} & p_{n+2,n+m+2} & & p_{n+2,n+m+k} \\
\cdots\cdots & \text{ZONE 2} & & & \cdots\cdots & \text{ZONE 5} & & & \cdots\cdots & \text{ZONE 8} & & \\
p_{n+m,1} & p_{n+m,2} & & p_{n+m,n} & p_{n+m,n+1} & p_{n+m,n+2} & & p_{n+m,n+m} & p_{n+m,n+m+1} & p_{n+m,n+m+2} & & p_{n+m,n+m+k} \\
\hline
p_{n+m+1,1} & p_{n+m+1,2} & \cdots & p_{n+m+1,n} & p_{n+m+1,n+1} & p_{n+m+1,n+2} & \cdots & p_{n+m+1,n+m} & p_{n+m+1,n+m+1} & p_{n+m+1,n+m+2} & \cdots & p_{n+m+1,n+m+k} \\
p_{n+m+2,1} & p_{n+m+2,2} & & p_{n+m+2,n} & p_{n+m+2,n+1} & p_{n+m+2,n+2} & & p_{n+m+2,n+m} & p_{n+m+2,n+m+1} & p_{n+m+2,n+m+2} & & p_{n+m+2,n+m+k} \\
\cdots\cdots & \text{ZONE 3} & & & \cdots\cdots & \text{ZONE 6} & & & \cdots\cdots & \text{ZONE 9} & & \\
p_{n+m+k,1} & p_{n+m+k,2} & & p_{n+m+k,n} & p_{n+m+k,n+1} & p_{n+m+k,n+2} & & p_{n+m+k,n+m} & p_{n+m+k,n+m+1} & p_{n+m+k,n+m+2} & & p_{n+m+k,n+m+k}
\end{array}\right]$$

In the matrix of transition probabilities $P(i, j)$, each ZONE (from ZONE 1 to 9) represents the transition probabilities between different types of node within the route-based graph, the values of first n columns are all zero, which means there is directed connection from any node to a room node. And the values of the last k rows are all zero, which means that there is not any directed arc from an external exit to any other node. Hence in matrix $P(i, j)$, the values in ZONE1, ZONE2, ZONE3, ZONE6 and ZONE9 are all zero. Let $t_i$ refer to the average time for an occupant from N (i) (i=1, 2 …n+m+k) to find the exit. According to the analysis in last example, see Figure 7.3.2, we can obtain the following set of equations:

$$\begin{cases} t_1 = \sum_{i=1}^{n+m+k} p_{1i}t_{1i} + \sum_{i=1}^{n+m+k} p_{1i}t_i \\ t_2 = \sum_{i=1}^{n+m+k} p_{2i}t_{2i} + \sum_{i=1}^{n+m+k} p_{2i}t_i \\ \ldots\ldots \\ t_n = \sum_{i=1}^{n+m+k} p_{ni}t_{ni} + \sum_{i=1}^{n+m+k} p_{ni}t_i \\ t_{n+1} = \sum_{i=1}^{n+m+k} p_{n+1,i}t_{n+1,i} + \sum_{i=1}^{n+m+k} p_{n+1,i}t_i \\ t_{n+2} = \sum_{i=1}^{n+m+k} p_{n+2,i}t_{n+2,i} + \sum_{i=1}^{n+m+k} p_{n+2,i}t_i \\ \ldots\ldots \\ t_{n+m} = \sum_{i=1}^{n+m+k} p_{n+m,i}t_{n+m,i} + \sum_{i=1}^{n+m+k} p_{n+m,i}t_i \\ t_{n+m+1} = \sum_{i=1}^{n+m+k} p_{n+m+1,i}t_{n+m+1,i} + \sum_{i=1}^{n+m+k} p_{n+m+1,i}t_i = 0 \\ t_{n+m+2} = \sum_{i=1}^{n+m+k} p_{n+m+2,i}t_{n+m+2,i} + \sum_{i=1}^{n+m+k} p_{n+m+2,i}t_i = 0 \\ \ldots\ldots \\ t_{n+m+k} = \sum_{i=1}^{n+m+k} p_{n+m+k,i}t_{n+m+k,i} + \sum_{i=1}^{n+m+k} p_{n+m+k,i}t_i = 0 \end{cases}$$

(Formula 7.3.5)

This set of equations equals the following:

$$(\mathrm{I-P(i,j)})\,\mathbf{t=b}$$

(Formula 7.3.6)

In Formula 7.3.6, $\mathrm{I}$ is a (n+m+k)x(n+m+k) identity matrix, $\mathrm{P(i,j)}$ is the matrix of transition probabilities of graph G. The unknown 14 x 1 vector **t** has elements $t_i$ (i = 1 ,2…n+m+k). And the (n+m+k) x 1 vector **b** has constant elements $b_i$ (i = 1, 2 … n+m+k), which can be represented as $b_i = \sum_{j=1}^{n+m+k} p_{i,j}t_{i,j}$ . The linear equations system in Formula 7.3.6 contain n+m+k equations and n+m+k unknown numbers. Hence, the linear system has only one solution if and only if the matrix $\mathrm{I-P(i,j)}$ is invertible.

Currently, we cannot ensure that the matrix I-P(i,j) is invertible. The author has generated some test cases to check if I-P(i,j) is an invertible matrix or not, and cannot find a counterexample. However, we cannot say this type of matrix must be invertible.

If matrix I-P(i,j) is invertible, we can easily obtain the solutions of the linear systems of the Formula 7.3.6. The first n values ($t_1$ to $t_n$ ) can be used to calculate the *Complexity time* according to the following Formula7.3.7

$$\textbf{\textit{Complexity time}} = \sum_{i=1}^{n} \left( \frac{Area_n}{\sum_{j=1}^{n} Area_j} \times t_i \right) \qquad \text{Formula 7.3.7}$$

In the mathematical analysis method, an occupant does not have to choose their next decision point randomly, i.e. the occupant will not have to perform a random route choice at a node with equal probability from all of the neighbours of the current node. An occupant can choose the next node according to a number of factors, for example, the node’s feature, arc length, direction etc. but we need to ensure that the sum of the probabilities is 1. However, the matrix of transition probabilities $P(i, j)$ is predefined before an occupant starts to search for an exit. Hence this model cannot reflect the travel history of the occupant. In other words, this analysis method cannot represent the occupant’s path memory ability. Therefore, this mitigates any invertible problem described in last paragraph since a simulation method is utilised to find a solution.

## 7.4 Building complexity calculated utilising a rule-based simulation

The simple mathematical analysis method cannot be utilized to calculate a building complexity measure for more complex models, which incorporates occupants with path memory. Hence, for complex buildings this mathematical analysis method will not be efficient to analyse the wayfinding process based on the assumption described in Chapter 6. Therefore, in this chapter, we will mainly apply a simulation method to obtain a *Complexity Time Measure*. The models in this chapter assume that the wayfinding task happen within a route-based graph of the building. In each model, by defining a set of behaviour rules which occur during the navigation process, the models are considered by applying the different route-based graph representation. These representations include the fine node route-based graph and the new 'route-based graph' as defined in Chapter 6. A suit of simple building plans are then used to verify the performance of these models.

## 7.4.1 Demonstration cases

For examining the efficiency of building complexity models, several examples have been employed to examine the different aspects of some simple buildings. These examples will start from Figure 7.4.1, this figure is an image of a typical building with one exit and six internal exits. In this building, there are six rooms, one corridor. In the route-based graph, the rooms and corridor will all be considered as room nodes. The areas from room 1 to room 6 are all 48 m2 (8m x 6m) and the area of the corridor (room 7) is 72 m2 (24mx3m).

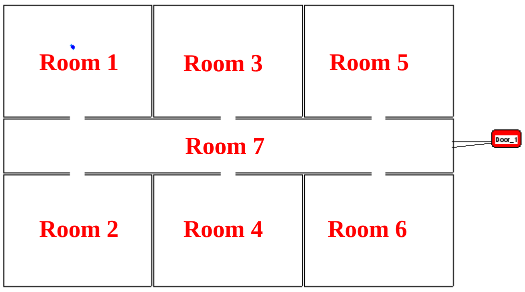
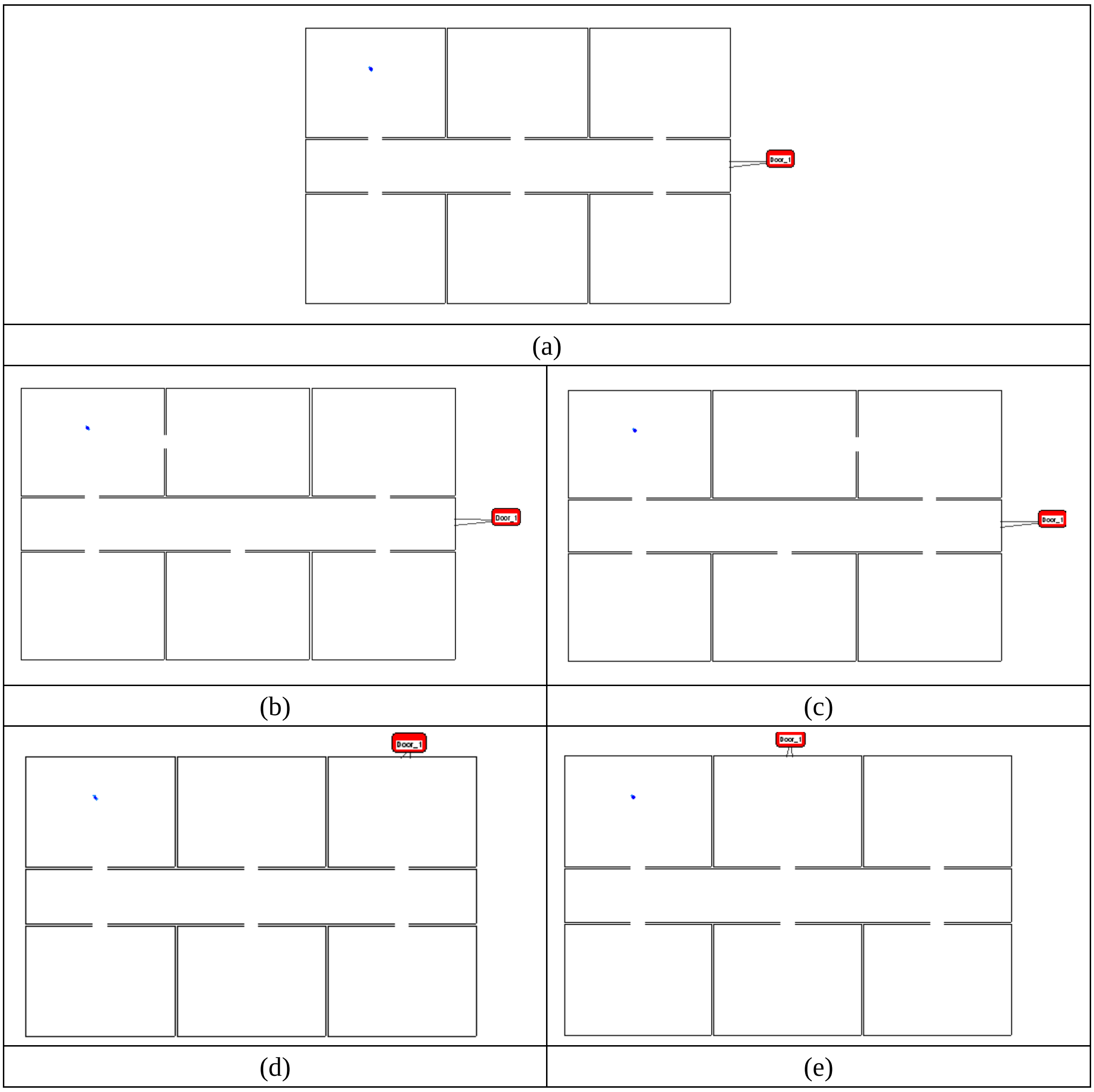


Figure 7.4.1: Building E

Firstly, for this building plan, an internal door location for Room 3 has been changed and all other geometry has been maintained (Figure 7.4.2 (b) and Figure 7.4.2(c)). Therefore these two examples are used to examine two buildings with the same number of rooms, the same room size but with different connections.

Secondly, the only external exit location has been changed and all other geometry condition maintained when compared to Figure 7.4.2 (a). Figure 7.4.2 (d) and Figure 7.4.2 (e) can be used to compare two buildings with different external exit position.



(a)

(b)

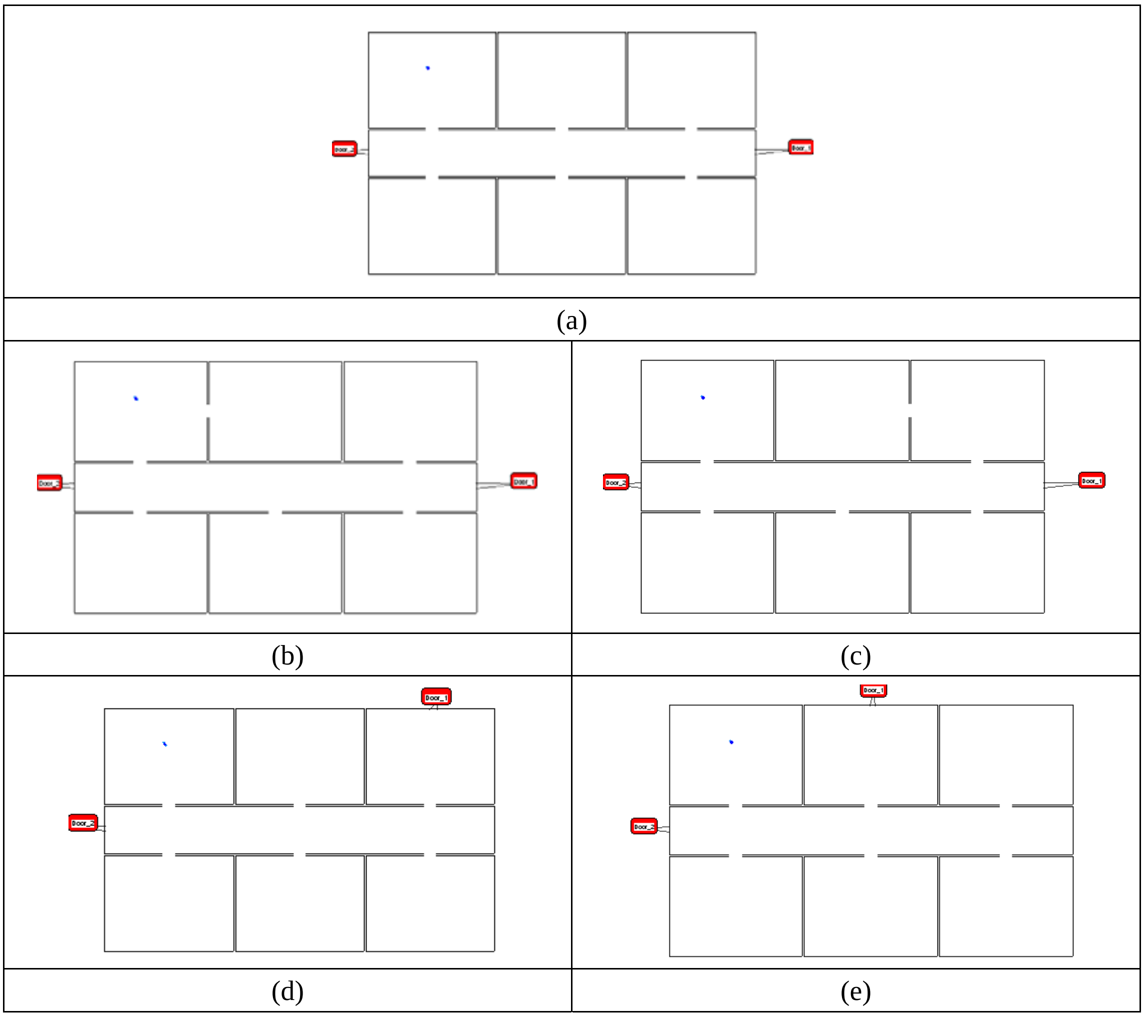
(c)

(d)

(e)

Figure 7.4.2: Different exit location and room connection for Building E

Finally, an external exit was added to each building for Figure 7.4.2 (a) to Figure 7.4.2 (e). These structures are shown in Figure 7.4.3 (a) to Figure 7.4.3 (e). In these examples an external exit on the left has been added to obtain Figure 7.4.3 (a) to Figure 7.4.3 (e). These cases can be used to compare buildings with two exits that are in different locations.



(a)

(b)

(c)

(d)

(e)

Figure 7.4.3: An external exit is add to buildings (Figure 7.4.2 (a) to Figure 7.4.2 (e)) respectively

These simple cases will be applied to analyse the building complexity in each of the models presented in this chapter. According to the analysis of the building complexity of this building plan, the models will then be modified by either improving their occupant searching behaviour rules or by applying different route-based graph representations.

### 7.4.2 Applying a rule base model and its implications on the building complexity

In this section, four building complexity models will be defined based on different route-based graph representations of a building. The route-based graphs will contain the fine node route-based and the new 'route-based graph' as defined in Chapter 6. In different models, the behaviour rules will be changed from simple to more complex by analysing the calculated results of the simple cases as defined in Section 7.4.1. Then the global complexity measure in the last model will be obtained by answering the following question:

> '*If an occupant is positioned at a random location within a building, on average how long does it take the occupant to find an available exit'*.

The global complexity measure in Model 4 will be called the *'Complexity Time Measure'*. When using '*Complexity Time Measure*' to compare two or more of structures with respect to evacuation ability, the building with the smallest value is the simplest to navigate.

## 7.4.2.1 Simple Model 1

### 7.4.2.1.1 The description of Model 1

In Model 1, a set of basic rules will be used on the fine node route-based graph representation to simulate the wayfinding process for a given person.

The basic idea of this model is described as: an occupant starts to evacuate from a building, with a randomly assigned start location and is assumed to have no knowledge of the building. This model is used to calculate the average travel distance to find an available exit using the simplest wayfinding behavioural rules. The model will be described as followed:

**Model 1**

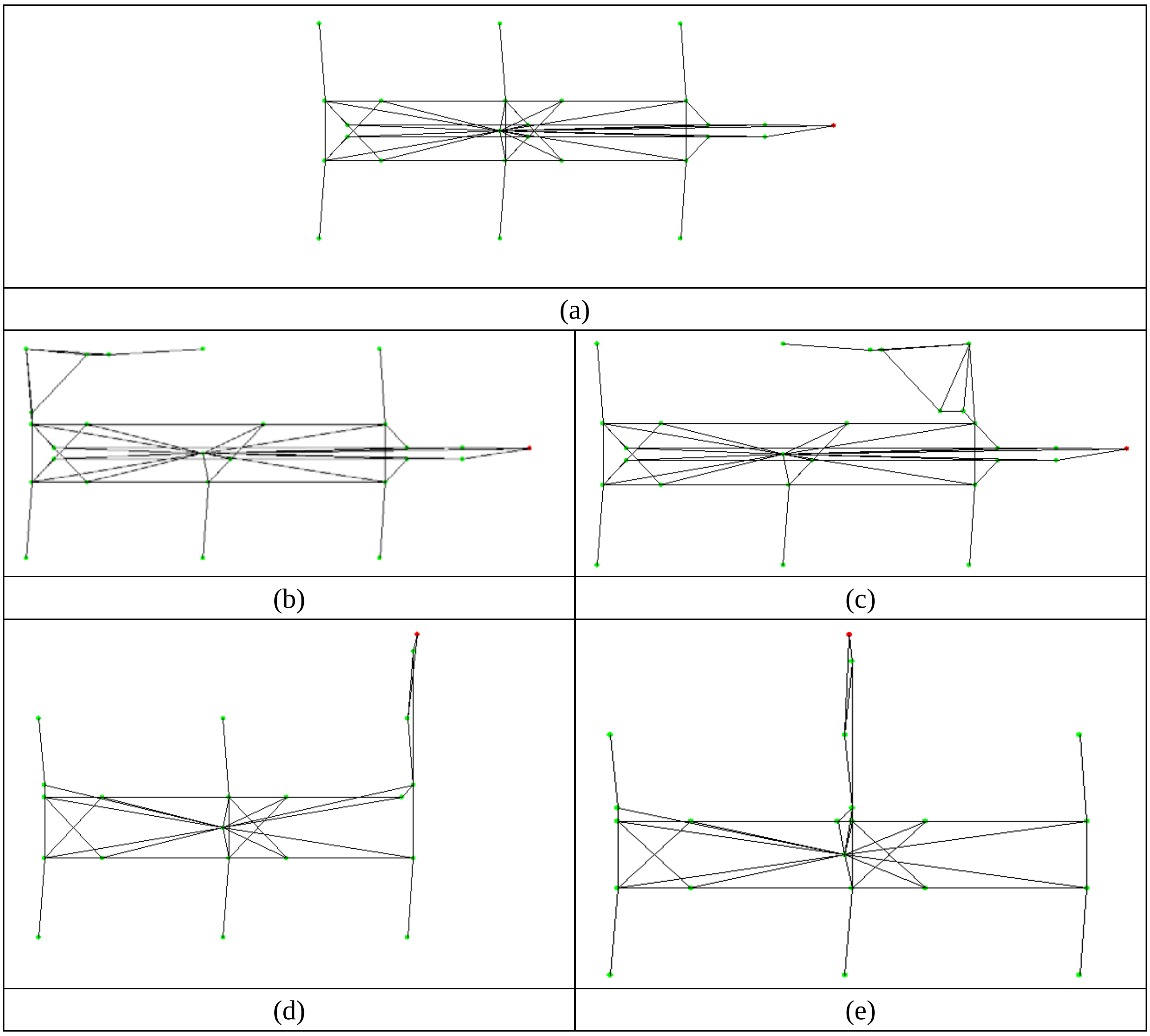
1) The simulation process is based on the fine node route-based graph representation of a building.

2) An occupant is placed on each room node in the fine node route-based graph.
3) For an occupant at the $i^{th}$ room node in the building, the following rules are used to calculate the average distance travelled to find an available exit.
   a) The direction of the initial step from the starting node is chosen from all the connected arcs at random.
   b) After the first step, all of the room nodes and the arcs connected with them will be removed from the graph, which means the evacuee never travels back to a room node during the wayfinding process.
   c) The evacuee then moves from node to node randomly, based on a uniform probability for all arcs.
   d) When the evacuee reaches an available exit, the wayfinding process ends and the travel distance is stored. This is repeated a 1000 times. The average distance D(i) is calculated. D(i) is used to indicate the distance to find an available exit for the occupant starting at the $i^{th}$ *room* node..
4) At each room node in the graph, this cycle is repeated 1000 times, then the average travel distance D(i) for each node is calculated.
5) All the travel distance D(i) can summed together to give the total distance. Then the average distance D(average) can be calculated.

If we assume there are the same probability of an occupant starting at a room node in an building, then the average number D(average) means the average travel distance to find an available exit.

### 7.4.2.1.2 Results analysis generated utilizing Model 1

In this section, the test cases as defined in Section 7.2.1 will be applied to analyse the efficiency of Model 1. Firstly**,** for the structure as shown in Figure 7.4.1, let’s examine the buildings with the same number of rooms of the same size, but with different connection (see Figures 7.4.2 (a) to 7.4.2 (c)). Here the fine node route-based graph representations of Figure 7.4.2(a) to 7.4.2(c) are shown by Figures 7.4.4(a) to 7.4.4(c).



(a)

(b)

(c)

(d)

(e)

Figure 7.4.4: Fine node route-based graph representations for Figures 7.4.2 (a) to 7.4.2 (e)

Applying Model 1, the results obtained for Figures 7.4.2 (a) to 7.4.2 (c) are shown in Table 7.4.1:

Table 7.4.1: Calculating results to comparing buildings with changing a room internal exit location by Model 1. Red numbers indicate maximum values. Blue numbers represent minimum values

| Figures | Figure 7.4.2 (a) | Figure 7.4.2 (b) | Figure 7.4.2 (c) |
|---|---|---|---|
| Room 1 | 112.202 | 109.227 | 98.7887 |
| Room 2 | 107.784 | 104.447 | 102.443 |
| Room 3 | 106.605 | 117.178 | 129.493 |
| Room 4 | 108.497 | 103.592 | 109.217 |
| Room 5 | 110.153 | 98.7355 | 124.956 |
| Room 6 | 110.649 | 104.67 | 107.471 |
| Room 7 | 96.6085 | 89.4298 | 94.3192 |
| **Average distance (Metres)** | **107.49** | **103.89** | **109.52** |
| **Number of Nodes** | **26** | **26** | **27** |
| **Number of Arcs** | **58** | **53** | **55** |

According to Table 7.4.1, these three buildings obtain similar values for the average travel distance. In all cases, Room 7 obtains the minimum value, hence is an good starting location with the quickest time to exit. For Figures 7.4.2 (b) and 7.4.2 (c), Room 3 obtains the maximum values after changing the position of door from the original case Figure 7.4.2 (a). However, from Table 7.4.1, one important problem should be noticed, that for such simple buildings the average travel distance obtained is far too large with respect to the area of the structure. This is a result of using simple wayfinding rules as applied by Method 1.

Another analysis will now be preformed which compares buildings with different external exit locations. These buildings are all the same except for different external door locations. The fine node route-based graphs representation of Figure 7.4.2 (d) and Figure 7.4.2 (e) are shown as Figure 7.4.4 (d) and Figure 7.4.4 (e) respectively.

Applying Model 1, the calculated results for Figure 7.4.2 (d) and Figure 7.4.2 (e) are shown as Table 7.4.2:

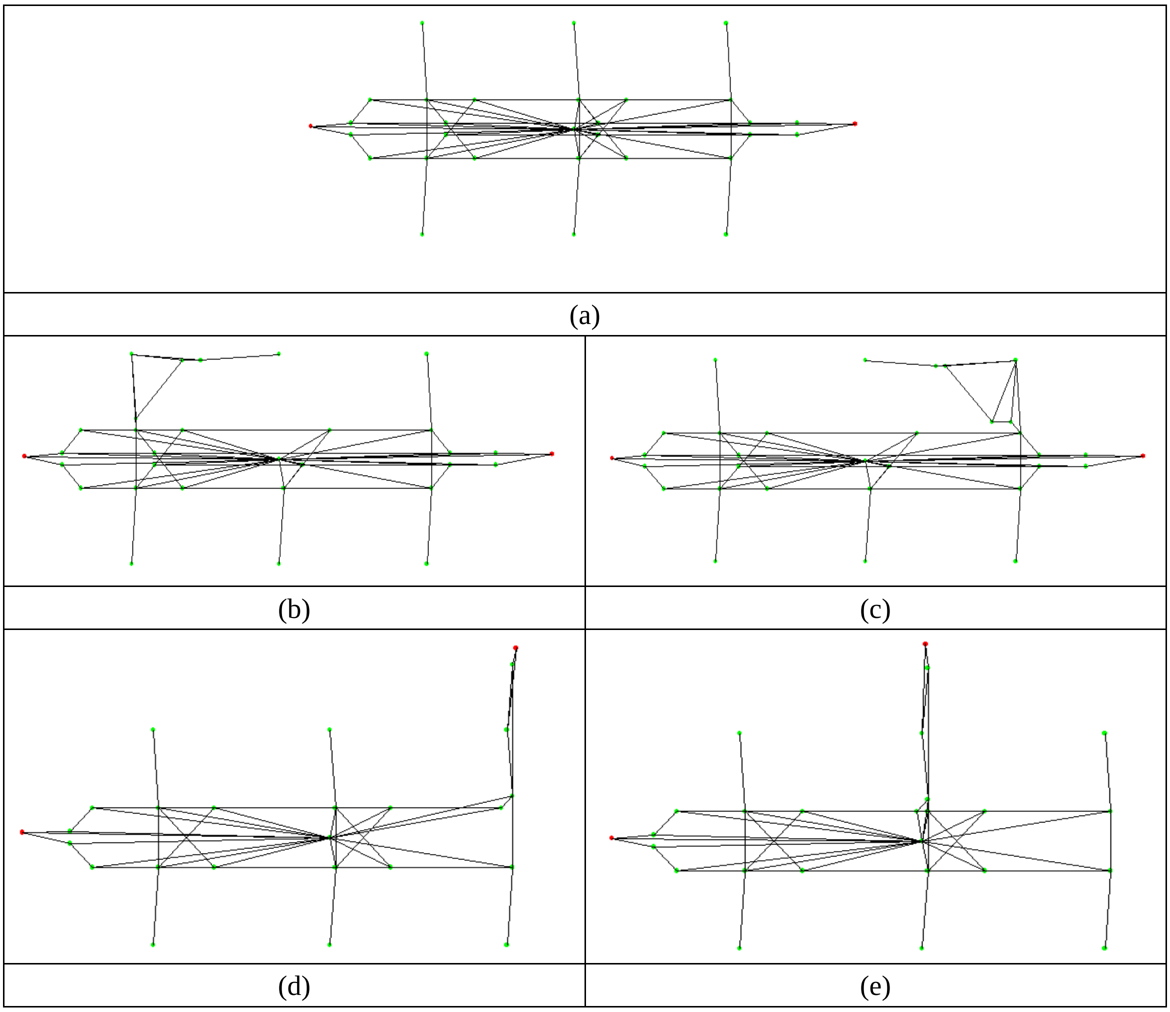
Table 7.4.2: Calculating results for comparing buildings with different external exit location by Model 1. Blue numbers represent minimum values

| Figures | Figure 7.4.2 (a) | Figure 7.4.2 (d) | Figure 7.4.2 (e) |
|---|---|---|---|
| Room 1 | 112.202 | 429.41 | 401.087 |
| Room 2 | 107.784 | 453.198 | 398.977 |
| Room 3 | 106.605 | 451.125 | 145.507 |
| Room 4 | 108.497 | 459.284 | 389.34 |
| Room 5 | 110.153 | 149.63 | 414.193 |
| Room 6 | 110.649 | 409.884 | 386.366 |
| Room 7 | 96.6085 | 401.67 | 359.713 |
| **Average distance (Metres)** | **107.49** | **393.45** | **356.45** |
| **Number of Nodes** | **26** | **21** | **22** |
| **Number of Arcs** | **58** | **43** | **45** |

From Table 7.4.2, the average travel distance of Figure 7.4.2 (a) obtained the minimum value, and Figure 7.4.2 (d) obtained the maximum value. For Figure 7.4.2

(d), Room 5 obtained the minimum value when the external exit is connected directly to it. For Figure 7.4.2 (e), room 3 obtains the minimum value when the external exit is connected directly to it. However, the problem is the average distance for these buildings is also large values given their total area. This is especially the case for Figure 7.4.2 (d) and Figure 7.4.2 (e).

Finally, the two external exits cases as defined in Section 7.4.1 will be used to compare the analysis of Model 1. The fine node route-based graphs representation of Figure 7.4.3 (a) to Figure 7.4.3 (e) are shown as the Figures 7.4.5 (a) to 7.4.5 (e) respectively.



(a)

(b) (c)

(d) (e)

Figure 7.4.5: Fine mesh route-based graphs for Figures 7.4.3 (a) to 7.4.3 (e)

Applying Model 1, the calculated results for Figure 7.4.3 (a) to Figure 7.4.3 (e) are shown in Table 7.4.3:

Table 7.4.3: Calculated results for comparing buildings with two different external exit location by Model 1. Blue numbers represent minimum values

| Figures | Figure 7.4.3 (a) | Figure 7.4.3 (b) | Figure 7.4.3 (c) | Figure 7.4.3 (d) | Figure 7.4.3 (e) |
|---|---|---|---|---|---|
| Room 1 | 106.742 | 97.436 | 87.1594 | 174.002 | 185.457 |
| Room 2 | 100.737 | 94.9017 | 93.5431 | 184.396 | 180.016 |
| Room 3 | 98.7041 | 111.32 | 118.887 | 183.101 | 63.5679 |
| Room 4 | 96.0225 | 90.8984 | 89.5905 | 187.514 | 188.512 |
| Room 5 | 98.5496 | 93.0651 | 106.897 | 63.755 | 180.457 |
| Room 6 | 101.498 | 93.701 | 95.7404 | 176.261 | 189.43 |
| Room 7 | 78.1548 | 72.0064 | 68.0687 | 156.626 | 155.599 |
| **Average distance (metres)** | **97.21** | **93.33** | **94.26** | **160.80** | **163.29** |
| **Number of Nodes** | **31** | **31** | **32** | **25** | **26** |
| **Number of Arcs** | **74** | **68** | **70** | **56** | **58** |

From Table 7.4.3, the first three figures obtained almost identical values, and the last two buildings obtained almost the same values. Comparing the results as shown in Table 7.4.3 with the results obtained in Table 7.4.1 and Table 7.4. We can see the results do not make sense since a two exits geometry should obtain better results than a single exit geometry. Some results shown in Table 7.4.3 are greater than the values show in Table 7.4.1. It is inconceivable that Figure 7.4.3 (d) and Figure 7.4.3 (e) are more complex than Figure 7.4.3 (a). Even for the buildings from Figure 7.4.3 (a) to 7.4.3 (c), the results also seem too high.

To solve the problems in Model 1, two approaches can be considered. One is to improve the route graph representation of the building; the other is to improve the behavioural rules used during wayfinding process. In the Model 2, the route-based graph is kept the same as in Model 1. A small improvement for the simulating rules will be outlined in the next section.

### 7.4.2.2 Model 2: Improving the behaviour rules

#### 7.4.2.2.1 The description of the Model 2

In Model 2, the graph representations of the buildings remains unchanged, which means the fine node route-based representation is still used in this model. However, some simple improvements to the rules have been done. In this section, two new rules are included in the wayfinding process.

1) The occupants step forward at random, and are never allowed to return to their previous node unless (2).

2) The occupants are allowed to return to their previous node if and only if there is no other option available.

These two extra rules allow the evacuees to have some local route memory ability.

For this model, a detail description of the rules is outlined below:

1) The simulation process is also based on the fine node route-based graph representation of a structure.
2) An occupant is placed at each room node in the route-based graph.
3) For each evacuee at the $i^{th}$ room node in the building, the following rules are used to calculate the average travel distance to find an available exit.
   a) The direction of the initial step from the starting node is chosen at random from the connected arcs.
   b) After the first step, all of the room nodes and the arcs connected with them will be removed from the graph; this means the evacuee will never travel to another room node during his wayfinding process.
   c) The evacuee selects the next node at random, and never returns to the last node they have visited unless it is the only option available.
   d) When the evacuee reaches an available exit, one trial ends and the travel distance is stored in memory. This cycle is repeated 1000 times. Then the average distance D(i) is calculated, D(i) is used to indicate the distance to

find an available exit for the evacuee starting at the *ith* node.

4) At each room node in the graph, this cycle is repeated 1000 times, then the average distance D(j)(j=1,2…N) is calculated, where N is the number of room nodes.
5) All the distances D(j) can be summed together to give the total distance. Then the average distance D(average) is calculated.

#### 7.4.2.2.2 The analysis of the results generated utilising Model 2

All the test cases defined in Section 7.4.1 will be used to analyse to efficiency of Model 2. In this model, the behavioural rules will also be applied to the fine node route-based graph representation of buildings. Therefore, all graphs generated for Model 1 test cases will be used in Model 2. Applying the new rules in Model 2, the calculated results for Figures 7.4.2 (a) to 7.4.2 (e) are shown in Table 7.4.4

Table 7.4.4.: Calculating results for comparing buildings with only one exit by Model 2

| Figures | Figure 7.4.2 (a) | Figure 7.4.2 (b) | Figure 7.4.2 (c) | Figure 7.4.2 (d) | Figure 7.4.2 (e) |
|---|---|---|---|---|---|
| Room 1 | 52.0348 | 50.7014 | 49.6514 | 271.588 | 278.96 |
| Room 2 | 51.4851 | 47.8661 | 52.0605 | 265.557 | 272.742 |
| Room 3 | 49.4004 | 54.9076 | 54.609 | 251.929 | 113.098 |
| Room 4 | 52.7567 | 48.5468 | 49.523 | 249.035 | 274.913 |
| Room 5 | 51.9007 | 46.6823 | 52.0837 | 92.5404 | 281.923 |
| Room 6 | 53.9534 | 47.648 | 46.6638 | 240.773 | 260.435 |
| Room 7 | 50.8958 | 47.1603 | 45.8078 | 246.144 | 271.548 |
| **Average distance (metres)** | **51.77** | **49.07** | **50.057** | **231.08** | **250.52** |

From Table 7.4.4, the first three buildings obtain similar values for the average distance. In all cases, Room 7 obtained the minimum value, hence is a good starting location with the quickest time to exit. For Figure 7.4.2 (b) and Figure 7.4.2 (c), Room 3 obtains the maximum values when changing the position of the door. The average travel distance in all of the cases is smaller than the ones obtained when applying Model 1.

Table 7.4.5 is the calculated results when applying the rules as described for Model 2 for the two exits test cases as defined in Section 7.4.1. All graph representations of these cases are the same as that in Model 1. According to Table 7.4.5, the first three Figures obtained almost identical values and the last two buildings obtained almost the same values. When comparing these results to Table 7.4.4, the Figure 7.4.3 (d) and Figure 7.4.3 (d) the travel distances are a lot larger than Figures 7.4.2 (a) to 7.4.2 (c).

Table 7.4.5: Calculated results for comparing buildings with two different external exit location by Model 2

| Figures | Figure 7.4.3 (a) | Figure 7.4.3 (b) | Figure 7.4.3 (c) | Figure 7.4.3 (d) | Figure 7.4.3 (e) |
|---|---|---|---|---|---|
| Room 1 | 52.6088 | 51.8657 | 47.5346 | 124.227 | 123.759 |
| Room 2 | 53.8028 | 51.3372 | 52.2962 | 120.468 | 126.681 |
| Room 3 | 53.48 | 54.7542 | 49.7202 | 124.239 | 57.9909 |
| Room 4 | 55.5462 | 48.9408 | 54.3796 | 124.46 | 128.821 |
| Room 5 | 55.4207 | 50.4284 | 52.9739 | 51.2405 | 125.89 |
| Room 6 | 55.3644 | 47.1908 | 51.112 | 109.077 | 131.701 |
| Room 7 | 48.6638 | 44.6081 | 43.6544 | 156.626 | 116.177 |
| **Average distance (metres)** | **53.55** | **49.87** | **50.23** | **111.46** | **115.85** |

Basically, the rules in Model 2 are a large improvement in regards to the calculated travel distance for the evacuee searching for an available exit. All value in this model is smaller than those obtained by Model 1. However, the buildings with two exits are also considered more complex than a single exit structure which is not a valid result.

During the process of generating the fine node route-based graph in buildingEXODUS, the waypoint nodes were generated based on the fine mesh already use for buildingEXODUS. However, the positions of the waypoint nodes are uncertain. That means there are a lot of choices for waypoints location. Hence different positions for waypoint nodes will influence the simulation results. At the same time this also influences the accuracy of the route graph representation. For example, Figure 7.4.3 (a) is a symmetrical structure, so the graph representation should also be a symmetrical graph, but Figure 7.4.5 (a) not symmetrical. By comparing the graph representation of Figure 7.4.3 (b) and Figure 7.4.3 (c), the same problem also exists.

### 7.4.2.3 Model 3: An improved Complexity model derived from the new 'route-based graph'

#### 7.4.2.3.1 Model 3 overview

This model uses the new 'route-based graph' representation of a building structure as defined in Chapter 6, and the wayfinding behavioural rules are the same as used in Model 2.

#### 7.4.2.3.2 The analysis of the results generated with Model 3

In this section, the test cases as defined in Section 7.4.1 are still used to examine the efficiency of the Model. The new 'route-based graphs' representations for Figures 7.4.2 (a) to 7.4.2 (e) are as shown in Figures7.4.6 (a) to 7.4.6 (e) respectively. And the Figures 7.4.7 (a) to 7.4.7 (e) are the new 'route-based graphs' for Figures 7.4.3 (a) to 7.4.3 (e). As described in Chapter 6, the arcs in the new 'route-based graph' are directed connections. The arrows to represent the direction have been omitted in these graphs, for simplicity.

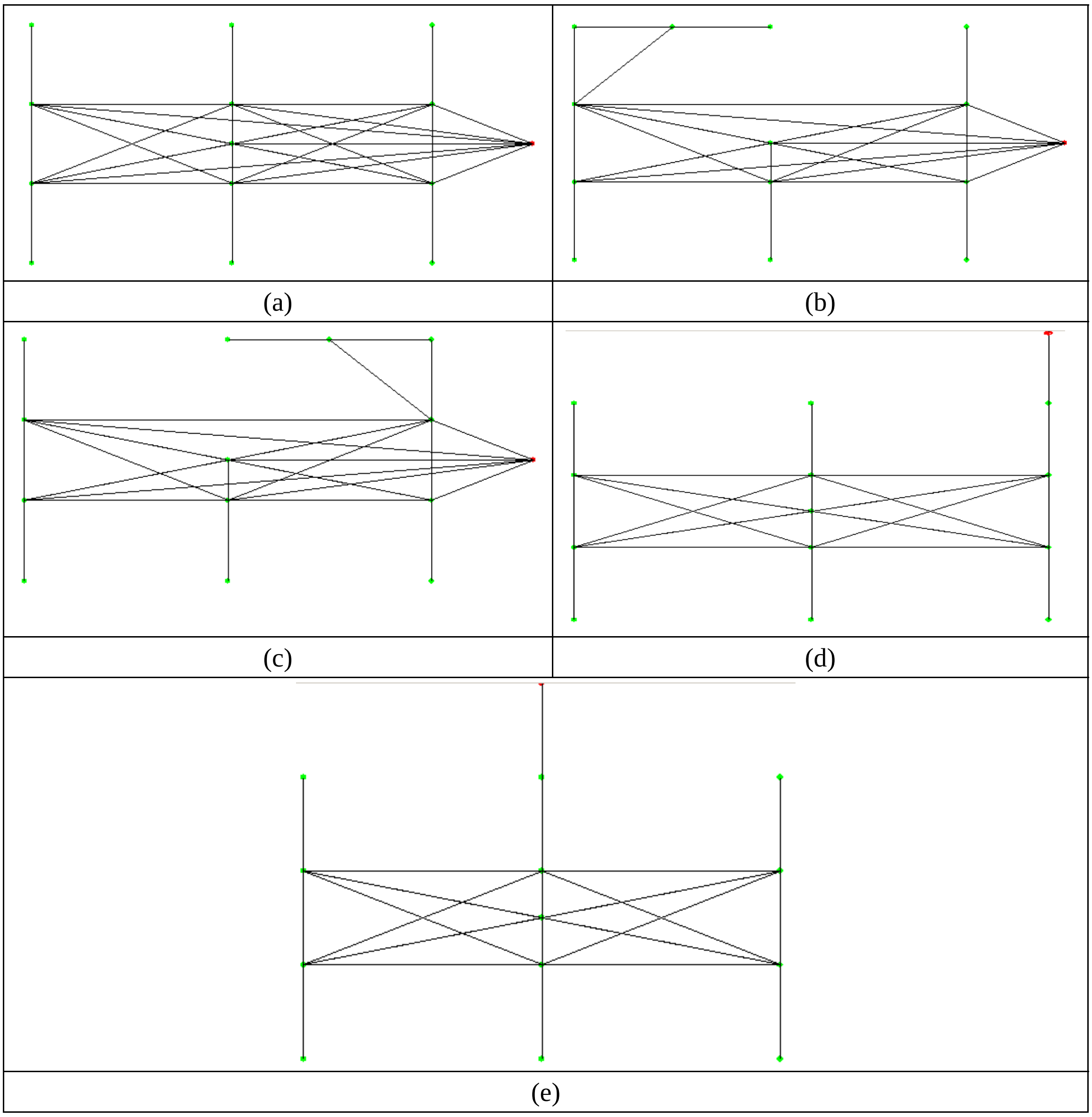
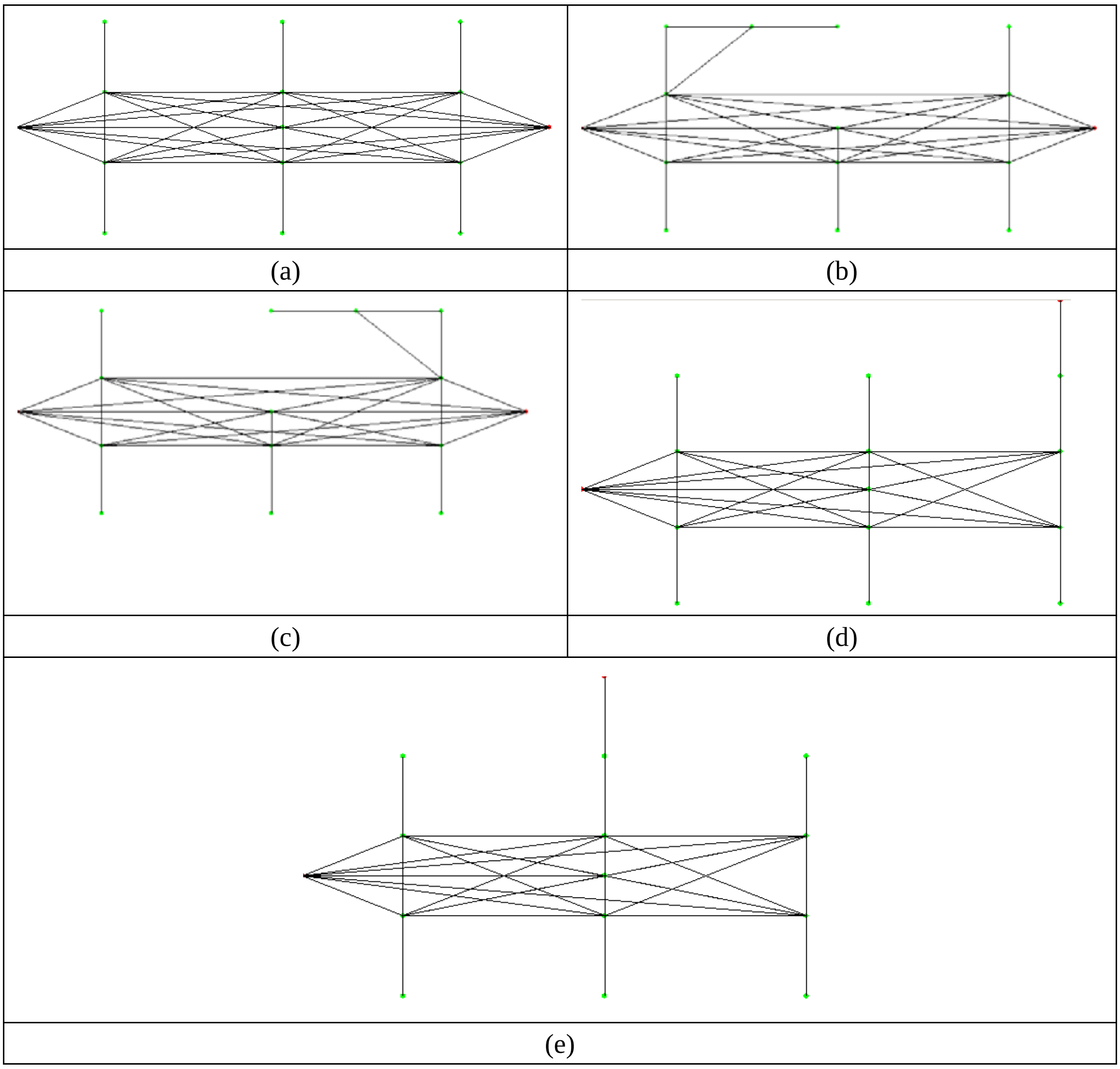


Figure 7.4.6: New ‘route-based graph representations for Figures 7.4.2 (a) to 7.4.2 (e)



(a)

(b)

(c)

(d)

(e)

Figure 7.4.7: New ‘route-based graph’ representations for Figures 7.4.3 (a) to 7.4.3 (e)

Applying Model 3, the calculated results for Figure 7.4.2 (d) and Figure 7.4.2 (e) are as shown in Table 7.4.6:

Table 7.4.6: Calculating results for comparing buildings with only one exit by Model 3

| Figures | Figure 7.4.2 (a) | Figure 7.4.2 (b) | Figure 7.4.2 (c) | Figure 7.4.2 (d) | Figure 7.4.2 (e) |
|---|---|---|---|---|---|
| Room 1 | 54.9786 | 55.3 | 53.5867 | 259.371 | 279.996 |
| Room 2 | 54.2733 | 51.2669 | 51.2001 | 271.508 | 276.539 |
| Room 3 | 52.4468 | 56.2997 | 53.9738 | 270.539 | 121.256 |
| Room 4 | 53.2989 | 47.5421 | 47.776 | 258.658 | 266.446 |
| Room 5 | 53.6057 | 48.8811 | 53.1358 | 110.584 | 261.74 |
| Room 6 | 52.7039 | 48.1648 | 46.4194 | 268.453 | 270.993 |
| Room 7 | 52.8838 | 47.2309 | 49.496 | 261.057 | 275.781 |
| **Average distance (metres)** | **53.45** | **50.66** | **50.79** | **242.88** | **250.39** |
| **Number of Nodes** | **14** | **14** | **14** | **14** | **14** |
| **Number of Arcs** | **34** | **29** | **29** | **29** | **29** |

According to the Table 7.4.6, the first three buildings obtain similar values for the average distance. In all cases, Room 7 obtained the minimum value. For Figure 7.4.2 (b) and Figure 7.4.2 (c), Room 3 obtains the maximum values when changing the position of door. The average travel distance in all of these cases have similar results when compared to Model 2 for the single exit cases.

Applying Model 3, the calculated results for Figure 7.4.3 (a) and Figure 7.4.3 (e) are as shown in Table 7.4.7:

Table 7.4.7: Calculated results for comparing buildings with two different external exit location by Model 3

| Figures | Figure 7.4.3 (a) | Figure 7.4.3 (b) | Figure 7.4.3 (c) | Figure 7.4.3 (d) | Figure 7.4.3 (e) |
|---|---|---|---|---|---|
| Room 1 | 34.1159 | 36.0022 | 33.7915 | 49.9161 | 50.4983 |
| Room 2 | 34.1204 | 32.8336 | 33.043 | 50.1218 | 49.3242 |
| Room 3 | 35.087 | 38.0636 | 39.4575 | 48.6773 | 23.1376 |
| Room 4 | 34.0533 | 33.1574 | 31.7113 | 49.3269 | 46.4363 |
| Room 5 | 35.6205 | 34.0377 | 36.1834 | 24.6742 | 50.4815 |
| Room 6 | 35.6801 | 33.5489 | 33.5527 | 49.7366 | 51.1721 |
| Room 7 | 30.481 | 29.8624 | 29.5531 | 43.7488 | 45.6512 |
| **Average value** | **34.16** | **33.92** | **33.91** | **45.17** | **45.24** |
| **Number of Nodes** | **15** | **15** | **15** | **15** | **15** |
| **Number of Arcs** | **42** | **36** | **36** | **36** | **36** |

From Table 7.4.7, the average travel distances for Figures 7.4.3 (a) to 7.4.3(e) obtain more reasonable values when compared with Table 7.4.6. The average travel distances for buildings with two exits obtain an obvious smaller valve than just only one exit. Especially for the last two buildings, the simulation results seem to obtain a reasonable value when compared to Model 2.

Basically, in Model 3, a lot of problems which occurred in Model 1 and 2 have been solved. However, there are two problems which still exist:

1) The average distance for Figure 7.4.2 (d) and Figure 7.4.2 (e) are far too large when compared to the value in Figure 7.4.2 (a) given that the location of the door is the only difference.

2) All the travel distances tend to be higher than a reasonable value as expected by the wayfinding process.

### 7.4.2.4 Model 4: Final model based on improving the algorithm utilised by Model 3.

#### 7.4.2.4.1 Overview of Model 4

The improvement over Model 3 is mainly based on the behaviour rules used during the wayfinding process. The rules which have been used in Model 3 are very simple and only consider the occupant’s local route memory ability. A better suit of rules should enable a more realistic simulation of human wayfinding behaviour within a building, given that the person has no knowledge of the building and is not utilising any signs. Hence the improved rules are based on the following aspects:

1) Route memory ability of the evacuees.

During the simulation, the occupant should never arrive at a bad node more than once during the wayfinding process. The occupant should record all bad nodes, so that after leaving a bad node it is removed from the new 'route-based graph'.

A bad node is defined as:

a) A node with only one arc, which is not an available exit node, which the occupant has travelled from.

b) An internal exit node which has been travelled to by the occupant and recognizes as not an available external exit, is only removed if the resulting graph remains connected.

2) The route chosen probability is based on the route length and the width of the exit or travel direction.

a) From the arcs which have never been travelled before choose the arcs according to their length and the width of their connected doors. For example, increase the probability of choosing shorter arcs and the wider doors.

b) If all the arcs have been used then choose an arc according to its length and the width of its connected door. For example, more probability of selecting an arc is given to shorter arcs and the wider doors.

3) In most situations, the travel time is a more important factor when compared to the travel distance during wayfinding process. Hence, the average wayfinding time will be used as the building complexity measure in Model 4. The travel time will be influenced by more factors than travel distance, for example, the number of the decision points, the number of arcs connected with the decision point, the width and type of door.

4) To achieve a reasonable result, the algorithms should consider the areas represented by room nodes, which will influence the probabilities of an occupant starting in any given room. We then calculate the average travel time using a weight sum of the

wayfinding times from each room. The average travel time are calculated based on the area and size of each room and is defined by the following equation.

$$Complexity\ Time = \sum_{i=1}^{n} \left( \frac{Area_n}{\sum_{j=1}^{n} Area_j} \times t_i \right)$$

Where n is the number of room nodes, and $t_i$ is wayfinding time from $i^{th}$ room.

In Model 4, several of the key points described above have been implemented. The main improvement comparing to Model 3 is that an occupant has global route memory. According to the history of wayfinding, an occupant will avoid travelling to a bad node as defined in point 1 above. The current model utilises the average travel time as the building complexity measure. In Model 4, the expected wayfinding time is calculated by dividing the travel distance by the speed of 1.5m/s which has been previously used in Chapter 5. This implies that the travel time is calculated based on the following assumptions: an occupant will start their wayfinding process immediately which means their response time is zero and all occupants walk at the same constant speed (1.5 m/s). An occupant makes their wayfinding decision instantaneously at each node which means the delay at each decision points has been omitted. Also, it is assumed that all doors are open. These assumptions are obviously unrealistic, but are acceptable for a scenario independent model. The time delays occurring at each decision point and an internal exit node could be quantitative analysis in further work. Changing the route choice probability based on the route length and the width of the exit or travel direction will also be left to further work.

Applying new wayfinding rules, a new building complexity measure is generated which is will be called the *Complexity Time Measure.* The *Complexity Time Measure* can be used to answer the following question:

> *If an occupant is positioned at a random location within a building, on average how long does the occupant take to find an available exit.*

When using *The Complexity Time Measure* to compare different buildings for evacuation ability the building which obtains the smaller value could be considered to be best for evacuation.

### 7.4.2.4.2 Model 4 results

In this section, the test cases as defined in Section 7.4.1 will be used to examine the efficiency of Model 4. The graph representations for Figures 7.4.2 (a) to 7.4.2 (e) are the same as the new 'route-based graphs' which were generated for Model 3, which are shown as Figures 7.4.6 (a) to 7.4.6 (e) respectively. And the Figures 7.4.7 (a) to 7.4.7 (e) are the new 'route-based graphs' for Figures 7.4.3 (a) to 7.4.3 (e).

Applying Model 4, the calculated results for Figure 7.4.2 (a) and Figure 7.4.2 (e) are as shown in Table 7.4.8:

Table 7.4.8: Calculating results for comparing buildings with only one exit by Model 4

| Figures | Figure 7.4.2 (a) | Figure 7.4.2 (b) | Figure 7.4.2 (c) | Figure 7.4.2 (d) | Figure 7.4.2 (e) |
|---|---|---|---|---|---|
| Room 1 | 38.5497 | 42.4395 | 37.7724 | 54.7046 | 53.2823 |
| Room 2 | 37.7092 | 39.1672 | 38.3013 | 54.9356 | 52.0533 |
| Room 3 | 38.7995 | 41.5673 | 41.1313 | 57.8003 | 28.3919 |
| Room 4 | 39.2882 | 39.4526 | 38.072 | 57.4786 | 55.6408 |
| Room 5 | 38.533 | 37.566 | 41.3952 | 31.3989 | 53.0179 |
| Room 6 | 37.9115 | 38.161 | 39.0899 | 57.5396 | 52.7513 |
| Room 7 | 36.4463 | 37.0065 | 37.7108 | 58.8813 | 56.1015 |
| **Average distance ( metres)** | **38.17** | **39.33** | **39.06** | **53.24** | **50.17** |
| ***Complexity Time Measure* (Seconds)** | **25.44** | **26.22** | **26.04** | **35.49** | **33.44** |

According to Table 7.4.7, the first three buildings almost obtain the same *Complexity Time Measure*, which means the average travel time has less influence when altering an internal exit position for Room3. The *Complexity Time Measure* of Figure 7.4.2 (d) and Figure 7.4.2 (e) are also in the same order. However, Figure 7.4.2 (d) and Figure 7.4.2 (e) are more complex than Figure 7.4.2 (a), which means they have exits in disadvantageous locations.

Applying Model 4, the calculated results for Figure 7.4.3 (a) to 7.4.3 (e) are as shown in Table 7.4.9:

Table 7.4.9: Calculated results for comparing buildings with two different external exit location by Model 4

| Figures | Figure 7.4.3 (a) | Figure 7.4.3 (b) | Figure 7.4..3 (c) | Figure 7.4.3 (d) | Figure 7.4.3 (e) |
|---|---|---|---|---|---|
| Room 1 | 32.1582 | 34.5222 | 29.4936 | 38.0256 | 37.1543 |
| Room 2 | 31.6792 | 30.5314 | 29.8875 | 36.6941 | 37.1197 |
| Room 3 | 29.2215 | 34.871 | 35.0687 | 36.1251 | 18.79 |
| Room 4 | 30.3085 | 29.6758 | 29.1777 | 37.123 | 36.6791 |
| Room 5 | 30.8027 | 29.2598 | 33.1449 | 21.2965 | 38.5608 |
| Room 6 | 31.7961 | 30.7213 | 30.2025 | 37.683 | 36.7102 |
| Room 7 | 28.8381 | 26.8233 | 27.8173 | 36.9898 | 34.8327 |
| **Average distance ( Metres)** | **30.68** | **30.91** | **30.69** | **34.84** | **34.26** |
| ***Complexity Time Measure* (Seconds)** | **20.45** | **20.61** | **20.45** | **23.23** | **22.84** |

From Table 7.4.9, the *Complexity Time Measure* for these two exits buildings are all smaller than the one exit building as shown in Table 7,4,8. The *Complexity Time Measure* for Figure 7.4.3 (d) and Figure 7.4.3 (e) are more than the *Complexity Time Measure* of the first three figures since the exit location in Figure 7.4.3 (d) and Figure 7.4.3 (e) are disadvantageously positioned.

Basically, all the simulation results applying Model 4 seems to obtain reasonable average travel distance and *Complexity Time Measures*. All the problems addressed in Model 3 have been resolved.

## 7.5 Validations based on test cases defined in Chapter 4

In this section, the test cases as defined in Chapter 4 will be used to verify the efficiency of the *Complexity Time Measure*. As described in Chapter 5, we cannot find any benchmark for verifying the validation of the *Complexity Time Measure* generated in this chapter. An evacuation simulation model, even though it is scenario dependant, it was still chosen as validation tool as used in Chapter 4 and 5, hence will be used again here. As described in Chapter 4, Case 1 is not suitable for comparing simulation results using buildingEXODUS. So in this section, the analysis will just be based on *Cases 2 to 5*. The *Complexity Time Measure* can also be compared with the results obtained based on *Distance Graph Method* as described in Chapter 4 and the *Global Complexity (PAT)* as outlined Chapter 5.

Figures 7.5.1 to 7.5.4 are the new 'route-based graph' representation of the Cases 2 to 5 respectively. For simplicity the room nodes have been omitted in these graphs. From Figures 7.5.1 to 7.5.4 we can see that the new 'route-based graphs' are extremely complex when compared to the room graph representation as generated in Chapter 4. Route-based graphs often contain a larger number of arcs and nodes than their room-based graph equivalent.

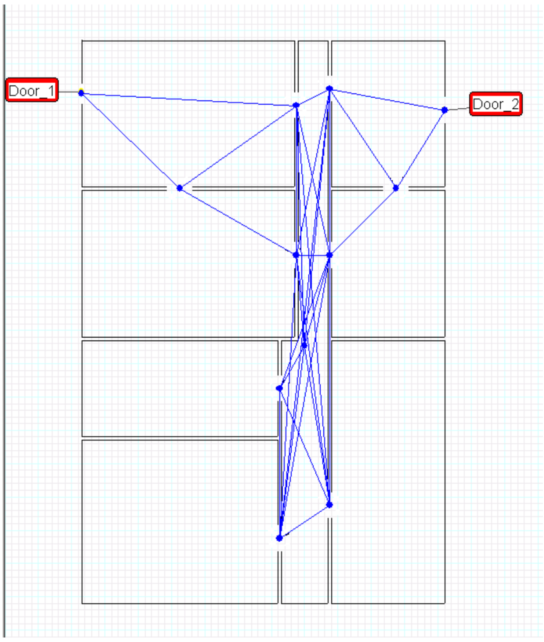


Figure 7.5.1: navigation graph of Case 2

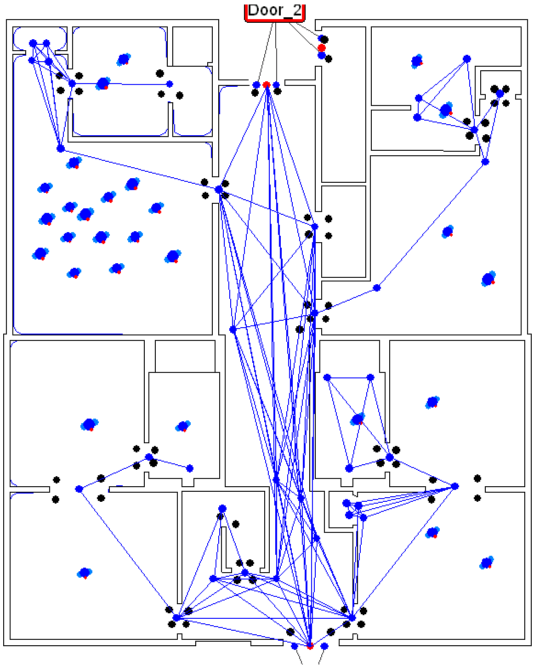


Figure 7.5.2: navigation graph of Case 3

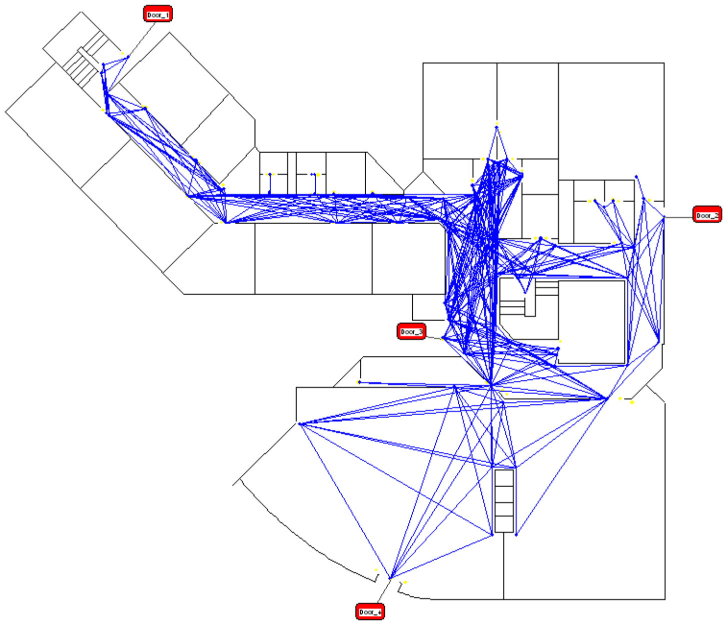


Figure 7.5.3: navigation graph of Case 4

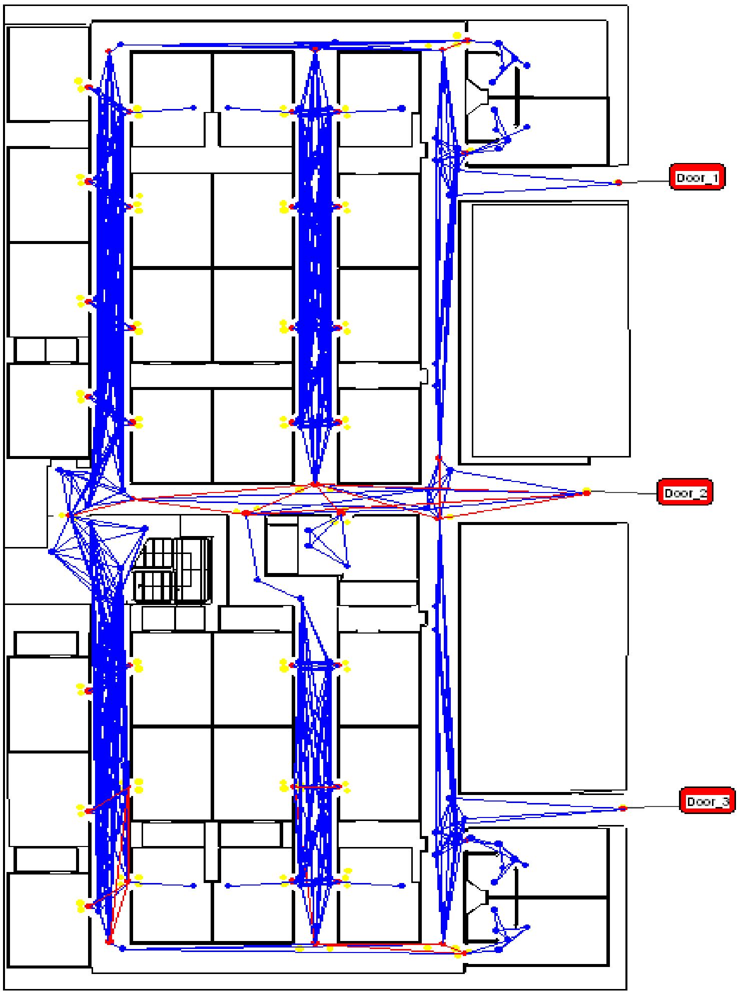


Figure 7.5.4: navigation graph of Case 5

In Table 7.5.1, the second column, shown in red, is the *Complexity Time Measure* for Cases 2 to 5 respectively.

Table 7.5.1 Comparing results for cases defined in Chapter 4

| Cases | *Complexity Time Measure* | Global Complexity (PAT) | Distance Graph Method | Simulation time(15 populations) | Simulation time(100 populations) |
|---|---|---|---|---|---|
| Case 2 | 33.37 | 24.2430 | 279.852 | 34.75 | 112.09 |
| Case 3 | 42.29 | 38.4492 | 982.711 | 45.14 | 54.25 |
| Case 4 | 101.15 | 83.9908 | 2994.75 | 61.79 | 77.59 |
| Case 5 | 166.47 | 91.2435 | 5718.05 | 47.56 | 61.76 |

According to Table 7.5.1, we see that the *Complexity Time Measure*, *Global Complexity (PAT)* and *Distance Graph Method* obtain the same complexity order for the buildings cases 2 to 5. These models all indicate that the building in Case 2 is the simplest and Case 3 is the most complex. When using the *Complexity Time Measure*, *Global Complexity (PAT)* and the *Distance Graph Method* to calculate the building complexity, these models assumed that the evacuees have no knowledge of the building. They mainly consider the complexity of the process of recognizing the building.

When comparing the *Complexity Time Measure* to evacuation simulation results, generated by buildingEXODUS, Cases 2 to 5, the results as shown in Table 7.5.1 indicate that they do not obtain the same order. The evacuation time for Case 5 is smaller than Case 4. However, as described in Chapter 4, the buildingEXODUS evacuation model has assumed every evacuee has complete knowledge of the building. Hence, the evacuees will always choose the nearest exit via the shortest path. Therefore, the *Complexity Time Measure* and the simulation in buildingEXODUS are being used to validate different aspect of the building. The results of the building complexity analysis were not always consistent with the result generated by buildingEXODUS. Moreover, from Table 7.5.1 the difference between the *Complexity Time Measure* and the evacuation time for simple buildings (for example Case 2 and

Case 3) are very small. However, the different between the *Complexity Time Measure* and the evacuation time are obvious for Case 4 and Case 5 since an occupant needs to spend more time to find an exit if he has no knowledge of a complex building.

When comparing the *Complexity Time Measure* to the *Global Complexity (PAT)* for Case 2 to 5, the *Complexity Time Measure* is always larger than the *Global Complexity (PAT).* This is mainly because these two models have been applied two different graph representations. The *Complexity Time Measure* was generated based on the route graph representation of these cases. However, the *Global Complexity (PAT)* is generated based on the room graph representation, which in most situations cannot express the real environment of the building. Also the arcs in the room-based graph cannot describe the real route the evacuee needs to travel. So such a simple outline of the structure and just specifying the connectivity between compartments is not enough for the required level of complexity to simulate the wayfinding process.

## 7.6 Limitations

The *Complexity Time Measure* developed in this chapter has been employed to compare the evacuation ability for different building. However, some limitations are present. The *Complexity Time Measure* is generated based on a set of behaviour rules used during the wayfinding process. The expected wayfinding time is calculated just by divided the travel distance by 1.5m/s as described in Chapter 5. This implies that the travel time is calculated based on the following assumptions: (1) an occupant will start their wayfinding process immediately which means the response time is zero, (2) all occupants walk at the same constant speed (1.5 m/s), (3) an occupant makes their wayfinding decisions at once at each node which mean the delay in decision points has been omitted, also, (4) we have assumed that all doors are opened. These assumptions are obviously unrealistic. Also, in the *Complexity Time Measure*, the probability for an occupant to choose a local route should be based on a lot of factors, for example, the local route length, the width of the door or travel direction etc, which are not implemented in this *Complexity Time Measure*, but could be included since they are scenario independent.

## 7.7 Summary

For comparing different building for the purpose of evacuation ability, The *Distance Graph Method* and the *Global Complexity (PAT))* have been developed. However, some limitations have been highlighted as described in Chapter 4 and Chapter 5. Therefore, a new graph representation has been defined in Chapter 6, which is called a new 'route-based graph'. In this Chapter, the *Complexity Time Measure* is outlined based on a set of behaviour rules, these rules are utilized on this new 'route-based graph' representation of a building. The *Complexity Time Measure* addresses the following question:

> *If an occupant is positioned at a random location within a building, on average how long does the occupant need to spend to find an available exit.*

When using The *Complexity Time Measure* to compare two or more structures with respect to evacuation ability, the building with the smallest value of complexity is considered to be the simplest to navigate. The results generated by The *Complexity Time Measure* for some of the test cases, presented in Chapter 4, have been validated against simulated results generated by buildingEXODUS and the calculated results obtained by the *Distance Graph Method* and the *Global Complexity (PAT))*. However, the *Complexity Time Measure* still has some weakness (see Section 7.6), which have been left as further work.

# Chapter 8 Conclusions and Further work

## 8.1 Introduction

This chapter summaries the worked done and concludes the thesis with answers to the original questions as posed in Chapter 1. It also identifies some new questions and possible improvements to the current research, which will require further investigation. Detailed conclusions have already been provided at the end of each chapter so the conclusions in this chapter will be of a more general nature.

Over the past two decades, more than 30 evacuation models have been developed and the corresponding simulation tools have been implemented to reproduce people's movement patterns. Despite the various aims of these models, the evacuation modelling approach has not addressed the more fundamental notion of building complexity for a given building. Any multi-compartment building may be regarded as a complex object or structure. However, evacuation models do not provide an understanding of why one building is better than another for evacuation; i.e. is less complex and hence easier to navigate.

There are techniques which attempt to compare different buildings for building complexity, for example Donegan's methods. In addition, there are also a number of other graph measures that are used to measure the relative accessibility of different spaces in a spatial system, e.g. Point depth entropy [HH87], Clustering coefficient [WS98][ST05]. However, these measures only provide local information relating to accessibility and do not provide a global measure of the evacuation capabilities of a building; i.e. allowing the engineer to make an overall assessment of the structure. Donegan's method was derived from information theory based on the concept of Shannon entropy; however, it can only be used to calculate the complexity of a tree graph and it does not take into consideration the important factor of 'travel distance'.

The main objective of this study was to develop a *‘building complexity’* model that can be applied to compare different buildings for evacuation wayfinding capability. When generating such a complexity measure for a building layout, there are two factors that need to be considered. The first is the representation of the building structure (i.e. how we select a graph capable of describing the building layout for wayfinding purposes). Second, a set of equations need to be developed that calculate a building complexity measure that is based on the specified graph representation of the building.

## 8.2 Contributions and summary of work

A review of the research background to of this thesis was presented in Chapter 1. The main aims and objectives of this thesis were discussed and outlined in Chapter 1 together with a number of questions related to the research carried out in this thesis~~.~~ The development of a ‘building complexity’ model has been identified as the main aim of the research presented. The associated research questions to the fundamental objectives as set out in Chapter 1 have been answered in the relevant chapters in detail. Therefore, this chapter will focus on the overall contributions of this research and provide an overall summary of answering the questions posed in Chapter 1.

### 8.2.1 Background Investigation and Research

Before the major research questions could be addressed the following question need to be considered.

**Question 1: Are there any existing graph measures which can be used to compare different buildings in relationship to building evacuation efficiency?**

A general review of available evacuation modelling techniques has been provided in Chapter 2, Section 2.2. However, none of these models provided a facility that could compare different buildings in relation to evacuation wayfinding capability. Chapter 2 also introduces the buildingEXODUS modelling in detail since the algorithms developed in this study have been compared with the results generated by this model for validating the building complexity measures presented in this thesis.

Chapter 3 focused on the various graph representation techniques for a building plan and associated graph measures. Three types of graph representation of a building plan were presented; room-based graph, visibility graph and axial map. This was then followed by a description of a number of graph measures which have been applied to analyse building plans. All these graph measures are primarily used to represent the relative accessibility of different locations in a spatial system and were not generated for the purpose of comparing the complexity of different buildings in terms of wayfinding. The only method that was identified which is available for comparing different buildings with respect to the evacuation wayfinding capability is Donegan’s method (see Chapter 3, Section 3.4). However, this method can only be applied to a tree graph, where in general this is not the case, since most buildings are far more complex with many circulation regions which form graphs with circuits. In the case of a graph with circuits Donegan’s method tries to calculate the complexities for all spanning trees of the graph. However, this is an NP-complete problem. In addition the Donegan’s method fails to consider other important factor, such as *travel distance*, *obstacles* and *travel time*.

### 8.2.2 The improvement of an existing building complexity measure from the theory and implementation

The second research question, addressed by this thesis, is:

**Question 2: How to extend Donegan's method to obtain a new algorithm to address the following two sub questions?**

- **Question 2.1: How can we introduce the concept of circuits into building graphs to calculate building complexity measures?**
- **Question 2.2: How can we introduce travel distance into building complexity to generate a new measure?**

In Chapter 4 *the 'Distance Graph method'* is presented which addresses the limitations of the Donegan's method. *The 'Distance Graph method'* was generated by introducing the concept of circuits and distance into the room-based graph of a building plan to calculate the building complexity measure. For generating *The 'Distance Graph method'*, there are several issues which have been addressed.

Firstly, some necessary complementary proofs are derived from Donegan's theories, which mainly include completing two proofs of Donegan's complexity theories, see Theorem 4.5.1 and Theorem 4.5.2 in Chapter 4, Section 4.5. These two theorems are important for subsequent work on subfloor networks [DP96]. However, these two theorems are not important for the extension of the Dongen's method to graphs that contain circuits.

Secondly, the Donegan's method was extended so that it could deal with complexity graphs which include circuits. For graphs with circuits, Donegan et al. try to solve this problem by generating all the spanning trees for a graph [DT99] [PD96] and then calculating the complexity for each of the spanning trees. From these set of spanning tree graphs they choose the maximum value as the complexity measure of original graph. In Chapter 4 (Section 4.5), a special spanning tree has been introduced which has been proved to represent the maximum complexity value. In other words, this spanning tree meets the requirements for calculating the complexity measure for any graph with circuits.

Thirdly, a distance measure has been incorporated into the complexity calculation, to form a new method which will be referred to as the '*Distance Graph method*'. This is achieved by defining two new variables $d^+$ and $d^-$. $d^+$ represents the distance travelled to gain positive instance of information and $d^-$ represents the distance travelled to gain negative information. $d^+$ and $d^-$ are used to replace $n^+$ and $n^-$ in Donegan's complexity calculation formula respectively (see Chapter 4, Section 4.5).

### 8.2.3 Development of new building complexity measures based on a room-based graph representation

The third research question answered by this thesis is:

**Question 3: Can we develop a new graph measure, based on a room graph, which answers the following question:**

> ***"What is the maximum distance the occupant needs to travel before successful egress from the building?"***

In Chapter 5 (Section 5.3), the development of new building complexity measures employing the room-based graph representation was described. Due to limitations of the Donegan's method and the '*Distance Graph method*', a new nodal complexity measure was produced.

As stated, in most evacuation models and the traditional safety codes, for example [TH94] [Ta91], the maximum travel distance (or the travel time) is a very important factor which influences evacuation ability. These vital factors for an occupant searching to find an available exit were not considered in Donegan's method and the '*Distance Graph method*'. So the main achievement of Chapter 5 (Section 5.3) is developing a set of new graph measures, based on a room graph, which answers the following question:

> '*For an occupant positioned at a random location within a building, what is the maximum distance or time the occupant needs to travel before a successful egression*?'

By comparing and analysing these new measures, the *Global Complexity (PAT)* has been shown to be a valid measure. The *Global Complexity (PAT)* measure considers some important factors such as wayfinding time, travel distance, and the areas of each room/compartment. In addition a mathematical comparison of the new nodal complexity against the *Distance Graph Method* has also been analysed in Chapter 5 (Section 5.2).

### 8.2.4 The Generation of a new 'route-based graph' representation for a building

The forth main question answered by this thesis is:

**Question 4: Is there another way to represent the building layout as a graph which overcomes the problem in room-based graph?**

A new kind of graph representation for a building structure is defined in Chapter 6, which is called a new 'route-based graph'. When comparing different buildings for the purpose of evacuation ability the *Distance Graph Method* and the *Global Complexity (PAT)* have been developed in Chapter 4 and Chapter 5. These models are all based on a room graph representation of a building structure in determining the building complexity. However a room-based graph is not capable of describing the actual routes taken during the wayfinding process. In most situations, the room-based graph cannot express the real environment of the building exactly, also the edges in the room-based graph cannot represent the actual route the evacuee will travel to an exit. So such a simple outline of the structure by just specifying the connectivity between compartments is not sufficient for the required level of complexity needed to simulate the wayfinding process.

The new ‘route-based graph’ defined in Chapter 6 (Section 6.6) has the capability of representing the routes that an evacuee could take during the wayfinding process. In the new ‘route-based graph’, the node types have been defined in three different categories. The first category represents the source locations which are known as room nodes. The second category represents the destinations which are described as the external exits nodes. And the last category navigation nodes that consist of two node types which represent the internal exits and the waypoints. There are three kinds of arcs defined within the new ‘route-based graph’. The first kind of route segments is employed to connect between room nodes to navigation nodes, and they are directed arcs which will point towards navigation nodes. The second kind of route segments represents the internal connections within the navigation nodes, and all of them are bidirectional. The last kinds of route segments are used to connect external exits to other kind of nodes, which are also directed arcs pointing to the external exits.

### 8.2.5 New building complexity models based on a route-based graph

The last question answered by this thesis is:

**Question 5: How can we apply a suitable technique to generate a measure of building complexity which can be used to answer the question *“If an occupant is positioned at a random location within a building, on average how far does the occupant needs to travel to find an available exit?”***

The *Distance Graph Method* and *Global Complexity (PAT)* presented in Chapter 4 and 5 are based on a tree room-based graph representation of a building plan. The new 'route-based graph', as defined in Chapter 6, includes many circuits for expressing the routes within the building environment. Even though, this research has sought a suitable method for breaking circuits, too much information provided by the edges would be lost. If we break the circuits in 'route-based graph' the resulting new tree graph would no longer denote the possible routes within the environment. Hence, the new 'route-based graph' as defined in Chapter 6 (Section 6.5) is not suitable for either "the *Distance Graph Method"* or "*Global Complexity (PAT)"*. Hence, the *Complexity Time Measure*, developed in Chapter 7, Section 7.4, is based on the route graph representation of a building plan.

In Chapter 7, Section 7.4, four new building complexity models were developed based on a random walk model applied to a route-based graph. The models presented in Chapter 7 assume that the wayfinding process occurs while navigating the route-based graph. For each model a set of navigational behaviour rules have been defined and each one was applied to the two different types of route-based graph representation, as described in Section 7.4.

The final model developed in Chapter 7 is the most important. This model applied the new 'route-based graph' representation of the building layout, see Section 7.4.2.4. A set of behaviour rules, are applied to this graph, and are used to generate the building complexity measure. This new method is referred to as the *Complexity Time Measure*. The *Complexity Time Measure* addresses the following question:

> *'If an occupant is positioned at a random location within a building, on average how long does the occupant need to spend to find an available exit?'*

In Chapter 7, Section 7.3, an analysis method has been applied to a directed graph where previously this was only possible in the case of an undirected graph [Gu98].

This analysis method was then applied to a directed graph to solve a simple wayfinding case to evaluate the building complexity of a building as represented by a new ‘route-based graph’.

### 8.2.6 Summary

The building complexity measures developed in this research are all standalone models, which mainly include the *Distance Graph Method*, the *Global Complexity (PAT)* and the *Complexity Time Measure*. These methods are all scenario independent methods, which are utilized to assess the different aspects of the evacuation efficiency of a building. They can also be used to understand why one building is easier to navigate than another. These methods can be used in conjunction with an evacuation model to find not just a building which is quick to evacuate, but is also a building which is simpler to evacuate. Hence, possibly reducing the likelihood of a person becoming lost on route to an exit or having to backtrack. A low scoring building should in theory help minimise the overall evacuation time and therefore reduce risk of people being exposed to any hazards or dangers unnecessarily. This research would be of use to a building designer or a fire safety engineer when planning to choose which layout of structure is best suited for egress.

## 8.3 Further work

In this section some new questions, which have arisen during the course of this research are discussed. Together with some suggested improvements to the models developed in this study.

### 8.3.1 The investigation of the multi-storey buildings

Currently, all complexity measures developed in this study are only available for a single storey building. However, in general, buildings consist of a number of floors. Hence, the following question needs to be answered:

> '*How can we generate a new complexity measure for a comparing multi-storey building for evacuation ability?* '

In Donegan's method [DP94] [DT98] [PD96] [DT99], they applied a vector to represent building complexity for multi-storey building. The basic idea is explained as: The stairwells on upper floor are considered as exit nodes and the stairwells on ground floor are considered as non-exit node [DT98] [PD96].

Based on this method, a k-storey building is considered, where *C(0), C(1), C(2)…C(k)* are used to represent the global complexity for each floor from ground to the $k^{th}$ floor. Then the complexity of the building is denoted by *(C (0), C (1), C (2)…C (k) ).* This vector representation method of a multi-storey building just supplies a mark of the complexity. It cannot be applied for comparing different building for the purpose of evacuation capability. Hence, a more reasonable method should be developed.

If we look at the *Complexity Time Measure,* as presented in this thesis, a possible solution to extending this method to a multi-story is outlined here. One method would be to do this in three stages.

1) The occupant needs to find a stairwell as his temporary destination.
2) The occupant will then travel through the stairwell to an exit floor.
3) Once on the exiting floor the occupant should then locate an available external exit. As such to cases need to be considered:
   - Case 1: If exit to the stairwell is directly connected to an external exit, the wayfinding time from the stairwell to exit can be calculated by a time delay.
   - Case 2: If stairwell exit not directly connected to an external exit, then the stairwell exit can be treated as a normal room node on the exiting floor.

The time taken during wayfinding process can be calculated as the sum of times taken during these three stages, as simulated using the *Complexity Time Measure*.

## 8.3.2 Additional information for new ‘route-based graph’

When generating a complexity measure for a building layout, two factors need to be considered. The first is the representation of the building structure. This means which kind of graph is suitable to describe the building layout. Second, a set of measures need to be generated to represent the building complexity based on the selected graph representation of the building. Hence, any improvement to building complexity measure should be based on these two key factors.

The new ‘route-based graph’ defined in Chapter 6 is capable of representing the actual routes that an evacuee will take during the wayfinding process. The *Complexity Time Measure* has been developed based on the new ‘route-based graph’ representation of a building. However, the description of such graph needs to contain more information that can be used by an occupant in deciding which neighbouring node or an arc to use. For example the probability of selecting any given arc or node could be influenced by the available width, exit size or obstacle value.

### 8.3.3 Improvement of the Complexity Time Measure

The *Complexity Time Measure* developed in Chapter 7 utilises the average travel time for the building complexity measure. In the *Complexity Time Measure,* the expected wayfinding time is calculated by dividing the travel distance by a constant speed of 1.5m/s. This means that the travel time is calculated based on the following assumptions: All occupants walk at the same constant speed of 1.5 m/s. In addition an occupant makes his wayfinding decision instantaneously at each node which means the delay at each decision points has been omitted. These factors could be incorporated into a scenario independent method to possibly produce a better complexity measure.

Further research, such as the quantitative analysis of the time delays occurring at each decision point could be utilized in the model. The changing of the route selection probability based on route length and width, exit width and/or travel direction could also be investigated.

# REFERENCES

[AE02] AEA Technology. A Technical Summary of the AEA EGRESS Code, AEAT/NOIL/27812001/002(R), Issue 1, 2002

[AG86] Asano T., Guibas L., Hershberger J. and Imai H., 'Visibility of disjoint pologons', Algorithmica 1, pp 49-63, 1986

[Be77] Best R.L., 'Reconstruction Of A Tragedy: The Beverly Hills Supper Club Fire', Southgate, Kentucky, May 28, 1977.

[Be79] Benedikt M.L., 'To take hold of space: isovists and isovist fields', Environment and Planning B 6(1). pp 47- 65, 1979

[BB07] Berrou J.L., Beecham J., Quaglia P., Kagarlis M.A., and Gerodimos A. 'Calibration and validation of the Legion simulation model using empirical data', Proceedings of the third International Conference on Pedestrian Evacuation Dynamics (PED) 2005, Vienna, 2007.

[BC95] Bulman E. And Clifford P., 'Customer Service. Interchangeable?', Passenger Terminal World, pp18-23,1995.

[BD82] Berlin G. N., Dutt A. and Gupta S. M., 'Modeling emergency evacuation from group homes', Fire Technology. Feb, 18(1), pp 38-48. 1982

[BF98] Boyce K., Fraser-Mitchell J. and Shields J. Survey Analysis and Modelling of Office Evacuation Using the CRISP Model. In T. J. Shields, Proceedings of the 1st International Symposium on Human Behaviour in Fire, pp. 691-702, 1998

[BP02] Bensilum M. and Purser D.A., Gridflow: an object-oriented building evacuation model combining pre-movement and movement behaviours for performance-based design. In 7th International Symposium on Fire Safety Science Worcester, MA: Worcester Polytechnic Institute, 2002

[BR04] Batty M. and Rana S., ‘The automatic definition and generation of axial lines and axial maps’, Environment and Planning B: Planning and Design, volume 31, pp615 - 640, 2004

[Ca00] Cappuccio J., ‘Pathfinder: A Computer-Based Timed Egress Simulation’, Fire Protection Engineering 8, pp 11-12, 2000.

[Ca07] Castle C.J.E., Guidelines for Assessing Pedestrian Evacuation Software Applications, Centre for Advanced Spatial Analysis (University College London): Working Paper 115, London, UK, 2007.

[CD00] Comeau E. and Duval R.F., “ Dance Hall Fire Gothenburg,Sweden, October 28,1998”, Fire Investigations, NFPA, 2000.

[CM97] Custer R. L. P. and Meacham B. J., Introduction to performance-Based Fire Safety Bethesda, MD: Society of Fire Protection Engineers, 1997.

[DK97] DeBerg M., Kreveld V. M., Overmars M. and Schwarzkopf O., Computational Geometry: Algorithms and Applications, ISBN 3-540-61270-X, Springer-Verlag, Berlin, 1997

[DL07] Donegan H. A., Livesey G., McMaster T. B. M. and McAleavy G., J Axiomatic considerations of a rule based mechanism for the determination of a building's egress capability, Artificial Intelligence Review, vol. 27(1), pp5-19, 2007

[DM94] De Floriani L., Marzano P. and Puppo E., Line-of-sight communication on terrain models, International Journal of Geographical Information Systems 8, pp329-342, 1994

[DP94] Donegan H.A., Pollock A.J. and Taylor I.R., ‘Egress Complexity Of A Building’, Proceedings Of Fourth International Symposium On Fire Safety Science, Ed: Kashiwagi,T.L, pp 601-612, ISBN 1-886279-00-4,1994

[DP96] Donegan H.A. and Pollock A.J., ‘A mathematical basic for egress complexity’, Appl.Math.lett., 9(2), pp75-80，1996

[DT98] Donegan H.A. and Taylor I.R., ‘How Complex Is The Egress Capability Of Your Design?’, Human Behaviour In Fire: Proceedings Of The First International Symposium, 31 August-2 September 1998, ISBN 1-85923-103-9, pp601-610,1998.

[DT99] Donegan H.A., Taylor I.R., Christie G. and Livesey G., ‘Illustrating some rule based algorithms of Egress Complexity from simple case studies’, Journal of Applied Fire Science,8(3): pp243-258, 1999

[EJ93] E R Galea, J M P Galparsoro and J Pearce, EXODUS: A simulation Model for the Evacuation of Large Populations from Mass Transport Vehicles and Buildings under Hazardous Conditions. CIB W14, Int. Symp. and Workshop Engineering Fire Safety in the Process of Design. Univ. of Ulster, Part 3 pp 11-25, 1993

[EM94] E R Galea and M. Owen, Predicting the Evacuation Performance of Mass Transport Vehicles, 'Fire Safety on Ships', Developments into the 21st Century, ISBN 0-907206-57-3, IMarE Conference, Vol 106,2, 239 – 247, London 1994

[EM96] E R Galea, M Owen and P J Lawrence, Computer Modelling of Human behaviour in aircraft fire accidents, Toxicology, vol 155, 1-3, pp 63-78, 1996.

[EP99] E R Galea, P J Lawrence, L Filippidis, Extending the Capabilities of the buildingEXODUS Evacuation Model to Cater for Hospital Evacuations, Proceedings of the Third International Conference on Fire Research and Engineering, 4-8 October 1999, Chicago, USA, pp 39-50, 1999.

[ER95] E R Galea, The Role of Evacuation and Fire Modelling in the Development of Safer Aircraft, Proceedings International Conference on Cabin Safety Research, pp 145-160, Atlantic City, Nov. 1995, Report: DOT/FAA/AR-95/120 pub: Office of Aviation Research, Washington, 1995.

[Fa91] Fahy R.F., ‘An Evacuation Model For High Rise Buildings’, Proceedings Of The Third International Symposium On Fire Safety Science, Eds: Cox G. and B.Langford, pp815-823, Elsevier, London, 1991.

[Fr01] Fraser-Mitchell J., Simulated Evacuations of an Airport Terminal Building, Using the CRISP Model. In 2nd International Symposium in Human Behaviour in Fire, pp89-100, Boston, MA, 2001

[Fr03] Fraser-Mitchell, J., ‘CRISP’ Fire Risk Assessment by Simulation, Presentation given at the University of Greenwich, Generic, 2003

[FS08] ‘Fire Statistics, United Kingdom, 2006’ ISBN: 978-1-4098-0046-0, Posted by Department for Communities and Local Government, May 2008, available on *http://www.communities.gov.uk/documents/statistics/pdf/ firestats2006.pdf*

[FS10] ‘Fire Statistics, United Kingdom, 2008’ ISBN: 978-1-4098-2660-6, Posted by Department for Communities and Local Government, November 2010, available on *http://www.communities.gov.uk/documents/statistics/pdf/1780609.pdf*

[Ga71] Garbrecht D., Pedestrian paths through a uniform environment, Town Plann Rev 41, pp71–84, 1971

[Ga73] Garbrecht D., Describing pedestrian and car trips by transition matrices, Traffic Quarterly 27, pp89-109, 1973.

[Go99] Golledge R. (Ed.), Wayfinding behaviour: Cognitive mapping and other Spatial Processes, John Hopkins University press, Baltimore, 1999

[Gu09] Guylène P., Evacuation from a single family house, Proceedings of the 4th International Symposium on Human Behaviour in Fire, pp255-266, ISBN 978-0-9556548-3-1, Robinson College, Cambridge, UK, 13-15 July 2009

[Gu98] Gunnar G. L, Theory and Methodology Models of Wayfiding in Emergency Evacuations , Journal of European Journal of Operational Research 105, pp371-389, 1998.

[GG00] Galea E.R., Gwynne S., and Lawrence P. J., Using Computer Simulation to Predict the Evacuation Performance of Passenger Ships, Paper 00/IM/61, CMS Press, ISBN 189999162X, 2000

[GG98] Gwynne S., Galea E. R., Owen M., Lawrence P. J., and Filippidis L., A Systematic Comparison of Model Predictions Produced by the buildingEXODUS Evacuation Model and the Tsukuba Dataset, Journal of Applied Fire Science, Volume 7 Number 3, pp 235 – 266, 1998.

[GG99] 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.

[GM78] Gobow H.N. and Myers E.W., ‘finding all spanning trees of directed, and undirected graphs’, SIAM J. Computing 7, pp280-287, 1978.

[GM87] Ghosh S. K. and Mount D. M. An output sensitive algorithm for computing visibility graphs, Proc. 28th Symposium on Foundations of Computer Science, pp11-19, 1987

[Ha96] Harrington S. S., TIMTEX: A Hydraluic Flow Model for Emergency Egress, MSci Department of Fire Protection Engineering, University of Maryland, 1996

[HtEn] *http://en.wikipedia.org/wiki/AutoCAD_DXF*

[HtPe] *http://people.hofstra.edu/geotrans/eng/ch1en/meth1en/ch1m3en.html*

[HtFA] *http://faculty.washington.edu/krumme/207/networks.html#networks*

[HtMa]*http://www.math.uchicago.edu/~may/VIGRE/VIGRE2006/PAPERS/ Casarotto.pdf*

[HH83] Hillier B., Hanson J., Peponis J., Hudson J. and Burdett R., Space Syntax: a different urban perspective, The Architects Journal, 30 November, 1983, pp 47-63 London, 1983

[HH84] Hillier B. and Hanson J., The Social Logic of Space (Cambridge University Press,Cambridge, UK), 1984

[HH87] Hiller B., Hanson J. and Peponis J., The syntactic analysis of settlements, Architecture and Behaviour 3(3), pp 217-231, 1987

[HH97] Hoffman N. A. and Henson D. A., Simulating Transient Evacuation and Pedestrian Movement in Stations, In 3rd International Conference on Mass Transit Management Kuala Lumpur, Malaysia, 1997

[HH98] Hoffman N. A. and Henson D. A., Analysis of the Evacuation of a Crush Loaded Train in a Tunnel, In 3rd International Conference on Safety in Road and Rail Tunnels Nice, France,1998

[HM98] Heskestad A. W. and Meland O. J., Determination of Evacuation Times as a Function of Occupant and Building Characteristics and Performance of Evacuation Measures, Proceedings of the 1st International Symposium in Human Behaviour in Fire, pp. 673-680, 1998

[HP93] Hillier B., Penn A., Hanson J., Grajewski T. and Xu J., Natural movement: or configuration and attraction in urban pedestrian movement, Environment and Planning B: Planning and Design 20, pp 29-66, 1993

[HW99] Hunt E. and Waller D., Orientation and Wayfinding: A Review, ONR Tech, Rep. N00014-96-0380, Office of Naval Research, USA. 1999.

[Je06] Jean-Paul R., Claude C., and Brian Slack, THE GEOGRAPHY OF TRANSPORT SYSTEMS, ISBN 0-415-35441-2, 2006

[Ka63] Kansky K., Structure of transportation networks: relationships between network geography and regional characteristics, University of Chicago, Department of Geography, Research Papers 84, 1963

[Ke95] Kendik E., Methods of Design for Means of Egress: Towards a Quantitative Comparison of National Code Requirements, Proceedings of the 1st International Symposium in Fire Safety Science, pp. 497-511, 1995

[Kl85] Klir G.J., Architecture of Systems Problem Solving, Plenum Press, New York, 1985.

[Kr79] Kruger M. J. T., an approach to built-form connectivity at an urban scale: system description and its representation, Environment and Planning B: Planning and Design 6, pp 67- 88, 1979.

[Ku00] Kuipers B., 'The spatial semantic hierarchy', Artificial Intelligence, 119, pp191–233, 2000.

[Ku08] Kuligowski E.D., Computer Evacuation Models for Buildings, SFPE Handbook of Fire Procetion Engineering, $4^{th}$ ed, pp 3-456 to 3-478. National Fire Protection Association: Quincy, 2008.

[KC94] Ketchell N., Cole S.S., Webber D.M., 'The Egress Code For Human Movement And Behaviour In Emergency Evacuation', Engineering For Crowd Safety , pp361-370,1994.

[KF05] Krieg-Bruckner B., Frese U., Luttich K., Mandel C., Mossakowski T., and Ross R.J., Specification of an Ontology for Route Graphs,  C. Freksa et al. (Eds.): Spatial Cognition IV, LNAI 3343, pp. 390–412, 2005.

[KF85] Kisko T.M. and Francis R.L., 'Evacnet+: A Computer Program To Determine Optimal Evacuation Plans, Fire Safety Journal 9, pp211-220, 1985.

[KF98] Kisko T.M., Francis R.L., and Nobel C. R., EVACNET4 User's Guide, Version 10/29/98 University of Florida, 1998

[KH07] Korhonen T., Hostikka S., Heliövaara S., Ehtamo H. and Matikainen K., Integration of an Agent Based Evacuation Simulation and the State-of-the-Art Fire Simulation, Proceedings of the 7th Asia-Oceania Symposium on Fire Science &Technology, 20.22 September, 2007.

[KH09] Korhonen T. and Hostikka S., Fire Dynamics Simulator with Evacuation FDS+Evac: Technical Reference and User.s Guide, ISBN 978-951-38-7180-2, VTT Technical Research Centre of Finland, 2009.
*Available on: http://www.vtt.fi/inf/pdf/workingpapers/2009/W119.pdf*

[KM03] Klupfel H. and Meyer-König T., Characteristics of the PedGo Software for Crowd Movement and Egress Simulation, In 2nd International Conference in Pedestrian and Evacuation Dynamics (PED), pp. 331-340, London, U.K.: University of Greenwich, 2003

[KM04] Kuligowski E.D. and Milke J.A., ‘A Performance-Based Design of a Hotel Building Using Two Egress Models: A Comparison of the Results’, Proceedings of the 3rd International Symposium on Human Behaviour in Fire, Belfast UK, pp399-410, 2004

[KP05] Kuligowski E.D. and 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.

[KR95] Kapoor S. and Ramesh H., ‘Algorithms for enumerating all spanning trees of undirected and weighted graphs’, SIAM J. Computering 24, pp 247-265,1995.

[Le87] Levin B.M., ‘Exitt-A Simulation Model Of Occupant Decsion And Actions In Residential Fires: User’s Guide And Program Description’, NBSr 87-3591, National Bureau Of Standards, Gaithersburg, Md, 1987.

[Le89] Levin B., ‘Exitt, A Simulation Model Of Occupant Decisions And Actions In Residential Fires’, Proceedings Of The Second International Symposium On Fire Safety Science, pp561-570,ISBN 0-891168648, 1989.

[Le79] Lee D.T., Proximity and reachability in the plane, Ph.D. thesis, Dept. of Electrical Engineering, University of Illinois at Urbana-Champain,1979

[Lo93] Lovász L., Random walks on graphs: a survey, Bolyai Society Mathematical Study, Combinatorics, vol. 2, pp.1-46. 1993

[LF00] Lo S.M. and Fang Z., A Spatial-Grid Evacuation Model for Buildings, Journal of Fire Sciences, 18, pp376-394, 2000.

[LF04] Lo S.M., Fang Z., and Zhi G.S., An Evacuation Model: the SGEM package, Fire Safety Journal, vol. 39, pp169-190, 2004

[LH03] L. Sheridan, H.J. Visscher and F.M. Meijer, Buildings regulations on fire safety in Europe, Proceedings of the CIB-CTBUH International Conference on Tall Buildings, pp 8-10, May 2003

[LW79] Lozano-Pérez T., Wesley, M.A., ‘An algorithm for planning collision-free paths among polyhedral obstacles’. Communications of the ACM 22, pp. 560-570, 1979

[Ma03] MacDonald M., STEPS Simulation of Transient Evacuation and Pedestrian Movements User Manual (Unpublished Work), 2003

[ME84] Mai S. W. and Evans D.J., ‘A parallel algorithm for the enumeration of the spanning trees of a graph’ Parallel Computing, Volume 1, Issue: 3-4, pp 275–286, ISSN: 01678191, 1984.

[ME88] Mark H. Overmars and Emo Welzl, New methods for computing visibility graphs, Proceedings of the 4th Annual Symposium on Computational Geometry, pp164-171, 1988.

[ME95] M Owen, E R Galea and P J Lawrence, The EXODUS evacuation model applied to building evacuation scenarios, Proc International conference, Fire Safety By Design, ISBN 0 9527398 3 6, Vol.3, pp.81-90, Compiled by G W Butler, 1995

[ME96] M Owen, E R Galea and P J Lawrence, The EXODUS evacuation model applied to building evacuation scenarios, Journal of Fire Protection Engineering, Vol.8(2), pp.65-86, 1996.

[MK01] Meyer- König T., Klupfel H. and Schreckenberg M., A microscopic model for simulating mustering and evacuation process onboard passenger ships, Proceedings of the International Emergency Management Society Conference, 2001 Available on: http://www.traffgoroad.com/downloads/tiems-paper.pdf

[MM97] M. Raubal, M. Egenhofer, D. Pfoser and N. Tryfona, Structuring Space with Image Schemata: Wayfinding in Airports as a Case Study, Lecture Notes In Computer Science; Vol. 1329 PP: 85 – 102, 1997

[MS71] March L. and Steadman P., ‘the geometry of environment’, RIBA Publications Ltd., London, 1971

[NM02] Nelson H. E. and Mowrer F. W., Emergency Movement, In P.J.Denno and W. D.Walton (Eds.), The SFPE Handbook of Fire Protection Engineering (Third ed., pp. 3-367-3-380), Bethesda, MD: Society of Fire Protection Engineers, 2002

[OG98] Owen M., Galea E.R., Lawrence J.P., and Filippidis L., The Numerical Simulation of Aircraft evacuation and its application to aircraft design and certification, Aeronautical Journal, pp 301- 312, 1998.

[Pa84] Passini R., Wayfinding in Architecture, Van Nostrand Reinhold publications, New York, 1984

[Po94] Poon L.S., EvacSim: A Simulation Model of Occupants with Behavioural Attributes in Emergency Evacuation of High-Rise Buildings, Proceedings of the 4th International Symposium on Fire Safety Science, pp. 681-692, 1994

[PB94] Poon L.S., ‘Evacsim: Simulation Model Of Occupants With Behavioural Attributes In Emergency Evacuation Of High Rise Buildings’, Proceedings Of Fourth International Symposium On Fire Safety Science, pp681-692, 1994.

[PC97] Penn A., Conroy R., Dalton N., Dekker L., Mottram C. and Turner A., ‘Intelligent architecture: new tools for the three dimensional analysis of space and built form’, Proceedings of the 1st International Symposium on Space Syntax University College London, London, pp 30.1-30.19,1997

[PD96] Pollock A.J. and Donegan H.A., Algorithms for the calculation of Egress Complexity, Fire Engineering and Emergency Planning-Research and Applications, R.Barham(Editor), pp39-46,1996

[PW98] Peponis J., Wineman J., Bafna S., Rashid M. and Kim S.H., ‘On the generation of linear representations of spatial configuration’, Environment and Planning B: Planning and Design, vol. 25, pp 559-576, 1998

[Re96] Reisser-Weston E., ‘Simulating Human Behaviour In Emergency Situations’, International Conference Of Escape, Fire And Rescue, In RINA, 19/20 Nov 1996.

[RB01] RP Darken, B Peterson, SPATIAL ORIENTATION, WAYFINDING, AND REPRESENTATION, Handbook of Virtual Environment Technology, Stanney K. M. (Ed.), pp493-51, 2001

[RD05] R Burioni and D Cassi, Random walks on graphs: ideas, techniques and results, Journal of Physics A: Mathematical and General, vol. 38, pp45-78, 2005.

[RK04] Richter K.F. and Klippel A., A Model for Context-Specific Route Directions, In: Freksa C., Knauff M., Krieg-Brückner B.(Eds.), Spatial Cognition IV. Reasoning, Action, and Interaction: International Conference Spatial Cognition, pp. 58-78. Springer, Berlin, 2004

[RT75] Read R.C. and Tarjan R.E., 'Bounds on backtrack algorithms for listing cycles, paths, and spanning trees', Network 5, pp237-252, 1975.

[Sc01] Schneider V., Application of the Individual-Based Evacuation Model ASERI in Designing Safety Concepts, In 2nd International Symposium on Human Behaviour in Fire, pp41-51, Boston, MA, 2001

[Sh48] Shannon C.E., the mathematical theory of communication, the bell technical journal, vol. 27, pp379-423,623-658, 1948

[St73] Steadman P., 'Graph theoretic representation of architectural arrangement', Architectural Research and Teching 2, pp161-172, 1973

[SE08] Steven M.V. Gwynne and Eric R. Rosenbaum, Employing the Hydraulic Model in Assessing Emergency Movement, SFPE Handbook of Fire Procetion Engineering, 4th ed, pp 3-373 to 3-396. National Fire Protection Association: Quincy, 2008

[SG94] Shestopal V.O. and Grubits S.J., ‘Evacuation Model For Merging Traffic Flows In Multi-Room And Multi-Storey Buildings’, Proceedings Of The Fourth International Symposium On Fire Safety Science, pp625-632, 1994.

[SH06] S. M. Lo, H. C. Huang and P. Wang, “A Game Theory Based Exit. Selection Model for Evacuation,” Fire Safety Journal, vol. 41, pp. 364-369, 2006.

[SK01] Schneider V. and Konnecke R., Simulating Evacuation Processes with ASERI, 1st International Conference on Pedestrian Evacuation Dynamics (PED), M. Schreckenberg and SD Sharma (Eds.), pp303 -314, 2001.

[SS84] Sharir M., Schorr A., ‘On shortest paths in polyhedral spaces’, Proceedings of the sixteenth annual ACM symposium on Theory of computing, pp 144-153, 1984

[ST05] Steyvers M., and Tenenbaum J., The Large Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth, Cognitive Science, 29(1), pp41-78, 2005

[ST97] Shioura A., Tamura A. and Uno T., ‘An optimal algorithm for scanning all spanning trees of undirected graphs’, SIAM J. on Computing, Vol 26 , pp678-692, 1997.

[Ta67] Tandy C.R.V., ‘The isovist method of landscape survey’, in Symposium: Methods of Landscape Analysis, Murray H.C.(Ed.), Landscape Research Group, London, pp9–10,1967,

[Ta76a] Tabor P., ‘analysing communication patterns’, In the architecture of form, Edited by L. March. Cambridge University Press, 1976

[Ta76b] Tabor P., ‘Analysing route patterns’, In The architecture of form, Edited by L. March. Cambridge University Press.1976

[Ta91] Tanaka T., ‘Study For Performance Design Means Of Escape In Fire’, Proceedings Of The Third International Symposium On Fire Safety Science, Eds: Cox G. and Langford B., pp729-738, 1991.

[To93] Tomomi M., ‘An Algorithm for Finding All the Spanning Trees in Undirected Graphs’, Technical Report METR 93-08, Dept. of Mathematical Engineering and Information Physics, University of Tokyo, Tokyo, 1993.

[TR96] Taylor I.R., ‘A Revised Interface For Evacnet+’, Proceedings Of The Seventh International Fire Science And Engineering Conference, Interflam '96, pp1010-1017, 1996

[Tu01] Turner A., ‘Depthmap: a program to perform visibility graph analysis’, Proceedings of the 3rd International Symposium on Space Syntax, Georgia Institute of Technology, Atlanta, Georgia, 2001,

[Tu04] Turner A., Depthmap 4:A Researcher's Handbook, June 2004
Available on: *http://www.vr.ucl.ac.uk/depthmap/depthmap4r1.pdf*

[TD01] Turner A., Doxa M., O'Sullivan D. and Penn A., From isovists to visibility graphs: a methodology for the analysis of architectural space, Environment and Planning B: Planning and Design 28(1)**,** pp103–121, 2001

[TH94] Tanaka T., Hagiwara I., ‘International Comparison Of Fire Safety Provisions For Means Of Escape’, Proceedings Of Fourth International Symposium On Fire Safety Science, Kashiwagi, T.L.( Ed), pp633-644, 1994.

[TM94] Thompson P. A. and Marchant E. W., Simulex; Developing New Computer Modelling Techniques for Evaluation, Proceedings of the4th International Symposium on Fire Safety Science, pp613-624, 1994

[TM95] Thompson P. A. and Marchant E. W., A Computer Model for the Evacuation of Large Building Populations, Fire Safety Journal 24, pp131-148, 1995

[TP05] Turner A., Penn A. and Hillier B., An algorithmic definition of the axial map, Environment and Planning B: Planning and Design **32**(3), pp425–444, 2005

[TW96] Thompson P. A., Wu J., and Marchant E. W., Modelling Evacuation in Multi-storey Buildings with Simulex, Fire Engineering, 56, pp7-11,1996

[VL09] Veeraswamy A., Lawrence P. and Galea E.R., 'Implementation of Cognitive Mapping, Spatial Representation and Wayfinding Behaviours of People within Evacuation Modelling Tools', Proceedings of the 4th International Symposium on Human Behaviour in Fire, ISBN 978-0-9556548-3-1, pp. 501-512, Robinson College, Cambridge, UK, 13-15 July 2009

[VM02] Vito Latora and Massimo Marchiori , The Architecture of Complex Systems, arXiv:cond-mat/0205649,2002.

*http://www.w3.org/People/Massimo/papers/2002/architecture_book_02.pdf*

[VS80] Ventre T.F., Stahl F.I., Turner G.E., NBSir81-23, 61, National Bureau Of Standards, Gaithersburg, 1980.

[We81] Weisman J., Evaluating architectural legibility: way-finding in the built environment, Environment and Behaviour 13, pp189-204, 1981

[We85] Welzl E., Constructing the visibility graph for n line segments in O ($n^2$) time, Inform. Proc. Let. 20, pp167-171, 1985

[Wi47] Wiener H., ‘Structural determination of paraffin boiling points’, Journal of the American Chemistry Society 69, pp 17-20, 1947

[Wi05] Willis K., Mind the Gap: Mobile Applications and Wayfinding. In Workshop for User Experience Design for Pervasive Computing, Pervasive 2005, Munich, Germany, 2005

[WB00] Werner S., Krieg-Bruckner B. and Herrmann T., Modelling navigational knowledge by route graphs, Spatial Cognition II, C. Freksa, C. Habel and K.F.Wender(Eds.), pp295–317, Lecture Notes in Artificial Intelligence, Springer, 2000

[WS98] Watts D. J. and Strogatz S.H., Collective dynamics of small-world networks, Nature vol. 393, pp 440-442, 1998,

# APPENDIX A

## The Forensic Investigation of Fire Incidents using Computational Fire Engineering Techniques and Experimental Data

**Jiang, H.[1], Gwynne, S.[1] Galea, E.[1], Lawrence P.[1], Jia, F[1]. and Ingason, H[2].**
**Fire Safety Engineering Group**
**University of Greenwich,**
**London SE10 9LS, UK**
http://fseg.gre.ac.uk/

### Abstract

This paper examines the application of evacuation modelling tools to forensically analyse a fire scenario ***similar*** to the tragic Gothenburg fire incident of 1998. It is not claimed that the analysis accurately reproduces the Gothenburg incident, as a key component required for such a forensic analysis, i.e. the evolution of the fire is not adequately represented within the evacuation model. However, the model predictions bare a striking resemblance to the events that took place during the actual incident. The model predictions correctly show that the evacuees experienced severe congestion during their attempted evacuation. While over predicting the number of fatalities, the model successfully predicted the fatality order of magnitude. Furthermore, the predicted location of the fatalities matched that found in the actual incident. In addition, the number of injuries sustained due to the interaction of the evacuating population with the deteriorating environmental conditions that was predicted in this scenario matched those produced during the actual incident. The analysis provides insight into the tragic event and an understanding of why so many people died at the Gothenburg incident. Clearly, evacuation and fire simulation models have an important role to play in fire investigation.

### 1. Introduction

The ability of Computational Fire Engineering (CFE) tools such as fire and evacuation models to simulate realistic incidents is important for engineering design applications as it provides a further means to validate these tools. However, a growing area of interest is the use of CFE tools for forensic fire analysis and incident investigation. In this paper, fire data - generated from small-scale experiments and fire simulation - reflecting an actual incident is used in conjunction with the buildingEXODUS evacuation model in an attempt to forensically analyse and better understand the outcome of an actual incident. Given that the outcome of the incident is known, comparisons can be made indicating which factors were likely to exert a strong influence on the outcome of the incident [1-21]. This form of analysis was originally outlined in a previous work [22]. However this previous analysis was based upon the Beverly Hills Supper Club incident where less information was available describing the deteriorating environmental conditions, limiting the analysis to a detailed qualitative investigation; in the case examined here, a larger set of quantitative information is available, particularly in relation to the environmental conditions.

[1] Fire Safety Engineering Group, University of Greenwich, UK.
[2] Swedish National Testing and Research Institute, Boras, Sweden.

APPENDIX A: *THE FORENSIC INVESTIGATION OF FIRE INCIDENTS USING COMPUTATIONAL FIRE ENGINEERING TECHNIQUES AND EXPERIMENTAL DATA*

When attempting to simulate an actual incident, great care must be taken to ensure that the initial and boundary conditions are represented as accurately as possible. However, the process involved in incorporating fire data is complex and subject to many assumptions and potential errors. In most cases it is *extremely* difficult to accurately represent these controlling conditions as they unfolded during the actual incident. The precise natures of these conditions are simply not known to the extent that is necessary to allow an accurate computer based reproduction of the actual incident. Thus when using models for forensic analysis it is essential that simulation results are viewed as being *indicative* rather than absolute. In many cases, computer simulation can only be used to rule out the occurrence of events rather than definitively confirming their existence; however, this in itself can still be of great value to the engineer.

The case selected for analysis in this paper is the Gothenburg Disco incident of 1998 [1]. Information concerning this incident was gathered from several sources, primarily the fire investigation incident report from National Fire Protection Association (NFPA) [1], the Swedish National Testing and Research Institute (SP) experimental report [2] and a number of relevant reports by Fire Alarm Central and other fire safety departments, as well as several website articles on the subject [3-12]. Fire hazard data used in the analysis was based on the 1/4 scale fire experiment conducted by SP as part of their analysis of the incident. This data could not be used directly in the evacuation analysis due to the reduced scale of the experiments and because many of the fire specific data required by the evacuation analysis was not collected. In reality, the fire hazard data eventually utilized by the buildingEXODUS forensic analysis may not closely replicate the original incident due to numerous approximations that needed to be performed to convert the data into a useful format. It should be emphasised that given the process involved in manipulating the fire hazard data that it is only possible for us to assert that we are modelling a *Gothenburg type* incident, rather than claiming to represent the actual event. However, this does allow us to identify the type issues associated with this incident and more generally in attempting to perform forensic analysis. This enables the model to be used to examine the hypotheses suggested in the NFPA incident report as well examining other aspects of the incident.

## 2. THE GOTHENBURG INCIDENT

### 2.1. THE INCIDENT

On the evening of October 29 1998, a fire developed in the premises of the Macedonian Association in Gothenburg Sweden. On the evening of the fire, some estimates of the number of people in the disco reached up to 400 people [1]; the building was approved by the local authorities for an occupancy of only 150 people. Witnesses confirm that the dance hall was very crowded, with some witnesses reporting that people were standing shoulder-to-shoulder and were unable to dance [1].

The fire started in a stairwell adjacent to the main hall. The disco was on the upper floor of an old industrial two-story building, served by two stairwells at each end of the structure (see Figure 1). There were two internal exits, one located at each end of the hall, each equipped with a single door that swung outward in the direction of travel and led to stairways (see Figure 1). The main stairway on the northwest end discharged directly to the exterior. The other stairway on the southeast end discharged into a corridor on the ground floor, which then led to the same external exit.

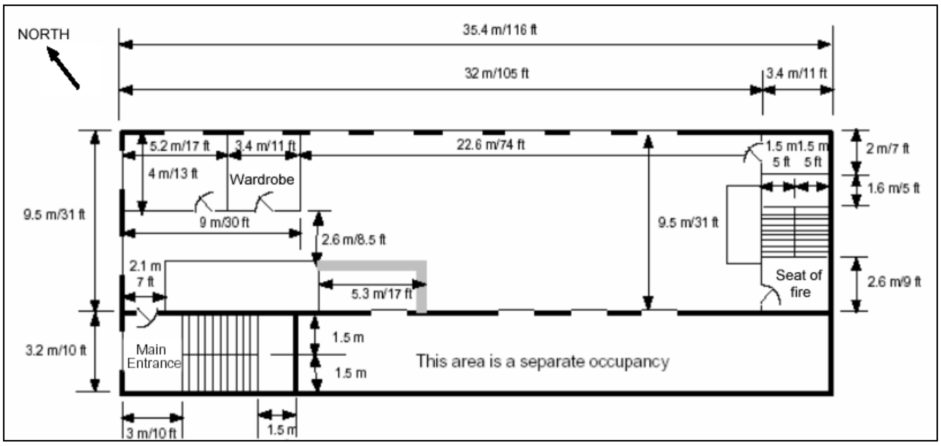
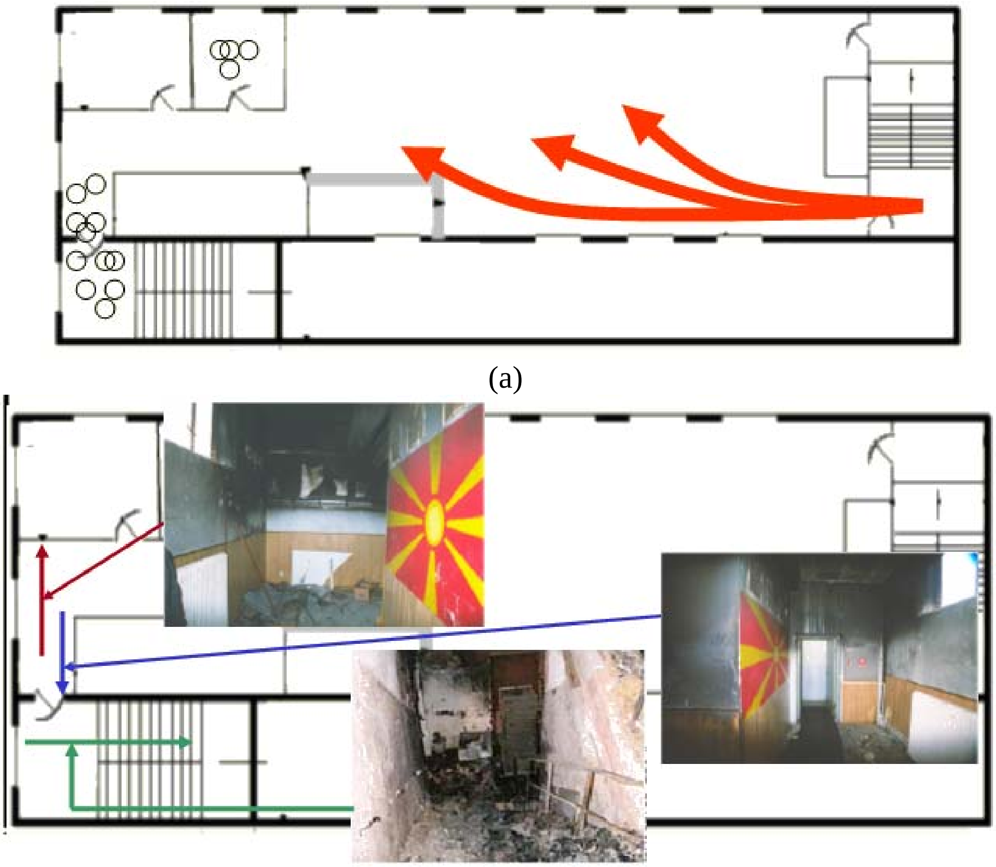


**Figure 1: Floor plan of Gothenburg Disco building in which the incident took place. Redrawn from [1].**

The stairwell located in the south-eastern end of the building was used to store approximately 40 chairs on the middle landing. The fire originated in this area amongst the chairs. Shortly before midnight, some time after the fire started, the door leading to the southeast stairwell was opened, allowing smoke from the fire to enter the main area of the building. It is unknown if the door was closed again after the fire was detected. This stairway effectively became impassable and was not available during the evacuation. The only viable means of egress from the disco was through the north-western exit, which was soon overloaded by the number of people simultaneously attempting to evacuate. The width of this entrance was only 82cm. Given the limited means of exit, a few people managed to escape through the windows. From some of the windows it was possible to jump down onto the roof of a lower building, however these windows were the first ones to be reached by the flames.

During this incident there were 63 fatalities of which 43 were found ‘piled’ around the internal exit leading to the main stairwell (see Figure 2(a)). Fire fighters reported that their access to the hall was blocked by a wall of bodies inside the doorway that reached the top of the doorjamb [1]. Other victims were found in a side room known as the ‘Wardrobe’, off of the main hall, where they had apparently sought refuge (see Figure 2(a)). In addition 180 people were injured. As can be seen from (see Figure 2(b)) the fire progressed into the main hall, with significant smoke and fire damage occurring throughout.

The NFPA concluded that the level of overcrowding, the lack of a fire alarm system and the ignition of combustible storage material in the stairwell contributed to the large loss of life in this incident [1].



(a)

(b)

**Figure 2: (a) Progression of the fire and the location of the fatalities. (b)Photographic evidence acquired from the incident. Redrawn from [1].**

## 2.2. THE BUILDING

The building had originally been constructed in the 1930s. However, the area where the fire occurred had been converted into a hall in 1990. In 1990 a series of violations had been noted by the Gothenburg Fire Brigade. These violations were corrected, and the building was allowed to re-open. The hall where the party was being held measured 32 metres by 9.5metres. There were two exits located at each end of the hall. Each exit was equipped with a door that had an opening approximately 820 mm wide (see Figure 1).

The main stairway on the north-western end of the structure discharged directly to the exterior. The other stairway on the south-eastern end of the structure discharged into a corridor on the first floor before reaching the exterior. There were several rooms on the northwest end of the hall. A stage was located on the southeast end where a disc jockey had set up his equipment. The building was constructed of a combination of concrete and masonry block. The ceiling was comprised of suspended acoustical tile. The composition of the interior finish in the hall itself varied. It was reported that there were decorations hung in the hall for the party and that there were a number of flags on the walls. There were no automatic fire sprinkler or fire alarm systems in the building. There were lighted exit signs at each end of the hall.

There were a series of windows on the northeast wall (see Figure 3). These windows measured 1.8 m by 0.8 m and the bottom of the windows was 2.2 m above the floor. There were a total of eight windows, six of which were in the hall itself. The other two were in the ancillary rooms off of the hall. On the south-western wall were five similar windows. These windows, however, were equipped with security bars to prevent intrusion.

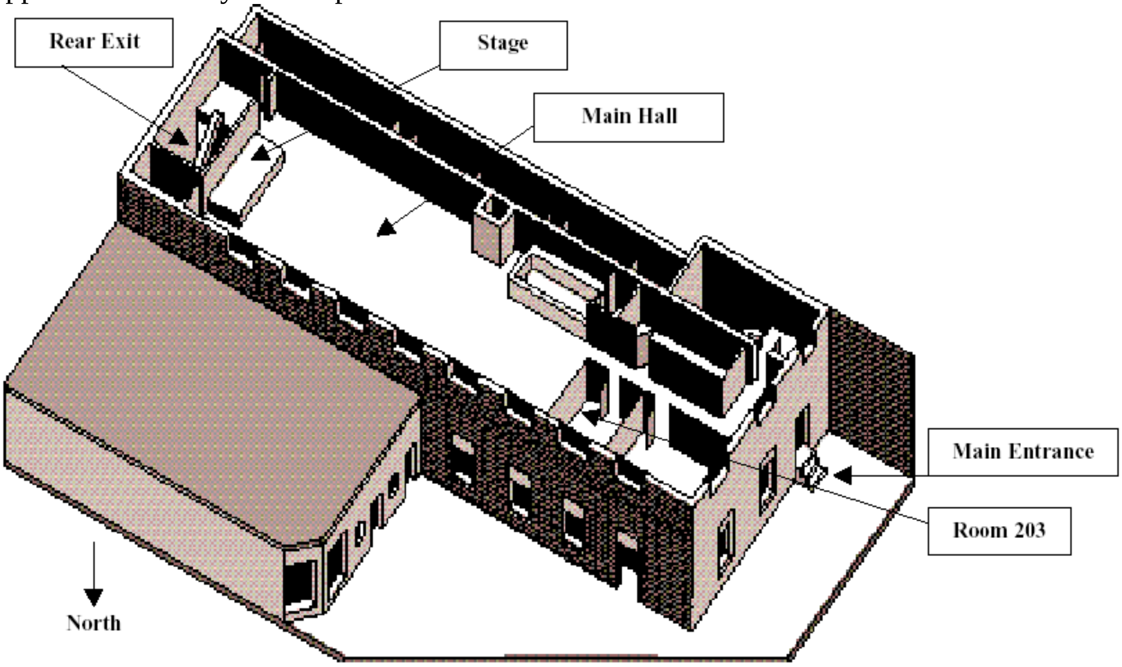


**Figure 3: Gothenburg Disco building. Redrawn from [1].**

## 3. THE INTRODUCTION OF THE SP EXPERIMENTS

The fire hazard data used in this paper is based on the small-scale fire experiments (1/4) conducted by the Swedish National Testing and Research Institute (SP) as part of the official investigation [2].

A total of 14 experiments were carried out. These tests demonstrated how quickly the smoke spread into the premises when the door to the stairwell was opened. SP also demonstrated that it was the combustible floor that constituted the primary fire load in the discotheque. The fire investigation carried out by SP concluded that the fire must have started on the intermediate landing of the escape route approximately 10 to 20 minutes before the door to the discotheque was opened (see Section 2).

As in the original structure, the model structure used during the experiments was made of brick and concrete. However, additional lighter materials (non-combustible calcium silicate boards) were used in the model in order to meet the thermo-physical scaling requirements. The fire was modelled using heaped wooden cribs located in the middle landing of the model structure. The various openings, and the position of the fire, were changed between experiments in order to gain a greater understanding of the evolving conditions. Numerous measurements were made throughout the model, using a variety of devices (thermocouples, etc.) producing data relating to the temperatures, gas concentrations, mass loss rates etc. (see Figure 4).

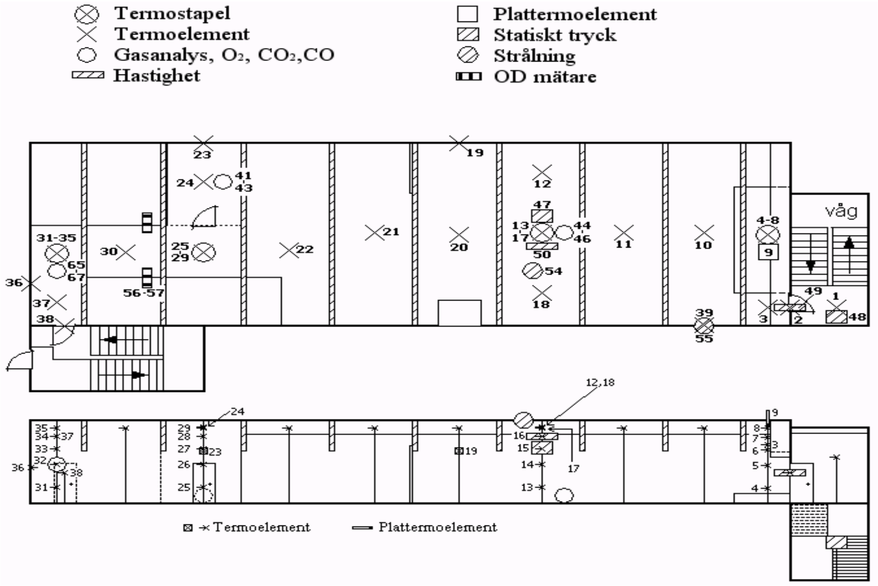


**Figure 4: ¼ scale floor plan of Gothenburg Disco building. Redrawn from [1].**

The experimental procedure enabled a great deal of information to be gathered describing the manner in which the conditions developed according to the location, the time period and (in some instances) the height within the room that was affected. The type of data produced relating to the environmental deterioration within the Main hall is presented in Figure 5.

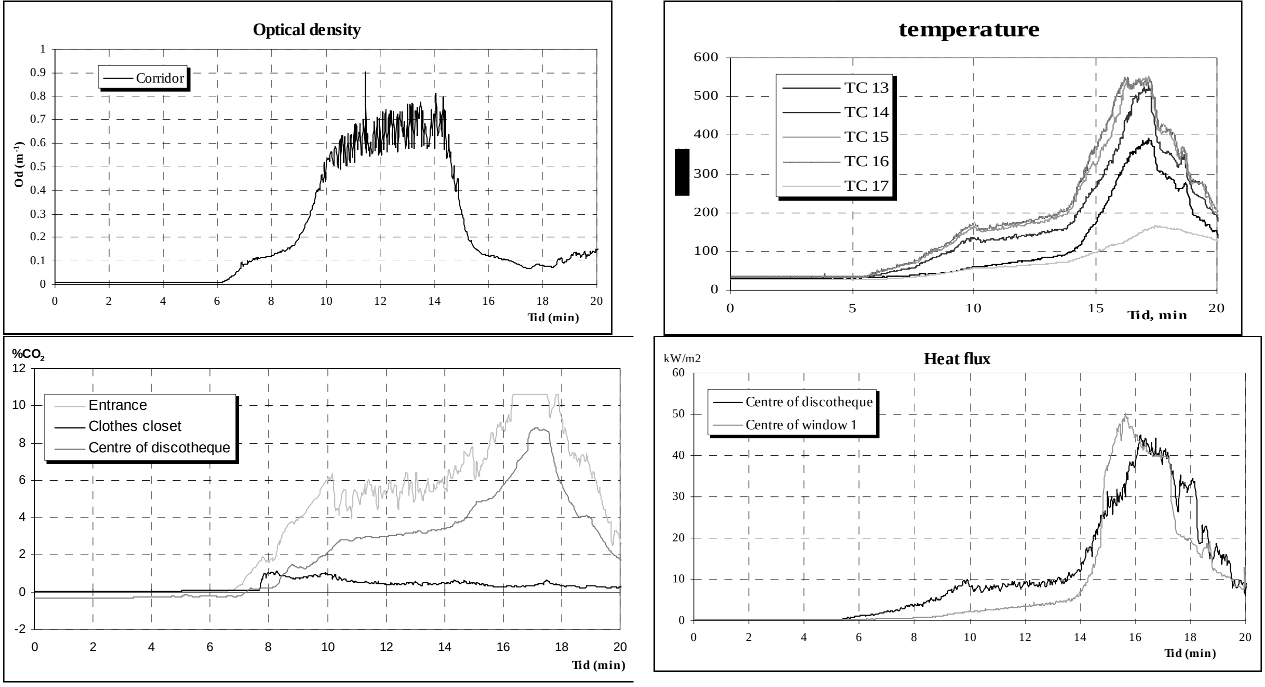


**Figure 5: Experimental results for the optical density readings, the temperature, the CO2 and the radiative fluxes produced.**

It should be noted that these experiments were not performed with the intent of utilizing the data provided in detailed evacuation analyses but to understand the fire dynamics produced during the incident. The data is therefore not ideal for evacuation analysis. An important difference between the experiments and the actual incident was the fire fuel involved. During the experiments, the fire was simply represented using wood cribs, whereas during the actual incident numerous seats were the primary material involved in the stairwell, which would have comprised of a number of different materials. Thus the toxic gas species produced and their concentrations in the experimental fire are unlikely to represent those generated in the actual fire incident. This is an important omission as toxic gases such as CO and HCN were the likely main cause of death in the majority of cases during the original incident. Furthermore, the experimental investigation studied a range of ventilation openings and fire locations so as to derive a better understanding of the possible fire dynamics. In this analysis, the results from only one of the 14 fire scenarios studies are used, deemed to most closely approximate the original conditions. During the experiments, the mass loss rate of the wood cribs was measured as were the temperatures, gas concentrations and optical densities at several locations throughout the structure. However, these measurements related specifically to the ¼ scale fire tests and clearly could not be used directly within the full-scale evacuation analysis. Therefore a procedure was developed in order to modify the data in order for it to better reflect the full scale conditions.

## 4. THE REPRESENTATION OF THE FIRE IN THE EVACUATION ANALYSIS

The evacuation analysis made use of the following fire hazard data: temperature levels, radiative heat flux, toxic gas levels (CO and $CO_2$ only), $O_2$ concentrations and smoke optical densities (converted into an extinction coefficient). The basis of this data was generated from the small-scale fire experiments. However, to make use of this data in the evacuation analysis the data had to be scaled/adapted in order to represent a full-size incident. Several methods were required in order to achieve this.

The toxic gas, smoke concentrations and $O_2$ levels were obtained directly from the small-scale experiments using both temporal and spatial scaling factors, suggested by those performing the original experiments [13]. Once again it must be emphasised that the gas concentration representation is a gross approximation due both to inaccuracies in the original experimental representation (i.e. using the wood crib rather than the actual fuel) and the scaling approach adopted. Temperature and radiative heat flux levels were determined using the CFAST zone model [14] (see Figure 6). CFAST was initially used to simulate the small scale experimental conditions. When a parameter set was determined that accurately reproduced the small-scale experiments (i.e. the initial conditions which led to the results produced during the experiment in question) these were then applied to the full-scale geometry in order to determine appropriate temperature and heat flux levels. Further details concerning the scaling approach can be obtained in another publication [2]. The relatively complex conversion process applied to the small scale experimental results in order to produce a useable set of data for the modelling task is outlined in Figure 6.

The type of data included within the buildingEXODUS model is shown in Figure 7. This data can be directly compared to that presented in Figure 5 to determine where simple linear transformations have been made between the two data-sets or where more sophisticated transformations have been made.

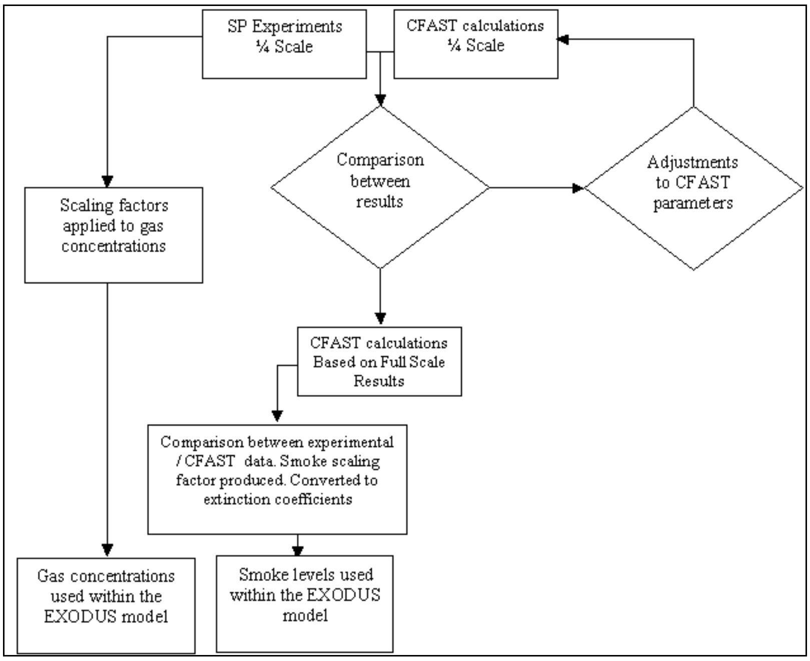

SP Experiments
¼ Scale
CFAST calculations
¼ Scale
Comparison between results
Adjustments to CFAST parameters
Scaling factors applied to gas concentrations
CFAST calculations Based on Full Scale Results
Comparison between experimental / CFAST data. Smoke scaling factor produced. Converted to extinction coefficients
Gas concentrations used within the EXODUS model
Smoke levels used within the EXODUS model

**Figure 6: The representation of the fire in the evacuation analysis**

It must be emphasised that the small scale experimental data used in this analysis is at best a crude approximation to the actual fire and furthermore, the scaling and fire modelling analysis undertaken to make this data appropriate for the full-scale analysis add greater uncertainty to the accuracy of the predicted fire atmosphere.

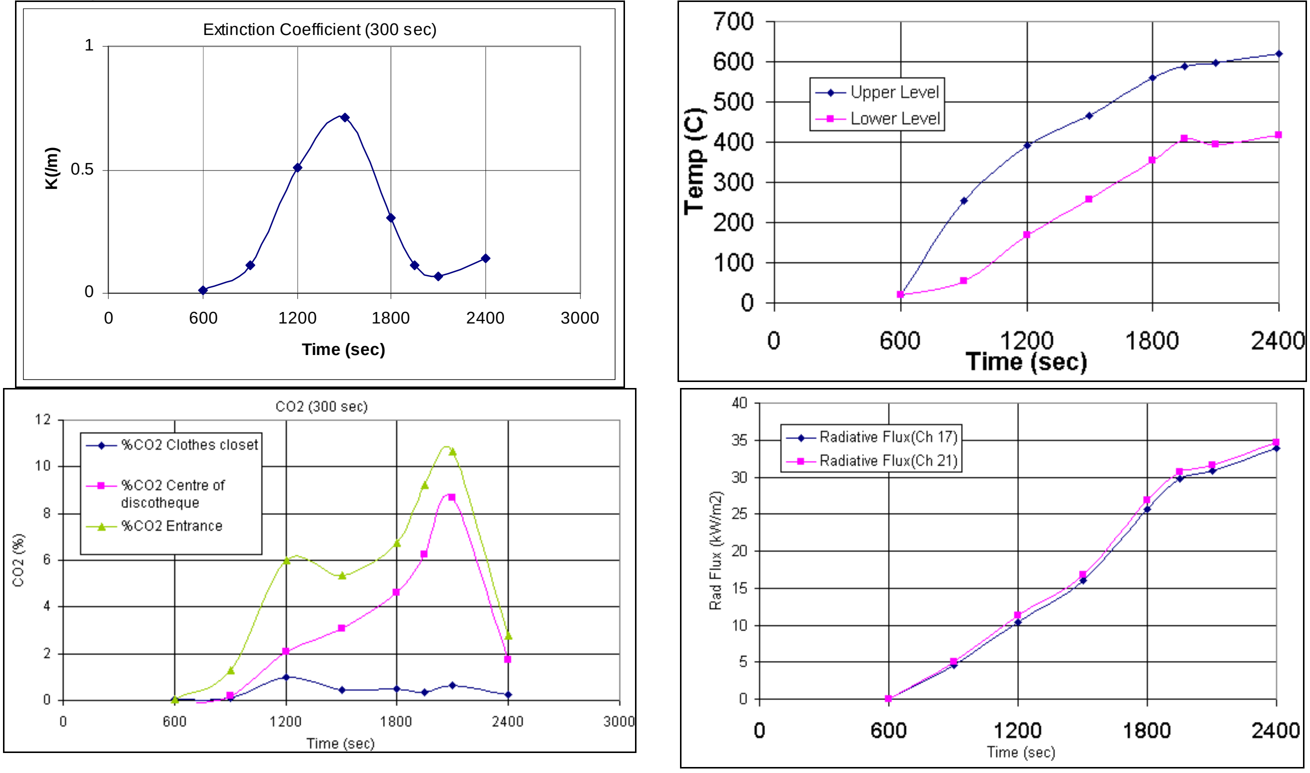


**Figure 7: Examples of the hazard data included within the EXODUS model**

## 5. THE BUILDINGEXODUS EVACUATION MODEL

buildingEXODUS is an evacuation modelling package developed by the Fire Safety Engineering Group at the University of Greenwich and designed to simulate the evacuation of large numbers of people from complex structures. The model comprises five core interacting sub-models: the Occupant, Movement, Behaviour, Toxicity and Hazard sub-models. The software is rule-based, with the progressive motion and behaviour of each individual being determined by a set of heuristics or rules. Architectural plans can be loaded straight in to the simulation suite to represent the structure or the user may avail themselves of a number of interactive design tools. On the basis of an individual's personal attributes, the Behaviour Sub-model determines the occupant's response to the current situation, and passes its decision on to the Movement Sub-model.

The Toxicity submodel determines the physiological impact of the environment upon the occupant. To determine the effect of the fire hazards on occupants, buildingEXODUS uses an FED toxicity model [15,16]. FED models assume that the effects of certain fire hazards are related to the *dose* received rather than the exposure *concentration*. Within buildingEXODUS, as the FED approaches unity the occupant's mobility, agility, and travel rates can be reduced making it more difficult for the affected occupant to escape. The buildingEXODUS toxicity model considers the toxic and physical hazards associated with elevated temperature, thermal radiation, HCN, CO, $CO_2$ and low $O_2$ and estimates the time to incapacitation. When occupants move through a smoke filled environment their travel speed is reduced according to the experimental data of Jin [17,18].

The thermal and toxic environment is determined by the Hazard submodel. This distributes hazards throughout the environment as a function of time and location. buildingEXODUS does not predict these hazards but can accept experimental data or numerical data from other models. The fire hazards are specified at two arbitrary heights that are intended to represent a nominal head height and crawling height. When occupants are considered to be erect, they are exposed to the hazards at head height (irrespective of their actual height) and when the occupants elect to crawl, they are exposed to the hazards at the crawl height. Simulations produced by buildingEXODUS can be replayed in three-dimensional virtual reality using the vrEXODUS software [19].

In this paper several terms are used to describe the results produced by buildingEXODUS. These include: *TET* (Total Evacuation Time, essentially the time for the last person to evacuate), *PET* (Personal Evacuation Time, evacuation time associated with an individual), *CWT* (Cumulative Wait Time, the amount of time spent by an individual in congestion during the evacuation), *Distance* (the distance travelled by a person during an evacuation), *FIH* (an individuals cumulative exposure to radiative and convective heat), *FIN* (an individuals cumulative exposure to narcotic gases), $FIH_c$ (an individuals cumulative exposure to convective heat) and $FIH_r$ (an individuals cumulative exposure to radiative heat).

## 6. Evacuation modelling assumptions

In this section the key assumptions and model parameters used in the evacuation analysis are described. Five simulation scenarios are presented in this paper. For each of the scenarios, 10 repeat simulations were performed using the same population for each of these scenarios in order to produce a distribution of results. The starting point for the evacuation analysis is taken as the time when the door to the fire stair is first opened, allowing the fire hazards to enter the main hall.

The fire is assumed to have developed for 10 minutes before the door was opened. It is assumed in each of the scenarios that once this exit was opened and the fire entered the room, that all of those in the room would have immediately responded to the incident, due to the perceived severity of the conditions. The simulation scenarios have been designed to examine the impact that specific events had upon the outcome of the evacuation.

In all of the scenarios the occupants were initially located in the main room of the geometry and were assumed to attempt to evacuate via the exits available. In the actual incident some individuals were eventually found in the Wardrobe room, whilst others leapt from some of the windows. Although this behaviour could have been represented within the model, in this instance, these behaviours have not been included. The main source of variation between the scenarios examined derives largely from the existence of the environmental decline, the population size, and the exit availability. In Scenarios 1 and 2, no hazard was modelled within the geometry. In Scenarios 3 to 5, the hazard was modelled as accurately as possible within the main hall (given the limitations of the data available). These scenarios then differed according to the population size and the exits available.

The environmental conditions were assumed not to deteriorate during Scenario 1 and *both* internal exits were assumed to be available. The evacuees responded instantaneously and moved towards their *nearest* internal exit. The use of the internal exits was then only dependent upon the evacuees' initial distance from them, according to the basic assumptions of the model. This scenario was therefore based on the assumption that the evacuees were aware of the availability and existence of the two internal exits (both of which led to the same external exit). This simulation was therefore designed to simulate the balanced usage of the internal exits available in non-hazardous conditions (see Table 1).

**Table 1: Scenario conditions**

| Scenario | Number of external exits | Number of internal exits | Environmental decline | Populatio n size |
|---|---|---|---|---|
| 1 | 1 | 2 | No | 400 |
| 2 | 1 | 1 | No | 400 |
| 3 | 1 | 1 | Yes | 400 |
| 4 | 1 | 1 | Yes | 150 |
| 5 | 2 | 2 | Yes | 400 |

The environmental conditions were also assumed not to deteriorate during Scenario 2, but in this case only the north-western internal exit was available, as was the case in the original incident. The evacuees again responded instantaneously and move towards the only available internal exit. This simulation was designed to simulate the usage of the only one internal exit available in non-hazardous conditions (see Table 1).

In Scenario 3, the environmental conditions were assumed to deteriorate, with the impact of the elevated temperatures, radiative fluxes, gas concentrations and smoke being modelled as accurately as possible. Within this scenario only the north-western internal exit was available, with the evacuees responding instantaneously, moving to the external exit through the only staircase available. This scenario therefore approximated the conditions evident during the original incident, including the hazard setting, the exit availability, the structure and the number of the evacuees (see Table 1).

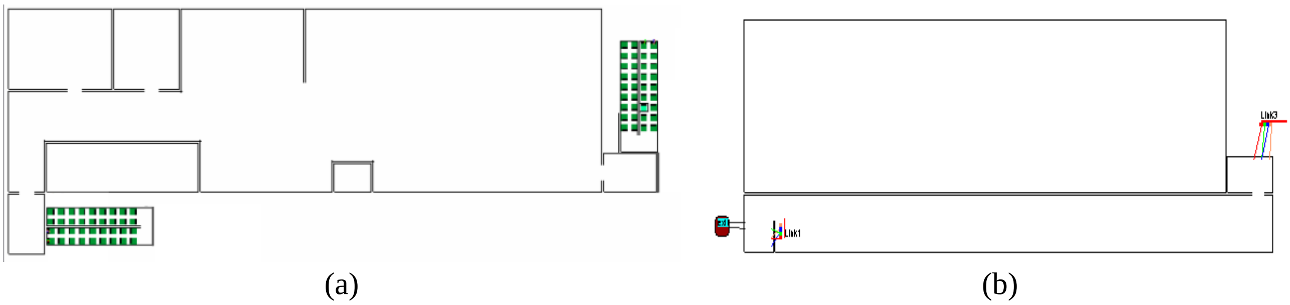


**Figure 8: Layout of the building as used in buildingEXODUS for Scenario 1to 4 showing (a) first floor and (b) ground floor**

In Scenario 4, the population size was reduced to 150 occupants in accordance with the maximum population size for which the premises were licensed. This scenario was designed to investigate the consequences of the incident had the premises been compliant at the onset of the fire. In all other respects this scenario was identical to Scenario 3.

In Scenario 5, the configuration of the structure was modified by adding an additional external exit and a stairwell leading directly to it, whilst including a population of 400 occupants. Other assumptions applied during this scenario were all the same as these used during Scenario 3. By changing the structure of the building, this scenario represents an attempt to examine the impact of having two functioning staircases, whilst catering for a population of 400 occupants. The option to add an additional staircase was adopted rather than making the existing rear staircase available as the hazard data related specifically to its location within this staircase. Although in the original design the evacuees only had access of one external exit, whereas in the new design, there are two external exits, the number of internal exits remains constant. This is important as once the evacuees had progressed beyond the internal exits they were deemed to be clear of the deteriorating conditions. It is therefore the number of internal exits and the subsequent conditions produced on the first floor that is of critical importance when comparing the results produced in the 'hypothetical' case presented in Scenario 5.

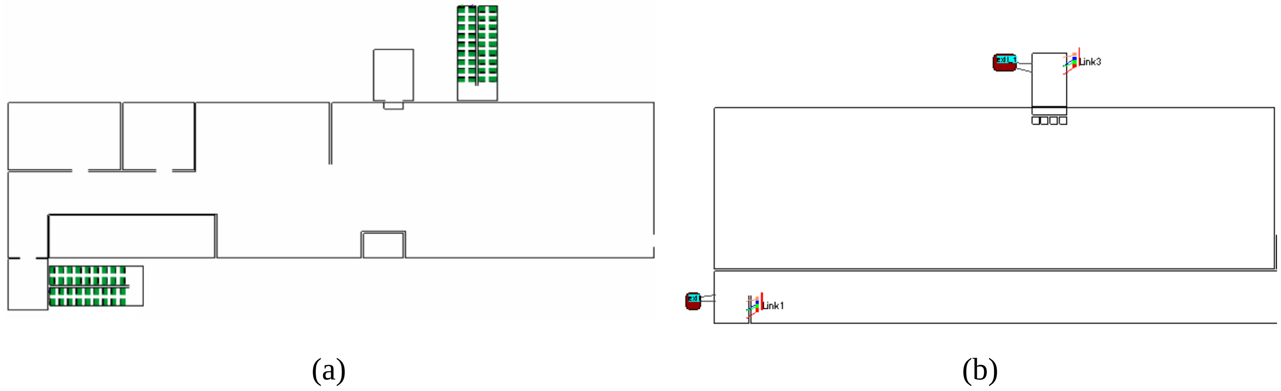


**Figure 9: Layout of the building as used in buildingEXODUS for Scenario 5 showing (a) first floor and (b) ground floor**

In Scenarios 1 to 4, the geometry layout used within the model adhered as closely as possible to the dimensions of the actual structure derived from the floor plans provided (compare Figure 8 with Figure 1). In Scenario 5, the geometry was modified, so that the geometry layout used within buildingEXODUS (see Figure 9) included an additional staircase which led directly to an external new exit. In all scenarios, the external exit was attributed with a unit flow rate of 1.33 occ/m/s. It should be noted here that this is the maximum flow rate that the exit is allowed to attain; it does not indicate that the exit would necessarily operate at this rate.

**Table 2:Basic characteristics of simulated population**

| Attribute | Specification |
|---|---|
| Age | 17- 29 Years |
| Gender | Male-Female |
| Response Time | Instant response |
| Mobility | 1.0 |
| Fast Walk | 1.2 - 1.5 metres/second |
| Walk | 90% * fast walk |
| Crawl | 20% * fast walk |

During Scenarios 1-3 and 5, the geometry was populated with 400 people who were distributed throughout the Main hall on the first floor. The population was generated from a buildingEXODUS population distribution with attributes varying as shown in Table 2. We assumed that the occupants had an instantaneous response time, i.e. they react immediately at the start of the incident; when the door to the fire stair was opened, the occupants of the main hall were immediately aware of the danger to which they were exposed. In Scenario 4, we assumed that only 150 evacuees were initially located in the Main hall.

### 6.1. USING THE HAZARD DATA

The fire hazard data described in Section 4 was imported into the buildingEXODUS model in order to facilitate the simulation process. This data related to the Smoke levels, the temperature, the radiative flux, the CO level, the $CO_2$ level and the $O_2$ level for the main Main hall.

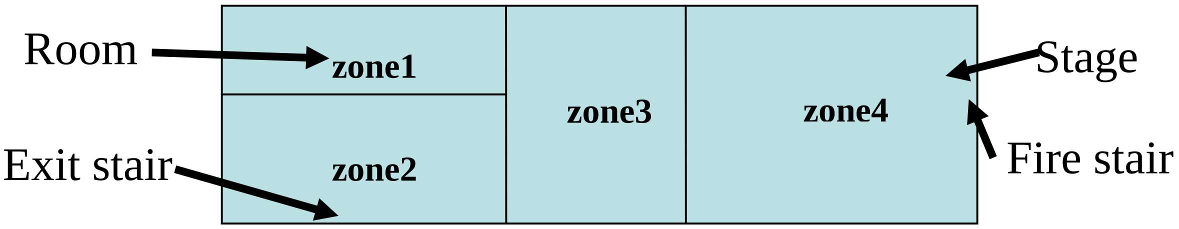


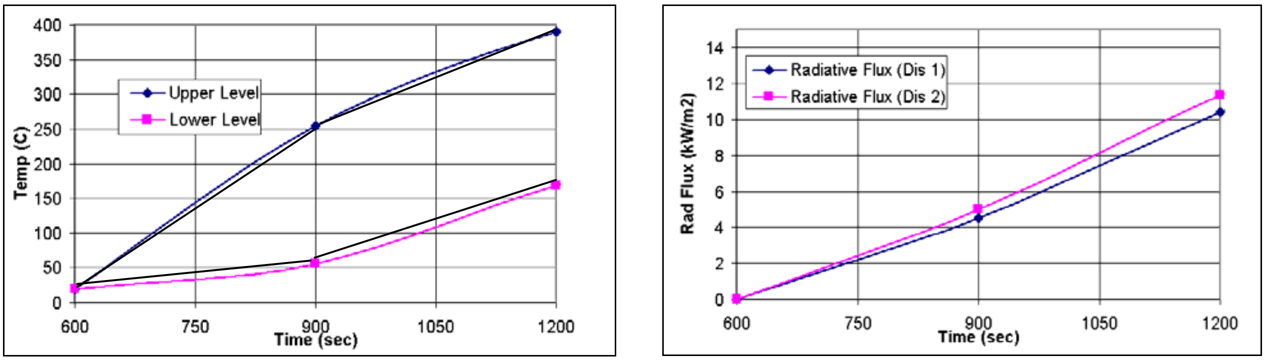
**Figure 10: Schematic of the hazard zones employed within the model.**

Before the simulations were performed, all of the hazards were determined at two heights (head height and knee height). Within the evacuation analysis, the Main hall was treated as a single zone for the temperature and smoke concentration distribution, two zones for the radiative flux distribution, and three zones for the toxic gas distribution (see Figure 10 and Table 3).

**Table 3: Hazard settings within each of the zones**

| | Zone1 | Zone2 | Zone3 | Zone4 |
|---|---|---|---|---|
| Temperature (upper) | *T1* | *T1* | *T1* | *T1* |
| Temperature (lower) | *T2* | *T2* | *T2* | *T2* |
| Smoke | *S1* | *S1* | *S1* | *S1* |
| Radiative Flux | *RF1* | *RF1* | *RF1* | *RF2* |
| CO | *CONC_A1* | *CONC_A2* | *CONC_A3* | *CONC_A3* |
| CO2 | *CONC_B1* | *CONC_B2* | *CONC_B3* | *CONC_B3* |
| O2 | *CONC_C1* | *CONC_C2* | *CONC_C3* | *CONC_C3* |

The fire hazard data is assumed to impinge on the evacuation analysis some 600 seconds after fire initiation (see Figure 11), and linear functions were created between the data points so that this data could be used within EXODUS (see Figure 11 (a)). This event corresponded to the opening of the door to the fire stair. In the FED model relating to heat developed by Purser applied within the EXODUS model, the incapacitation level criteria was used rather than the pain criteria; i.e. evacuees are assumed to succumb once they are deemed to be incapacitated rather than when they first experience pain. In addition, the relevant behavioral options were selected to enable the people to crawl when the environmental conditions where deemed appropriate [19].



(a) (b)

**Figure 11: Calculated (a) temperature and (b) radiative flux levels (at two locations in the Main hall) used in the buildingEXODUS analysis.**

## 7 RESULTS and DISCUSSION

### 7.1 Scenario 1

In Scenario 1, there was no internal exit biasing, with evacuees responding instantaneously in clear environmental conditions. The two internal exits appeared equally attractive to the evacuees within the geometry. Therefore the adoption of exits was *entirely* dependent upon the patron's starting location. This led to an unrealistic representation of exit usage, as in reality the internal exit leading to the rear stairwell (the south-eastern internal exit) was inoperative due to the failing conditions. However this scenario is useful in that it that it enables us to establish the evacuation time under more optimal conditions, where both internal were active and available.

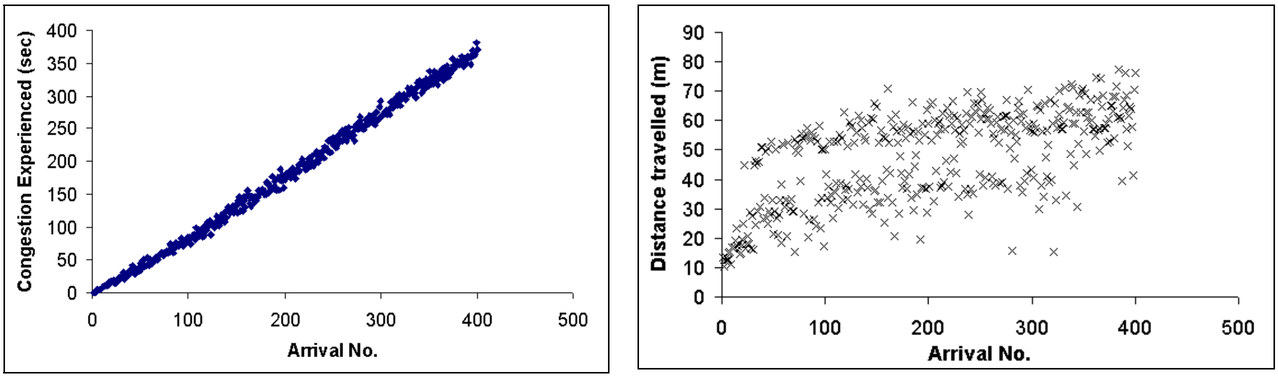
The average evacuation time for this scenario is **6 minutes 46 seconds**, with the individual evacuating on average after **3 minutes 31 seconds**, after spending **2 minutes 49 seconds** in congestion, therefore spending over 80% of their evacuation involved in congestion (see Table 4). It is apparent that there is an approximately linear increase of the congestion experienced in relation to arrival number of the evacuee (see Figure 12 (a)), indicating that the later that the individual arrived, the more congestion they were likely to experience. The evacuees experienced a large degree of congestion around the internal exits and the only external exit due to their simultaneous response and the limited flow rate of the exits. The time spent in congestion compares to a modest average travel distance of 46.5m (see  Figure 12 (b) and Table 4) which would have taken the average evacuees under 35 seconds to complete in clear air and with free movement (assuming an average travel speed of 1.35m/s). From Figure 12 (b), it is also apparent that there is a distribution of distances travelled and that this distribution appears to be independent of the arrival number of the evacuee, indicating that once the evacuation was underway, that the distance travelled was not a significant discriminator in the time of arrival. However the distances travelled do form to clusters, determined by the internal exit adopted by the evacuees and the subsequent egress routes adopted. It is also possible to establish the last times for the evacuees to leave the Main hall, as opposed to the structure as a whole. In Table 4 it is evident that on average the last evacuee to pass through the front (the north-western) internal exit spent 3 minutes 54 seconds leaving the first floor, while the last evacuee to pass through the rear (the south-eastern) internal exit spent 3 minutes 30 seconds leaving the first floor. The population is relatively evenly split between the two internal exits (with on average 185 people using the south-eastern internal exit and 215 people using the north-western internal exit). The differences in the times produced are due largely to the geometry of the approach to the exits, providing more of a hindrance to the evacuees at the south-eastern internal exit than was the case on the approach to the north-western internal exit.

**Table 4: Evacuation results for Scenario 1.**

| **Avg. Evac. time (secs)** | **Avg Cong. Exp (secs)** | **Avg. Dist. Trav.(m)** | **Avg Ind. Time (secs)** | **Last out (Int. 1) (s)** | **Last out (Int.2) (s)** |
|---|---|---|---|---|---|
| 406.9<br>[403.8-410.4] | 169.4<br>[166.6-171.3] | 46.5<br>[45.3-47.6] | 211.3<br>[207.9-214.0] | 233.7<br>[200.7-266.6] | 209.5<br>[193.0-226.0] |

It is apparent that a significant difference exists between the time to clear the first floor and ground floor, with an average discrepancy of over almost three minutes between the two clearance times. This difference is due to the congestion that developed around the single external exit, which was overloaded by evacuees arriving from the two internal exits (i.e. the convergence of two distinct egress routes into one).

This results produced during this scenario demonstrate that the main hall could have been cleared in under four minutes and that the structure as a whole will have been cleared in under seven minutes, assuming the availability of all of the intended egress routes and an instantaneous response.



**(a)** **(b)**

**Figure 12: (a)Congestion experienced by evacuees during Scenario 1 (b) Distances travelled by evacuees during Scenario 1. Arrival numbers rather than evacuation times are used to allow for clearer presentation.**

### 7.2 Scenario 2

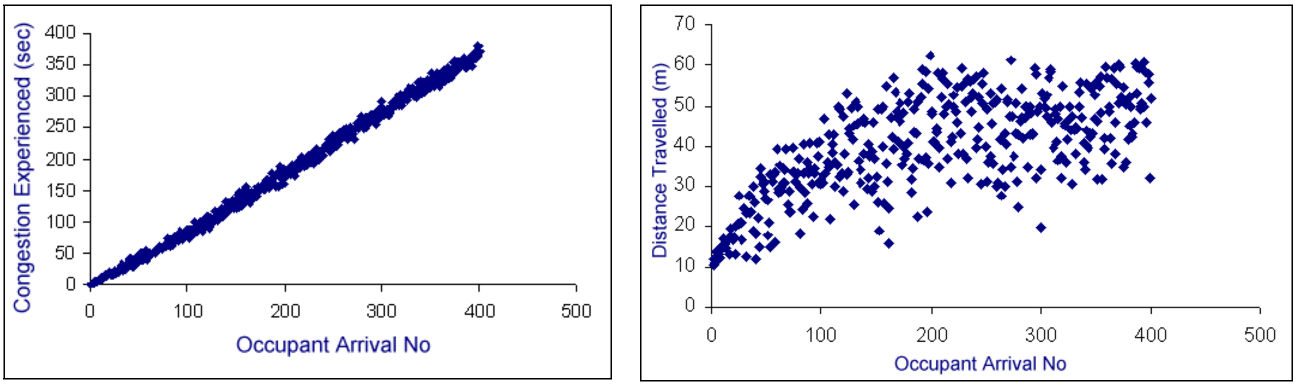
Again in this scenario no hazard was simulated, but this time only one of the internal exits, the front (north-western) internal exit, was deemed available. The evacuees again responded instantaneously and moved towards the only available internal exit.

**Table 5: Average evacuation statistics for the 10 repeat simulations in scenario 2 (range in average values generated across the10 simulations is shown in brackets)**

| Avg. Evac. time (secs) | No. Fatalities | Avg Cong. Exp (secs) | Avg. Dist. Trav.(m) | Avg Ind. Time (secs) | Last out (Int. 1)(s) | Avg FIH | Avg FIN |
|---|---|---|---|---|---|---|---|
| 413.2<br>[409.4-416.6] | 0<br>[0-0] | 179.5<br>[177.9-181.2] | 38.8<br>[38-39.8] | 214.4<br>[212.2-216.8] | 395.2<br>[392.3-397.8] | 0<br>[0-0] | 0<br>[0-0] |

The model predictions are presented in Table 5. The average overall evacuation time for this scenario is **6 minutes 53 seconds**, with an individual taking on average **3 minutes 34 seconds** to evacuate. During this time the individual spent an average of **2 minutes 59 seconds** in congestion: 84% of their overall evacuation time. These figures are similar to those produced during Scenario 1, with a slight increase in the levels of congestion experienced, due to the extreme overloading of the single available internal exit from the room. In this scenario, the average distance travelled was approximately 39 metres, 8 metres less than that evident in Scenario 1, due to the removal of the more distant of the two internal exits. From Figure 13(a), it is apparent that there is also an approximately linear increase in the congestion experienced in relation to the time that an evacuee eventually evacuated, as was also the case in Scenario 1. In contrast, there appears to be a less obvious relationship between the distance travelled and the arrival time, indicating that this factor is less important in determining the arrival time of the evacuees than the level of the congestion experienced.

There is a much smaller time difference between clearing the internal and the external exit (approximately 20 seconds). This is due to the extensive overloading of the internal exit, leading to serious congestion developing. The development of this congestion made further advancement more difficult and caused a more dispersed arrival of the individuals at the external exit, as opposed to the more laminar arrival evident in Scenario 1, leading to congestion building up around the external exit. In this instance, the congestion built up primarily at the internal exit rather than the external exit.



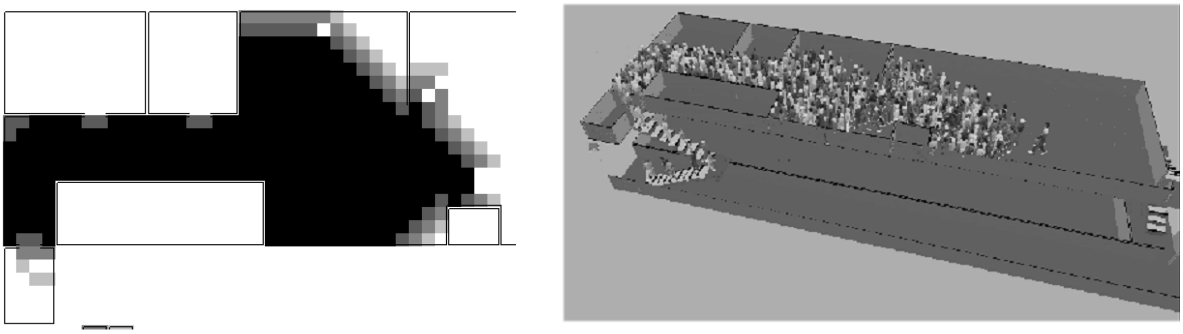
(a) (b)

**Figure 13: (a) Congestion experienced by evacuees during Scenario 2 (b) Distances travelled by evacuees during Scenario 2.**

This scenario represents the evacuation time that might be expected had the fire not entered into the main hall, but had still prevented the use of one of the previously available egress routes.

### 7.3 Scenario 3

The conditions represented within Scenario 3 represent an attempt to approximate those evident during the original incident given the limitations of the data available: the development of the hazard, the geometry of the structure and the estimated number of people present. It should be remembered when examining the results produced that there were only a few means of comparison: the number of fatalities, the recollections of the survivors and the findings of the NFPA [1].



(a) (b)

**Figure 14: Population density diagrams produced by buildingEXODUS one minute into the simulation (a) in population density mode where black areas indicate a population density in excess of 4 people/m$^2$ and (b) via the virtual reality interface.**

The overall model predictions are presented in Table 6. The average total evacuation time for this scenario is **5 minutes 17 seconds**. After approximately five minutes the first fatalities began to occur. The average individual evacuation time was **2 minutes 46 seconds**, **2 minutes 13 seconds** of which was spent in congestion (indicating that on average 80% of an individual's time was spent in congestion; 4% less than that experienced during Scenario 2 due to the reduced number of interacting survivors and the removal of the extreme congestion levels to which some of the fatalities would have been exposed from the average figure produced). These figures suggest that extremely high levels of congestion occurred, very rapidly choking access to the only available exit. This is confirmed by viewing the developing congestion levels using the population density option within buildingEXODUS (see Figure 14(a)) as well as the results produced within the virtual reality interface (see Figure 14(b)). These predictions are in line with the eye witness accounts that suggest severe congestion occurred in the vicinity of the staircase during the evacuation [1-12]. The average effective travel speed achieved by occupants during this evacuation can be calculated from Table 6 as 0.24 m/s.

Conditions and experiences varied considerably amongst the survivors (see Figure 15 and Figure 16). If two particular individuals are selected from one of the simulations we can determine their experiences. Consider the 50th person to evacuate from the building. This person evacuated the building after 65 seconds spending 38.5 seconds (59% of the evacuation time) in congestion, and was essentially unharmed by exposure to the developing fire atmosphere. If we now consider the 200th person to evacuate, we see that this person required just over 215 seconds to evacuate, spending nearly 174 seconds (81%) in congestion. In addition this individual suffered from exposure to the developing fire atmosphere, recording an FIN value of 0.1 and an FIH value of 0.25. This would suggest that this person was likely to have suffered burns to exposed parts of the body.

**Table 6: Average evacuation statistics for the 10 repeat simulations in scenario 3 (range in average values generated across the10 simulations is shown in brackets)**

| Avg. Evac. time (secs) | No. Fatalities | Avg Cong. Exp (secs) | Avg. Dist. Trav.(m) | Avg Ind. Time (secs) | Avg FIH | Avg FIN |
|---|---|---|---|---|---|---|
| 317.3<br>[314.7-319.6] | 96.2<br>[94-99] | 133.0<br>[131.7-134.2] | 37.0<br>[35.9-37.8] | 166.3<br>[165.3-167.5] | 0.2<br>[0.2-0.2] | 0.1<br>[0.0-0.1] |

We also note from Figure 16 that the first 150 people to evacuate the building managed to do so without suffering severe exposure to heat (FIH ~ 0.0) and only light exposure to the narcotic gases (FIN < 0.07); however, on average the survivors were affected by the deteriorating environmental conditions, recording an average FIH value of 0.2 and an FIN value of 0.1.

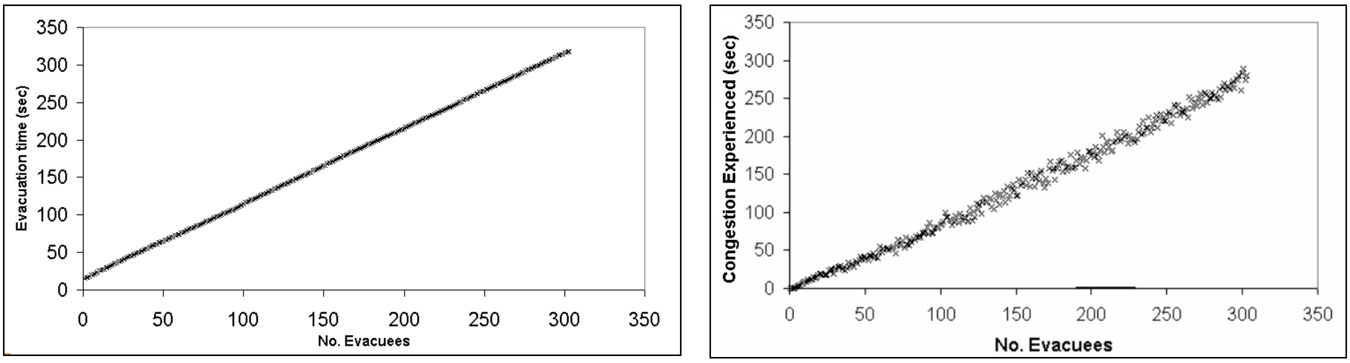
**Table 7: Distribution of FIH values amongst survivors for a single simulation**

| FIH range | Frequency | Cumulative % |
|---|---|---|
| 0.1 < FIH | 173 | 57 |
| 0.1 < FIH < 0.4 | 58 | 76 |
| 0.4 < FIH < 0.6 | 30 | 86 |
| 0.6 < FIH < 0.8 | 26 | 95 |
| 0.8 <FIH < 1.0 | 15 | 100 |

It should be remembered that these values represent averages over 10 simulations. Within any one simulation there may be more extreme values produced amongst the survivors. Presented in Table 7 is the distribution of FIH values amongst the survivors for a particular simulation from

Scenario 3. In this case while there are 98 fatalities (slightly above the recorded average), 41 survivors are likely to have suffered serious burn injuries (i.e. FIH > 0.6).

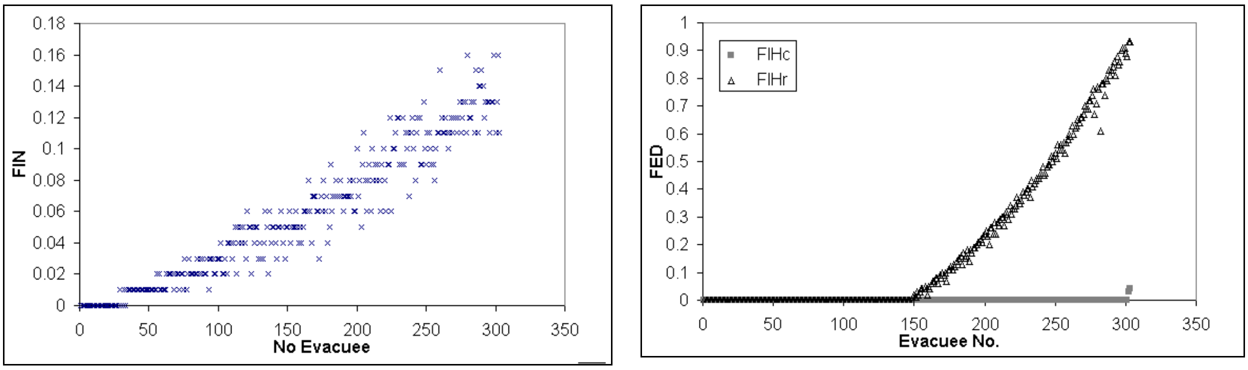
If we assume that individuals with an FED level in excess of 0.1 will have suffered some form of injury [20-21], then on average between 150-160 of the survivors are predicted to have sustaining some form of injury (see Figure 15(a) and (b)) including smoke and toxic gas inhalation and burns; this would not have included injuries relating to crushing or falling.



(a) (b)

**Figure 15: (a) The arrival times of the evacuees (b) the congestion experienced by the evacuees.**

This compares favourably with the 180 injuries that were recorded during the actual incident. Furthermore, on average these simulations suggest that we can expect some 96 fatalities (see Table 6). This compares with 63 fatalities reported in the actual incident. While there is a discrepancy between the predicted and observed fatality levels, they are of the same order of magnitude, indicating the seriousness of both the simulated and actual incidents..



(a) (b)

**Figure 16:( a) The impact of the narcotic gases upon the evacuees. (b) the impact of the temperature and the radiative flux upon the evacuees.**

On an official web site concerning the Gothenburg fire, it was reported by The National Board of Health and Welfare for Sweden that,

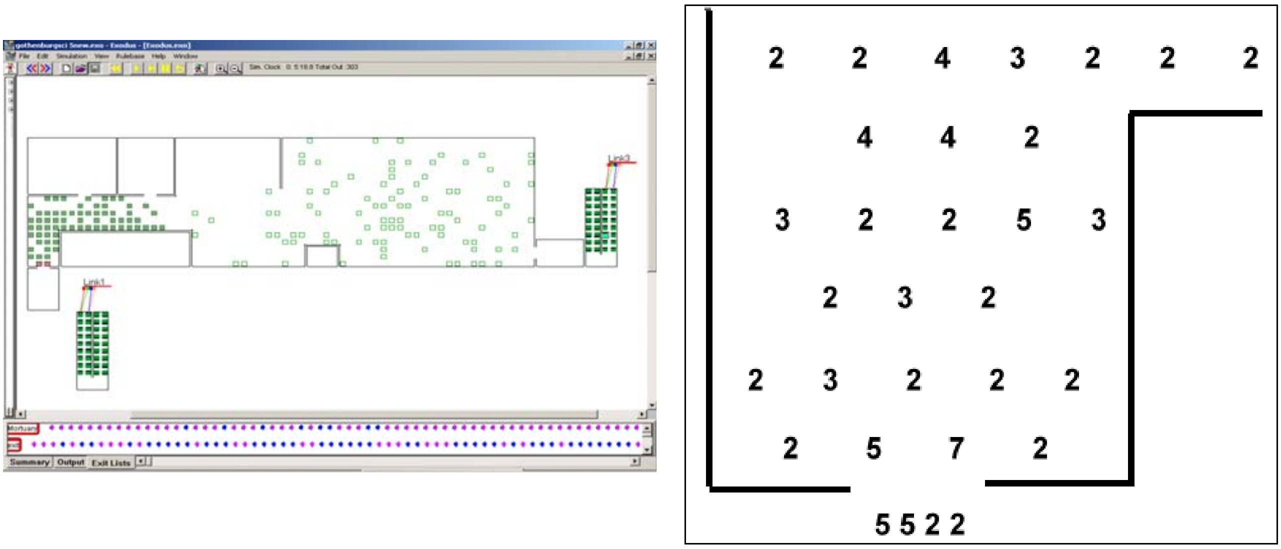
"Most of those who died at the scene had more or less severe burns, but the most common cause of death was carbon monoxide poisoning. Many also had high levels of cyanide in their blood, which in itself could also have been the cause of death. Those whom the smoke divers succeeded in bringing out at an early stage were suffering from mild smoke poisoning and slight injuries. Of those rescued later, most had severe fire gas poisoning, were unconscious and had more or less severe burns." [3].

While the official reports recognized that heat-related injuries did occur, the most important component identified that led to the high level of fatalities was the occupant's exposure to toxic fire gases. The predicted (i.e. simulated) number of fatalities and injury levels was strongly dependent on the accuracy of the imposed fire atmosphere. This in turn was dependent on the experimental data and the nature of the zone modelling and the scaling that was applied. The main cause of death in these simulations was the evacuee's exposure to heat (largely through radiation), whereas the incident reports identified CO and HCN exposure as a significant contributory cause of death. Within the simulated conditions, those evacuees that initially survived, who might otherwise have quickly succumbed to the narcotic gases, due (possibly) to the reduced levels of CO and HCN in comparison to the original incident, may have eventually succumbed to heat exposure. As has already been identified, the fire experiment did not accurately represent the generation and spread of the fire gases. Indeed, HCN was not even represented in the fire experiment. Thus we cannot expect to accurately reproduce the nature of the fire fatalities produced in this incident.

While the toxic gases *contributed* to the predicted fatality levels, the most significant factor driving the number of fatalities was the radiative heat flux generated by the fire (see Figure 16(b)). This problem was exacerbated by the extreme congestion produced during the evacuation. Indeed the average congestion experienced by the fatalities was far in excess of that experienced by the survivors (the average congestion experienced by the fatalities was 281 seconds as compared with 133 seconds for the survivors), indicating the importance of this factor in determining survivability.

Another aspect of the buildingEXODUS simulations concerned the predicted location of the fatalities. In Figure 17 the locations of the fatalities from one of the simulations is presented. These locations were consistent with all of the simulations examined. It is immediately apparent that the fatalities occurred around the (north-western) internal exit at the top of the stairs, as was reported in the actual incident.

Furthermore, the model predicts that in the immediate vicinity of the main stairwell, bodies were pilled up in some places seven high. This is remarkably similar to reports produced by the fire fighters who reported finding bodies 'piled' around the (north-western) internal exit leading to the main stairwell. Fire fighters reported that their access to the hall was blocked by a wall of bodies inside the doorway that reached the top of the doorjamb [1].



(a) (b)

**Figure 17: Fatality locations within the main floor of the building predicted by the buildingEXODUS simulations (a) Starting (open squares) and final locations (filled squares) of the fatalities and (b) the number of bodies pilled up on the floor in the vicinity of the main staircase.**

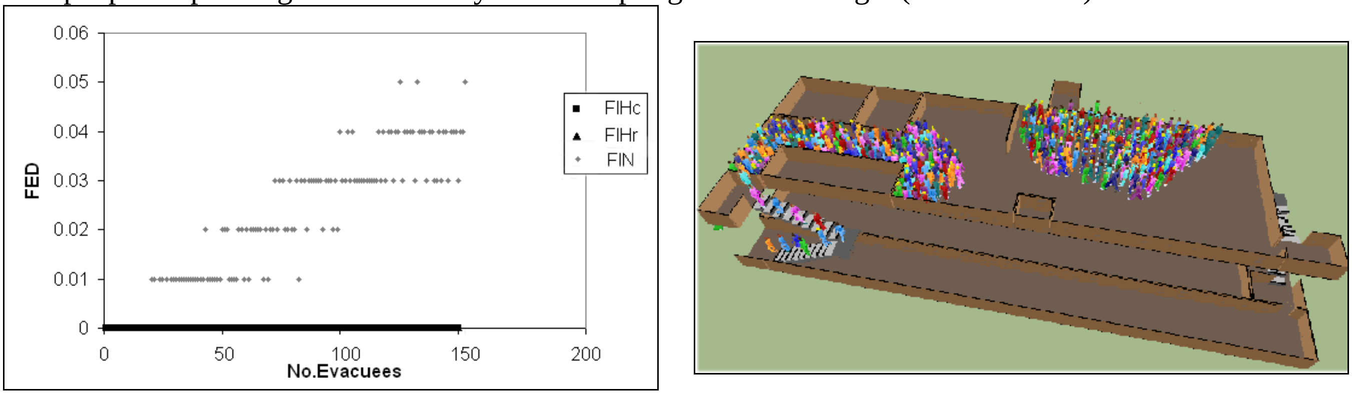
### 7.4 Scenario 4

This scenario was investigated using the same fire conditions as in Scenario 3 but with a population that did not exceed the regulatory compliant population of 150 people; i.e. the population size was 150. Through the examination of this scenario we can determine whether the conditions would have been survivable if the permitted number of residents were located in the main hall. We are also able to examine one of the factors identified as contributing to the tragic outcome in the NFPA report [1]

**Table 8: Average evacuation statistics for the 10 repeat simulations in scenario 4 (range in average values generated across the10 simulations is shown in brackets)**

| Avg. Evac. time (secs) | No. Fatalities | Avg Cong. Exp (secs) | Avg. Dist. Trav.(m) | Avg Ind. Time (secs) | Avg FIH | Avg FIN |
|---|---|---|---|---|---|---|
| 164.3<br>[160.3-174.6] | 0<br>[0-0] | 55.3<br>[54.1-56.2] | 36.7<br>[36-37.5] | 88.8<br>[87.0-89.9] | 0<br>[0-0] | 0.02<br>[0.02-0.02] |

The model predictions are presented in Table 8. The average overall evacuation time for this scenario is **2 minutes 44 seconds**, a 48% reduction in relation to the evacuation time generated during Scenario 3. All of the evacuees completed the evacuation before the environment conditions seriously deteriorated. This is supported in Figure 18(a), where it is apparent that all of people evacuated from the building without suffering severe exposure to heat (FIH ~ 0.0) and only a light exposure to the narcotic gases (FIN < 0.06). The entire population therefore escaped without suffering serious injury. On average an individual evacuated after **1 minute 29 seconds** while spending only 55 seconds in congestion (approximately 60% of their evacuation time, as compared with the 80% of the individual's time spent in congestion during Scenario 3), where the single exit route was overloaded to a greater degree, simply due to the larger number of people responding simultaneously and attempting to use the single (north-western) internal exit.



(a) (b)

**Figure 18:(a) The impact of the narcotic gases, temperature and the radiative flux upon the evacuees, (b) Virtual reality representation of the evacuation in Scenario 5**

It is therefore apparent that the reduction of the population to within the limits for which the structure was licensed would have had a dramatic impact upon the evacuation process, leading to a complete absence of serious injuries or fatalities. In addition, the conditions under which the population evacuated had improved significantly with less congestion experienced as well as less exposure to inhibiting environmental conditions. These findings support those presented in the NFPA incident report [1].

### 7.5 Scenario 5

In Scenario 5 the same fire conditions were again employed. However, in this case, an additional staircase has been made available, located on the north-eastern wall (see Figure 9(a)). It has been assumed that this staircase had identical dimensions to the staircase that was unavailable (located on the east wall), due to presence of the fire and that, as with Scenario 3, the staircases were equally attractive to the evacuees. This scenario was therefore been designed to examine the evacuation of the structure assuming the capacities of the available routes were equivalent to those of the original staircases, whilst a fire still existed in the staircase on the south-eastern wall; to achieve this scenario the additional staircase was provided. Effectively this produces conditions which approached those that would have evolved had an identical fire to that assumed in Scenario 3 (i.e. an approximation to the original incident) evolved outside of the main hall but had not interfered with any of the escape routes. We were therefore able to examine one of the factors identified as contributing to the tragic outcome in the NFPA report [1]: the impact of the storage of combustible material in one of the stairwells.

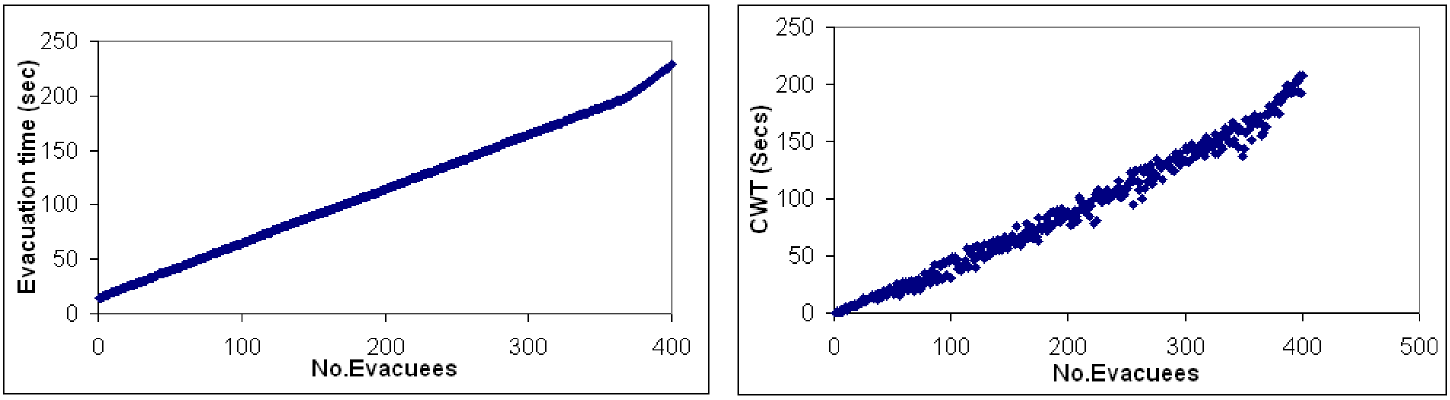
The results for this scenario are presented in Table 9. The average overall evacuation time for this scenario was **3 minutes 42 seconds**. On average an individual evacuated after **1 minute 55 seconds** while spending **1 minute 30 seconds** in congestion (approximately 78% of their evacuation time). Again the levels of congestion evident within the structure were a determinant in the evacuation times produced and are comparable to those evident in Scenario 3 (see Figure 18(b) and Figure 19(b)); however, it should be borne in mind that the absolute time spent in congestion was much less during Scenario 3. The evacuation times produced reflect the balanced use of the exits. The south-eastern internal exit was used by an average 192 people, and cleared in

188.1 seconds, while the north-western internal exit was used on average by 208 people, taking 205.7 seconds to clear. This is comparable to the internal exit usage evident in Scenario 1 where the original staircases were in use, while the evacuation is completed relatively quickly compared to Scenario 3, where only a single internal exit was available, showing a 30% reduction. All of the evacuees had completed their evacuation prior to the deterioration of the environmental conditions to the point where fatalities or serious injuries would have occurred. From Figure 20, we note that the population evacuated the building with only a modest exposure to the narcotic gases (FIN < 0.07) and with less than 40 people affected by elevated temperature levels, recording FIH values between 0.1 and 0.3, indicating the expectation of some injuries.

**Table 9: Average evacuation statistics for the 10 repeat simulations in scenario 5 (range in average values generated across the10 simulations is shown in brackets)**

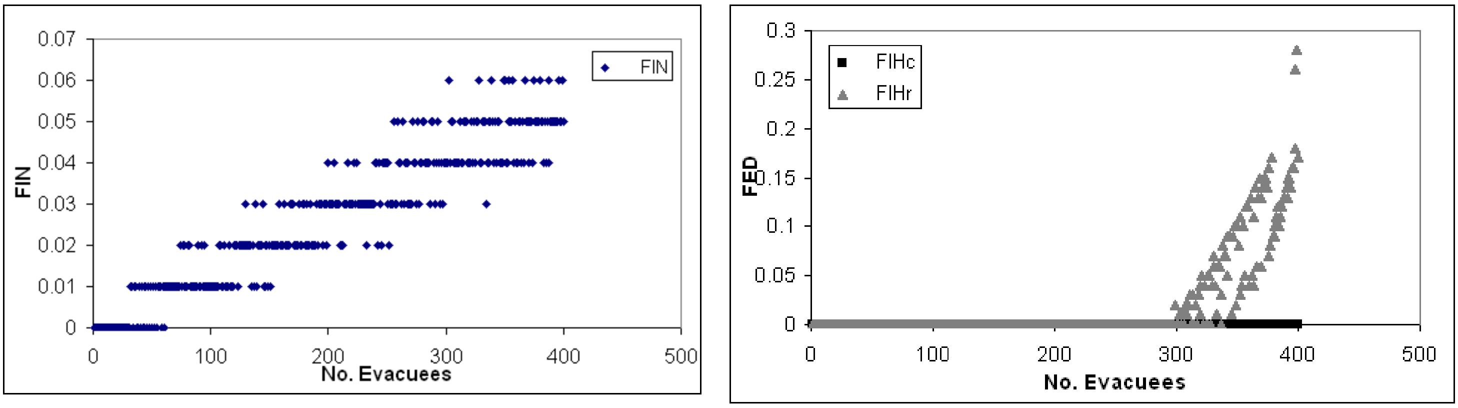
| Avg. Evac. time (secs) | No. Fatalities | Avg Cong. Exp (secs) | Avg. Dist. Trav.(m) | Avg Ind. Time (secs) | Avg FIH | Avg FIN |
|---|---|---|---|---|---|---|
| 221.2<br>[214.9-229.6] | 0<br>[0-0] | 89.9<br>[88.8-91.1] | 25.7<br>[25-26] | 115.2<br>[114.5-116.4] | 0.02<br>[0.02-0.02] | 0.03<br>[0.03-0.03] |

This scenario has therefore demonstrated that it was possible to evacuate the structure even with a population of 400, had the fire not been located in the stairwell blocking off one of the main evacuation routes, again indicating the tragic consequences of the decision to store the additional seating in the stairwell. This conclusion is valid based on the assumption that the additional 'hypothetical' staircase provided *equivalent* egress capacity to that which would have been available had the fire been located elsewhere within the geometry but had allowed access to the two original internal exits. In this instance, as mentioned previously, the external configurational differences evident between the 'hypothetical' and original designs would have not influenced the exposure of the evacuees to the deteriorating environmental conditions, supporting the validity of the assumption mentioned above.



(a) (b)

**Figure 19:(a) Arrival times of the evacuees (b) Congestion experienced by the evacuees.**



(a) (b)

**Figure 20: (a) Impact of the narcotic gases upon the evacuees (b) Impact of the temperature and the radiative flux upon the evacuees.**

The results generated during this scenario are important in that they demonstrate the catastrophic consequences of the location of the fire to the subsequent development of the fire and to the likelihood of the survival of the evacuees. If the fire was similar in development (i.e. entered the room from outside of the main hall after having a significant and uninterrupted period of development) and was broadly comparable in its nature, the results produced during the simulations have demonstrated that the consequences need not have been as devastating as was the case during the original incident, given that sufficient means of egress was available. However, the entrance of the well-developed fire into the main hall in conjunction with the loss of an egress route combined to produce conditions that quickly became untenable for a large proportion of the population.

## 8. CONCLUDING COMMENTS

In this paper we have attempted to use CFE tools to forensically analyse a fire scenario similar to the tragic Gothenburg fire incident. It is not claimed that the model accurately reproduces the Gothenburg incident as a key component required for such an analysis, i.e. the evolution of the fire, was not adequately represented within the model. This was due to a number of factors. Firstly, inaccuracies were introduced by the experimental fire tests due both to scale and discrepancies between the experimental conditions and the original fire conditions. Secondly, the methods used to scale up the small-scale experimental fire data are acknowledged to have been crude. In addition, the representation of the incident within the buildingEXODUS evacuation model required a number of simplifications to be made that would also have affected the result produced. All of these issues should be considered when analysing the model predictions.

Given these provisos, where a direct comparison was made, the model predictions bore a striking resemblance to the events that took place during the actual incident. The model predictions correctly demonstrated that the evacuees experienced severe congestion during their attempted evacuation, especially around the approach to the only means of escape: the staircase located in the north-western corner of the building. This fact was highlighted by the NFPA in their assessment of the incident [1]. This congestion was produced by the overloading of the single available egress route caused by the simultaneous arrival of a large number of evacuees and the deteriorating conditions.

While over-predicting the number of fatalities, the model successfully predicted the order of magnitude of the number of fatalities. Furthermore, the predicted location of the fatalities matched that found in the actual incident. These occurred at the location of greatest congestion: the (north-western) internal exit located at the top of the staircase. In addition, the number of injuries predicted in this scenario matched those produced during the actual incident.

It should be acknowledged that the major cause of the fatalities in these simulations differed from that of the actual incident. Due to the inaccuracies in the model fire representation, the major cause of death in the computer simulations was radiative heat, while in the actual incident the inhalation of toxic products was cited as the primary cause of death.

In addition to examining the original scenario, the model was also used to investigate the two of the main findings of the NFPA incident report: the overcrowding evident in the main hall and the location of the fire due to the storage of the chairs in the eastern stairwell. In both cases, the results produced by the model supported the findings of the incident report and these factors were found to have an enormous influence over the ability of the patrons to evacuate without succumbing to the conditions.

Given the provisos made in regard to the data supplied, the assumptions made and the procedure outlined, it is contended that the results produced are satisfactory and have successfully captured the major events and influences of the original incident. Finally, the major conclusion of this work is that the high death toll reported in this incident was a result of the severe level of overcrowding experienced in the hall combined with the loss of one of the means of egress. Not only has the application of the model enabled this conclusion to be arrived at, but has also established the processes and events which led to these outcomes. This analysis has provided insight into the tragic event and an understanding of why so many people died at the Gothenburg incident. Clearly, evacuation and fire simulation analysis of this type has an important role to play in fire investigation.

The use of CFE tools in order to forensically analyse actual fire scenarios is a useful additional tool that may allow the engineer to better uncover the underlying causes of the outcome of such scenarios. However, the nature and extent of this analysis is enormously sensitive to the data available. In an earlier publication, involving the forensic analysis of the Beverly Hills Supper Club incident [22], only limited data was available concerning several key aspects of the original incident, excluding the routes adopted by the evacuees and a comprehensive understanding of the development of the fire. However, relatively detailed information relating to the experiences of the evacuating population was available, through the collection of data via interviews and questionnaires. This data-set precluded a detailed *quantitative* comparison, compelling the analysis to instead concentrate upon the evacuee’s behaviour (and its consequences). The insights produced were still valuable, but were tempered by the limitations of the original data-set.

In this instance, a more comprehensive (although still flawed) understanding of the fire evolution was available through the performance of experimental trials and the application of modelling techniques. However even here, the very nature of this process would have introduced discrepancies that would have certainly influenced the results produced. It is important that these limitations are acknowledged when incorporating secondary data and extrapolating additional information from it; indeed it is good practice to describe the limitations of the modelling procedure irrespective of the nature of the scenario being examined. However, this does not minimize the impact (and usefulness) of the use of Computational Fire Engineering tools in such situations, without which the type of insight described here would have not been possible, removing the capacity for support to be provided for other forms of investigation (e.g. material evidence, expert option, etc.), as well as indicating further avenues of investigation (e.g. highlighting influential events, potential solutions, etc.) that otherwise might have been missed.

## 9 Acknowledgements

The authors would like to acknowledge the assistance provided by SP through the provision and interpretation of the experimental data.

## 10 References

[1] Comeau, E. and Duval R.F., “Dance Hall Fire Gothenburg, Sweden, October 28, 1998”, Fire Investigations, NFPA, 2000.

[2] Ingason,h., Wickström, u., van Hees, P.,”The Gothenburg discotheque fire investigation”, Proceedings of Interflam 2001, the 9th International Fire Science and Engineering conference, 17-19th September, Scotland, Interscience Communication, pp 965-976, 2001.

[3] http://www.sos.se/sos/publ/REFERENG/0112314E.htm

[4] http://writer-tech.com/pages/summaries/summgothenburg.htm

[5] http://pdm.medicine.wisc.edu/thore.htm

[6] http://www.geocities.com/framemethod/goteb.html

[7] http://www.crismart.org/pdf/Gothenburg_Fire.pdf

[8] http://www.wsws.org/news/1998/oct1998/swe-o31.shtml

[9] Wickstrom, U. and Ingason, H., “The Gothenburg Discotheque Fire Reconstructed on a Model Scale”, BrandPosten, Fire Technology, Sp Swedish National Testing and Research Institute, 22, June, 2000.

[10] Wickstom, U.,“The Discotheque Fire – Why did so many die?”, BrandPosten, Fire Technology, Sp Swedish National Testing and Research Institute, 22, June, 2000.

[11] Ingason, H., “The Discotheque Fire – Model Trials Confirm the Hypothesis ” , BrandPosten, Fire Technology, Sp Swedish National Testing and Research Institute, 22, June, 2000.

[12] Thureson, P.,“The Discotheque Fire – the stairwell was full of combustible material”, BrandPosten, Fire Technology, Sp Swedish National Testing and Research Institute, 22, June, 2000.

[13]Personal communication with H. Ingason of SP, Sweden.

[14] Peacock, R.D., Forney, G.P., Reneke, P., Portier, R., and Jones, W.W., “CFAST, The Consolidated Model Of Fire Growth And Smoke Transport”, *NIST Technical Note 1299, U.S. Dept. of Commerce, Technology Administration, National Institute of Standards and Technology, 1993.*

[15] Purser, D.A., "Toxicity Assessment Of Combustion Products", *The Sfpe, Handbook Of Fire Protection Engineering (3rd Edition), Ed:Dilenno, P.J., Drysdale, D., Beyler, C.L., Walton, W.D., Custer, R.L.P., Hall, J.R., and Watts, J.M.W., National Fire Protection Association,Quincy,Ma, pp (2-83)-(2-171), 2002.*

[16] Purser, D.A. "Modelling Time To Incapacitation And Death From Toxic And Physical Hazards In Aircraft Fires", *Conference Proceedings No: 467, Aircraft Fire Safety, NATO-AGARD Sintra, Portugal 22-26 May, pp41-1 - 41-12, 1989.*

[17] Jin, T And Yamada, T., "Irritating Effects From Fire Smoke On Visibility", *Fire Science And Technology, Vol. 5, No.1, pp79-90, 1985.*

[18] Jin,T., "Visibility Through Fire Smoke", *Report Of Fire Institute Of Japan #42,Sept 1976.*

[19] Galea, E.R. Gwynne, S. Lawrence, P.J. Filippidis L., Blackshields, D. and Cooney, D., “buildingEXODUS version 4.0: User Guide and Technical Manual”, University of Greenwich, 2003.

[20] Purser, D., "People And Fire", *Inaugural Lecture Series, 17th February, University Of Greenwich, ISBN 1-86166-117-7, 1999.*

[21] Purser, D., “Human Tenability”, Technical Basis for Performance-Based Fire Regulations, Engineering Foundation Conference, 7-11 January, San Diego, United States, 2001.

[22] Gwynne, S., Galea, E.R. and Lawrence, P.J. , “Forensic Analysis of an Incident Using an Evacuation Model”, CMS Press, London 2004.