The University of Nottingham

Genetic Algorithms and GIS Data for Decision Making in Planning Water Distribution Networks

Karwan Ghazi Fendi

BSc., MSc.

Thesis submitted to The University of Nottingham for the degree of Doctor of Philosophy

December 2011

Contents

List of Figures

List of Tables

Abstract

This thesis is concerned with the optimal design of Water Distribution Networks (WDNs). The design involves finding an acceptable trade-off between cost minimisation and the maximisation of numerous system benefits. The primary design problem involves cost-effective specification of a pipe network layout and pipe sizes in order to satisfy expected consumer water demands within required pressure limits. The design of a WDN has many variable parameters such as position and size of the water sources, position and the size of the pipes and position of the treatment plants. However, the layout is constrained by the location of existing facilities such as streets and buildings and other geographic features. The total costs may consist of the cost of network materials such as pipes, construction works and system operation and maintenance. The problem may be extended to consider the design of additional components, such as reservoirs, tanks, pumps and valves. Practical designs must also cater for the uncertainty of demand, the requirement of surplus capacity for future growth, and the hydraulic reliability of the system under different demand and potential failure conditions.

The thesis reviews the literature related to water distribution networks, their design and optimisation. It then presents a Genetic Algorithm (GA) formulation to assist in developing the design of a water distribution network.

The main aim of this research is to investigate the possibility of combining GAs and GIS in the design optimisation. A decision mechanism is developed which enables the model to reach a meaningful solution and provide a practical design technique for WDNs. The aim is also to provide an experimental analysis of the combined GA and decision mechanism to solve the problem in hand and to assess the robustness of these techniques when applied to different instances.

An initial prototype model is presented for the design of a WDN which is used to determine the necessary features of the 'final' model. These features include the world in which the model will be built, the design of the fitness function, chromosome representation, and GA operators.

The research mainly concluded that the initial model prototype was useful to determine the necessary features and to produce the final model which enables a variety of necessary factors to be explicitly included in the design of WDNs.

This initial model suggested that the final model should include the decision mechanism, which is a matter of policy management and hydraulics, and hydraulic principles which allowed to compare the behaviour of different parameters and to simulate the functioning of the network under different scenarios. Water allocation and distribution policies can be applied according to the importance of the demand area and the ability of the system to deliver sufficient water amounts. These policies link essential hydraulic and institutional relationships as well as water uses and users and allocation decision-making process.

It was also found that the representation of the world layout is important. The world is described in GIS in terms of models that define the concepts and procedures needed to translate real-world features into data. The important aspects in the chromosome representation are the node positions, the links. In this case, a chromosome must contain the three-dimensional node coordinates, the connection between nodes, the head required to pump the water. The best model parameters were extracted to be used in real-life situations. The result of tests on an example world demonstrated that the model was successful, and the potential exists for the use of this formulation in more complex and real-world scenarios.

Publications

While pursing this research programme, two publications are produced. The following publications are based on the research conducted for this thesis:

FENDI, K.G., SALIH S.S., 2011. 'Avoid Desired Landscapes in the Construction of Water Distribution Networks Using Genetic Algorithms and GIS'. Paper presented at East Midlands Universities Association Postgraduate Conference on Perspectives on Landscape. Nottingham Trent University, 13^{th} September, Nottingham, UK.

FENDI, K., AHMAD AL-HADAD, B. and MAWDESLEY, M.J., 2010. Using genetic algorithms to assist in the design of a water distribution network. In Computing in Civil and Building Engineering, Proceedings of the International Conference, W. TIZANI (Editor), 30 June-2 July, Nottingham, UK, Nottingham University Press, Paper 204, p. 407, ISBN 978-1-907284-60-1

Acknowledgement

This thesis is dedicated to my parents, who taught me the value of live and education. I am deeply and forever indebted to them for their continued support and strong faith in me.

I would like to express my gratitude to *Dr Michael Mawdesley* for his supervision, advice, criticism, and guidance throughout this research. I am grateful to him for the long discussions that helped me sort out the technical details of this work.

I also gratefully acknowledge *Dr Nawfal Hussein* for his support and playing a significant role in my life.

It would not have been possible to pursue this research without the help and support of many kind people. Their good advice, support and friendship has been invaluable on my academic and a personal live, for which I am extremely grateful. Their support and care helped me overcome difficult times and stay focused on my research.

None of this would have been possible without the love, concern and patience of my parents, brothers and sisters.

The most special thanks belong to my wife *Sulin*, for her love, support and company during the research period. I also credit my little one, *Reejwan*, for inspiring and amazing me every day.

List of Abbreviations

2-D	Two Dimensional
3-D	Three Dimensional
ACO	Ant Colony Optimisation
ANN	Artificial Neural Network
CFG	Control Flow Graph
DBF	Demand Benefit Factor
DFS	Depth-First Search
DSS	Decision-Support System
EA	Evolutionary Algorithm
FAO	Food and Agricultural Organisation
FORM	First-Order Reliability Method
GA	Genetic Algorithm
GUI	Graphical User Interface
GIS	Geographic Information System
GP	Genetic Programming
LP	Linear Programming
MGA	Messy Genetic Algorithms
MSV	Marginal Swap Value
NLP	Non-linear Programming
POS	Public Open Space
PSO	Particle Swarm Optimisation
PTDS	Percentage of Total Demand Satisfied
RMV	Random Mask Value
RNSGA-II	Robust Non-dominated Sorting Genetic Algorithm II
SA	Simulated Annealing
SMGA	Structures Messy Genetic Algorithms
UN	United Nations
UNEP	The United Nations Environment Program
VuF	Visited But Not Finished
WDN	Water Distribution Network
WDS	Water Distribution System

Chapter 1

Introduction

1.1 Background

A water distribution system is an essential infrastructure that conveys water from a single water source or multiple water sources to current and future consumers. A typical water distribution system consists of pipes, pumps, reservoirs, tanks, and valves.

In this thesis, the term Water Distribution System (WDS: singular and WDSs: plural) will be used as general term to denote a water supply system from the source of water to the consumers. The term Water Distribution Network (WDN: singular and WDNs: plural) will be used to denote a part of the system.

Globally, water demand is increasing while the resources are diminishing (Dinar et al., 1997). It is thus of increasing importance that the existing water resources be allocated and distributed more efficiently. The distribution system is designed to reliably distribute bulk water supplies to the suburbs and cater with demand patterns, pressure limitations, velocity limitations, quality assurance, and maintenance issues at minimum costs. To achieve these objectives, particular combinations of reservoir storage, network layout, water mains, and pumping are used, depending on the system service area topography and size. This can be named as the optimal design of a water distribution system. The design optimisation of WDNs, in this context, is the least-cost design of the WDN. The least-cost refers to the minimisation of the implementation and operation costs of the network in terms of the cost of the pipes to be installed, which involves least-cost routing; and energy costs to pump the water with the required quantities to current and future consumers.

The simulation of hydraulic behaviour in WDNs is a complex problem because WDN design problems require solving a set of non-linear equations (Walski et al., 2003). The solution process involves simultaneous consideration of head loss, energy, and continuity equations.

Optimisation techniques have been used to provide a decision mechanism for improving WDNs including, amongst others, network design and rehabilitation, model calibration, leakage detection, and pump scheduling. These tools represent an attempt to provide further assistance to practitioners through the means of such intelligent methods. In addition, evaluating alternative solutions adds to the power of the process as part of a decision-support system (DSS). These decision-support systems encompass a wide range of computer applications and can be structured into three major phases (Ford, 2007):

1. Intelligence: this phase determines whether there is a problem or an opportunity for change.
2. Design: in this phase, decision alternatives are found.
3. Choice: the best alternative is indicated in this phase.

An optimisation algorithm generates potential solutions that are optimal or near-optimal with respect to the problem objectives. These solutions are iterative and the time taken to compute the solution is largely dependent on the size and configuration of the network model. The ability of evolution algorithms to converge rapidly on an optimal or near-optimal solution has made such algorithms popular subjects for research. Despite considerable improvements in computer power, there is still a need to improve the performance of these algorithms to allow for more complex optimisations to be undertaken with acceptable efficiency and effectiveness (Ford, 2007).

Optimisation for WDNs design is currently not yet part of standard engineering practice, although many optimisation tools are already incorporated into certain design programs (Walski et al., 2003). Other difficulties with the inclusion of optimisation are that the model is a simplified version of the real-world and the inability of algorithms to fully cover the design process. All this because of data uncertainty / variability and the complex non-linear interactions between components, and that the use of some search heuristics fails to always find globally optimal designs (Weise, 2009). Therefore, some level of optimisation is essential to enable effective decision-making. The optimisation model must be broad enough to address the wide range of important criteria for WDS design but simple enough to be used by a non-expert individual. Any approach should be efficient, holistic, and systematic, in order to enable correct decision-making.

Genetic algorithms (GAs) are one of the best-known types of evolutionary techniques (Michalewicz, 1999) in which the solution search space is searched by generating candidate solutions with certain probabilities that evolve over the course of the search process. Several applications of GAs have been presented by Holland (1975) and Goldberg (1989). GAs proved to be a powerful approach with a proven ability to identify near-optimal solutions for WDNs (Savic et al., 1997; Halhal et al., 1999). For instance, Savic et al. (1997) and Halhal et al. (1999) used GAs to achieve an optimal solution for the WDS design problem, considering minimisation of the cost as the sole objective.

Geographic information systems (GIS) and spatial decision-support systems have increasingly been used to generate alternatives to aid decision makers in their decisions. The ability of GAs to search for a solution space and selectively focus on promising combinations of criteria makes them ideally suited to such complex spatial decision problems. Many researchers have used GAs to solve spatial problems in different application domains.
It is possible to integrate GAs with the GIS and be user friendly. In addition, an algorithm for a GIS problem has to be robust, is that it should be possible to integrate the algorithm in a GIS environment, where it can be used and extended easily by the user.

Finally, WDNs are very costly to install and maintain (Walski et al., 2000), and it is often the case that optimisation can achieve large cost savings, as this shall be demonstrated in a real-world application case study towards the end of this thesis. Any methodology which makes WDS design easier and more comprehensive is worth consideration, especially if it can produce designs, which are both cheaper and more reliable.

1.2 Aims of the Research

The aim of the research is to investigate the possibility of combining GAs and GIS in the WDN design optimisation. This is done with the aid of a decision mechanism that enables the model to reach a meaningful solution and provide a practical design technique for water distribution networks.

The aim is also to provide an experimental analysis of the developed GA to solve the problem in hand and to assess the robustness of these techniques when applied to different instances.

In this thesis, the focus is primarily on the GA optimisation and on the characteristics of the problem that affect it. This includes investigations to the representation of the genetic material employed and the operators that act on it in order to promote the efficient convergence of the population. This is done through carrying out a set of experiments to investigate the best combinations of GA parameters that may be applied to real-world problems.

1.3 Objectives of the Research

In order to achieve the aims of the research as set out in section 1.2, the following objectives have been formulated:

1. To provide a background to what is required in planning with particular references to aspects which might affect or be affected by water supply.
2. To provide a background to the use of networks and theories related to networks which have been applied to WDNs and similar fields of WDNs.
3. To provide an understanding of WDNs including:
   - The planning process
   - WDN components
   - WDN features
   - The types of WDNs
   - The hydraulic theory
4. To provide wide literature review and critical analysis of work done by others to model and optimise WDNs.
5. From the important features identified through the achievement of previous objectives, to produce an initial prototype model for the design of a WDN which can be used to determine the necessary features of the 'final' model and how these might be included.
6. From objective 5, to produce the 'final' model.
7. To test the model to determine its practicality and the best values for all the parameters.
8. To demonstrate the applicability of the model on a hypothetical real-world layout.

1.4 Thesis Outline

This thesis is arranged in eight chapters with three supplementary appendices. The thesis adopts the following structure:

Chapter One - Introduction

The introduction provides a general background to the research area and the problem to be addressed. It also presents a statement of the research aims and objectives and the layout of the thesis. Finally, it presents some of the research limitations.

Chapter Two - Planning and Water Networks

It provides a background to planning and a foundation for understanding some planning concepts. The chapter introduces basic concepts behind networks and introduces a search technique to find cycles in network graphs. The chapter also provides an introduction to water distribution systems and some related terms. It constitutes a literature survey of fluid mechanics for WDNs providing the foundation required to understand pipe hydraulics and hydraulic network simulation theory.

Chapter Three - Optimisation of Water Distribution Networks and GIS

It introduces the concept of evolution algorithms (EAs) and relates the design and implementation of the application of EAs to WDS design optimisation problems with a particular emphasis on GA approaches to the optimisation of WDS. In addition, the chapter introduces a framework in which optimisation algorithms can be combined with GIS to assist in the design of WDNs.

Chapter Four - Initial Model Formulation

This chapter defines the initial mode prototype of the problem to be examined, namely, the problem of the identification of the near-optimal WDN layout and the amount of delivered water to the consumers. This chapter presents a new methodology for the optimisation applications that involve network simulation. The chapter provides the initial steps involved in the model formulation that introduced the fitness function, encoding, and GA operators.

Chapter Five - Model Development

The actual optimisation model implementation used in this study is developed. Novel extensions to the classical problem formulation presented in chapter four are presented in this chapter with the aim of adopting a wider range of real-life scenarios and improving the model in terms of the algorithm performance and the quality of the results. A decision mechanism is introduced in this chapter and included in the 'final' model.

Chapter Six - Tests and Results

In this chapter, experimentation is undertaken of five 'manufactured' example worlds. These worlds allow for several scenarios to be tested in respect of different GA operators and model parameters. This chapter demonstrates the applicability of the model formulation introduced in the prior chapters through their application to a number of small-scale world problems where only two land access costs are introduced, namely, low cost areas and high cost areas. Many experiments are carried out in this chapter to extract the best model parameters to be used.

Chapter Seven - Application of the Model - An Example

This analysis is extended in chapter eight where a larger-scale problem is adopted. This chapter reapplies the methodologies presented to a real-world layout which represents a more complex problem. This is to demonstrate a wider applicability of the model to optimisation problems.

Chapter Eight - Conclusion and Further Work

The final chapter details the conclusions that can be drawn from this research and describes possible avenues for further research work.

There are four appendices. The first appendix provides an introduction and the basic knowledge to genetic algorithms and geographic information systems. Appendix B presents some definitions related to genetic algorithms, water distribution networks, and networks. Appendix C contains an illustrative example of removing cycles from a directed graph. The last appendix (appendix D) presents the important aspects of the computer model used to demonstrate the work carried out in this research.

1.5 Limitations of the Research

All PhD theses have limitations. These are brought about by the nature of the problem, but may be influenced by the previous work in the area, the existence of suitable computer packages and other tools, and the availability of appropriate data.

This work also has boundaries. The proof of these are as follows, where their necessary explanations are discussed throughout the thesis.

1. The individual term 'planning' is a broad concept. It is used in this research in the context of regional and urban planning to understand the relationship between planning and the design of WDNs and the planning of them.

2. The ability of GIS to handle and process geographically referenced data which describe both the location and characteristics of the spatial features on the Earth's surface, made GIS to be used widely. GIS is used in many fields such as scientific investigations, land-use planning, emergency planning, market analysis, facility management, and military applications. Because GIS is a wide subject area, some GIS concepts of interest are used to demonstrate this work, namely, spatial and attribute data, and raster and vector models. These concepts were developed and used to produce a geographically aware computer program. In addition, no commercially distributed or open-source GIS software is used in this work because none of these software, to the author's knowledge, would be able to meet the specific objectives of this research.

3. There are many examples of developed software provided in this thesis. These software are routinely used for operational investigations and the design of WDNs. Some of these software are used to simulate WDNs and others are used in the design optimisation of them. None of these software, to the author's knowledge, would be able to meet the specific needs of this research. Therefore, external software are not used in this work.

4. Although WDSs vary in size and complexity, they all have the same basic purpose to deliver water from the source to the customer. Water use may vary over time both in the long-term (annually) and the short-term (daily). In addition, water networks may vary over time in the long-term. System growth or degradation may occur because of population growth, territorial extension, acquisition, or wholesale agreements between water supply utilities. These changes in the network and the water use are not considered in design optimisation of WDNs.

Chapter 2

Planning and Water Networks

2.1 An Introduction to the Chapter

This chapter should provide the reader with a foundation for understanding the concepts and terminologies used in the remainder of this thesis.

This chapter is divided into three main parts. Each part is followed by a summary to that part.

The first part starts with an overview to planning which includes urban and regional planning. It also discusses the relationship of planning with decision-making and sustainable development. Using geographic information systems (GIS) with planning is also discussed. The fields of urban and regional planning are discussed at the end of this part.

Because water distribution networks are examples of networks in the context of graph theory, the second part of this chapter is an introduction to networks. It introduces directed and undirected graphs. The representation of the graphs are also discussed. This part also introduces the presence of cycles in networks and how they can be detected. A method of detecting cycles in graphs is demonstrated.

The third part of this chapter is devoted to water distribution networks. It starts with an overview to water distribution networks then an introduction to water supply planning. The different components, features and types of WDNs are also discussed. The principles of pipe flow and hydraulics are demonstrated and the some relevant equations are presented. The topics of pumping, water demand and demand estimation, are discussed. Problem identification involves various forms of analysis, which also might include data simulation and modelling. Modelling and simulating WDN are discussed.

2.2 Urban and Regional Planning Concepts

Planners need to address the best use of a community's land and resources for residential, commercial, institutional, and recreational purposes; this section introduces planning-related terms and the need for planning.

The individual term "planning" is a broad concept and has a generic meaning. Planning can be defined as the predetermination of a course of action aimed to achieve some goal (Hayes-Roth and Hayes-Roth, 1979; Cambridge Advanced Learner's Dictionary, 2011). Ward (2004) defined planning as an organisational activity to create and maintain a future plan without ignoring past arrangements. Other researchers explained planning as the process of strategic choice, making decisions and policies to guide public and private sectors to create a desired future (Friend and Hickling, 2005; Sendich, 2006). For the rest of this thesis, the term "planning" and "planners" will refer to the disciplines related to urban and regional planning.

Planning is related to the areas where people work and live by preserving environmental qualities and protecting people's interests. Planning provides decision-makers with the required information they need to make decisions. These decisions will be used to avoid future planning problems and attempt to solve existing problems. It provides a framework to develop economic, social, environmental, architectural, and political activities.

Planners aim to shape the growth development by avoiding dense development or overly scattered development (Levy, 2002). This means that residents have a ready access to shopping centres, schools, recreational and cultural facilities. A convenient traffic network is important to avoid excessive congestion. Consequently, the separation of incompatible land-uses and activities as well as the location of public facilities such as schools and social service centres is essential. The larger and more complex the community, the more difficult the decision-making becomes.

Sustainable development and decision-making have become important concepts in today's urban and regional planning fields. Many planners have begun to advocate for the development of sustainable cities. The relationship between planning and decision-making as well as sustainable development are discussed in section 2.2.1 and section 2.2.2 respectively. Regardless of the community size, planners deal with spatially related information such as zoning, land-use, addresses, and water and transportation networks. Planners also study and keep track of multiple urban and regional indicators, forecast future community needs, and plan accordingly to guarantee a good life quality. The uses of GIS in planning is discussed briefly in section 2.2.3.

More significantly for this work, the primary focus of the water supply planning is to make recommendations on a range of alternatives that should be evaluated to ensure the best service for a region. This will be discussed in details in section 2.4.2.

2.2.1 The Decision-Making

Decision-making is defined as making a choice about something after thinking about several possibilities (Cambridge Advanced Learner's Dictionary, 2011). Decision-making is a subjective process because the perception regarding a problem may differ from a person to another (Malczewski, 1999). The whole decision-making process relies on information about alternatives in such a way that decision-makers can receive their supportive and required information among a set of specified alternatives.

The general objective of decision-making tools is to assist decision-makers to select the best possible alternative from a set of feasible alternatives using user defined priorities. The process of development involves making decisions. Therefore, the planning is, as a process of strategic decision-making, making choices among possible development alternatives. In addition, planning process selects one or more effective lines of action.

A group and/or an individual can make decisions. The framework of the decision process is summarised by Jankowski (1995) to four distinguished steps:

1. Problem definition: it defines a problem driven from the discrepancy between the present state and the desirable state (a solution to a problem).

2. Search for alternatives and selection criteria: it defines the alternatives that have to be taken under consideration and criteria for evaluating these alternatives.

3. Evaluation of alternatives: the impact of each alternative on every criterion is assessed.

4. Selection of alternatives: it deals with ordering the alternatives from most desirable to least desirable and either the top alternative is selected or a group of desirable alternatives.

Geographical or spatial data defined as unorganised data that are associated with a location. When the data are organised, analysed and interpreted, they then become information. In this stage, the data can be useful for effective decision-making in urban and regional planning (Yeh, 1991; Jones et al., 1997; Fedra, 1999). For example, the population density refers to a particular location. Geographical and spatial data are interchangeable. Here, facts, results of observation, remote-sensing images, census figures and statistics can be considered geographical data. According to Jankowski (1995), geographic information systems (GIS) are carried out in the second step in searching for feasible alternatives. Most decision-making problems solved with GIS are related to a search for suitable sites, routes and land-use (Jankowski, 1995). More information about GIS can be found in appendix A.

Most significant decisions tend to have a level of complexity, which require the use of organised algorithms that can accommodates multiple criteria. In addition, the citizens of a region or a city may want to know that a variety of plans have been equally considered before a final decision is made (Healey, 2006). Thus, decision-making requires effective methods such as optimisation techniques to solve complex planning problems. Chapter 3 demonstrates more information about the possible uses of optimisation techniques such as mathematical and meta-heuristic optimisation techniques.

2.2.2 Sustainable Development

Sustainable development is a broad concept. The United Nations Environment Program (UNEP) defines it as the development that ensures a balanced use of resources for better quality of life in which the environment today does not restrict their use by future generations.

Concerns about environmental quality, social equity, economic vitality, and the threat of climate change had converged to produce a growing interest in the concept of sustainable development. Sustainable development must facilitate economic development while fostering environmental protection. The concept of sustainable development was introduced in the early 1980s in order to reconcile conservation and development objectives (Wheeler, 1998). One definition of sustainable development was put forth in 1987 by the Brundtland Commission: "Sustainable development is the development that meets the needs of the present without compromising the ability of future generations to meet their own needs." (Brundtland, 1987).

Today, sustainable development is widely viewed as the development that improves the standard of living and quality of life, while at the same time protecting and enhancing the natural environment and honouring local culture and history. It is not only stressing the importance of resources for economic growth, but also draws attention to the need to develop methods that emphasise the potential between economic development and environmental improvement (Mikolajuk and Yeh, 1999).

Sustainable development has a worldwide attention and international support. Efforts are being made all over the world to increase the sustainability of development patterns. In addition, decision-makers and regional planners are increasingly being asked to take a leadership role in sustainable development efforts and apply their expertise to analyse sustainable development issues.

2.2.3 Using GIS in Planning

Spatial planning is required to satisfy the environmental, social, economic and physical requirement of the residents on short and long-term basis. However, the conventional systems of urban and regional planning have been found to be inappropriate in promoting sustainable urban development (Feeney et al., 2001). Therefore, tools are required. A new trend is emerging where GIS systems are combined with analytical, mathematical models to form spatial decision-support systems (Feeney et al., 2001; Birkin et al., 2005).

Successful implementation of GIS for a sustainable urban and regional planning is largely dependent on many factors. The first requirement is the automation of the GIS database. As data are costly to collect, store and shift through large quantities of unnecessary data, the most cost-effective approach is only to collect the data required for the specific task, in this case for urban and regional analysis and planning (Yaakup and Sulaiman, 2002). The collected data then need to be integrated using GIS. The urban and regional GIS should be organised to facilitate query and analysis. The GIS should be able to perform spatial modelling, so that alternative scenarios can be generated.

More importantly for this work, the design of water distribution networks (WDNs) and planning of them may heavily depend on GIS due to the natural geo-referenced representation of objects (houses, parks, roads), distribution lines (pipelines), and water resources (main pumping station and intakes). Integrated frameworks for combining GIS and optimisation process can be designed to optimise WDNs. In these integrated systems, GIS have been used to handle the spatial data and provide a decision and management functions which are parts of the planning process. Using GIS with optimisation techniques are discussed in chapter 3.

2.2.4 Urban Planning

Many early settlements were located near water sources, fertile lands and accessible transport routes to support the population in that area. The first ideas of town planning started to grow in the last decades of the nineteenth century (Levy, 2002; Ward, 2004).

The term "Urban Planning" refers to plan a geographical area distinct from rural areas (Levy, 2002; Brunn et al., 2003; Ward, 2004). Although the most common term is "Urban Planning", other authors used different terminologies to refer to the same concept such as "Town Planning", "City Planning" and "Urban Design Planning" (Steger, 2002; Balling, 2003; Sendich, 2006).

Ward (2004) defined urban planning as the science concerned with managing and shaping the future of a city growth both physically and socially. The future plans include dealing with the existing physical structures and past urban arrangements, which cannot be ignored (Ward, 2004). It includes the economic development of a city and addresses the social ties among societies.

Urban planners are mainly concerned with standards and rules of physical environment and social development plans. They need to address the best use of a community's land and resources for residential, commercial, institutional, and recreational purposes. Water supply, traffic congestion, overcrowding, and other economic and social matters are urban problems which planners should consider when they start to implement urban development plans.

Urban planning has many fields. These fields will be summarised in the following subsections.

2.2.4.1 Urban Design

Urban design is the art of making places for people and the design of public areas. It involves the management and design of buildings, spaces, and landscapes including the connections between people and places (Steger, 2002; Sendich, 2006).

Urban design differs with urban planning as it works through planning schemes and other development controls. In addition, it falls between the professions of urban planning and landscape architecture or architecture. For example, urban designers deal with the large scale of cities and long-time frames. Landscape architects are concerned with improving the ways in which people interact with the landscape and reducing the negative impacts of human use on landscapes (Steger, 2002).

2.2.4.2 Land-Use Planning

Because land-use planning is a complex process, decisions should answer questions related to the selection of various land-uses across an area. United Nation Food and Agricultural Organisation (UNFAO) defined land-use planning as the systematic assessment to select the best land-use options to meet people needs and conserve natural resources (Food and Agriculture Organization of the United Nations, 1993). Levy (2002) defined land-use planning as the human modification of the natural environment to the built environment in order to exploit the land for agriculture, industrial, residential or other purposes. Stewart et al. (2004) defined land-use planning as the process of allocating different uses (such as housing activities, recreational activities or manufacturing industries) across a geographical region.

Land-use practices have a direct impact on natural resources such as water, soil, plants and animals. In addition, a sustainable neighbourhood plan provides residences with schools, shopping facilities, water supply, parking structures and playgrounds (Steiss, 2002). Several researches highlighted the importance of the relationship between land-use and transportation planning (Balling, 2003; Lowry, 2004).

Cities may have the following land-use elements:

1. Public open spaces (parks, squares, or water ponds) in every neighbourhood.
2. Down-town or city centres with intimate and mixed uses.
3. Industrial, commercial and business parks with landscape greenbelts, wide roads, controlled traffic access points, and sign controls.
4. Residential areas with traffic-calming features, and separation of residential uses.
5. Preserved agricultural areas and natural features (natural topography, wetlands, and floodplains).

2.2.4.3 Transportation Planning

Transportation planning is the system that exists to improve the individual accessibility by sitting the transportation facilities (streets, motorways, bike lanes and other public transportation lines). Land-use integrates and shapes the demand for the transportation. For example, highways have been built because the increasing population caused serious congestion and delays. Additionally, the rapid increase of vehicle ownership created new plans and designs for motorway networks to facilitate efficient vehicle movements from an existing city to another (Levy, 2002).

Transportation planning assists governments to provide an adequate transportation system at reasonable costs. This process involves behaviour studies of the existing system, future travel demands and the interaction between the existing system and future demands. It is worth mentioning that estimating travel movements requires some geographical information database with respect to area divisions (Possiel et al., 1995).

2.2.4.4 Other Urban Planning Fields

1. Community Development:
   Community development also called "Community Building". It is the process of building communities on a local level with the aid of economy building and strengthening social ties (Lowry, 2004).

2. Economic Development and Social Planning:
   Economic development means controlling and directing the economic decisions and activities by the government (Healey, 2006). The economics of some cities depends on other services such as banking, accounting and advertising (Brunn et al., 2003). The income generated by industrial goods will be spent in the city where the industrial employees will spend this income on shopping, fuel, entertainment and other daily expenditures.

2.2.5 Regional Planning and Its Principles

Regional planners address environmental, economic, and social issues of a community as it grows and changes. Regional planning is introduced in this section followed by the principles of regional planning. Many researchers in this field highlighted several terms in regional planning such as "Regional Level" (Yeh, 1991; Possiel et al., 1995), "Regional Agencies" (Sanchez and Wolf, 2007) and "Regional Strategic Planning Process" (Blair et al., 2007).

Regional planning is the planning of the socio-economic development and addressing the physical and environmental impacts on a geographic area that exceeds the boundaries of a city or a town (Levy, 2002; Sendich, 2006). These aspects affect the placement of various infrastructures and the shape of regional zoning. The Planning of regional zones should conserve landscape areas, reduce pollution and facilitate transportation among regions (Possiel et al., 1995). Regional transportation requirements should be evaluated and designed according to the economic growth and different land-use schemes for a region. Regional planning can be applied to significantly big units of land, which were originally shaped by geological structure, surface relief, climate, vegetation, animal life and partly reformed by human being activities (Sendich, 2006).

Regional planning is closely related to urban planning. The relationship between regional and urban planning can be described by the efficient development of urban areas while preserving open spaces, agricultural lands and environmentally sensitive areas unsuitable for intensive development. Sensitive areas may include habitats of threatened and endangered species, streams and their buffers, steep slopes, and floodplains. In addition, critical and sensitive areas may also include historic structures or archaeological structures.

Principles of Regional Planning

Regional planning is the science of the efficient placement of infrastructure and zoning for a region which addresses environmental, social and economic issues. The term "Region" in the planning context can be administrative and is likely to include settlements and character areas. In 2001, The US Midwest Regional Planning Organisation outlined the principles of regional planning into three phases which are proposed chronologically (Midwest Regional Planning Organization, 2001):

- Phase I: Organisation and Coordination
  This phase develops a framework for regional planning. This stage is a general statement on the development, organisation and protection of some areas. This means that a well-balanced system of settlements and open spaces should be developed in the entire territory. For example, studies will be conducted to avoid regional development in floodplains and earthquake faults; However, these areas can be utilised as public parks or unimproved farmlands. In addition, green belt lands and transportation corridors can be assigned, and new infrastructure should be considered.

- Phase II: Technical Assessment
  This stage aims to provide an understanding of the current pollution levels, identify the principal sources, determine which regions have the same problems and estimate the impact of future strategies on the region. The

ecological functions of the rural areas should also be maintained with a view to their importance for the entire territory. Assessments should include the protection, conservation and development of the natural surroundings and landscape, including water bodies and forests.

- Phase III: Strategy Development and Implementation
  The purpose is to reach the goals which the regional strategies needed to achieve and implement regional plans. Midwest Regional Planning Organization (2001) pointed that the objectives of this phase include: 1) reaching consensus about the regional strategies needed to make reasonable progress toward the national visibility goal in class I areas, and 2) adopting and implementing system plans which reflects the regional strategy.
  Easy access between all regions for population and goods transport should be ensured. Therefore, building codes, zoning roles and polices will promote best use of land with cost-effective decisions (Brunn et al., 2003).

2.2.6 Summary of Planning

The core target of planning is to secure the citizen's quality of life, maintaining and/or improving living conditions in cities as well as rural areas. At the same time, regions and cities must keep pace with the increasing modernisation developments of today's world. The planning concept was introduced in the first part of the chapter. Decision-making and sustainable development are based on the best available information and coordination at the local, regional, and global levels. The section discussed the relationship of planning with other fields such as decision-making and sustainable development.

The use of GIS in planning was introduced in this section. GIS is found to be a very good tool for planners, enabling them to integrate a variety of data from multiple sources and to perform spatial analyses that previously might have taken more time.

Urban and regional planning are essentially concerned with shaping the future plans and deal with the consequences of past planning decisions. They involve consideration and evaluation of a wide range of complex issues. This first part demonstrated and discussed urban and regional planning. The aim was to explore the principles and the fields of urban and regional planning.

The next section discusses some principles of the networks in the context of graph theory. This knowledge is helpful to understand the behaviour and the design of water distribution networks.

2.3 The Theory of Networks

A network is defined as a collection of 'n' vertices (nodes) connected by 'm' edges (links).

The study of networks, in the form of mathematical graph theory, is one of the fundamentals of discrete mathematics (Gross and Yellen, 2004). The systems taking the form of networks (also called "graphs" in much of the mathematical literature) are many in the world (Newman, 2003). Examples include the Internet, the World Wide Web (WWW), social networks or other connections between individuals, organisational networks and networks of business relationships between companies, neural networks, distribution networks such as blood vessels or postal delivery routes, WDNs, and many others.

Network models are an important category of mathematical programs that have numerous practical applications. Part of their success is the direct mapping between the real-world, the network diagram, and the underlying solution algorithms. Table 2.1 presents some of real-life examples of networks.

Table 2.1: Real-Life Examples of Networks

Nodes	Links	Flow
Cities	Highways	Vehicles
Switching centres	Telephone lines	Telephone calls
Pipe junctions	Pipes	Water

Networks have also been studied in the social sciences. Typical network studies in sociology involve the circulation of questionnaires. Typical social network studies address issues of centrality (which individuals are connected to others or have influence) and connectivity (whether and how individuals are connected to each other through the network).

The recent years had a new movement in network research, with shifting away from the analysis of small scale graphs to consideration of large scale graphs. This new approach has been driven largely by the availability of computers and communication networks that allow to gather and analyse data on a scale far larger than previously possible. Where studies used to look at networks of tens or in some extreme cases hundreds of vertices, it is common now to see networks with millions of vertices (Dorogovtsev and Mendes, 2002).

For networks of tens or hundreds of vertices, it is a relatively easy to draw a picture of the network with actual points and lines. Specific questions can be answered about network structure by examining this picture using human eye. However, this approach is useless with a network of a million vertices where drawing a meaningful picture of this network is impossible.

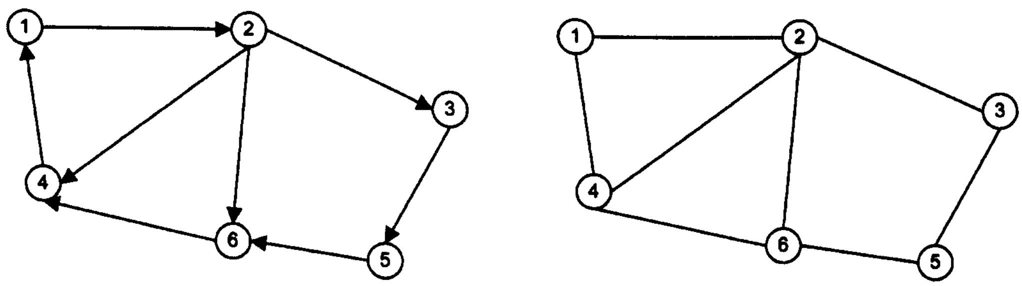
A set of vertices joined by edges is only the simplest type of network; there are many ways in which networks may be more complex than this. For instance, there may be more than one different type of vertex in a network, or more than one different type of edge.

2.3.1 Directed and Undirected Graphs

Graphs may be undirected such as those in a social network of people (how well two people know each other) or they can be directed, pointing in only one direction, (telephone calls between individuals).

A directed edge is one in which there is a single direction of the relation where the edge moves from node v to node u. A network that has this type of edges is called a "directed network"; if it does not, then it is called "undirected graph" meaning the edges can be in both directions.

For some real-life problems, an edge might infer a one way connection from one node to another. For example, when modelling the Internet as a graph, a hyper-link from web page v linking to web page 'u' would simply mean that the edge between 'v' to 'u' would be unidirectional. That is, that one could navigate from 'v' to 'u', but not from 'u' to 'v'. Graphs that use unidirectional edges are said to be directed graphs and are used in this work. An example of directed and undirected graphs is given in figure 2.1. Further definitions associated with graphs and networks can be found in appendix B.



(a) Directed Graph (b) Undirected Graph

Figure 2.1: Undirected and Directed Graphs

On this figure, two graphical representations of networks are shown. Nodes are represented by the circles and the lines that connect them represent the edges.

A number of points can be made about directed and undirected graphs:

- A directed graph can be transformed to an undirected graph by "erasing arrows" on all edges and removing duplicate edges.
- An undirected graph can be transformed to a directed graph by adding an edge (u, v) for every edge (v, u).
- The round-trip transformation undirected $\Rightarrow$ directed $\Rightarrow$ undirected does not loss information, but the round-trip transformation directed $\Rightarrow$ undirected $\Rightarrow$ directed can loss information.

A graph is represented so that:

$$G = (V, E)$$

where $V = \{v_1, v_2, ..., v_n\}$ is the set of n vertices and $E = \{e_1, e_2, ..., e_m\}$ is the set of m edges. It is said that two nodes are connected or are neighbours if there is an edge between them (Newman, 2003).

2.3.2 Adjacency Matrix

The usual way to work with a network is through its adjacency matrix (Gross and Yellen, 2004).

The adjacency matrix of M is a two-dimensional $n \times n$ array, and shall be called "adj_arr". An entry in this matrix $M(i, j)$ represents the value of the edge between nodes v_i and v_j. i.e. If the edge (v_i, v_j) is in M, $adj\_arr[i][j] = 1$. If there is no such edge in M, $adj\_arr[i][j] = 0$.

An example of the adjacency matrix for a directed graph, which represent the example graph shown in figure 2.1(a), is presented in figure 2.2.

Destination Node

Source Node	1	2	3	4	5	6
1	0	1	0	0	0	0
2	0	0	1	1	0	1
3	0	0	0	0	1	0
4	1	0	0	0	0	0
5	0	0	0	0	0	1
6	0	0	0	1	0	0

Figure 2.2: The Adjacency Matrix for The Directed Graph in Figure 2.1(a)

If the network is undirected then M is symmetric. The entries of M^2 are the number of paths of length two between nodes, in general M^m gives the paths of length m between the nodes (Gross and Yellen, 2004). An example of the adjacency matrix for an undirected graph, which represent the example graph shown in figure 2.1(b), is presented in figure 2.3.

Destination Node

Source Node	1	2	3	4	5	6
1	0	1	0	1	0	0
2	1	0	1	1	0	1
3	0	1	0	0	1	0
4	1	1	0	0	0	1
5	0	0	1	0	0	1
6	0	1	0	1	1	0

Figure 2.3: The Adjacency Matrix for The Undirected Graph in Figure 2.1(b)

The advantages of representing the edges using adjacency matrix are:

1. Simplicity in implementation: a two-dimensional array is needed for the network representation.

2. Creating/removing edges is easy: the array values need to be updated.

The drawbacks of using the adjacency matrix are:

1. Increased memory: the creation of adjacency matrix requires to declare $n \times n$ matrix, where n is the total number of nodes.
2. Redundancy of information, i.e. to represent an edge between u *to* v and v *to* u, it requires to assign two integer values in the adjacency matrix.

In most of programming languages including Visual Basic, the representation of the adjacency matrix can be implemented by a two-dimensional array of numeric, boolean, or alphabet values.

2.3.3 Adjacency List

Adjacency list is an array of linked nodes. In other words, each node in the adjacency list maintains a list of adjacent nodes.

For the given directed graph example in figure 2.1(a), the edges will be represented by the below adjacency list:

Table 2.2: The Adjacency List Representation for The Directed Graph in Figure 2.1(a)

Node	Node Adjacency List
$1 \Rightarrow$	2
$2 \Rightarrow$	3, 4, 6
$3 \Rightarrow$	5
$4 \Rightarrow$	1
$5 \Rightarrow$	6
$6 \Rightarrow$	4

For the undirected graph in figure 2.1(b), the edges will be represented by the below adjacency list:

Table 2.3: Adjacency List Representation for an Undirected Graph 2.1(b)

Node	Node Adjacency List
1 $\Rightarrow$	2, 4
2 $\Rightarrow$	1, 3, 4, 6
3 $\Rightarrow$	2, 5
4 $\Rightarrow$	1, 2, 6
5 $\Rightarrow$	3, 6
6 $\Rightarrow$	2, 4, 5

It can be noted that with an undirected graph, an adjacency list representation duplicated the edge information. For example, in the adjacency list representation in table 2.3, node 2 has node 4 in its adjacency list, and node 4 also has node 2 in its adjacency list.

Each node has precisely as many nodes in its adjacency list as it has neighbours. An adjacency list is a more compact representation and therefore memory efficient representation of a graph where data are never stored more than needed. Specifically, for a graph with V nodes and E edges, a graph using an adjacency list representation will require $(V + E)$ instances for a directed graph and $(V + 2E)$ instances for an undirected graph.

The downside of an adjacency list is that adding/removing an edge to/from adjacent list is not so easy as for adjacency matrix. In addition, the determination whether an edge exists from node 'u' to 'v' will require searching '$u's$' adjacency list. For dense graphs, 'u' will likely have many nodes in its adjacency list.

WDNs can be represented as graphs where pipes represent the edges and pipe junctions represent the nodes. In WDNs, the direction is given by the direction of the water flow. Therefore, only directed graphs will be considered in the remainder of this thesis.

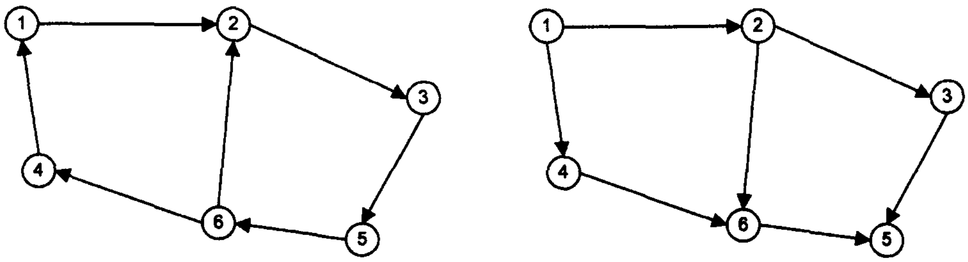
2.3.4 Cycles in Networks

A network cycle is defined as the circle in which a directed graph starts from a point and circles back to the same node (point).

A cycle exits when, starting from some node 'v', there is some path that travels through some set of nodes v_1, v_2, v_3, ..., v_k that then arrives back at 'v'.

Acyclic graph is formed by a collection of nodes and directed edges, each edge connecting one node to another, such that there is no way to start at some node 'v' and follow a sequence of edges that eventually loops back to 'v' again.

An example of cyclic and acyclic graphs are presented in figure 2.4.



(a) Cyclic Graph (b) Acyclic Graph

Figure 2.4: Cyclic and Acyclic Graphs

Figure 2.4(a) consists of two cycles formed when a path travels through the following nodes:

Cycle 1: $1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 5 \Rightarrow 6 \Rightarrow 4 \Rightarrow 1$

Cycle 2: $2 \Rightarrow 3 \Rightarrow 5 \Rightarrow 6 \Rightarrow 2$

When a graph is explored, the same path cannot be taken twice. For instance, when path $1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 5 \Rightarrow 6 \Rightarrow 4 \Rightarrow 1$ is explored, path $2 \Rightarrow 3 \Rightarrow 5 \Rightarrow 6 \Rightarrow 4 \Rightarrow 1 \Rightarrow 2$ is considered the same path but with a different starting node.

The cycles may violate some conditions in WDN analysis. Analysis of water distribution system includes determining quantities of flow and head losses in the pipelines. In any pipe network, the following two conditions must be satisfied:

1. The algebraic sum of pressure drops around a closed loop must be zero; i.e. there can be no discontinuity in pressure.
2. The flow entering a junction must be equal to the flow leaving that junction; i.e. the law of continuity must be satisfied. This is shown in figure 2.5.

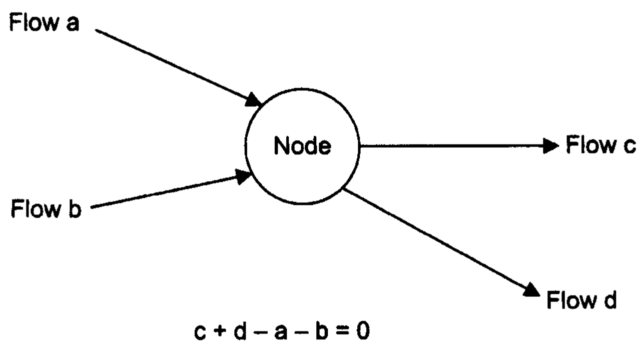


Figure 2.5: Flow Conservation Concept at Node

The flow conservation equation can be written as:

$$\sum_{outflows} X_i - \sum_{inflows} X_i = 0 \tag{2.1}$$

Where X_i is the i^{th} node in the system.

In this work, the flow quantities through the piping system are subject to a decision rule (this decision is explained in details in section 5.3.3). This rule is violated with the presence of these cycles. Therefore, cycles in water networks need to be detected and removed from the solution. This will produce a directed acyclic network (graph).

The detection of cycles in graphs are discussed in section 2.3.5 and the technique used to remove these cycles is discussed in section 5.3.1.

Cycles in directed graphs can be removed to form acyclic graphs. Removing cycles can be implemented either by removing one of the edges forming the cycle or reversing the direction of the edge. The removed or the reversed direction edge may be randomly selected or selected according to a specific rule. The process of detecting and removing cycles is repeated until acyclic graph is produced. A basic flowchart for removing cycles is shown in figure 2.6.

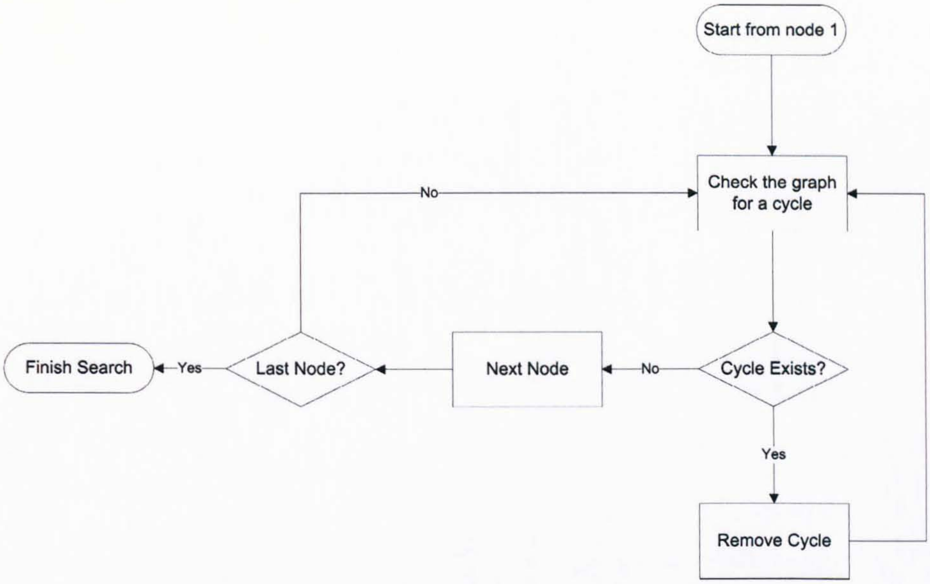
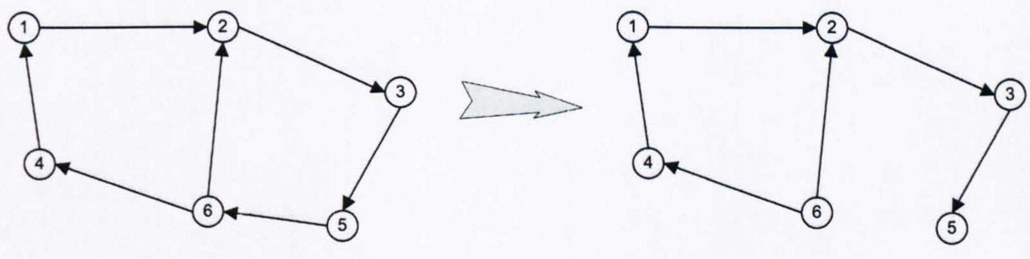


Figure 2.6: Basic Flowchart for Removing Cycles

The process start with the first node in the sequence then the graph is checked for cycles. If a cycle exists, the algorithm will remove it and keep checking the network for existing cycles from the current node and will remove them until no cycle exists. If cycles do not exist, the algorithm moves to the next node in the sequence and repeat the same procedure to remove cycles. The search is terminated when the last node in the sequence is reached.

For the cyclic graph shown in figure 2.4(a), cycles may be removed if a randomly selected edge in the cycle is removed. For example, in case edge $5 \Rightarrow 6$ is randomly selected and removed from the graph, this will produce acyclic graph as shown in figure 2.7.



(a) Cyclic Graph (b) Acyclic Graph

Figure 2.7: Removing Graph Cycles

The process of removing graph edges continues until acyclic graph is produced.

2.3.5 Depth-First Search (DFS)

Depth-First Search (DFS), also called "Backtracking", is a technique which has been used for finding solutions to problems in artificial intelligence and combinatorial theory. The DFS can be used for:

- Testing connected components of a graph.
- Finding a Spanning Tree (see appendix B.3 for Spanning Tree definition).
- Finding paths between two nodes of a graph or reporting that no such path exists.
- Finding cycles in a graph or reporting that no such cycle exists.

Several researchers have been worked on the subject of cycle detection (Tarjan, 1973; Sreedhar et al., 1996). A depth-first search algorithm by Hopcroft and Tarjan (1973), abbreviated as DFS, is used to determine whether a directed graph contains a cycle or not.

The idea behind DFS is to explore a graph from a given node by first exploring all reachable nodes (children) from that node. To avoid looping through cycles, it marks each node upon first visiting it so that the children of a previously visited node is not explored. DFS concept and advantages are presented in the following sections.

Depth-First Search Concept

DFS is a systematic way to find all the nodes reachable from a given source node. In DFS, edges are explored out of the most recently discovered node, and DFS explore as far as possible along each branch before backtracking. The process continues until all the nodes reachable from the chosen source are discovered. If any node in the graph remains undiscovered, that one is selected as the new source and DFS is repeated again. More commonly, DFS is implemented recursively. This action allows visiting all the paths that exist in a graph.

To keep track of progress, DFS marks each node. Each node of the graph is in one of three states:

1. Unvisited (Undiscovered)
2. Visited but not finished (shall be abbreviated as "VuF")
3. Visited (have found everything reachable from it) i.e. fully explored.

The DFS forms a depth-first graph comprised of more than one depth-first graphs. Each graph is made of edges (u, v) such that 'u' is "VuF" and 'v' is "unvisited" when edge (u, v) is explored. The flowchart shown in figure 2.8 demonstrates the basic idea behind DFS.

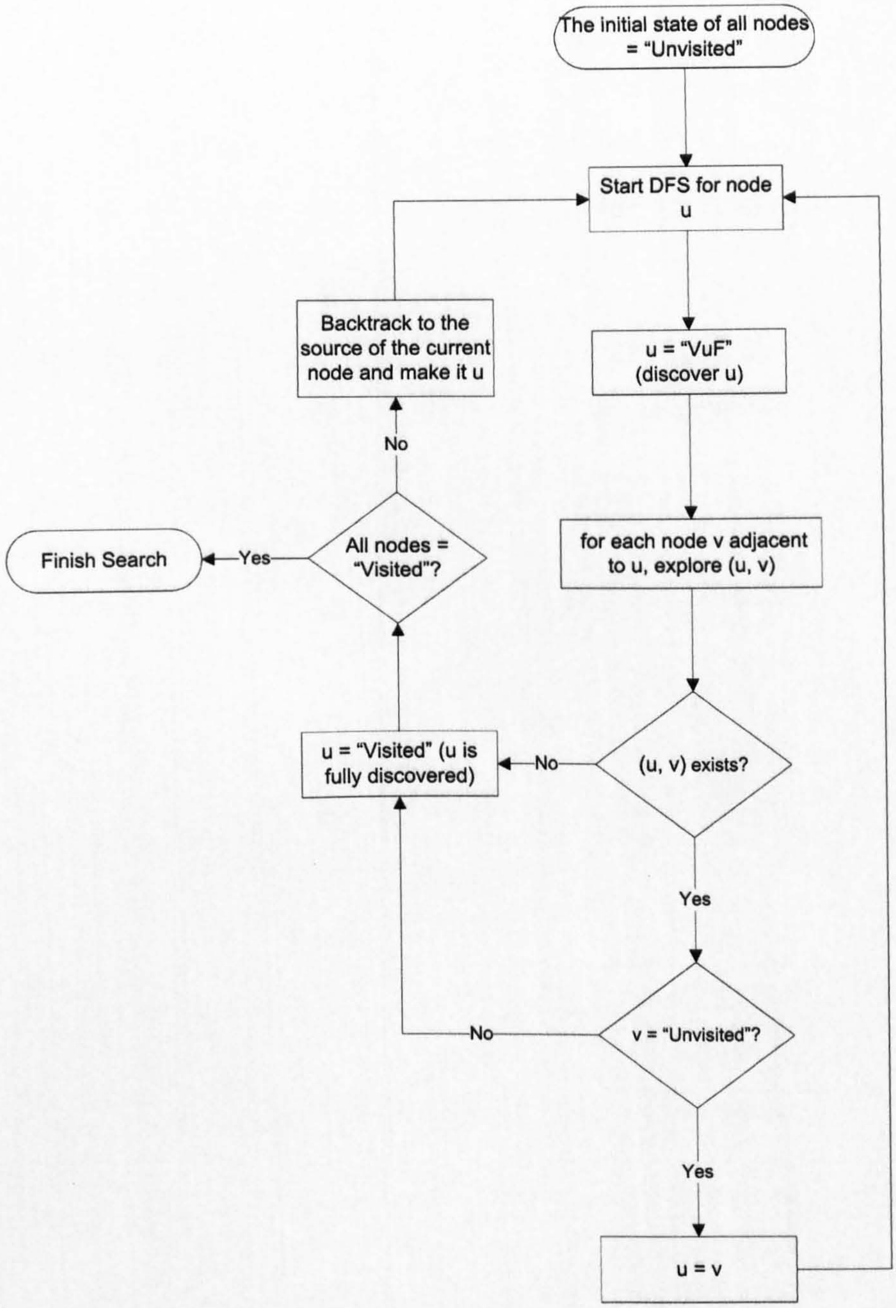


Figure 2.8: Basic Depth-First Search Flowchart

The initial state of all nodes in a network is marked as "Unvisited". Any node in the network can be selected as the source node to start with. This starting node is marked as "VuF". Now, the source node becomes current node 'u' which needs to be discovered. Then, the network is traversed by considering edge (u, v) from current node 'u'. If edge (u, v) leads to an "Unvisited" node, the algorithm makes 'v' the current node and the above routine is repeated (DFS algorithm). This procedure continues until the algorithm reaches a deadlock, meaning there is no edge is going out of current node 'u' which is a dead end. To exit this, the algorithm backtracks along the edge that brought the algorithm to node 'u' and goes back to a previously visited node 'v'. The algorithm again makes the node 'v' current node and start repeating the above procedure for any edge that missed earlier. If all of '$v's$' edges lead the algorithm to "Visited" nodes, the algorithm again backtracks to the node it originated from to get to node 'v', and repeat the computation at that node. DFS continues to backtrack the path that it has traced so far until it finds a node that has yet unexplored edges.

When DFS has backtracked all the way back to the original source node, it has built a DFS tree of all nodes reachable from that source. If there still undiscovered nodes in the network then it selects one of them as a source for another DFS tree.

A detailed example of DFS algorithm on a directed graph can be found in appendix C.

Using DFS has some advantages. Memory requirement is only linear with respect to the search graph. The reason is that the algorithm only needs to store a stack of nodes on the path from the root to the current node. If DFS finds solution without exploring much in a path, the time and space it takes will be very less.

2.4 Water Distribution Networks (WDNs)

2.4.1 An Overview to Water Distribution Networks

The primary purpose of a WDS is to deliver water to the individual consumer in the required quantity and at sufficient pressure. WDNs typically carry potable water to residences, institutions, and commercial and industrial establishments. Although few municipalities have separate distribution systems, such as a high pressure system for fire fighting or a recycled waste water system for non-potable uses, most municipal WDSs must be capable of providing water for potable uses and for non-potable uses, such as fire-fighting and irrigation.

Water systems typically consist of one or more sources of supply, appropriate treatment facilities, and a distribution system. Sources of supply include surface water, such as rivers or lakes; groundwater; and, in some instances, seawater. They usually consist of a network of interconnected pipes to transport water to the consumer, storage reservoirs to provide for fluctuations in demand, and pumping facilities.

WDNs are usually designed to meet peak demands; in parts of the network this creates low-flow conditions that can contribute to the deterioration of chemical water quality. During low demand periods these tanks are filled as water is pumped into the system. During the peak demand periods, water flows from the tanks back into the system to augment flows and maintain pressure (Loucks et al., 2005). Many systems also include auxiliary pumps which operate only during peak demand periods. The following sections demonstrate more details about WDN.

2.4.2 Water Supply Planning

Water is essential for all means of life. It is the mission of municipal and other domestic water providers to provide high quality water to their customers at a reasonable cost. Common criteria, such as water quality, delivery, technical support, price, and supply continuity, have not been adopted by municipal water providers when evaluating the reliability of the raw water supply that will be developed (Loucks et al., 2005). In addition, water has limited natural resources, however, it can be reused by water purification and disinfection processes (Swamee and Sharma, 2008).

Planning of water supply has a great importance to the economic development and the quality of life for all citizens because it shapes the future of places where the people live and work (Samuels et al., 2006). Water supply planning requires a great cooperation and coordination with the other planning disciplines such as land-use planning and transportation planning (Loucks et al., 2005). In addition, further considerations related to water supply such as urban, industrial, and agricultural water consumption should be carefully studied by the planners so that sufficient water amounts are available for each sector (Lund and Israel, 1995).

Urban water demands for domestic uses are a large share from the regional water demands and, therefore, a large water share of the regional water available. Economic potential to transfer water from water sources to demand points is an important factor to choose between water source alternatives to supply a specific area (Samuels et al., 2006; Brooks and Wolfe, 2007).

Water resource management and water supply problems become increasingly a limiting factor in the development of many urban communities (Swamee and Sharma, 2008). Water resource planning tries to solve the problem between balancing water demands and available water resources. Planners and managers are decision-makers who are responsible for solving water supply problems. Therefore, in terms of urban planning processes, local water authorities, and municipalities must consider the future improvements of WDNs and waste water over a specified planning period which indicates a significant challenge to water authorities. Future improvements may include (Loucks et al., 2005):

- Upgrading water treatment plants,
- Searching new water resources,
- Amending pipelines, and
- Constructing new water tanks.

The whole improvement system should be able to supply sufficient quantities of water and solve waste water system problems.
In addition, new approaches for long term water planning are required for sustainability (Gleick, 1998). There are many factors affecting the sustainability of WDN, however, the key ones are the use of energy and the use of water as a resource (Savic and Walters, 1997*a*). This means that sustainable water supply systems should be designed and operated so as to:

- Minimise energy use;
- Minimise the number and consequences of pipe failures;
- Make most effective use of the existing assets; and
- Still meet customers' needs in terms of water quality and quantity.

Water supply planners should estimate current and future water demands and develop water resource plans to ensure a secure, efficient and sustainable service provision. Water demands can be defined as the total amount of water required for all potential uses of water resources within a given area. The demand of water resources may include public supply for domestic, municipal, commercial, agricultural, industrial and mining (Samuels et al., 2006; Brooks and Wolfe, 2007). The demand is seasonal and highly influenced by the weather (Brooks and Wolfe, 2007).

The quantity of the delivered water is related to the technical constraints such as pumping and pipe sizes. In addition, transferring clean water and monitoring water quality is very important for health and safety. Reliability tests can be achieved by undertaking regular tests for chemical and physical characteristics of supplied water at the points of use. Furthermore, reliability constraints usually emerge the designs of looped networks to ensure redundancy (Simpson et al., 1994; Halhal et al., 1997; Skok et al., 2002; Schütze et al., 2003; Prasad and Park, 2004). If one solution to a problem is dominating the other solution, a ranking method can be used to emphasise the dominated objective in the solution (Prasad and Park, 2004).

Finally, planning of WDN improves the quality of life for all citizens. The overall goal of planners and managers is to provide a reliable and inexpensive supply of water with an assurance of water quality. The optimisation of WDNs is discussed in chapter 3.

2.4.3 Water Distribution System Components

Water utilities construct, operate, and maintain water supply systems. The basic function of these water utilities is to obtain water from a source, treat the water to an acceptable quality, and deliver the desired quantity of water to the appropriate place at the appropriate time. WDSs are made up of components that connect source sites to demand areas. A typical water distribution system consists of four main components (Schütze et al., 2003; Bhave, 2003; Loucks et al., 2005; Swamee and Sharma, 2008):

1. Water collection points (source sites).
2. Water treatment plants and distribution storages. Treatment plants and water storages could be pumping stations, which depends on the topography.
3. Water transport facilities (Water transmission mains) such as canals or main pipes from source sites to water treatment plants and storage tanks.
4. A water distribution network to transfer the treated water to demand areas.

The components of the WDN may be different from a region to another according to the nature and the purpose of the system. For illustration, piping systems used in water supply systems can be classified in several categories such as transmission lines, in-plant piping systems, distribution mains and service lines (Walski et al., 2003; Bhave and Gupta, 2006). In this thesis, water collection sites, pumping mains and water distribution system are considered.

A particular WDN is unique in source, layout, and topography of the service area, pipe materials, and connections at user points. The layout of a WDN is highly constrained by (Savic and Walters, 1997*b*; Afshar et al., 2005):

- The existing patterns of streets and highways.
- Buildings.
- Existing and planned subdivisions of urban areas.
- Possible locations of water tanks.
- Location and density of demand areas.

The distribution system may also include pumping stations in order to pump water from intake points to demand points. Although the users within a community may be close to each other, communities themselves are spread apart and the water sources may also be far away from the demand areas.

The choice between a pumping and a gravity system or a combination between them depends on the topography. For instance, pumping stations should be established if the demand areas are higher than the source or tank locations so that the tapping points at different elevations can get water.

It may also be necessary to provide pressure reducing valves to the lower elevation areas. Furthermore, check valves may be necessary to maintain the selected direction of the flow. Valves have the ability to shut off the pipes during maintenance and replacement periods, reduce pressure, maintain flow in selected direction, allow air to enter pipelines while emptying, and release air from pipelines during filling.

Pipes are the most common means to transport water and link the network components because they are available commercially with uniform diameters. Pipes may be made of cast or ductile iron, steel, concrete, pre-stressed concrete, asbestos cement or PVC (Bhave and Gupta, 2006). Finally, meters are can be attached to the system to measure the water flow among different zones throughout the WDN.

2.4.4 Water Distribution System Features

A WDS consists of a network of pipes, reservoirs, pumps, valves, and other hydraulic elements. Its purpose is to supply good quality water to customers within specific pressure levels under various demand conditions.

A water network is typically represented as a two-dimensional plan of existing and/or potential pipelines. Individual pipes are linked together to form pipelines, which may meet at nodes (or junctions). Although water may exit the pipeline at any point along its length via service lines, demand is usually grouped at the nodes (also known as demand nodes). Pipelines may also be connected to reservoir, pumps, and valves. A water source is usually connected to one pipe. A reservoir has two pipes, one to refill it and one for the outflow. A simple WDN is shown in figure 2.9, including a reservoir with a pump, a tank, six numbered junctions connected by pipes, and a valve.

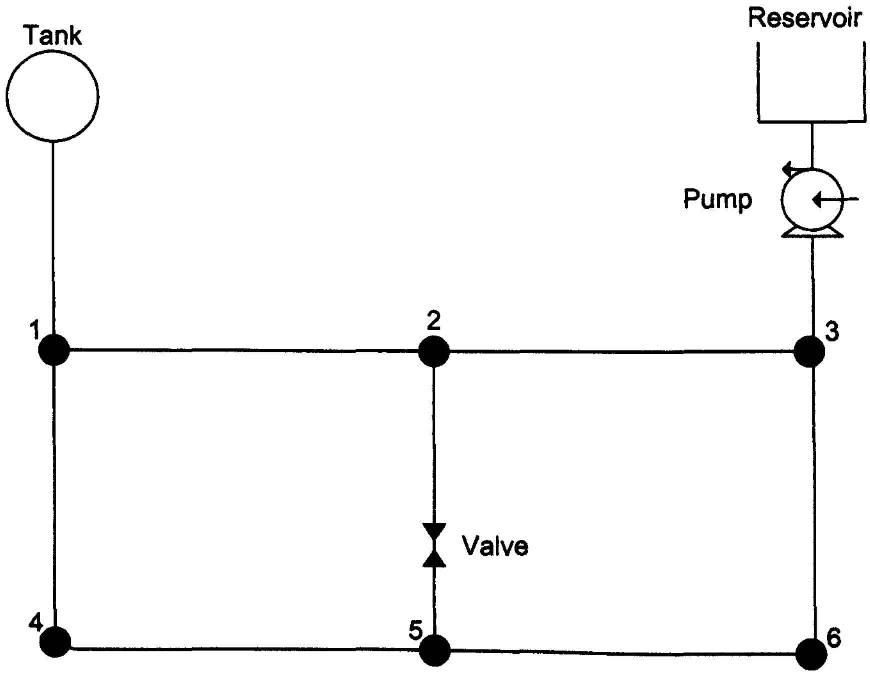


Figure 2.9: A Typical Water Distribution Network

Pipes are ordinarily straight and cylindrical, both because this is the easiest and most reliable way to manufacture them, and because a cylindrical section is best suited to handle fluid pressure and makes the most economical use of pipe material.

The size of a new pipeline may typically be one of a discrete set of commercially available pipe sizes, each associated with a different cost. In addition, existing pipes in a system under rehabilitation may undergo several operations including, cleaning, parallel pipe installation (duplication), replacement, or removal.

Tanks maybe used as balancing agents to provide additional flow during times of high demand, and fill during periods of low demand. Tanks also assist in providing consistent pressures across the WDN. They may be placed at multiple points in a system, and come in a variety of sizes and installation costs.

Pumps may also be placed throughout a system to add energy where necessary, though this is typically near the water source in the form of a pump station. There are various pump types, each with different installation and running costs. Pumps are also associated with operating curves with different wire-to-water efficiencies at different pressure and flow conditions.

Valves are essential for the safe functioning of the pipeline and pump sets. These include air valves, non-return valves, pressure relief valves, ...etc. Air valve allows trapped air to escape from the pipe line and ensures steady flow of water. Non return valve is fitted immediately after the pump. This ensures safety from water returning with high velocity due to unforeseen stoppage of power failure.

The optimisation of WDNs proceeds by considering alternative sizes for, and operations on, pipelines and other system elements and, for each network configuration, calculating the hydraulic properties of the network such as flow and pressure values. In the search of configurations, each system element would take on each of its possible attribute values, generating multiple combinations. Combinations grow exponentially as the number of network elements increase (the search complexity is $f(\alpha^{\kappa})$, where 'κ' is the number of formative elements and 'α' is the number of design options for each element). The optimisation of WDNs are discussed in details in chapter 3.

It is assumed that a fixed inflow and outflow to the system is known in advance. If a system can satisfy peak demand, then it will also be able to cater for reduced demand, but care must be taken to respect maximum pressure limitations, justifying the need for a static zero-flow simulation. If tanks are to be designed, then it is essential to conduct an extended period analysis, simulating the tank inflows and outflows, in order to design for effective tank operation (Walski et al., 2003).

Each component in a WDN is associated with an elevation. Demand nodes are also associated with lumped demand quantities (or loads), expressed in terms of flow out of the network. Reservoirs and tanks (sources) are associated with a volume of water in storage, a potential energy expressed in units of pressure that depends on the source elevation, and a maximum flow rate at which they can feed the network.

2.4.5 Types of Water Distribution Networks

Walski et al. (2003), Bhave and Gupta (2006) and Committee on Public Water Supply Distribution Systems (2006) classified WDNs according to their layout into three types:

2.4.5.1 Serial Networks

Serial networks are the simplest type of distribution networks which do not have branches and loops. They usually have one source and one or more intermediate nodes and one sink (See figure 2.10).

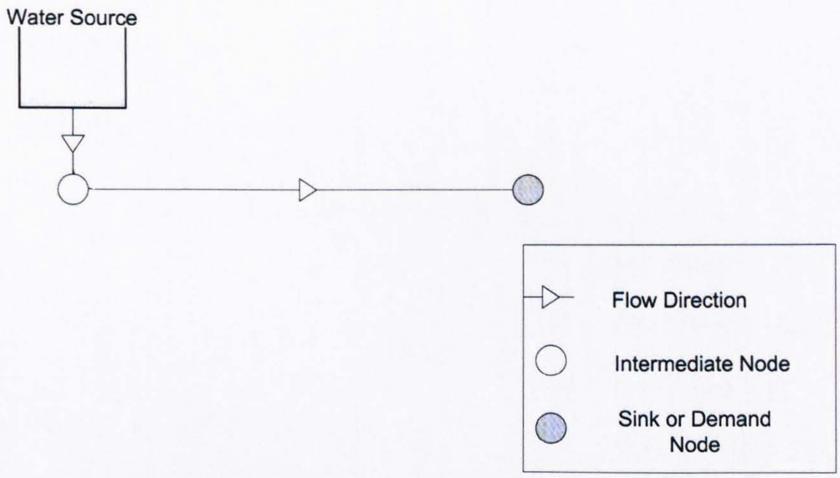


Figure 2.10: A Typical Serial Network

2.4.5.2 Branched Networks

Also called "Dead-end Networks", they are similar to that of a tree branches and have no loops. A tree branch exists when smaller pipes branch from larger pipes throughout the service area. They consist of several serial networks and usually have one source and one or more of intermediate nodes. Branched networks have many sink nodes. This type of network is usually used in rural areas, small communities and also for industrial water supply purposes. Figure 2.11 is a typical example of branched networks.

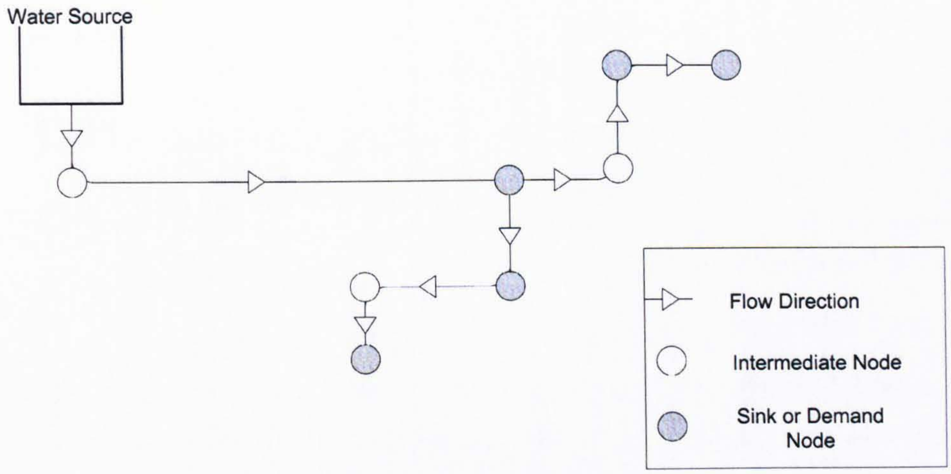


Figure 2.11: A Typical Branched Network

2.4.5.3 Looped Networks

They are connected pipe loops throughout the service area as shown in figure 2.12.

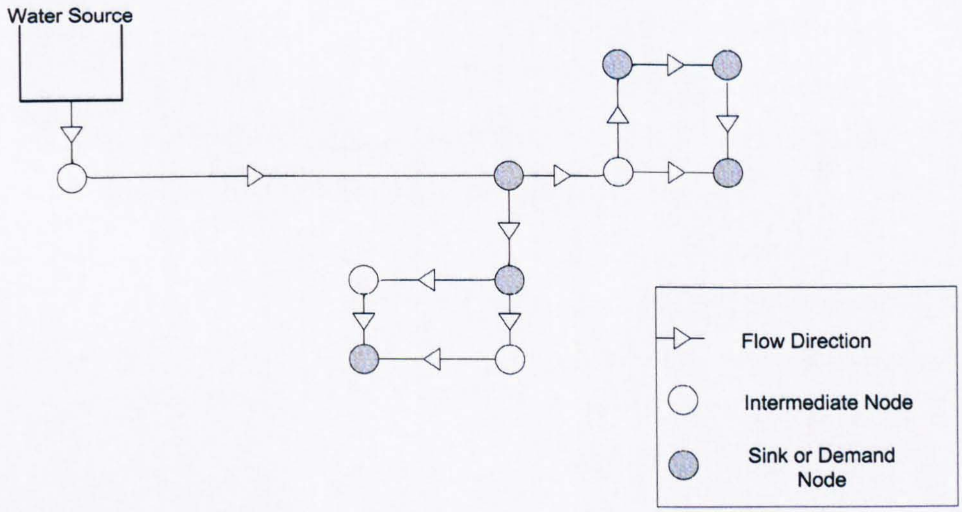


Figure 2.12: A Typical Looped Network

They take the water from one or several water sources. A looped network has several pathways that the water can flow from the source to the consumer and must have at least one sink. These types of networks are widely used in large municipal and metropolitan areas.

Although looped networks are more costly than branched ones, they provide a high degree of reliability because of the following reasons:

1. Water can be provided to the demand points through a number of pipes so that the required flow rates can be maintained at the demand points.

2. A link break or closing a pipe for replacement on a looped network can be isolated with little impact on consumers. Alternative routes from source nodes to demand nodes are available and only consumers between the closed valves will be affected.

3. Looping behaviour of the network keeps water moving to reduce some of the problems associated with the water stagnation (slow movement of water) during normal working conditions especially in dead ends. Additionally, it increases fire fighting capabilities.

Most of urban water supply systems are a combination of both looped for reliability and branched portions are for infrastructure cost savings. In some cases, a distribution network may contain a combination of the three types of networks (Walski et al., 2003; Bhave and Gupta, 2006; Committee on Public Water Supply Distribution Systems, 2006).

Distribution networks consist of many loops modelled by non-linear equations and complex design criteria. For example, the Hazen-Williams equation (see section 2.4.7.6) is widely used in pressure pipes to find the velocity or the discharge at any node (Walski et al., 2003; Bhave and Gupta, 2006; Shihu et al., 2010).

2.4.6 Water Demand

It is the amount of water required by the customer. Generally, water demands are generated from residential, industrial, commercial development, community facilities, fire flow demand, and account for system losses (Swamee and Sharma, 2008).

The estimation of the water demand is an important part of the WDN design methodology. It is difficult to estimate water demand accurately because a number of factors affect the water demand such as economic and social factors, land-use, and pricing. Demand estimation refers to the process of assigning water demand quantities (or loads) to new or existing WDNs, and potentially predicting future demands, thus determining the required system capacity for WDN design or rehabilitation. To estimate water demands, comprehensive studies should be conducted.

The residential forecast of future demand can be based on number of houses, census record, and population projections.

The residential and commercial facilities have a wide range of water demand. This demand can be estimated based on historical data from the same system or from other similar systems.
This demand is the water required to meet the non-emergency needs of users in the system. This demand type typically represents the metered portion of the total water consumption.

System losses is the portion of total consumption that is "lost" due to system leakage, theft, un-metered usage, irrigation of public parks, or other causes. Therefore, the estimation of system losses is difficult.

Fire flow demand is a computed system capacity requirement for ensuring adequate protection is provided during fire emergencies. The fire flow demand can be estimated using formulas or local guidelines.

WDNs are subject to water balance constraints, which means that all input water to the system must equal all water exiting the system (accounting for changes in storage). The water resources required to satisfy consumer demand are higher than the consumer demand itself. An estimate of demands must consider:

1. Ordinary consumer demands (including non-revenue consumers).
2. Water losses, such as leakage, unauthorised usage, and metering inaccuracies.
3. Fire flow demands.

A simplified water balance constraint for a given period may be expressed as:

$$V_{inflow} = V_{demand} + V_{losses} + V_{fireflow} + \Delta V_{storage}$$

where V_{inflow} denotes the total input amount, V_{demand} denotes the sum of water demands, V_{losses} denotes the total losses, $V_{fireflow}$ denotes the total water used for fire-fighting, and $\Delta V_{storage}$ denotes the change in storage volumes.

2.4.7 Flow Hydraulic Principles

Pipe flow principles covers the hydraulics and the basic principles of flow such as continuity equation, equations of motion and Bernoulli's equation for close conduit. Another important area of pipe flows is to understand and calculate resistance losses and form losses due to pipe fittings (e.g. elbows, valves, expanders, and reducers).

The flow hydraulics of fluid through a pipeline is complex in nature and needs particular consideration in head loss computations. Some important concepts in hydraulics are presented in this section.

2.4.7.1 The Pressure

Pressure, 'p', can be defined as a force applied normal or perpendicular to a body that is in contact with a fluid (Walski et al., 2003). In the English system of units, pressure is expressed in pounds per square foot (lb/ft^2), but the water industry generally uses lb/in^2, abbreviated as "psi". In the SI system, pressure has units of N/m^2 , also called "Pascal".

Specific weight, 'γ', is the weight per unit volume of the fluid, related to the fluid density, 'ρ'.

$$\gamma = \rho g \tag{2.2}$$

Where $g = 9.81$ denotes standard gravitational acceleration. For water at 4^oC, its specific weight is 9810 N/m^3 and its density is 1000 kg/m^3.

Pressure varies with depth; for instance, for fluids at rest, the variation of pressure over depth is linear and is called the "hydrostatic pressure distribution". Pressure head (which will be referred to as "head" throughout this thesis) is the energy resulting from the water pressure (see figure 2.13). The head 'h' is given by:

$$h = \frac{p}{\gamma} \tag{2.3}$$

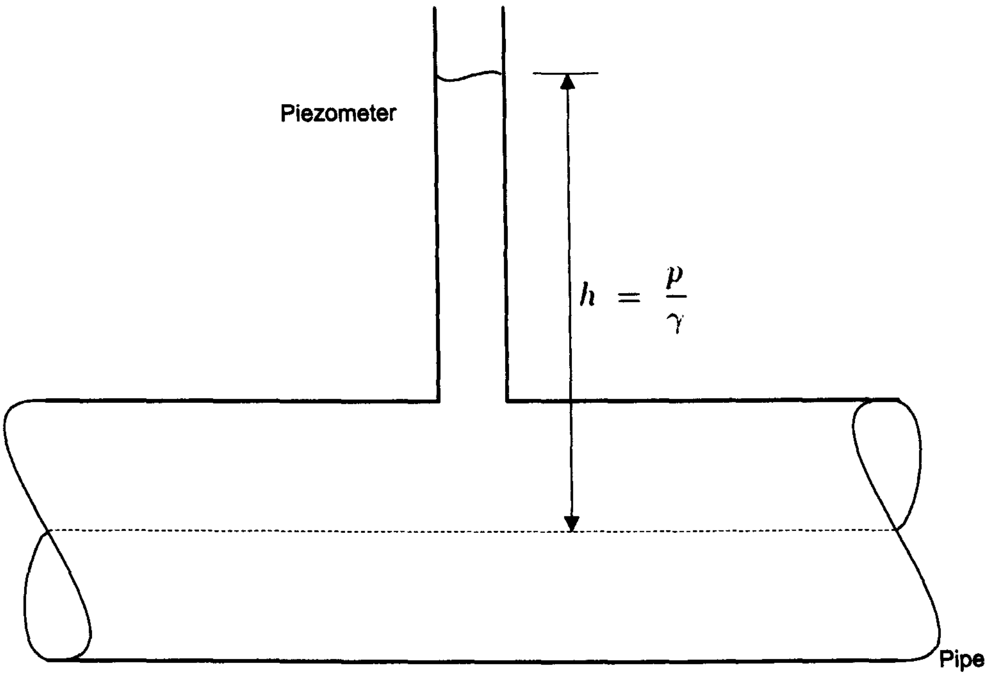


Figure 2.13: Piezometer Attached to a Pipe Showing the Head Concept

For a static fluid on a horizontal plane, the pressure, 'p', everywhere on this plane is constant, that is:

$$\frac{p}{\gamma} + Z = constant$$

$\frac{p}{\gamma} + Z$ is known as the "*hydraulic head*", which is constant throughout any incompressible static fluid. 'z' is the elevation above an arbitrary datum. Hydraulic head is a measure of the total energy per unit weight above a datum. It is measured in units of height (m). Pressure and elevation at two different points, 1 and 2, in the fluid must satisfy:

$$\frac{p_1}{\gamma} + Z_1 = \frac{p_2}{\gamma} + Z_2$$

Hydraulic head may be used to determine a hydraulic gradient between two or more points. Fluid always flows down a hydraulic gradient from a higher to a lower total head (hydraulic head plus velocity head).

2.4.7.2 The Discharge (Flow-Rate)

Discharge or flow rate, 'Q', is the volume rate of flow that passes a given section in a flow stream such as a pipe section. It has SI system units of m^3/s and English system units of ft^3/s. In most cases, hydraulic models deal with the average velocity in a cross section of a pipeline, the flow rate can be found by using the following formula:

$$Q = V\,A$$

Where V is the average velocity and A is the cross sectional area of the pipe.

The cross sectional area of a circular pipe can be directly computed from the diameter 'D', so that the discharge equation can be rewritten as:

$$Q = \frac{\pi}{4}\,D^2\,V \tag{2.4}$$

2.4.7.3 Hydraulic Energy Concept

The energy at any point within a hydraulic system is often expressed in three parts, as shown in figure 2.14:

1. Pressure head (pressure energy) $= \frac{p}{\gamma}$
2. The amount of energy depends on the fluid's movement or velocity head (kinetic energy) $= \frac{V^2}{2g}$
3. Elevation (potential energy) $= Z$

Where: p = pressure (N/m^2, lbs/ft^2)

γ = specific weight (N/m^3, lbs/ft^3)

Z = elevation (m, ft)

V = velocity (m/s, ft/s)

g = gravitational acceleration constant (m/s^2, ft/s^2)

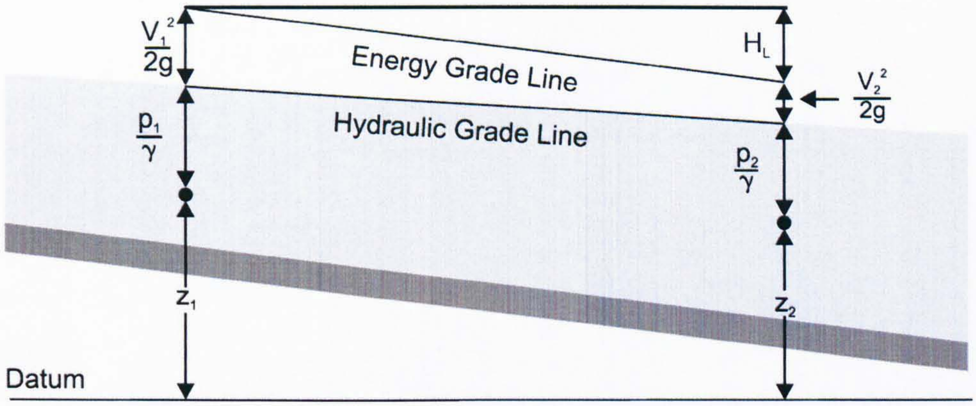


Figure 2.14: The Energy Principle (Source: (Walski et al., 2003))

In a hydraulic system, a fluid can have all three types of energy associated with it simultaneously. The total energy associated with a fluid per unit weight of the fluid is called "total head" (H). the units of total head are feet or meters.

$$H = Z + \frac{V^2}{2g} + \frac{p}{\gamma} \tag{2.5}$$

Each point in the system has a unique head associated with it. A line plotted of total head versus distance through a system is called the energy grade line (EGL). The sum of the elevation head and pressure head yields the hydraulic grade line (HGL), which corresponds to the height that water will rise vertically in a tube attached to the pipe (piezometer) and open to the atmosphere. Figure 2.14 shows the EGL and HGL for a pipeline.

2.4.7.4 The Head Losses

Energy (or head) losses (H_L) in a system are due to a combination of several factors. The primary cause of head loss is because of two mechanisms:

1. The internal friction between fluid particles travelling at different velocities and the pipe walls.
2. Localised areas of increased turbulence and disruption of the streamlines, such as disruptions from valves and other fittings in a pressure pipe, or disruptions from a changing pipe section.

Head losses along the pipe wall are called "friction losses" due to friction, while losses due to turbulence within the bulk fluid are called "minor losses".

The rate at which energy is lost along a given length of channel is called the "friction slope", and is usually presented as a unit-less value or in units of length per length (ft/ft, m/m).

In addition to pressure head, elevation head, and velocity head, energy may be added to a system by, for example, a pump (h_p), and removed from the system by friction or other disturbances. These changes in energy are referred to as "head gains" and "head losses", respectively. Because energy is conserved, the energy across any two points in the system must balance. This concept is demonstrated by the energy equation:

$$Z_1 + h_1 + \frac{V_1^2}{2g} + H_G = Z_2 + h_2 + \frac{V_2^2}{2g} + H_L \tag{2.6}$$

Where H_G is the energy gain and H_L is the head loss.

2.4.7.5 The Friction Losses

Water flow in a pipe network satisfies two basic principles, conservation of energy and conservation of mass. Conservation of energy describes the relationship between the energy loss and pipe flow while the continuity equation is a statement of the conservation of mass in a system.

There are many equations that approximate the friction losses associated with the flow of a liquid through a given section. Commonly used methods include:

- Manning's equation
- Chézy's (Kutter's) equation
- Hazen-Williams equation
- Darcy-Weisbach (Colebrook-White) equation

These equations can be described by a generalised friction equation (Walski et al., 2003):

$$V = K\,C\,R^{\alpha}\,S^{\beta}$$

Where: V = average velocity of flow

K = factor to account for empirical constants, unit conversion,... etc

C = flow resistance factor

R = hydraulic radius = $\frac{A}{P}$

where:

A = the cross sectional area of the flow (m^2)

P = the wetted perimeter of the pipe (m).

S = friction slope (head loss per length of pipe) = $\frac{h_L}{L}$; where h_L is the head loss and L is length of the pipe.

α & β = exponents

The empirical Hazen-Williams equation is most frequently used in the design and analysis of pressure pipe systems (Walski et al., 2003; Bhave and Gupta, 2006; Shihu et al., 2010). This is equation is used in this work.

2.4.7.6 Hazen-Williams Equation

The empirical Hazen-Williams equation is used frequently in water supply engineering for flow of water through pipes. The Hazen-Williams equation is (Walski et al., 2003):

$$V = 0.849\, C_{HW}\, R^{0.63}\, S^{0.54} \tag{2.7}$$

Where: V = average velocity of flow (m/s)

C_{HW} = Hazen-Williams coefficient

R = hydraulic radius (m)

S = slope of the energy line = $\frac{h_L}{L}$

For circular pipes the velocity can be obtained from equation 2.4 and $R = \frac{D}{4}$. The discharge formula can written as:

$$Q = 0.285\, C_{HW}\, D^{2.63}\, S^{0.54} \tag{2.8}$$

By substituting $\frac{h_L}{L}$, equation 2.8 on simplification becomes:

$$h_L = \frac{10.68\, L\, Q^{1.852}}{C_{HW}^{1.852}\, D^{4.87}} \tag{2.9}$$

Where L and D are in meters and Q is in m^3/s.

2.4.7.7 The Continuity Equation

Conservation of mass in distribution networks states that, for a steady flow at a junction, the sum of mass flow rate entering a junction must equal to the sum of the mass flow rate leaving it. This relationship holds for the entire network and for individual nodes. A node is included in a network model at (1) a demand location and/or (2) a junction where two or more pipes combine. One mass balance equation is written for each node in the network as:

$$\sum_j Q_{in} - \sum_j Q_{out} = Q_{demand} \tag{2.10}$$

Where Q_{in} and Q_{out} are the flows in pipes entering or exiting the node j and Q_{demand} is the user demand at that location. These demands are uncertain since they are estimated from the local user base that cannot be predicted exactly since they vary nearly continually. In addition, the demand is typically represented as a lumped demand for users near the node.

2.4.8 Pumping Systems

A pump is a device to which mechanical energy is applied and transferred to the water as total head. The head added is called 'pump head' and is a function of the flow rate through the pump (Walski et al., 2003).

Pumping systems are essential to supply water at required pressure and quantity where topography is flat or undulated. External energy needs to be added to a hydraulic system to overcome elevation differences, friction losses, and minor losses. Pumps have two primary purposes: (1) adding energy to the WDN in order to overcome friction and gravity so as to satisfy consumer hydraulic requirements, and (2) filling tanks. Suitable pumping system need to be designed for this purpose.

Pump design refers to the process of determining the capacity and number of pumps in order to satisfy the hydraulic requirements of a WDN. These pumps would typically be located in a pumping station, positioned after the water treatment facility. There are typically at least two pumps in a pumping station (perhaps three or four in large systems) in order to provide a level of redundancy. This arrangement means that if the duty pump fails to start, the standby can be used to avoid production loss. These are often sized similarly in order to simplify maintenance (Mays, 2000).

To avoid accumulation of very old water in the standby system, the allocation of the duty and standby pumps should be alternated from time to time. This will also ensure that the standby pump remains functional and will spread the wear over both (or all) pump sets. Where possible pumps should start up slowly to reduce the scouring effect of a sudden increase in water velocity that may lead to dirty water.

Pump capacities and number may be adjusted during the course of optimisation in order to achieve feasible pressures. Pumps should be able to satisfy a peak hour demand scenarios.

While pumps are a common component of systems, where the source is at a sufficiently high elevation, the system may not have any pumping. Booster pumping may be required to provide adequate pressure in certain portions of a system when there is significant variation in elevation or use rate (Mays, 2000). However, pressure reducing valves serving just the opposite purpose may be needed.

A pump station is installed where water must be lifted from a low level to a high level. The flow may also be pressurised to a higher hydraulic grade instead of installing a high level reservoir. Typically, all pumps used to lift water more than a few metres are centrifugal pumps (Walski et al., 2003).

Incorporating pump design as a cost objective requires pump capital costs and the present value of energy costs over the WDN lifetime. This requires simulation of pumping over the course of an extended period analysis (typically maximum day or maximum week), in order to determine pump schedules.

2.4.9 Modelling of Water Distribution Networks

Simulations are mathematical methods which approximate the behaviour of the real WDNs. Simulation and modelling are widely used by water authorities around the world for planning and management of water supply systems. These water supply systems consist of 'nodes' such as reservoirs, demand centres, diversion weirs, stream junctions and pipe junctions, and 'carriers' such as rivers and pipes. They supply irrigation and urban water demands, meet environmental flows, and provide releases for hydro-power generation and other uses.

Real-world distribution networks are very complex and difficult to design because the real-world involves a number of different constraints such as terrain, buildings and streets (Savic and Walters, 1997*b*; Afshar et al., 2005). Therefore, mathematical models are required to analyse the network. In addition, gathering data is required to analyse the network for system designs and operations.

Modelling becomes one of the important parts of designing and operating WDNs. Computer models can be used to find the relationship between the hydraulic flow and the head using numerical methods. These methods can be used to assess the reliability of the network to deliver the required amounts of water to demand points. For example, EPANET software is an example of one of the hydraulic simulators which checks the hydraulic constraints in WDN. The model should be designed to include the important elements so that the results obtained from the analysis describe the behaviour of the real-world network (Bhave and Gupta, 2006).

Simulations are increasingly used to analyse many design problems in urban water systems. These models can be represented in graphical user interfaces (GUI) environment which simplify the use of models and support further analysis.

The selection of specific models depends on the research objectives and requirements. For example, Savic and Walters (1997*a*) and Dandy and Engelhardt (2006) demonstrated mathematical models for both cost and reliability criteria of a system. The two criteria (cost and reliability) are often conflicting because, for instance, cost minimisation tends to eliminate any redundancy in the networks which may lead to less reliable solutions.
Other models have been presented to joint layout and pipe size or cost and pipe size optimisation for a given level of reliability (Eusuff and Lansey, 2003; Afshar et al., 2005). These studies emphasised that the efficiency and the accuracy of the mathematical models used to model systems' behaviour have an important role in the design and the operation of the network systems.

Identifying basic modelling tools are necessary for analysing the networks. In addition, these tools support decision-making for efficient planning, design and operation of WDNs (Savic and Walters, 1997*b*).

Basic modelling components should be identified and studied to allow the complex WDNs to be examined under various operating conditions. Walski et al. (2003) demonstrated some of the modelling components:

- Fluid Properties (e.g. density, viscosity, compressibility).
- Friction losses and minor losses.
- Energy gains Pumps (e.g. pump operating point, pump flow-head relationship).
- Water quality modelling (e.g. transport in pipes, mixing at nodes).

A WDN contains various components and modelling them will indicate how these components are interconnected. For example, a node is a feature of a specific location within the system and the links represent the relationship among nodes. Some modal elements are listed in table 2.4 (Walski et al. (2003)):

Table 2.4: WDN Modal Elements

Element	Type	Modelling Purpose
Pipe	Link	Transfers water to the system
Junction	Node	Removes demand or adds inflow from/to system
Storage Tank	Node	Provides water to the system
Pump	Node or Link	Lifts water to higher elevations
Control valve	Node or Link	Controls flow in the system

These elements should be named and labelled carefully and should include enough information to identify the element because a name that seems clear and easy to understand today may be confusing for future users.

Before modelling WDN, it is necessary to gather information about how to describe the network. To construct a model, data can be gathered from maps such as system maps, topographic maps and city maps and other existing records. Maps usually include information about existing pipelines, buildings, streets, and other city features. Records or databases are non graphical data and may contain surveys, legislations, and population. Some systems may combine both graphical and data records such as computer aided drafting software (CAD) and GIS. Other considerations regarding constructing models should be carefully studied such as model representation, tanks, node junctions, pipes, pumps, valves, and other types of controls.

To create an effective model, the engineer should study the goals of regional planning authorities, local economic development councils, local city councils, county boards, and the various budget groups associated with them. In addition to these goals, consideration must be given to demographic and transportation studies, which provide information on future locations, land-use, and future population densities. These and other items should be evaluated and incorporated into the calibrated model as future scenarios (Walski et al., 2003).

2.4.10 Summary of Water Distribution Networks

The primary objective of a water supply system is to take water from the best available source, disinfect it to ensure water of good quality, and distribute it to meet the requirements of the town or the city. Part two of this chapter started with an overview of WDNs the planning of them. Then, the WDN components, features, and the types of WDNs were presented. This chapter also summarized the hydraulic principles which are used in the design of WDNs. Pumping and demand are were also discussed. It is necessary to design a modelling system to simulate WDN parameters. To identify and solve WDN related problems, modelling the WDN was discussed.

2.5 Chapter Summary

This chapter featured a background on topics required in the development of a more realistic WDN model.

The chapter introduced planning with an overview of the need for planning with the other topics such as decision-making, sustainable development, and the use of geographic information systems in planning. Urban and regional planning fields and principles were also introduced.

In the second part of this chapter, some of the basic ideas behind graph theory and the study of network structure were demonstrated. This allows the formulation of basic network properties for the optimal design of WDNs. Two types of networks were introduced followed by demonstration of some related topics. The presence and detection of cycles, using depth-first search, in networks were discussed.

Essential topics in WDS were presented. This include WDN features and components, water demand, and pumping as well as the types of water networks and basic theory of WDN hydraulics. The WDN modelling and design process was also presented.

Chapter 3

Optimisation of Water Distribution Networks and GIS

3.1 An Introduction to the Chapter

This chapter focuses on the optimisation of water distribution networks which involves both non-evolutionary and evolutionary algorithms with special attention to GA optimisation. In addition, the chapter discusses the use of GIS in the design optimisation of water distribution networks.

The chapter consists of a short introduction to the process of optimisation and a review of the WDN optimisation techniques that have been reported in the literature. A wide literature review of using GAs in the design of WDNs is presented.

In addition to discussing the use of GIS as a tool to assist in the design of WDNs, this chapter also describes a framework in which GAs can be integrated with GIS to assist in the design of water distribution networks.

3.2 An Overview to Optimisation

Optimisation can be defined in many ways. It may refer to the process of finding the values of decision variables to provide the maximum or minimum of one or more desired objectives (Weise, 2009). Pelikan (2005) defined the optimisation as a performance process to provide potential solutions to a problem. Others referred to optimisation as the process of trying variations on an initial concept and using the information gained to improve the idea (Fletcher, 2000; Andrew et al., 2009).

People use optimisation for simple things, such as travelling from one place to another, as well as for other decisions such as finding the best combination of study, job and investment. In addition, optimisation finds many applications in engineering, science, business and economics.

Optimisation techniques can be used to help improving the efficiency of design and operation of many applications such as manufacturing and engineering. It has attracted the interest and attention from many civil and construction engineers in academia and industry for the past decades (Fonseca and Fleming, 1993; Haupt and Haupt, 2004; Garca-Pedrajas et al., 2006).

Perhaps, the most common use is as an aid to decision-making. An optimisation model forms part of a larger system, which people use to help them make decisions. The task of decision-making entails choosing between various alternatives. The choice is governed by the desire to make the best decision. The measure of goodness of the alternatives is described by an objective function or performance index. Optimisation theory and methods address the selection of the best alternative through a given objective function (Chong and Źak, 2001).

A number of meta-heuristic algorithms exist to find good possible solutions to the optimisation problems. Heuristic algorithms help to decide which one of a set of possible solutions is to be examined next. Many of these algorithms are based on the fundamental principles of the natural evolution such as population, selection and mutation. Some widely used algorithms are genetic algorithms (proposed by Holland (1975) and Goldberg (1989)), particle swarm optimisation introduced by Kennedy and Eberhart (1995), and simulated annealing (proposed by Kirkpatrick (1984)).
Linear programming (Dantzig, 1998) and tabu search (proposed by Glover (1977)) are other types of optimisation techniques. These are non-heuristic mathematical optimisation techniques.

Computers are able to evaluate candidate solutions hundreds of thousands times faster than the manual process. They are used to assist reducing the time of the optimisation process since the variable influencing the idea can be input in an electronic format.

3.3 Optimising Water Distribution Networks

Safe piped potable water supply is one of the basic human needs (Loucks et al., 2005). With the increasing population growth in cities, the stress on water infrastructure has increased phenomenally (Quesne et al., 2007). The purpose of a WDN is to supply water to consumers with the required pressure under various demand conditions.

A WDN requires extensive planning at the design stage and maintenance during operations to ensure that good service is provided to the customers in a reliable and economical way. The planning and maintenance of a WDN involve determination of demand, optimal network layout, optimal sizes of other system components such as pipes, pumps, valves and optimal operation conditions. Water quality should be maintained at all points in the system, including the point of delivery. The huge range of possible combinations of pipe materials, routes, diameters, pumping station locations and capacities makes the task of engineers and managers involved in the design optimisation very hard. In addition, complexity of the operational parameters such as pressure zone boundaries, control valve settings, and pump operating schedules has a significant impact on service quality and costs (Mays, 2000; Bhave, 2003).
In real-world applications, additional WDN design features may add more complexities to the design of WDN. These complexities may include:

- Multiple loadings: in reality, various demand areas (e.g. industrial, residential, retail) have different peak values at different times of the day. Therefore, in the design of a WDN, it is necessary to consider several loading conditions (Bhave, 2003; Swamee and Sharma, 2008).

- Uncertainty: real-life problems involve uncertainty in one way or another. There could be uncertainty in measurement, parameter estimation and process choice in a model. Developing methodologies, which take uncertainty into account when predicting the behaviour of a system is of a great importance (Babayan et al., 2005). Kapelan et al. (2003) reported a technique for accommodating uncertainty in design constraints when using GA optimisation technique. This study incorporates the uncertainty in nodal demands and pipe roughness conditions in a multi-objective optimisation model in the design of a WDN.

- Network operations: WDNs must be operated properly so that it performs at an acceptable level of service. The water utility industry investigates the integration of multi-purpose control systems in an effort to reduce operating costs and provide more reliable operations (Mays, 2000).

  Boulos et al. (2001) presented a mathematical model for determining least-cost pump operation. The proposed operational management model aimed to meet target hydraulic performance requirements of the WDN for a given time period (normally 1 day). The proposed model uses GAs to determine the least-cost pump scheduling / operation policy for each pump station in the water distribution system while satisfying target hydraulic performance requirements. The model was useful and the results obtained indicate that the model can reduce the cost of energy consumed for pumping in a complex WDN while maintaining satisfactory levels of service.

- Water quality: To meet pressure and flow requirements, quality constraints and the hydraulic laws that govern the network behaviour should be considered. The objective of meeting water quality conflicts with meeting pressure and flow requirements in which pollutants need to be carried in the pipes and mass conservation is maintained at the nodes. The physics of the system should include the continuity of water and of pollutant mass at nodes.

  Ostfeld and Shamir (1996) demonstrated a methodology which integrates the optimal design and reliability of a multi-quality water supply system. In multi-quality water distribution systems, water of different qualities is taken from different sources, treated, conveyed, and supplied to consumers. Their system was designed to sustain some predefined failure scenarios such that any component failure should maintain the desired levels of service in terms of pressure, quality and quantity of supplied water to consumers.

- Reliability: The reliability requirement is usually addressed by considering branched or looped layout for the networks to be designed (Alperovits and Shamir, 1977; Quindry et al., 1981; Murphy et al., 1993; Wu et al., 2001; Boulos et al., 2001). The WDN reliability can be defined as the ability of a network to provide an adequate supply to consumers under various operating conditions (Xu and Goulter, 1999). The adequacy of supply is measured in terms of requirements of pressure head, water amount, and water quality. These requirements may not be fulfilled in the case of a component failure (failure of pipes, pumps and valves) and performance failure (demand on the system being greater than the design value) (Mays, 2000). In the reliability assessment, these two types of failure should be considered together because they are strongly related (Ostfeld and Shamir, 1996). Any failure or scheduled maintenance of any of the network components would lead to a part of the network being cut off from the source nodes. On the other hand, the reliability depends on the network layout in which the number of independent paths from source nodes to each of the consumption nodes is considered as a measure of reliability (Afshar et al., 2005).

– Rehabilitation: as water distribution systems age, the physical condition of the elements continually worsen. They gradually lose carrying capacity, eventually failing to fulfil their specified function of distributing potable water continuously and within prescribed pressure limits. The frequency of leaks and breaks increases, resulting not only in loss of water, but also in contamination through cracked pipes (Halhal et al., 1997). Hence, the decision process for rehabilitation, replacement, and/or expansion of existing systems to meet current and future demands forms a major area of interest (Walters et al., 1999). Rehabilitation allows a WDN to operate efficiently and economically within the defined operating requirements over an extended period. Optimising WDN rehabilitation includes water network reinforcement, expansion and reconditioning to maintain the level of service required by demand areas. In addition, it may consists of pumping installations and storage tanks Walters et al. (1999).

In order to find a low cost design in practice, experienced engineers have traditionally used trial and error methods based on their intuitive engineering sense. However, their approaches have not guaranteed optimal or near-optimal designs, which is why researchers have been interested in optimisation methods (Goulter, 1992; Weise, 2009). The rest of this section will discuss some of the mathematical optimisation and heuristic optimisation methods used in the design of WDNs.

3.3.1 Mathematical Optimisation Techniques

The earliest models for WDN design were developed for water networks in 1960s. Karmeli et al. (1968) used Linear Programming (LP) to optimise the lengths of pipes and Schaake and Lai (1969) who introduced the problem of New York City Tunnel network used Dynamic Programming (DP) to search for a optimum design of the WDN.

One of the efforts for the WDN design optimisation was described by Alperovits and Shamir (1977) who developed a linear programming gradient methodology for a least-cost implementation of a two-loop network. In this context, least-cost refers to the minimisation of the implementation cost of the network in terms of the cost of the pipes to be installed. This methodology reduces the optimisation to a series of problems regarding the possible flow routes to each point in the network. The applicability of this approach is constrained by the complexity of the network involved and most of the subsequent work has concentrated on reducing the computation complexity of optimising such networks.

This innovative approach was adopted and further developed by many researchers such as Quindry et al. (1981), Goulter et al. (1986), Kessler and Shamir (1989) and Fujiwara and Khang (1990).

Quindry et al. (1981) adopted linear optimisation gradient search based on the assumption that the ratio of the quantity flowing in a pipe to its cost is independent of the diameter. Their technique can be used to identify critical areas in the design and to suggest improvements in the design to reduce costs. This method allowed for a larger system to be considered. However, a shortcoming of this model was the inability to design and analyse a complete WDN. The limitations included the number of loading conditions that could be analysed, and the types of components that could be designed.

Goulter et al. (1986) study considered reliability constraint and assumed poison distribution for the failure of individual pipes. In addition, the study developed and assessed two quantitative approaches to the incorporation of reliability measures in the least-cost design of WDNs. The authors suggested that the minimum cost design for a given layout and single loading case should be a branched network (a network with no loops). While this may be true, loops are an essential feature of actual distribution systems as they provide alternative flow paths in the case of pipe failure or for maintenance purposes. One can achieve a degree of redundancy in pipe network optimisation by ensuring that the layout has appropriate loops and by specifying minimum diameters for all pipes.

Alperovits and Shamir (1977) study has been adopted and restated in more concise and complete form by Kessler and Shamir (1989); the decision variables in this study were (1) lengths of all pipe segments in the network and (2) flow rates and directions in all pipes. They presented a two stages method. In the first stage, parts of the variables are kept constant while other variables are solved by linear programming (LP). For a given set of flows, the corresponding sets of heads are determined by LP. In the second step, search is conducted based on the gradient of the objective function. Flows are modified according to gradient of the objective functions. Although the problem is non-linear and the gradient information may not be attained in many instances, they solved the problem by linearising the formulation.

Fujiwara and Khang (1990) proposed an iterative technique based on a two-phase search method for the optimal design of a new WDN as well as for the parallel expansion of existing ones. The main feature of the method is that it generates a sequence of improving local optimal solutions. The objective of optimisation was to specify an assignment of pipe diameters in order to minimise costs, although reliability will also be considered. Only a single fixed head source was used. The minimum head requirement at all nodes was fixed to a specific value. A single demand loading condition is enforced and the set of commercially available diameters and associated costs are adopted. This linear approximation method do not guarantee optimality to a non-linear problem, produce unrealistic output solutions (so-called split-pipe solutions where a pipeline is divided into several sub-lengths of different diameters) or those having a lack of redundancy (no loops), and also often suffer from efficiency problems.

However, Savic and Walters (1997*b*) pointed out that the optimum solution obtained by the stated methods might contain one or two pipe segments of different discrete sizes between each pair of nodes because the methods are based on a continuous diameter approach in which the pipe links are calculated and once the solution is obtained, the nearest commercial sizes are adopted. They stated that the split pipe design should be altered into only one diameter, and that the altered solution should then be checked to ensure that the minimum head constraints are satisfied. In addition, Cunha and Sousa (2001) indicated that the conversion of the values obtained by the stated methods into commercial pipe diameters could worsen the quality of the solution and might not even guarantee a feasible solution.

The computational efficiency of mathematical programming methods is limited to continuous solutions which are not favoured from an engineering point of view (Alperovits and Shamir, 1977; Quindry et al., 1981; Kessler and Shamir, 1989; Fujiwara and Khang, 1990; Afshar and Marino, 2005).

3.3.2 Meta-Heuristic Optimisation Techniques

Meta-heuristic techniques are employed in situations where classical optimisation techniques would otherwise struggle to achieve good results with acceptable runtime. In order to overcome the drawbacks of mathematical methods, researchers such as Simpson et al. (1994), Cunha and Sousa (1999) and Lippai et al.

(1999) began to apply meta-heuristic algorithms, such as the Genetic Algorithm (GA), Simulated Annealing (SA), Ant Colony Optimisation (ACO), and Particle Swarm Optimisation (PSO) to water network design.

Appendix A provides an overview to GAs. Optimisation using GAs to solve planning problems are presented in section 3.4 and discussed in details for the design of WDN in section 3.5. The other aforementioned optimisation techniques will be briefly presented in this section.

3.3.2.1 Simulated Annealing (SA)

Simulated Annealing (SA) is an optimisation technique used to find a good solution by trying random variations of the current solution. It applies the mutation operator familiar to the genetic algorithm to a single solution repeatedly. The idea behind the method of SA is derived from the analogy of heating and cooling of materials in order to increase their strength, as is frequently done in metal-work.

This technique stems from thermal annealing which aims to obtain perfect crystallisations by a slow enough temperature reduction to give atoms the time to attain the lowest energy state. Initially, a high 'temperature' allows the mutation to vary widely to the values of the decision variables. As the 'temperature' cools during the progress of the optimisation, the freedom of the mutation to vary the values is constrained. SA was first used for WDN design by Loganathan et al. (1995). They proposed an outer flow search optimisation procedure to identify lower cost design solutions. In this approach, each pipe network is subject to an outer search scheme that selects alternative flow configurations in an attempt to find an optimal flow division among pipes. For each selected set of pipe flows, a SA is used to find the associated optimal pipe diameters and energy heads. The SA as an iterative optimisation algorithm used to improve the cost of the WDN design in which the current cost is compared with the cost of the new iterate. The new iterate is used as the starting point for the subsequent iteration if the cost less than the previous cost.

In 1999, Cunha and Sousa (1999) achieved high quality solutions to two benchmark problems from the literature. They described the application of the SA technique to WDS problems, namely, the two-loop network introduced by (Alperovits and Shamir, 1977) and the network introduced by (Fujiwara and Khang, 1990) of Hanoi city. They proposed a scheme whereby the neighbourhood of a solution y is defined as any solution x' differing in configuration by a single pipe whose size is above or below its current size. The cost is defined as a function of pipe diameters and the objective is to minimise the initial capital cost with the constraints on nodal heads. In the proposed simulated annealing, energy cost is considered and the network configuration is defined by the pipe diameters. In each step of the algorithm, a change in configuration is produced and the cost is evaluated. The new configuration is then chosen in the neighbourhood of the current configuration. The application was successful in obtaining lower cost solutions than the solutions previously reported in the literature. The study suggested the consideration of reservoirs, pumps (and other devices), as well as multiple-loading conditions for future research.

3.3.2.2 Ant Colony optimisation (ACO)

Ant Colony optimisation (ACO) is a population based search technique for the solution of combinatorial optimisation problems. This method is inspired by the observation of the foraging behaviour of real ant colonies.

A colony of ants locates food by sending out foragers who initially explore their surroundings in an essentially random manner. Once a food source is located, the ant returns to the nest, while depositing a chemical substance called pheromone on the ground. The ants deposit the pheromone on the ground in order to mark some favourable path that should be followed by other members of the colony (Maier et al., 2003; Dorigo et al., 2006).

ACO exploits a similar mechanism for solving optimisation problems (Dorigo et al., 2006). In the optimisation technique, other ants can detect these pheromone trails and choose to follow, with higher probability, paths which are marked by greater pheromone concentrations. Ants can reinforce trails by leaving their own pheromones, and may choose alternative paths with a smaller probability. This reinforcement is a positive feedback mechanism which allows more ants to locate food. A negative feedback mechanism is provided by pheromone decay (Walski et al., 2003). The exploration of new paths helps to find improved solutions. Shorter paths between destinations will increase in pheromone intensity due to shorter traversal times, allowing more ants to travel along them in a given time period. These simple mechanisms constitute a form of indirect communication which enables the ant colony to solve the difficult problem of finding a shortest path to a food source.

The ACO approach was first applied to the domain of WDN optimisation by Maier et al. (2003) who used this technique to tackle a number of water network rehabilitation problems. This work is notable for finding the best solution for the New York City tunnels optimisation problem introduced by Schaake and Lai

(1969). This was improved upon by (Zecchin et al., 2005) who conducted an extensive parametric study to determine guidelines for assigning values to the various parameters in the algorithm specifically for WDN design optimisation.

Following on from an earlier work of (Zecchin et al., 2005), Zecchin et al. (2007) present a comparative study of the performance of five ACO algorithms. In these models, the cost is taken as a linear function of the length of the pipes. The pheromone information is modified using a pheromone penalty factor to discard choices that result in violation of hydraulic constraints.

3.3.2.3 Particle Swarm Optimisation (PSO)

Particle swarm optimisation (PSO) technique shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GAs). The system is initialised with a population of random solutions and searches for optima by updating generations. However, unlike GA, the mechanisms for improving the fitness of the population are quite different to those of evolutionary algorithms. PSO has no evolution operators such as crossover and mutation. In PSO, the individual solutions, called "particles", move through the problem space by following the current optimum particles. Each individuals is associated with a current velocity 'v', a memory of its previous best position called p_{best}, knowledge of the global best position g_{best} and, in some cases, a local best position l_{best}, within some neighbourhood. The concept consists of, at each time step, the individuals in the population use retained information about the best solution they themselves have encountered in the solution space. The velocity of each particle is changed toward its p_{best} and l_{best} locations (local version of PSO). The velocity is weighted by a random term, with separate random numbers being generated for acceleration toward p_{best} and l_{best} locations. When improved locations are being discovered, these will then come to guide the movements of the population. The process is repeated until a satisfactory solution is eventually discovered.

PSO was first applied to WDS optimisation by Vairavamoorthy and Shen (2004). They used PSO to minimise the capital cost ensuring adequate pressures at all nodes. The efficacy of the PSO was tested on two widely considered water benchmarks in the literature, namely, Hanoi network and New York City Tunnel. They compared their results to those who applied GA to the same benchmarks. For Hanoi network, although some previous research have produced lower costs using other optimisation techniques, these solutions were found to be infeasible because the required pressure values were not met. For the problem of New York City Tunnel introduced by, PSO obtained better solutions achieved by others optimisation techniques reported in the literature.

More recently, PSO is applied to WDS by Montalvo et al. (2008). They developed an adaptation of the original algorithm whereby solutions are checked using several of the fittest particles, and any solutions are randomly regenerated with a new position and velocity. This adaptation improves population diversity and global convergence characteristics. Furthermore, they employed an adaptive inertial weight which has the effect of accelerating the search in the initial stages, and gradually reducing the importance of the current velocity, ideally such that the particle converges to an optimum. Finally, they adapted the algorithm to accommodate discrete velocities variables in order to create discrete step trajectories for these variables. The authors tested their algorithm on the New York City Tunnel and Hanoi WDS benchmarks, and achieved large computational savings by a magnitude better than previous methods whilst closely approximating known global optimum solutions. PSO main drawback is its inability to maintain desired levels of population diversity and the balance between local and global searches and hence near-optimal solutions are prematurely obtained.

Finally, Table 3.1 summarises some other researches who used optimisation models for designing WDNs.

Table 3.1: Other Optimisation Methods in the Design of WDN

Author/Year	Optimisation Technique	Decision Variables
Geem (2009)	Harmony Search	Pipe Size Pump
Eusuff and Lansey (2003)	Shuffled Frog Leaping Algorithm (SFLA)	Pipe Size
Cui and Kuczera (2003)	Shuffled Complex Evolution (SCE)	Demand Reservoir capacity Supplied Water
Vasan and Simonovic (2010)	Differential Evolution	Pipe Size Network Layout
Mohan and Babu (2010)	Honey-Bee Mating	Pipe Size Hydraulic Head
Ostfeld and Shamir (1996)	Integer Non-Linear Programming	Pipe size Pump
Samani and Mottaghi (2006)	Integer Linear Programming	Pipe Size Pipe flow velocities Nodal pressure

3.4 Optimisation Using Genetic Algorithms

Evolution programs (Michalewicz, 1999), of which GAs are probably the best-known types, are general, artificial evolution search methods based on natural selection and mechanisms of population genetics. They emulate the natural processes of evolution being based on survival and reproduction of the fittest members of the population, the maintenance of a population with diverse members, the inheritance of genetic information from parents and the occasional mutation of genes. These algorithms are best suited to solve combinatorial optimisation problems that can not be solved practicably using more conventional operational-research methods. Thus, they are often applied to large, complex problems that are non-linear with multiple local optima. The degree of difficulty of solving a certain problem with a dedicated algorithm is related to its computational complexity, i.e., the amount of resources such as time and memory required to finish the process. The computational complexity depends on the number of input elements needed for applying the algorithm (Weise, 2009). GAs do, however, provide a basis on which it is possible, via specialisation or hybridisation, to build very efficient optimisation algorithms (Beasley et al., 1993*b*).

To solve optimisation problems, GA requires a data structure to represent the solutions, called "chromosomes". This makes the GAs to be applied to broad areas. Their ability to deal successfully with a wide range of problem area, including those which are difficult for other methods to solve make them quite powerful (Coley, 1999; Sivanandam and Deepa, 2008). For more information about GAs, see appendix A and appendix B for GA-related definitions.

Because of the complexity involved in setting up a GA to operate effectively, this form of optimisation is not well suited to 'trivial' problems, i.e. those for which the number of possible solutions is small (Sivanandam and Deepa, 2008). For large problems such as those associated with networks, the exhaustive search of all options is very hard and unlikely to be feasible. For example, in a WDN design problem, through exhaustive search the designers would analyse every scheme in the design space until the best design is found. The number of possible WDN designs is calculated as:

$$N = X^a \times Y^b \tag{3.1}$$

where:
N = number of possible designs
a = number of nodes

b = number of pipes

X = number of possible node locations

Y = number of available diameters

For example, a water network with 50 nodes, 1,000 possible node locations, 50 pipes, and 10 possible diameters (including the option of not employing a pipe) would have:

$$\begin{aligned} N &= 1,000^{50} \times 10^{50} \\ &= 10^{150} \times 10^{50} \\ &= 10^{200} \textit{ possible network designs} \end{aligned}$$

Even with current high-performance computers available, exhaustive search would require long running times. If one billion design evaluations (i.e. network simulation runs) could be performed in a second, the time needed to evaluate all possible schemes would take billions of years to analyse 10^{200} design schemes.

GAs are suited for parallel computations especially if a parallel computer is available and where each processor will be responsible for evaluating a separate fitness function at the same time (Haupt and Haupt, 2004). This implementation assumes a population on each of the computer and migration of individuals among the computers. This implementation allows for larger simulations to investigate the real potential of the proposed methods.

Despite of the GA optimisation advantages, optimisation using GAs have some disadvantages and optimisers tend to use their previous experience methods to employ GAs in problem solving. Haupt and Haupt (2004) and Van Zyl et al. (2004) discussed some of drawbacks of the GA optimisation:

- The convergence is not guaranteed in the solution.
- GAs require more function evaluations than other search heuristics to find fit solutions.
- GA optimisation process is time consuming. For a large population sizes, a GA optimisation may take up to few days using personal computers.

Long running times to solve a problem may be problematic for some reasons. These reasons may be related to long period simulation time which is undesirable process (Van Zyl et al., 2004). For example, short simulation periods will make it possible to utilise an optimised model in emergency situations where time is very crucial.

GA optimisation is a powerful approach with a proven ability to identify near-optimal solutions (Savic and Walters, 1997*b*; Halhal et al., 1999). However, the correct operation of a GA optimisation depends on careful configuration and parameter tuning. Too high or low rates of genetic operators ('crossover' and 'mutation') may sometimes result in stagnation.

GAs are the most appropriate optimisation algorithms to solve many real-world problems. They have the ability to begin the solution with a group of plans, and iteratively change the entire group to produce a final generation of 'good' solutions particularly in very large and complex search spaces (De Jong, 2006).

Efficiency is desirable as it allows larger scale planning problems to be analysed and for the interactive use of the planning tools by decision-makers. Both of these factors are significant for the operational use of the planning tools. For instance, the objectives involved in the sustainable development include efforts to (Balling et al., 1999; Bui, 1999):

- Minimise traffic congestion.
- Maximise economic development.
- Minimise physical change.
- Preserve open spaces and historical sites.
- Provide adequate utility infrastructure.
- Control air pollution.

It is hard to develop plans that achieve all objectives. Many objectives may compete with each other, and the solution to one problem may not be suitable for another one. In addition, it is very difficult for planners to search through and analyse all possible plans at the same time (Balling et al., 1999). It is therefore necessary to make trade-offs based on the relative importance of one objective against another. The relative importance for the objectives can be compromised using weighting factors.

GA applications have been increasingly used in many fields because they are able to deal successfully with a wide range of planning, scientific and engineering problems (Mitchell, 1996; Herrera et al., 1998; Sivanandam and Deepa, 2008). Some applications of GA are as follows:

- Genetic programming: GAs have been used to evolve computer programs (Koza, 1990) for specific tasks. In addition, GAs are used to design other computational structures such as designing neural networks (Garca-Pedrajas et al., 2006; Leng et al., 2006) and sequence scheduling and planning (Mawdesley and Al-Jibouri, 2003; Mira et al., 2003).

- Robotics: GAs have been widely used in machine learning and robot trajectory applications (Wang and de Silva, 2008; Gaham and Bouzouia, 2009).
- Classifier systems: GAs have been used in classification and prediction tasks. For example, predicting and analysing non-linear dynamical systems and weather prediction (Radcliffe, 1992; Wu et al., 2009; Canellas et al., 2010).
- Strategy planning: GAs have been used in to model process and innovation systems (Cova and Church, 2000; Matthews, 2001; Balling, 2003; Lowry, 2004; Stewart et al., 2004; Zegras et al., 2004).
- Others: evolution, artificial life, social systems, immune systems, creating images and evolving music.

3.5 Using GAs to Assist in the Design of WDNs

Solving optimisation problems related to WDNs is recognised as non-deterministic polynomial-time (NP)-hard problem (Eusuff and Lansey, 2003; Amit and Ramachandran, 2009; Vasan and Simonovic, 2010). This problem has conventionally been approached using a number of techniques such as linear programming, non-linear programming, and dynamic programming.

However, the success of these procedures has been limited and very few have actually been applied to real WDNs. Limited acceptance of these techniques in engineering practice is partly because:

1. Such techniques are generally quite complex involving a considerable amount of mathematical sophistication (e.g., requiring extensive expertise in systems analysis and careful setting up and fine tuning of parameters).
2. These techniques are dependent upon the number of pipes, nodes, pumps, and storage tanks being considered along with the duration of the operating period.
3. They are generally subject to simplification of the network model and its components. This simplification is to accommodate the non-linear nature of the network hydraulics.

4. They tend to be time-consuming resulting in added costs and inefficient use of the computer hardware and software.

5. They may be easily trapped at local optima and may not lead to the global optimal solution.

Evolution algorithms represent a proven alternative strategy for approaching complex problems. GAs are increasingly used in water network design for problems that are difficult to solve using traditional optimisation techniques.

The benefits of GAs stem from their ability to converge rapidly on an optimal or near-optimal solution, having analysed only a small fraction of the number of possible available solutions. In addition, GA can consider the design of large networks and produce discrete pipe diameters and alternative solutions.

Simpson et al. (1994) presented a methodology for applying GA technique to the optimisation of WDN design. Results are compared with the techniques of complete enumeration and non-linear programming. They applied the optimisation techniques to a case study pipe network. They concluded that complete enumeration may be only used for simple problems and for networks with few pipes. Non-linear optimisation is an effective technique when applied to a small network and the method only generates a single solution which may be a local optima. In addition, non-linear algorithms perform on the basis of continuous variables, pipe diameter for example.
The GA technique generates a class of alternative solutions close to the optimum. One of these alternative solutions may actually be preferred to the optimum solution. In addition, GA found the global optimum in relatively few evaluations compared to the size of the search space.

Gupta et al. (1999) developed a methodology based on genetic algorithms for lower cost design of new, and expansion of existing, WDN. The results have been compared with those of non-linear programming technique through application to a case study which has been previously attempted by several researchers. The hydraulic simulator called "*Analis*" which is based on graph theory, was used in both Non-Linear Programming (NLP) and GA solutions to calculate pressure heads, flows and velocities in the design of branched, looped and combined systems. The optimal cost obtained from the NLP technique was higher than the solution achieved by some researchers. A comparison between the results of the GA and NLP techniques for augmentation of several networks showed that the GA in general provided a lower cost solution, than that obtained from the NLP technique. The differences in costs, however, was small which shows both techniques are efficient.

Reca and Martínez (2006) developed a new computer model called Genetic Algorithm Pipe Network Optimisation Model (GENOME) with the aim of optimising the design of new looped irrigation WDNs. The model is based on an improved GA method. The model had been tested and validated by applying it to the least-cost optimisation of several benchmark networks reported in the literature. Their model had shown good results when applied to well-known benchmark networks adopted by earlier works. Reca and Martínez (2006) compared their model results to the earlier model by Alperovits and Shamir (1977) of two-loop network. The second benchmark network selected was the pipeline network for water supply in Hanoi (Vietnam) proposed by Fujiwara and Khang (1990). The results were compared with the best results obtained in previous works. They concluded, however, for practical use and for complex WDNs, improvements should be introduced in order to achieve a faster and more precise convergence to the global optimum.

The objectives of water supply planning are related to the designs of existing facilities such as streets, lakes and buildings. This plan should aim to minimise initial and operational costs of the system providing a certain level of reliability and water quality. Environmental, social and political issues can be added to the optimisation process. To solve any problem using GAs in water distribution systems, Halhal et al. (1997), Skok et al. (2002) and Schütze et al. (2003) discussed some practical issues to consider when solving a problem. They categorised these issues as:

- Cost parameters (e.g. Investment, pipeline losses, maintenance).
- Route constraints (e.g. physical obstacles, existing network sections, high cost passages).
- System reliability (e.g. quality, quantity , redundancy)
- Technical constraints (e.g. pumping, pipe sizes)

Design engineers may depend on cost optimisation techniques for WDN to achieve maximum benefits with minimum costs. Therefore, studies conducted using GAs in the design and management of water networks concentrated to a great extent on cost designs of WDN (Geem, 2006).

GA studies included minimising the cost of construction, operation and maintenance and maximising the reliability of modelling and operation of WDN (Halhal et al., 1997; Savic and Walters, 1997*b*; Dandy and Engelhardt, 2001, 2006). Although these two criteria are conflicting and one may dominate the other, decision-makers need to find the optimum solution to the problem and represent a compromise between the two criteria. For example, the reliability of a system can be maximised by meeting the minimum allowable pressure constraints across the network under nodal demand and pipe size conditions (Taher and Labadie, 1996; Tolson et al., 2004).

In most cases, optimisation methods can be used in conjunction with the simulation models for urban water supply to find the best technical, environmental, and financial solutions. These methods become increasingly effective in the design and planning of urban infrastructure. Yet they are constrained by the complexity and non-linearity of WDN designs (Loucks et al., 2005).

A significant number of researchers have applied GA technique to certain aspects of the design of WDNs. These aspects will be discussed in the rest of this section.

3.5.1 Network Design and Rehabilitation

Walters and Lohbeck (1993) studied the case of pipe networks with one demand and no constraints on minimum pipe diameters. They applied GAs to the layout and design optimisation of a branched network. They showed that GAs effectively converge to near-optimal network layout. However, they neglected the strong coupling which exists between pipe sizing and layout determination for pipe networks.

In the same year, Murphy et al. (1993) presented applications of a GA to a real-world WDN optimisation problem. Their study, like some other studies, restricted to considering the hydraulics and availability requirements, leading to the so-called optimal pipe sizing of the pipe networks. The reliability requirement is usually addressed by considering a predefined fixed, usually looped, layout for the networks to be designed. In this paper, they demonstrated the ability of a GA to produce an optimal design layout for a new housing development using discrete commercially available pipe diameters.

This theme is returned by Simpson et al. (1994) who described one of the first applications of GAs to a conventional supply system. In this problem, a small network (one reservoir, one tank and nine demand nodes) is to be upgraded through the provision of new pipes and the duplication or cleaning of others. They stated that for a given layout of pipes and specified demands at the nodes, the combination of pipe sizes that gives the minimum cost should be found, subject to hydraulic constraints. The objective is to minimise the cost of implementation whilst meeting a minimum pressure criterion at each node.

Dandy et al. (1996) proposed an improved GA for pipe network optimisation to find a low cost feasible discrete pipe diameters. They used a variable power scaling of the fitness function. The introduced exponent into the fitness function is increased in magnitude as GA iterations proceed. In addition, they introduced a bitwise creeping mutation operator. They compared their model performance with those of traditional GA formulation. Their results indicated better performance than the traditional GA.

Similarly, Gupta et al. (1999) presented an improved GA implementation for WDN design. Instead of employing binary strings to represent discrete pipe diameters, this work substitutes the actual diameters into the chromosome instead to avoid the encode/decode cycle associated with the use of binary strings. The GA itself is influenced by several heuristic modifications, including the initial formulation of the network into different diameter groups using expert judgement. The results for six related case study scenarios are contrasted with those obtained from the authors' own "WATDIS" software which employs non-linear programming (NLP). It is noted, however, that both the GA and the NLP software require several trials to identify near-optimal solutions. In the case of the GA this is to accommodate different scenarios of initial formulation for the pipe diameter ranges whilst the progress of the NLP software is dependent on the initial conditions as it has a tendency to identify local optima.

Improvement in a WDS performance can be achieved through replacing, rehabilitating, duplicating or repairing some of the pipes or other components (e.g. pumps, tanks) in the network, and also by adding completely new components. It is likely that funding will be available to modify or add only a small number of components in a network at any time. The decision problem is to choose which components to add or improve (and how to improve them) to maximise the benefits resulting from the changes to the system whilst minimising the costs.

Savic and Walters (1997*b*) introduced the "GA*net*" software and applied it to previously published case studies in the design and rehabilitation of WDNs. These examples illustrate the potential of GA*net* as a tool for WDN planning and management. The first problem deals with the design of a completely new network for the city of Hanoi, Vietnam (Fujiwara and Khang, 1990). The second example considers the problem of optimally expanding an existing network, namely, the two-loop network problem studied by Alperovits and Shamir (1977). The availability of GA*net* with its sampling capability enables the engineer to concentrate on the quality of the solutions rather than on the mathematical characteristics of the model used. The quality of the network designs is assessed on the basis of (1) the investment costs; and (2) how well the minimum head constraints are satisfied. A comprehensive comparison of the results obtained in previous work, using various optimisation techniques is presented and illustrates the sensitivity of such analysis to the minor variation of some modelling parameters between different researchers. In particular, variation in the constants used to derive the Hazen-Williams coefficient is highlighted. Such differences in behaviour are shown to have consequences for the behaviour of the optimisation algorithms employed.

Afshar et al. (2005) addressed the change from an existing layout to a cheaper one. The new layout may include all possible connections in the network to satisfy a predefined level of reliability. In addition, minimum pipe diameters and associated lengths are also studied (Taher and Labadie, 1996; Afshar et al., 2005).

Halhal et al. (1997) demonstrated a GA technique termed 'Structured Messy Genetic Algorithms' (SMGA) with a particular application to the rehabilitation of WDS. The Messy Genetic Algorithm (MGA) of Goldberg (1989) differs from a conventional GA in that the MGA operates with variable chromosome lengths allowing the algorithm to progressively build up a solution as it runs which constrains the search space. In this instance, integer-coded genes are employed. The SMGA by Halhal et al. (1997) extends the approach of the MGA by employing an initial population, each member of which contains each single decision element. For instance, in the small rehabilitation problem presented in this paper, there are eight possible decisions to be made on each of 15 pipes which would lead to the SMGA having a starting population of $8 \times 15 = 120$ individuals. Each individual represents each possible decision using one intervention. Furthermore, the authors describe one of the first applications of multi-objective optimisation to WDS problems. The SMGA is shown to perform better than a conventional GA for the presented problems, producing a better classification of individuals than the standard GA.

This technique is later applied to the 'Anytown' WDN benchmark system (Walski et al., 1987) by Walters et al. (1999). This problem is a network reinforcement for increasing demand scenarios, which is constrained not only by infrastructure but also by pumping costs. The SMGA as presented by the authors is seen to improve the best published results for this benchmark by between 4% and 5%. Wu et al. (2001) seek to optimise the same network arrangement solved by Simpson et al. (1994) using a similar MGA. They demonstrated an improved performance in terms of GA convergence to near-optimal solutions with the MGA on the New York City tunnels problem over that achieved by the 'Improved GA' of Dandy et al. (1996).

Dandy and Engelhardt (2001) reported the use of GAs to optimise pipe replacement schedules for single and multiple time horizons. The algorithm described includes the use of a hybrid selection scheme in which tournament selection is combined with the conventional roulette wheel selection technique (Holland, 1975). In addition, uniform crossover (Syswerda, 1989) and the creep mutation of Dandy et al. (1996) were employed to operate on an integer coded chromosome. The authors demonstrated that this approach employed on a problem with a large solution space (approximately 1×10^{100}) produces good results by identifying pipes requiring replacement within the required budget. The results were validated by comparison with a simplified asset model for which decisions were made on a case-by-case basis.

Kadu et al. (2008) present a modified GA for undertaking optimal design of WDN employing techniques to reduce the optimisation search space. A real coded chromosome is employed with single point, uniform and multi-parent crossover. In addition, non-uniform and neighbour mutation with a critical path technique are used to reduce the search space. The different genetic operators are selected at random during the operation of the algorithm. The algorithm is demonstrated on several familiar networks from the literature including the Hanoi network where the algorithm is shown to match the best known result (Cunha and Sousa, 1999) and to better the result achieved for the stricter problem introduced by Savic and Walters (1997*b*). For performing the hydraulic analysis, the authors introduced "*GRA-NET*" which is a hydraulic solver.

3.5.2 Network Reliability Optimisation

The design of WDNs requires extensive planning to ensure that good service is provided to the customers in a reliable and economical way. The advent of multi-objective GAs made it possible to handle both the minimisation of cost and maximisation of reliability as two independent objectives. Reliability refers to the ability of the water network to provide an adequate supply to the consumers, under both normal and abnormal operating conditions (Xu and Goulter, 1999).

Goulter et al. (1986) suggested that the minimum cost design for a given layout and single loading case is a branched network (i.e., a network with no loops). In practice, however, loops are an essential feature of actual distribution systems as they provide an alternative flow path if there is pipe failure or for maintenance. To achieve a degree of redundancy in pipe network optimisation, the layout should have appropriate loops and by specifying minimum diameters for all pipes.

Tolson et al. (2004) presented an approach that links GAs as the optimisation tool with the First-Order Reliability Method (FORM) (Madsen et al., 2009) for accurately estimating network capacity reliability and identifying the most critical node in the network. FORM estimates the reliability of a system, α, by computing reliability index, β, using:

$$\alpha = 1 - \Phi(-\beta) = \Phi(\beta)$$

where $\Phi(\,)$ = standard normal cumulative distribution function. β can be interpreted as the minimum distance between the mean-point of 'n' random-variables and the failure surface. These 'n' random-variables influence the load, the resistance, and the performance of the system. The limitation of FORM is the repetitive calculations of the first-order derivatives and matrix inversions even for small WDNs. Uncertainty in nodal demands and pipe roughness conditions are considered in this paper. Correlations between nodal demands are shown to increase costs of a WDN designed to meet a specific reliability level.

Hybrid optimisation techniques are increasingly being employed to accommodate additional concepts to improve the reliability of optimized solutions particularly by considering uncertainty in design criteria. Kapelan et al. (2003) report a technique for accommodating uncertainty in design constraints when optimising using GAs. This approach embeds a stochastic optimisation cycle within the operation of a conventional single or multiple-objective GA. The stochastic cycle evaluates samples for Probability Density Functions (PDFs) obtained for the stochastic variables (in this example, uncertain future demands) and aggregates statistics on the network performance. It provides a measure of reliability of the network under the uncertain constraints.

Babayan et al. (2005) and Kapelan et al. (2005) present differing approaches to the optimisation of WDS design / rehabilitation under conditions of uncertainty. The former approach employs the reformulation of the stochastic problem in deterministic terms. Babayan et al. (2005) discussed the design of WDNs using a GA model with demand uncertainty. The randomness in the system in this model is due to the uncertain nature of nodal demands. The objective is to minimise initial cost with minimum nodal head constraints. The original stochastic model is converted to a deterministic model with standard deviation as a natural measure of variability. The deterministic model is then coupled with a GA solver to find optimal solutions.

Kapelan et al. (2005) incorporate the uncertainty in nodal demands and pipe roughness conditions in the multi-objective optimisation model for the design of a WDN. The objectives are to minimise the total design cost and maximise system robustness. System robustness is defined as the probability of simultaneously satisfying minimum pressure head constraints at all nodes in the network. The problem is solved using Robust Non-dominated Sorting Genetic Algorithm II (RNSGA-II). In RNSGA-II, a small number of samples are used for each fitness evaluation compared to the full sampling approach. It is shown that RNSGA-II is capable of identifying robust Pareto optimal solutions with reduced computational effort.

By comparison, the latter approach promotes a sampling-based technique for accommodating uncertainty that is independent of the model under consideration. Both of these methodologies employ single objective GAs to perform the optimisation. Prasad and Park (2004) used Non-dominated Sorting Genetic Algorithm (NSGA) of Srinivas and Patnaik (1994) to arrive at the trade-off between cost and network resilience. In this instance, the design of WDN arrives at a trade-off curve between the cost of the network and its reliability and optimal solution can be chsen based on cost constraints.

3.5.3 Pump Optimisation

The design of a WDN involves the determination of the network layout, pipe sizes, pump sizes, tank sizes and locations, valve locations and pump and valve operating schedules. For networks with pumps, the cost of pumping should also be considered. The utility of the network could be defined as a reliability measure, which accounts for the performance of the network under uncertain conditions.

One of the earliest applications of a GA to a pipeline optimisation problem is presented by Goldberg and Kuo (1987). They demonstrated the efficacy of applying the GA methodology to pipeline optimisation problems. In this case, the optimisation is applied to a 40 pump, serial pipeline and seeks to minimise the cost of pumping (in terms of power consumption) whilst meeting constraints of maximum discharge pressure and maximum and minimum suction pressure. The GA is implemented in terms of a simple binary string and employs single-point crossover and simple mutation. The results are compared with the optimal solution determined using mixed-integer programming and the GA is found to have performed well.

The operation problem involves determining the optimal pumping schedule for a given network so as to minimise the cost of operation, while ensuring adequacy of level of service under different loading conditions. Pump scheduling is the process of choosing which of the available pumps within a WDS are to be used and for which periods of the day the pumps are to be run (Savic et al., 1997).

Walters et al. (1999) extended the model of Halhal et al. (1997) by including sizing and operations of storage tanks and pumping installations as decision variables. Halhal et al. (1999) applied the structured messy GA (SMGA) for the optimal design and scheduling of investment for the rehabilitation of a WDN. The aforementioned models used GA for small WDNs. Wu et al. (2001) apply GA for large scale WDNs containing pipes, tanks, valves, and pumps. A messy GA is used for improving the efficiency of the optimisation procedure. A network solver is integrated with the GA optimiser to solve the hydraulic equations at each iteration.

GA techniques are used to solve optimal pump scheduling problems. The pump scheduling problem can be formulated as a cost optimisation problem which aims to minimise costs of supplying water, whilst keeping within physical and operational constraints (e.g. maintain sufficient water within the system's reservoirs to meet the consumer demands on the required time-scale).

Savic et al. (1997) proposed a GA technique for pump scheduling in water supply systems. They considered two objectives in the paper (1) minimisation of energy and (2) pump switching which is introduced as a measure of maintenance cost. This paper presents several improvements of the single-objective GA and the results of the further investigation into the use of multi-objective GAs for solving the pump scheduling problem. The multi-objective approach used in this work deals with both the energy cost and the pump switching criterion, at the same time. The performance of the algorithm is tested for different demand profiles and additional requirements and compared to that of the single-objective GA. Both techniques improved by using hybridisation with a local-search method.

Boulos et al. (2001) presented a management model for optimal control and operation of WDS. They developed an optimal operations model, H_2ONET Scheduler, for real-time control of multi-source, multi-tank WDSs. The proposed model determines the least-cost pump scheduling /operation policy for each pump station in the WDNs while satisfying certain hydraulic performance requirements. H_2ONET computes the optimal pump schedule for each pump or pump group based on the specified control time-scale such that the overall energy cost is minimised. Pumps are grouped together based on their known characteristics such as location, pumping capacity and control components (e.g. storage tanks). The developed model has been applied to several WDNs of different sizes and degrees of complexity. The resulting model can be used to evaluate various rate schedules, optimise storage / pumping trade-offs, improve operational efficiency, and assure reliable operations.

Prasad (2010) presented a GA method for the design of WDN with various network elements such as pipes, pumps, and tanks. The objective of the study is minimisation of total cost, which includes capital cost and energy cost. In this study, all existing pipes are considered for duplication, or cleaning and lining. Upgrading the existing pumping station is allowed through the addition of a maximum of two new pumps with the same characteristics as the existing ones. In addition, it proposed a new approach for tank sizing which does not require explicit consideration of some of the operational constraints, such as the limits on maximum and minimum tank levels. To demonstrate the efficacy of the model, it has been applied to 'Anytown' WDN problem proposed by Walski et al. (1987) to find the least-cost design that satisfies pressure criteria while supplying future demands. Two cases were studied, one considering all constraints except pressure constraints and the other considering all constraints. The obtained design results were 15% cheaper than those proposed by other researchers (e.g. (Walters et al., 1999) and (Babayan et al., 2005)) under similar performance conditions but with different tank sizing methods.

3.5.4 Network Calibration

WDS calibration can be defined as the process in which a number of WDS model parameters are adjusted until the model behaves as closely as possible to a real WDS. Calibration of pipe network systems involves determining the physical and operational characteristics of an existing system. This is achieved by determining various parameters that, when input into a hydraulic simulation model, will yield a reasonable match between measured and predicted pressures and flows in the network. GAs have been used for calibrating WDS by modifying system parameters and pipe roughness's in order to match model results with data obtained from the field (e.g. (Savic and Walters, 1997*a*)).

Vitkovsky and Simpson (1997) presented an analysis of the application of GAs to the calibration, both for fitting roughness values to pipes and for transient calibration. They studied the uncertainty in developing computer models is the condition of the interior of old pipes in the network. The decision variables were the pipe friction factors and lumped leak coefficients, which encoded within a GA string and represented by discrete values within a continuous range. The objective function was to minimise absolute differences between measured and model predicted nodal head values. Calibration parameters considered were pipe friction factors and lumped leak coefficients (i.e. leak areas multiplied by leak discharge coefficient).

De Schaetzen et al. (2000) utilised the GA in a different fashion for calibration. Their study attempted to find the optimal arrangement and density of sampling points for the analysis. The optimisation is formulated in order to maximise a fitness function, by employing an integer-based chromosome with each gene representing a potential sampling point location. The results are compared to those produced through expert judgement and are found to be a useful tool for deriving a likely set of candidate sampling points. A similar approach is adopted by Meier and Barkdoll (2000) where the objective function is to realise the maximisation of the length of pipes in the network that have non-negligible flows when the sampling is being performed for the proposed sampling point distribution. This method was applied to a network model for a small town in Ohio (USA) and shown to perform well matching the optimal solutions produced by enumeration in a series of validation tests. The GA search technique was verified by comparing results to global optimal solution found by enumeration technique.

3.5.5 Water Quality Optimisation

Genetic Algorithms are used by Munavalli and Kumar (2003) to optimise the rate, timing and concentration of chlorine dosing in a WDN. Given predefined dosing locations, this methodology seeks to optimise chlorine dosing so as to minimise the maximum concentration found in the network whilst continuing to maintain the minimum level of chlorine residuals at all nodes in the network. Using hydraulic results from hydraulic software (EPANET) for extended period simulation of a network, the authors apply their quality model for chlorine decay. The employed GA includes decision variables (chlorine dosage) uses a conventional binary string implementation. The performance of simple GA operators (reproduction, crossover, and mutation) is further enhanced by incorporating improvements such as creep mutation in which mutation permutes the variable by the smallest possible amount in a given direction. The results obtained from this study for the three sample networks studied show that GAs are effective in estimating near-optimal solutions within a reasonable number of evaluations.

Broad et al. (2005) addressed the problem of multi-objective WDN design taking into account both water quality and reliability targets using the analysis of theories useful for modelling. They compared the performance of optimisation methods on the New York City Tunnels problem introduced by Schaake and Lai (1969) using two approaches. In the first approach, a traditional GA was linked with a hydraulic simulator, whilst in the second approach, the simulator was substituted by an Artificial Neural Network (ANN) (i.e. ANN is linked to a GA optimiser). The authors used a meta-modelling approach to optimise a water distribution design problem that includes water quality in order to minimise the initial network cost with water quality constraints (see equation 3.2).

$$C_{j,min} \leq C_j \leq C_{j,max} \tag{3.2}$$

where C_j is the residual chlorine concentration, $C_{j,min}$ is the minimum allowable residual chlorine concentration at node j, and $C_{j,max}$ is the maximum allowable residual chlorine concentration at node j. For the ANN meta-model, the decision variables (pipe diameters and chlorine dosing rate) are taken at the input layer and the constrained variables (pressure and chlorine residual at node j) the output layer. The ANN was trained to approximate the nodal pressures and chlorine concentrations given pipe diameters and water demands. To find optimal or near-optimal solutions when using a GA, a simulation model is needed to check whether the constraints are violated for a given set of decision variables (if so, a penalty cost must be applied). Under these circumstances, methods that are capable of achieving a satisfactory level of performance with limited number of function evaluations represent a valuable alternative. However, there is a distinct lack of literature addressing this issue in relation to the design of WDN.

GAs can be combined with GIS-based methods to solve complex problems and optimise water distribution systems. Potential uses of GIS with GAs for the planning of WDN are explained in section 3.7.

3.6 Applying Geographic Information Systems to Water Networks

The design of water supply networks may heavily depend on GIS due to the natural geo-reference representation of objects (e.g. houses, parks, roads), distribution lines (e.g. pipe lines), and water resources (e.g. main pumping station and intakes). More information about GIS can be found in appendix A.

Integrated frameworks for combining GIS and optimisation process can be designed to optimise WDNs. In these integrated systems, GIS is able to handle the spatial data and provide both decision and management functions. GIS can be used in WDNs as a mapping tool to store, retrieve, and analyse spatial and thematic data. Information regarding real-world objects stored in GIS on thematic layers linked together geographically.

As WDNs consist of several interconnected components, they are particularly suited to a GIS vector model which provides a natural data representation, storage, retrieval, and analysis. Representation in a GIS vector model is described by feature codes, associated attributes, and geographic (x, y) coordinates. In addition, GIS brings spatial dimensions into the WDN database to present an integrated view of the world (McKinney and Cai, 2002). For example, the location of a node in a network, which is a point, can be described by a single (x, y) coordinates; pipelines which are linear features can be stored as a collection of point coordinates and network loops can be stored as a closed loop of coordinates (polygon features).

Assigning land parcels to nodes is frequently performed as a GIS-based procedure, with polygons representing land parcels superimposed on a network graph of the WDN, and the allocation of polygons to nodes.

Although the design of WDN has many variable parameters such as layout, pressure requirements and pipe characteristics and cost, GIS can be used in the following tasks (Walski et al., 2001):

1. Vicinity analysis: this identifies distances between different network components such as proximity distance between nodes, users (demand areas), storage tanks, and water sources.

2. Overlay analysis: this obtains certain information to determine, for example, all street junctions in the service area.

3. Network analysis: it performs the necessary network management functionalities such as hydraulic and water quality analyses, fire flow computation, pump scheduling, and selective scenario management. For instance, network analysis identifies all households in the area impacted by a water main break.

4. Visualisation: it displays and relates real-world objects graphically.

The benefits of GIS may extend beyond the process of building a model of a WDN (Savic and Walters, 1999; Walski et al., 2001), to include:

- Record Maintenance: reports, map production, and updating for operational use.

– Demand generalisation: expansion and rehabilitation of water networks, finding new sources, and risk management.

– Operational asset management: maintenance of data and applications, leakage information, and pumps.

GIS modelling tools of a great assistance to planners, engineers, and city managers helping them to comply with water supply regulations and design of distribution systems. More importantly, GIS and hydraulic network solvers can be combined with optimisation applications to provide a flexible and powerful tool to support decision-making and solve complex problems in WDNs. One of these integrated optimisation environments, for example, is GA*net* which is used for optimal design and modelling of WDN problems (Morley et al., 2001). The authors described an architecture for an integrated optimisation application, GA*net*, which comprises a GA application, a GIS and a hydraulic network solver. This software is developed to be used by experienced GA users, water supply modellers and engineers. Although it is difficult to determine the success of a decision-support tool, the optimal solution appears easily in this software and, therefore, it can be used as a decision-support tool.

Figure 3.1 shows the basic idea behind combining GIS and optimisation. This framework is problem dependent and can be extended to involve various planning and management schemes.

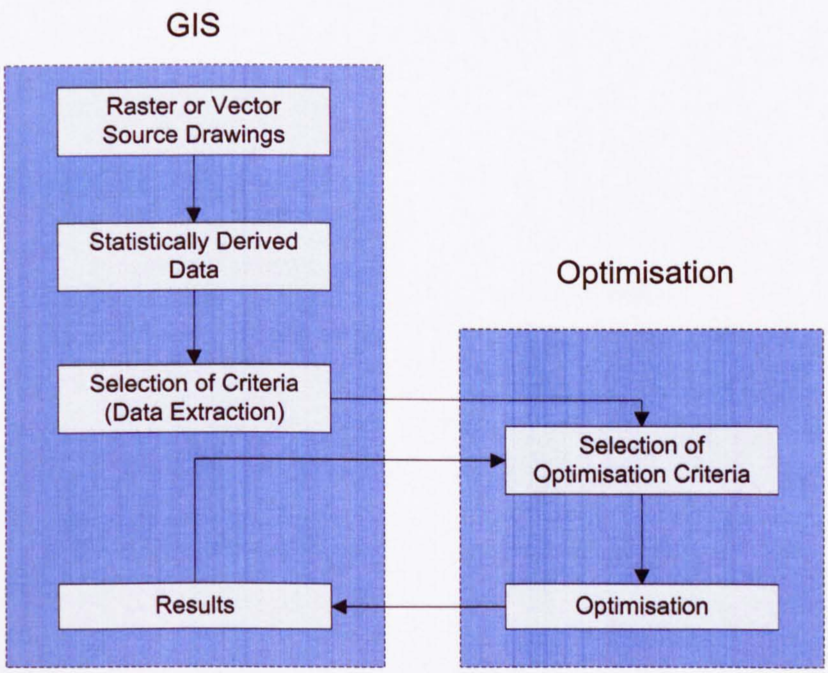


Figure 3.1: GIS and Optimisation Framework

Figure 3.1 proposes a simple framework for a GIS and an optimisation technique. Since GIS does not aim to analyse the information instantaneously, it may provide extensive and productive geo-referenced data (raster or vector) combined with statistically derived data (e.g. water demands, population) for planning purposes. GIS data are stored in GIS databases and used to define utility's data model and the location of actual data. The next step is the selection of criteria or extraction of data from GIS databases. The extracted data are prepared at this stage for the next stage (the optimisation stage). This step also involves selection of several parameters such as demand, existing pipes segments, and head requirements. The exported GIS data may be modified or manipulated to accommodate the required objectives.

After the data extraction and preparation stage, GIS data are optimised in a separate process according to selected optimisation criteria in the optimisation part of the framework. This stage optimises the solution and the results of the optimisation will be returned to the GIS so they may be used again with the geographical data. The whole process terminates until an acceptable solution is found.

The external optimisation package may be commercial software or can be written in one of the modern programming languages such as Visual C++ or Visual Basic. This package allows the design of a WDN to be performed outside the GIS environment. These packages can share water distribution data with the GIS database, allowing the optimisation package to be part of the management and planning system. These capabilities can greatly assist in the decision-making processes for network design, rehabilitation requirements, and reliability. Hydraulic network solvers with optimisation capabilities can build a link to any GIS database structure using attribute mapping. For example, a pump or valve can be represented as either a node or link in the GIS database, while still being linked with the hydraulic solver. Further, this package can use more than one database (different databases can be linked to each other) to extract information and rewrite new information to the same databases or use a new GIS database. Some of these data (data from other GIS databases) will only be transferred to the utility GIS database as a final solution to the problem (Skok et al., 2002; Morley et al., 2004). This allows the user flexibility and functionality in developing specialised user-defined reports. These reports can be customised to meet any combination of modelling criteria for different network variables and time periods. However, meaningful analyses of the water network may be achieved with limited GIS analysis tools. Many commercially available GIS applications have little or no computational analysis engines or have extensive modelling and water network analysis capabilities.

Evolutionary computational methods, especially GAs, are increasingly employed in the design of WDNs. Numerical optimisation using GAs can be successfully adapted in this field. Some examples of existing roles for GAs in WDN design are outlined in section 3.7.

Morgan and Goulter (1985) and Taher and Labadie (1996) formulated their design optimisation as a linear programming problem for which a software package was developed. A common theme for these papers, and several others, is the direct integration of the network optimisation software with spatial information. In this instance, the spatial component is used to perform network analysis for pressure zone distribution, node demand allocation, and least-cost routing.

Makropoulos and Butler (2003) employed a random search technique to resolve a spatial optimisation problem obtained from the aggregation of water demand management scenarios developed through fuzzy inference rules. The work discussed the development of a prototype spatial decision-support system supporting strategic planning and providing examples from a particular application in water demand management. This work seeks to produce optimum strategies for maximising water-saving whilst respecting investment constraints. The results supported the case of using a spatial decision-support system based on approximate reasoning to engineering expertise for urban water management applications according to user characteristics and site constraints.

More recently, Ho et al. (2010) conducted one of the first attempts to manipulate earthquake data in the break-event Artificial Neural Network (ANN) prediction model. The authors developed a methodology based on the integration of an ANN model and a GIS to assess water leakage and to optimise pipeline replacement. They derived and analysed their pipeline break-event data from Taiwan Water Corporation Pipeline Leakage Repair Management System. Pipe diameter, pipe material, and a magnitude of 3+ earthquakes were employed as the input factors of ANN, while the number of monthly breaks was used for the prediction output. Spatial distribution of the pipeline break-event data was analysed and visualised by GIS. They compared their findings to the traditional processes for determining the priorities of pipeline replacement. Their methodology was more effective and efficient even in situations where the break-event records are unavailable.

3.7 Using GAs and GIS in the Design of WDNs

The ability of genetic algorithms to search a solution space and selectively focus on promising combinations of criteria makes them ideally suited to such complex spatial decision problems. Decisions are often evaluated by the quality of the processes on which they are based. It is in this context that GIS and spatial decision-support systems have increasingly been used to generate alternatives to aid decision-makers in their decisions.

Data preprocessing is the reorganisation of the spatial data in GIS to facilitate the implementation of GA. Different GIS software products may have different types of data organisation and management. The spatial data provided by the GIS database should be preprocessed before they can be used for network analysis with GAs (Morley et al., 2001). Section 3.6 highlighted the importance of integration between optimisation and GIS in the design of WDNs.

Atkinson et al. (1998) illustrated some of the advantages of the integration of a GIS and a GA solver including the speed of processing and a common user interface. This optimisation software for water networks has highlighted the absence of an agreed standard for representing water network infrastructure and operating conditions. It was necessary to translate the representation of network infrastructure from the network-modelling software into a form that can be understood both by GIS applications and by a hydraulic network solver that can be automated for the optimisation purposes.

The 'GA*net*' user interface of Morley et al. (2001) takes this integration a step further and integrates the interface of the GIS application into its own. The authors adopted a simple mechanism for linking GIS information into a GA application by reading the data through some common file format. The plan produced using GA*net*' proposes the construction of 85 new pipes in addition to 750 existing pipes in the system. Six new pumping stations are proposed and three old pumping stations are identified for expansion with a total of 42 new pumps. Seven new ground reservoirs and elevated tanks are proposed. The objectives function determines the cost of a solution by summing the cots of network pipes and adding pumps and reservoirs to the new system. The authors concluded that 'GA*net*' should be viewed as a decision-support tool rather than a modelling tool.

De Schaetzen and Boulos (2002) presented a GIS-based decision-support system, called 'H_2OMAP Utility Suite', for use in WDN planning and management. It links a hydraulic network simulator with spatial technology and optimisation technique to address network modelling and asset management.

H_2OMAP Utility Suite reads GIS data, extracts necessary modelling information, and automatically constructs, loads, calibrates, and optimises a network model. It also runs and simulates various modelling conditions, locates system deficiencies, and determines the most cost-effective improvements for optimum performance. The optimisation model uses GA for solving network model calibration, field sampling design, pump scheduling, and network design and rehabilitation problems in an optimal fashion. The integrated approach offers a virtual geo-spatial environment to assist water industry executives and professionals in formulating, evaluating and prioritizing facility management and infrastructure security strategies.

A different theme is proposed by Bartolin and Martinez (2003). They presented an ArcView GIS (by ESRI) extension called "GISRed" used in modelling and calibrating WDNs using GAs. This extension can be used as a decision-support tool in the design and calibration of WDNs. The 'GISRed' is capable of simulating, analysing, and retrieving the network resilience under certain conditions using integration of hydraulic solver engines. To construct a model, the 'GISRed' tool incorporates node and link themes in the ArcView GIS window. This window contains all the necessary features and references to represent a WDN model. The 'GISRed' offers a module based on GAs to fine-tune some of the network parameters for network calibration. The GA module offers the ability to manipulate parameters such as the number of generations, selection type, population size, crossover and mutation. The GA module has been successfully applied to calibrate the WDN (~1,200 km in length and serves ~400,000 consumers) of Valencia (Spain).

The integration of GIS and GA have been used to find the best route of water pipelines. Ebrahimipoor et al. (2009) presented a study to find the best route for constructing a water pipeline between a dam and the city of Qom in central Iran. The main aim of their study was to investigate the possibility of using GAs and shortest path algorithm in geographic information systems to solve pipeline routing problem. The required GIS information are gathered using topographic databases and land-cover maps, pre-processed, and converted to GIS database for GA use. In this paper, the authors compared the route proposed by the GA with the existing routes. The route proposed by GA proved to be 20% more cost effective than the existing routes by reducing the length of pipelines and crossing expensive paths such as bridges.

3.8 Chapter Summary

A broad introduction to the problem of WDN optimisation has been provided in this chapter with a focus on GA optimisation. Optimisation methods used in WDN modelling were reviewed in more detail. The review covered both non-evolutionary and evolutionary algorithms.

Several practical WDN design issues were examined, with regard to their ability to provide a GA optimisation design algorithm which can be applied to real water systems. Some of the most important topics discussed included network design and rehabilitation, the requirement of loops for redundancy, the analysis of design pumps, and network calibration.

The application of GIS to WDN design was also presented. The chapter demonstrated how GA and GIS can be integrated to assist in the design of WDNs. Appendix A covers an introduction to these two techniques.

The next chapter will introduce the early work performed in this research.

Chapter 4

Initial Model Formulation

4.1 An Introduction to the Chapter

The previous chapters discussed the use of GA and GIS in the design optimisation of the WDN. None of these studies tackled the problem of minimising network pipe length and land access cost whilst maximising the amount of water delivery to a demand area.

It was, therefore, necessary to undertake some initial basic experiments to test the potential compatibility of GAs for use in the problem. To achieve its goal, this experimentation would need to consider a representation of the problem, a fitness function and the operators necessary for GA use.

This chapter will first describe the steps involved in the initial problem formulation. This involves the definition of the problem and the identification of the parameters as well as chromosome representation and world representation.

It is important to note that this overview of the procedure does not constitute detailed operating instructions. It is rather a simple overview of procedures and options. Chapter 5 includes some examples of more developed formulation and other aspects described in that chapter.

4.2 Initial Design of the Fitness Function

It can be noted from chapter 3 that WDNs design optimisation is complex involving a large number of different components. Water supply planners use a range of models to support decision-making.

Generally, WDNs consist of the following components: water sources, demand areas, valves, and pumps all connected by pipes. A source is usually connected to one pipe. One or more pipes can be connected to a demand area. WDNs supply water to consumers with adequate head and with as little pipe leakage as possible. Pipe leakage increases with higher pressure. Therefore, a suitable compromise must be found by optimising the schedules of the control devices, pumps and valves, in the network. This involves the determination of pipe sizes from a set of commercially available diameters which ensures a feasible least-cost solution. Various methods with different degrees of success have been devised by different researchers to solve this problem.

The optimisation output in this work is defined as a cost. Since the cost is something to be minimised, optimisation becomes minimisation of an objective function, say, f(x).

The design of WDN is formulated as a least-cost optimisation problem with a selection of pipe sizes as the decision variables. The nodal demand, and minimum head and velocity requirements are assumed to be known in this chapter. In a general form, the optimisation of WDNs can be stated as:

Find the least-cost combination of pipe sizes while satisfying the following conditions:

1. Conservation of mass: inflows and outflows must balance at each node,
2. Head loss in each pipe follows a known function of the flow in the pipe, its diameter, length, and hydraulic properties,
3. Pressure head and flow requirements: Minimum pressure must be provided at network locations for a given set of demands, and
4. Acceptable pipe sizes: diameters are selected from an admissible set.

It was necessary to devise an initial fitness function for the problem. This function was simple enough to aid in the determination of GA use to address the problem. It was also important to select aspects from the WDN design optimisation that could be reasonably expected to be known at an early stage of the design process and could be expressed numerically.

The objective function is used to provide a measure of how individuals have performed in the problem domain. In the case of a minimisation problem, the fit individuals will have the lowest numerical value of the associated objective function.

An important issue associated with GAs is the issue of chromosome fitness. In the simplest case, the chromosome fitness value is set to be equal to the raw objective function value. In order to improve performance of GA, another function, the fitness function (ff), is used to transform the objective function value into a measure of relative fitness (De Jong, 1975) by scaling the raw value of the objective function, thus:

$$F(x) = \alpha\,(f(x)) \tag{4.1}$$

Where 'f' is the objective function; 'α' is a linear scale that transforms the value of the objective function, and 'F' is the resulting relative fitness.

Scaling may be done using a linear or power law or some other function. In cases when fitness calculation is very time consuming or very complex, an approximate scaling value may be used to drive the GA search process (De Jong, 1975; Beasley et al., 1993*a*). Since the fitness function in this work depends on many design aspects, an approximate scaling value is used for each component of the fitness.

Since the main objective of the design of WDNs is to supply water in sufficient quantities at a reasonable cost to existing and future consumers, each solution is evaluated using a fitness function that is specific to the problem being solved. Several fitness functions have been tested and developed in this work. As an example, a fitness function might consist of the cost of pipes and their installation and the benefit from the supply of water. This ff is used to demonstrate the initial GA formulation. The chosen objective function (OF) can be written as:

OF = Cost of pipes & installation + Land Access Cost − Payment to the supplier

The optimal solution is one which minimises the fitness function. In order to calculate the fitness, some cost figures are required. These figures can be obtained from industry such as material cost and construction cost for discrete pipe diameters.

The three aspects of the fitness function will be discussed in this section.

4.2.1 Material and Installation Cost

The cost of the pipes and their installation is a cost unit multiplied by the length of the pipe. Different diameter sizes have different material and installation costs.

$$C_{MI} = \sum_{i=1}^{n} L_i \times C_{MI}(D_i) \tag{4.2}$$

where C_{MI} is the material and installation cost of the pipe, L_i is the length of the i^{th} pipeline, $C_{MI}(D_i)$ is the material and installation unit price of the pipeline i under diameter D_i, D is the internal diameter of the selected pipe, and n is the total number of pipes in the system.

Water flow in a pipe network satisfies two basic principles, conservation of mass and conservation of energy. Conservation of mass states that, for a steady system, the flow into and out of the system must be the same. This relationship holds for the entire network and for individual nodes. A node is included in a network model at (1) a demand location and/or (2) a junction where two or more pipes combine. One mass balance equation is written for each node in the network. Hence, equation 4.2 describes the total materials and installation cost of the pipes in the network subject to:

$$Q_i^{in} - Q_i^{out} = Q_k\,, \qquad k = 1,\ldots,J \tag{4.3}$$

$$D_{min} \leqslant D_i \leqslant D_{max}\,, \qquad i = 1,\ldots,N \tag{4.4}$$

Where J is the number of existing nodes in the network; Q_i^{in} is the flow rate into pipe i; Q_i^{out} is the flow rate out of pipe i; Q is the demand at a junction node; Q_k is the required demand at consumption node k; D_{min} and D_{max} are the minimum and maximum commercially available pipe diameters; and N is the number of commercially available pipe diameters.

The last constraint requires that the optimal pipe diameter should be between the maximum and minimum commercially available pipe diameters.

The demand is represented in this work as a lumped demand for each demand area. This is because the these demands are uncertain since they are estimated from the local user base that cannot be predicted exactly and they vary almost continually.

Figure 4.1 provides an example of pipe representation by Cartesian coordinates. The pipe starts with node i which has the coordinates $(x_i,\, y_i)$ and ends with (x_j, y_j) for node j.

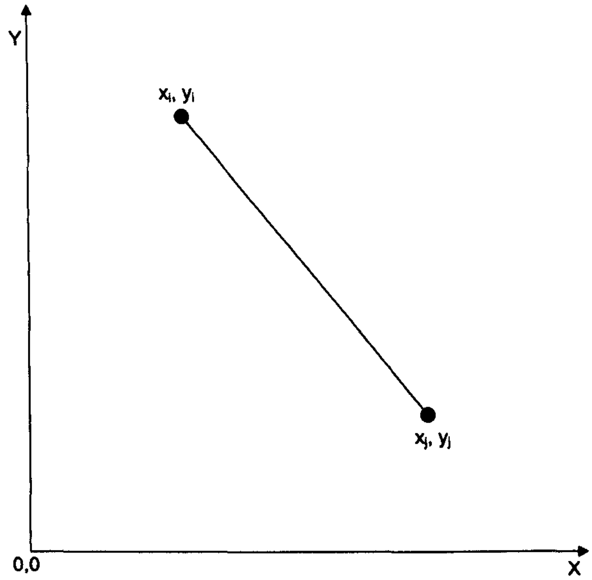


Figure 4.1: An Example of Representation by Node Coordinates

The distance d, which represent the length of the pipe, between two terminal nodes is taken to be a straight line distance (see section 4.3 for more details). Thus, for two nodes i and j this can be calculated as:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{4.5}$$

Where d_{ij} is the straight line between the two terminal nodes i and j.

4.2.2 Land Access Cost

Any area represented or modelled for GA assessment throughout this and the following chapters is referred to as the 'world'.

The land access cost is the cost of crossing a specific area. This cost element represents the cost of disruption caused by the construction and operational processes of the WDN at each grid cell within the world. The construction of WDN in populated urban settings significantly increases costs and ensures the indirect social costs associated with interruptions to the flow of traffic and obstacles to both businesses and the public sectors. Some of the additional variables that influence the cost are:

- Topography, e.g. slope and grading.

- Geology, e.g. rock and soil.
- Land-use, e.g. water bodies and roads

Furthermore, the potential costs of environmental impacts of the proposed pipeline should be considered and may involve the following:

- Land-use and recreation
- Vegetation and wildlife
- Water use and quality

The cost values are preferably based on empirical data, or else on decision-maker judgement.

Cell based datasets (e.g. grids) are well suited to represent traditional geographic phenomena that vary continuously over space such as cost, elevation, and slope (Heywood et al., 2006). Land access cost is assigned to each cell in the raster grid where each cell in the raster grid represents the features in that part.

Hence, land access cost is the cost of crossing a specific feature in the world. This cost can be expressed by:

$$C_{LA} = \sum_{i=1}^{n} \sum_{j=1}^{m} LS_{ij} \times C_{ij} \tag{4.6}$$

Where C_{LA} is the land access cost; C_{ij} is the access cost of the cell where the pipe segment passes; LS_{ij} is the length of the j^{th} pipe segment in a cell of the i^{th} pipeline; n and m are the number of pipelines and number of pipe segments in i^{th} pipeline respectively.

The land access cost calculation uses the cost to move through each cell as the unit of distance. To begin with, the land access cost calculation requires another grid defining the cost to move through each grid cell. The cost of each grid cell in the cost grid represents the sum of different costs. GIS is used to incorporate location based information into the calculation of a land access cost calculation.

The input for the cost calculation consists of two grid maps: (1) a map showing only the real-world features that function as sources (source-grid), and (2) a map of the matrix consists of different cost types (cost-grid). On this cost-grid, every cell has a resistance value (cost), depending on its land-use type.

Figure 4.2 is an example of four types of land-uses. Each land-use has a different land access cost.

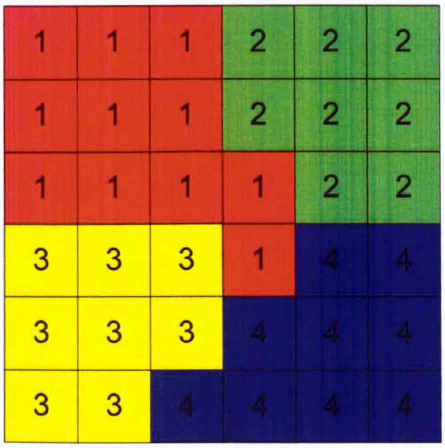


Figure 4.2: Source Grid With Associated Cost Grid

The calculation separates the surface into a single raster and derives for each cell a cost value for the access over the particular cell. The optimisation algorithm is an iterative process which finds the least-cost path between two grid locations, where the cost of crossing raster cells between the two locations is associated with the respective cost value of each cell. An example of a crossing pipe is shown in figure 4.3.

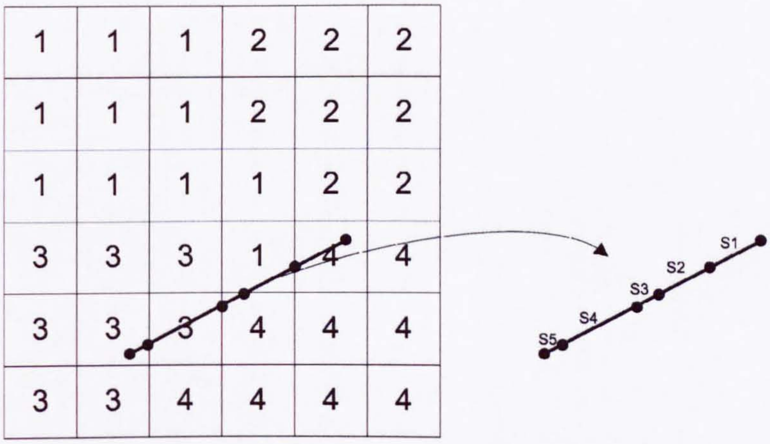


Figure 4.3: Example of Pipe Crossing the Area

The pipeline intersects with the vertical and horizontal grid lines in four locations. Therefore, the pipeline will have five length segments (S1 to S5). The total cost of this pipeline will be:

$$Pipeline\,Cost = S_1 \times 4 + S_2 \times 1 + S_3 \times 4 + S_4 \times 3 + S_5 \times 3$$

4.2.3 Payment to the Supplier

The third part of the fitness function is the payment in which the consumers have to pay to the supplier. The payment will be for the amount of water delivered to demand area multiplied by the unit cost of the supplied water.

The importance of demand areas is taken into consideration when the water is delivered to them. The amount of supplied water depends on the capacity of the source and the required demand. In this chapter, a single water source is used and assumed to be sufficient to deliver water to all demand areas.

The expression can be written as:

$$P(DW) = \sum_{d=1}^{l} DW_d \times C_{water} \tag{4.7}$$

Where $P(DW)$ is the payment to the supplier; DW_d is amount of water delivered to demand area d; C_{water} is the unit cost per cubic meter of delivered water; and l is the number of demand areas in the system.

The fitness function of an individual solution is defined in terms of its 'cost'. From equations 4.2, 4.6, and 4.7, the fitness function can be formulated as:

$$ff = \alpha_1 \times C_I(D) + \alpha_2 \times C_{LA} - \alpha_3 \times P(DW) \tag{4.8}$$

Where $C_I(D)$ and C_{LA} is the cost, and *P(DW)* is the payment, are as described above. α_1, α_2, and α_3 are coefficients selected by the decision-maker.

It is Possible to consider sub-sets of the general problem and to examine the effects of each of the aspects of the fitness function. This can be achieved by making some coefficients equal to *zero*.

4.3 Initial Chromosome Representation

In order to apply optimisation algorithms in an efficient manner, it is necessary to find representations which allow for iterative modifications with bounded influence on the objective values, (i.e. exploitation). The problem space, i.e., the representation of the solution candidates, should be chosen in a way which allows the construction of a correct solution to the problem in hand.

'Chromosome' is the name given in GAs to the representation of a possible solution to the problem being considered. There are normally several possible alternative ways to represent a solution.

In GA implementation, the choice of chromosome representation is crucial since it influences the GA's performance significantly. Solving any optimisation problem by means of GAs starts by deciding on which coding scheme to implement. Various schemes exist such as binary, integer, and real coding. For more information about the representation see appendix A.

The encoding strategy in this research uses chromosomes whose genes are represented by real numbers, i.e. it is a real valued chromosome with a fixed length. The principal advantage of such a representation is that the decoding stage to obtain the phenotype is unnecessary since the genes are already in the required form. This has a clear performance advantage for the algorithm, reinforced by the fact that floating point genes are easier to manipulate than their bit string equivalents and occupy significantly less memory space. Many researchers such as Janikow and Michalewicz (1991), Herrera et al. (1998), and Ortiz-Boyer et al. (2007) found experimentally that such real number encodings out-performed their binary equivalents both in computation performance and also in their flexibility for the implementation of GA operators. Real encodings are of significant benefit to the engineering design optimisation field as there are typically a large number of parameters to describe the design options that can be chosen.

The mechanisms of crossover and mutation that are normally associated with the bit string representation are appropriate for use with real number encoding. However, the representation allows new forms of crossover to be considered. Similarly, the mutation operations can be devised for use with real numbers. It is possible to add or subtract from the existing value of the gene or average the value with a random number, rather than simply replacing a gene with a newly generated random version.

The representation proposed in this chapter is designed to create rectilinear networks that contain loops and consist of a single source and multiple sinks (demand areas). These source and demand nodes together form a set of nodes that will be referred to as 'basic nodes'. All links between nodes are assumed to be straight line segments. Therefore, to obtain a rectilinear layout, some 'dummy' nodes are required in addition to the basic nodes to produce the necessary elbows, and intersections in the final layout. In this thesis, the term "intermediate node" is used. For a complete description of the intermediate nodes, see (Davidson and Goulter, 1991).

The important aspects in the initial chromosome representation are the nodes and their positions and the links. In this case a chromosome must contain:

1. Two-dimensional node coordinates
2. The connection between nodes
3. Node type

4.3.1 Two-dimensional Node Coordinates

The nodes of the network must be positioned within the world. It is, therefore, necessary to include their two-dimensional coordinates (x, y) and their type. The coordinates are represented as integer values. Integer values of node coordinates are selected because achieving sub-meter accuracy in real-life applications is hard. Therefore, integer coordinate values limit the search within the boundaries of the world. In addition, searching through an infinite number of unnecessary possibilities can be easily avoided.

This part of the chromosome in the algorithm is represented by a sequence of integers. The gene of the first locus is always reserved for the source node. The other genes in the sequence represent other nodes in the network.

4.3.2 The Connection Between Nodes

The connection between the nodes represents the pipes along which the water will flow. In the initial model defined here, the properties of a connection are limited to:

- The start and end of nodes: pipes are assumed to be straight lines between the terminal nodes.
- The pipe diameter: for this problem formulation, only standard pipe diameters are allowed. The *zero* value in the chromosome means that no connection exists between entire nodes.

4.3.3 Node Type

The node type determines whether nodes are a source of water, demand areas, or intermediate node. For basic nodes, the source and the demand areas, it is necessary to know water availability or requirement respectively.

Several types of representations have been investigated. An example of a chromosome proposed in the initial problem formulation is shown in figure 4.4.

To Node

From Node	1	2	3	4	5	6
1	0	0	0	0	0	D4
2	0	0	D2	D1	0	0
3	0	0	0	D3	0	0
4	0	0	0	0	D4	0
5	0	0	0	0	0	0
6	0	D1	0	D3	0	0

Connectivity Matrix

X	Y
X1	Y1
X2	Y2
X3	Y3
X4	Y4
X5	Y5
X6	Y6

Node Coordinates

S
I
D
D
D
I

Node Type

Figure 4.4: A Typical Example of Initial Proposed Chromosome Representation

The above chromosome represents an example of a water network consists of six nodes and seven pipe connections, where D1, D2, D3, and D4 represent integer values of different pipe diameters from a discrete list of available pipe sizes; node types of 'S' represent source node, 'I' represent intermediate node, and 'D' is the demand node.

The set of all nodes in a network can be represented as a matrix in which each row of the matrix corresponds with a start node and each column corresponds with an end node. The term "Connectivity Matrix" is used throughout this thesis to refer to the connection between the nodes.

Evolution programmes use a random search to create the starting population of solutions. As a result of this random initiation of the procedure and the fact that program operators are blind search techniques, most solutions created by the algorithm may contain redundant links that should be removed in the interest of minimising length or cost. For example, if an algorithm is used to compute the flow on the links, the links that should be removed can be identified as those links which carry no flow.
In addition, nodes are randomly scattered within the world. These positions are adjusted during the evolution process in the favour of minimising length or cost.

4.4 World and Network Representation

In many ways, GIS presents a simplified view of the real-world. The processes involved are not always straightforward because realities are irregular and constantly changing. In addition, the perception of the real-world depends on the observer. For example, a surveyor might see a road as two edges to be surveyed, the roadwork authority might regard it as an asphalt surface to be maintained, and the driver will see it as a highway.

The water network and the service area features in this research are described in two different representations:

1. Point representation, such as pole and nodes, illustrates a location of a point on the surface of a map of a given scale. The points are represented by an x - y coordinate pair.
2. Line representation, such as roads, pipes and rivers, illustrates the location of a feature whose shape is too narrow to define an area on a map of a given scale.
3. Polygons representation, such as lakes, parks, and other land-uses, illustrates the location of a shape of a significant area on a map of a given scale.

The complexity of the world and its interpretations suggests that GIS system designs vary according to the preferences of their creators. The process of interpreting reality by using both a world and a data model is called "Data Modelling".

A world can be described in GIS only in terms of models that define the concepts and procedures needed to translate real-world observations into data that performs meaningful analysis in GIS. The set-up of the real-world model determines which data need to be acquired. The basic carrier of information is called "the entity". Further details about GIS and their models can be found in appendix A.

In this work, the world is represented using world data, raster grid, and area information. The solution to the problem, i.e. the WDN, is represented by extracting information directly from the solution candidate (chromosome).

4.4.1 World Data

The world data, available from GIS databases, are the information which describes the world for which the model will be built.

This information includes the boundary of the world which can be of any size. The world includes terrain, rivers, roads, buildings and water sources. It might also include land-use areas such as residential, industrial and agricultural areas. The world is always assumed to be rectangular.

4.4.2 Area Information

The area information includes area code, the importance, access cost, and water demand of the area. The area code is matched with the grid type in order to associate the area information with the related grid information.

Any area in the world, whether existing or prospective, could be of any shape or size and positioned in any orientation.

The property type of the area information might be defined by the user or might be a physical property. Table 4.1 lists area information properties and their descriptions.

Table 4.1: Area Information Components

Property name	Property Type	Description
Area code	User defined	A name which denotes the area type. e.g. D1 = Residential, D2 = Industrial, D3 = Retail, D4 = Lake ... etc.
Importance*	User defined	The priority of the demand area on a scale of 1 to 5. The importance defines the water right for each demand area.
Land access cost	User defined	The cost of crossing any particular part of the world is determined by the features in that part.
Water demand	Physical	The amount of water required (demanded) by each area.

*More information about the importance can be found in section 5.3.2

The representation of areas could be achieved by the use of a series of rectangles of varying sizes. The accuracy of such a representation would be dependent on the size of the these rectangles.

4.4.3 Raster Grid

A grid is a type of two-dimensional pattern. The term 'grid' usually refers to a series of intersecting axes or to the points of intersection of these axes (DeMers, 2005).

The raster grid divides the entire study area into a regular grid of cells in specific sequence. There are several points to make about the grid:

1. The default sequence of the grid is row by row from the top left corner.
2. Each set of cells has some associated values to form a layer (one set of cells and associated values is a layer).
3. There may be many layers in a database, e.g. ground elevation, land-use, and land-cover.
4. The grid is space-filling since every location in the world corresponds to a cell in the raster grid.

A rectangular grid is placed over the whole world. The grid size determines the minimum dimensions of features that can be modelled and is equal for the whole system. The smaller the grid size, the more details can be modelled but the larger the problem for a given size of world. The boundary and the grid system are represented by arbitrary lines. World data for each grid element can be extracted from a GIS database.

More features, smaller features, or a greater detail can be represented by a raster with a smaller cell size. Moreover, smaller cell sizes result in larger raster datasets to represent an entire surface. This requires greater storage space, which often results in longer processing time. This concept is illustrated in figure 4.5.

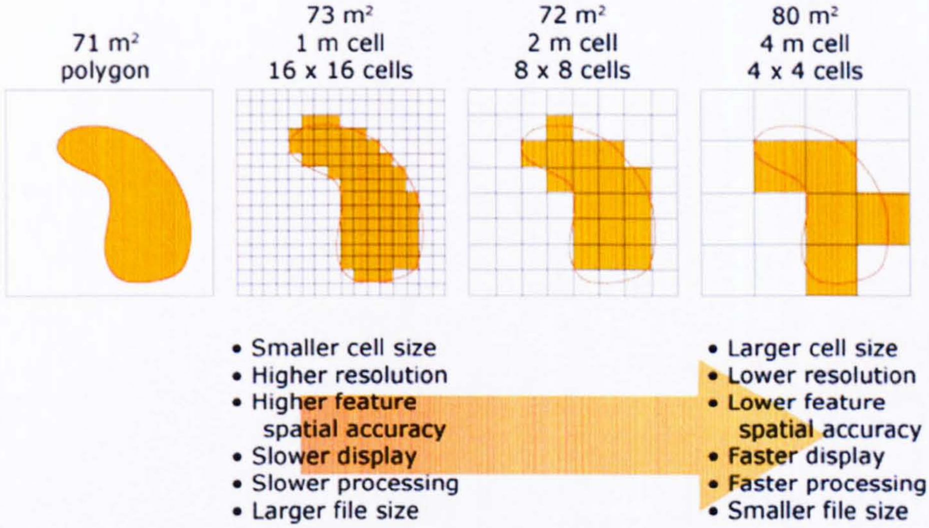


Figure 4.5: Raster Grid Resolution (Source ESRI (2011))

Cell numbering follows a specific pattern. An example of this pattern is shown in figure 4.6.

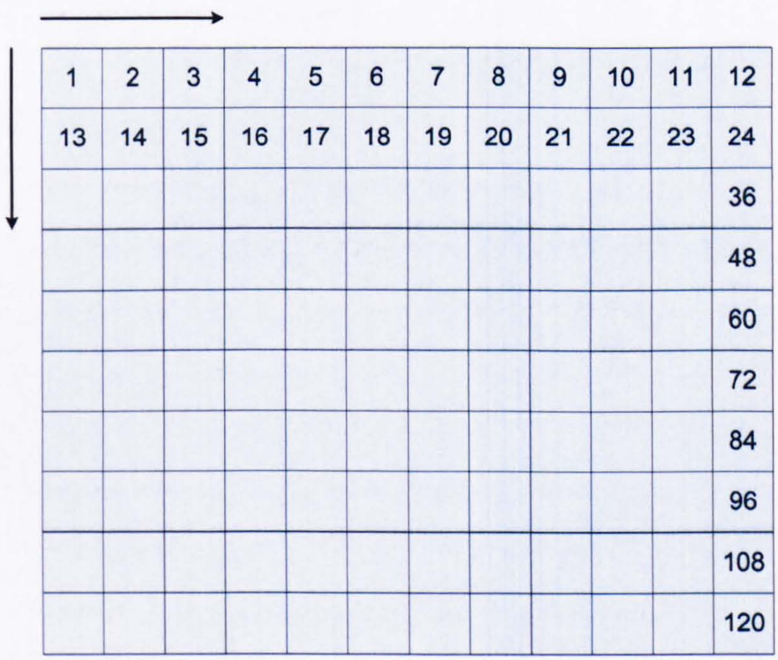


Figure 4.6: An Example Showing the Raster Grid Cell Sequence Pattern

The sequence starts from the top left cell of the grid and moves to the right increasing by one per cell until the end of the row. The sequence continues to the next row until the bottom right of the grid.

Such an implementation may require large amounts of memory and large computation power. In addition, the solution to the problem is iterative and

the time taken to compute the solution is largely dependent on the size and configuration of the network model itself. In the example shown in figure 4.9, information would have to be stored for all of the $50 \times 50 = 2,500$ cells of the world. This would had a particular constraint in the late 1990*s* when the performance of personal computers was insufficient to perform large input data structures. Today's high-performance computers can perform complex computation algorithms in relatively short time-scales.
Despite the drastic improvements in computer power, there is still a need to improve the performance of the GAs to allow for more complex optimisations to be undertaken with acceptable efficiency and effectiveness.

In this work, the existing ground level for each grid is defined. Each grid has been assigned a unique code which represents the grid type. For example, a grid type may be a demand, roadway or stream. A cost is incurred for placing a water pipe across any given grid element. The cost depends on the world features in that element.
Table 4.2 lists raster grid properties and their meanings.

Table 4.2: Raster Grid Components

Property name	Description
Grid size	Square dimensions for each grid cell. e.g. grid size might have dimensions of 10m x 10m. The whole system has the same grid size.
Grid type	A name which denotes the grid use. e.g. D1 = a grid belongs to a residential area, River = a grid represents river.
Grid access cost	A number which represents the cost of crossing the entire grid. e.g. access cost to place a pipe across a river is more expensive than placing a pipe line in a flat area.
Ground level	Existing ground level of the entire grid above an arbitrary datum.

4.4.4 Water Network Representation

Information, data and maps needed to analyse the WDN are today available and the use of information technology and GIS is becoming common in water industry in order to model systems and support management. In particular, knowledge of the exact location of demands (e.g., the number and position of connections to the demand areas along mains of WDNs) permits a more detailed analysis of the hydraulic system.

The model uses two primary model element types. These two types are represented by specific model elements:

1. Links (linear features):
   - Pipes
   - Pumps
2. Nodes (point features):
   - Reservoirs (source locations)
   - Demand locations
   - Pipe junctions

All links are connected between two terminal nodes. All intermediate nodes form a connection between two pipes. A demand node is a node that is located inside a demand area and can work as an intermediate node. Reservoirs, in model terms, are large water supply bodies with known locations.

Unique identifiers are required for each component (tied to each spatial feature by the GIS); connectivity must be specified (typically by specifying the start and end node to each link). The spatial locations of these various elements are stored in the GIS database. It is also convenient to store the coordinates of the point features in data tables.

Each model component type has unique values associated with it (e.g., pipes have a length and diameter); these values change over the simulation run. Other input datasets such as demand amounts are required.

Although the geographical location of the node is required by the model, it is also necessary to assign (x, y) coordinates to the nodes so that the network can be visualised. The corresponding integer values of the node number are assigned to nodes for the visualisation.

4.5 Initial Population

Networks consist of a set of nodes, where all of the nodes share some connection to other nodes. A pipe network, like a tree, is a collection of nodes and edges (pipes), but has no rules dictating the connection among the nodes. These connections are generated randomly among nodes in the initial population.

The process of generating random pipes among nodes results in directional flow pipes. Unidirectional edges are drawn as an arrow, showing the direction of the pipe flow resulting in branched or looped water networks.

Connections to the source node are not allowed in the initial population. The coordinates of the source node are fixed. In addition, it is always assumed to be node number one in the sequence.

If bidirectional flow occurs at any stage of the GA process due to crossover or mutation operators, one direction will be removed as cycles are not permitted in this work.

A link is composed of a pair of nodes; i.e. the links have direction and point from one node to another. A demand area is connected to the source if the direction of flow is going through consecutive nodes until it reaches to the node inside the demand area.

Random coordinates, except the source node, are assigned to the other nodes in the system constrained by the world boundaries.

4.6 Initial GA Operators

Standard GA operators are used in the initial problem formulation. The individuals in a population are selected for reproduction according to their fitness values. Selection, crossover, and mutation used in the initial problem formulation are will be explained in this section.

4.6.1 Selection

In GAs, selection is usually the first operator applied to population. Selection allocates reproductive opportunities for each individual in the population. The fitter the individual, the more times it is likely to be selected for reproduction.

Several selection mechanisms have been suggested in the literature, although there are no general guidelines on which to use on a given problem.

For model demonstration in the initial testing, two parent chromosomes are selected from the population according to their fitness values for reproduction.

4.6.2 Crossover Operator

Crossover chooses a location on the two selected individuals (parent 1 and parent 2) and exchanges contents between them to produce two new offspring (children).

Two conventional types of crossover are developed and tested in the initial problem formulation, namely, single point crossover and two point crossover. In this chapter, the single-point crossover is used and illustrated in figure 4.7.

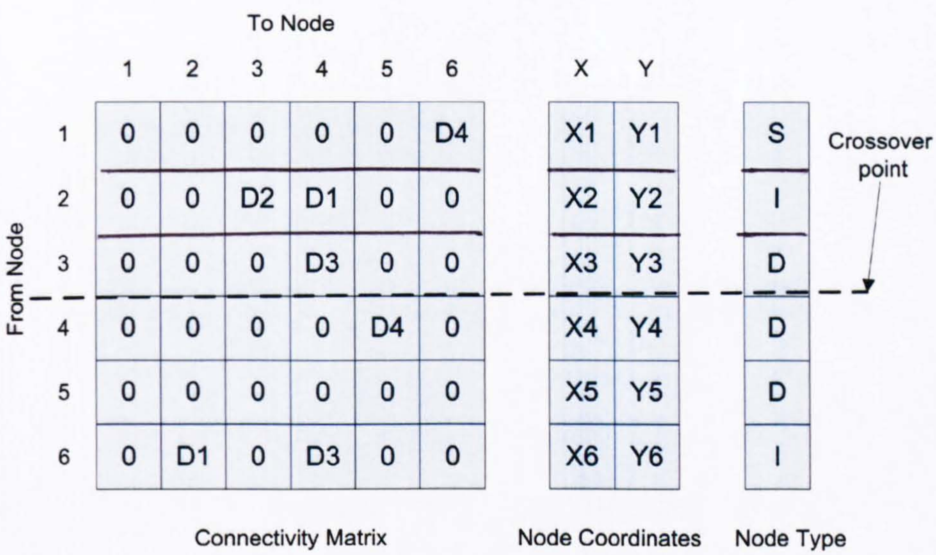


Figure 4.7: Single-Point Crossover

In the above crossover example, node 3 is randomly selected to be the crossover point. All the genetic materials above the crossover point from the first parent and below this point from the second parent will form the first child. The rest of the genetic materials will form the second child.

The crossover exchanges information according to a randomly chosen crossover point of connectivity and coordinates matrices between individuals. The crossover point can be fixed at a specific node for the whole GA run or may vary for each generation.

A crossover probability of 100% is used in this work. This indicates that all the selected chromosomes are used in reproduction. If no crossover is performed, offspring will be the exact copy of parents.

4.6.3 Mutation Operator

After the offspring have been created, they can then be permuted. Mutation randomly alters the value of a number within the individual. Different mutation rates are assigned to each matrix. The mutation may change the position of the nodes or alter a cell in the connectivity matrix. Figure 4.8 shows an example of the mutation.

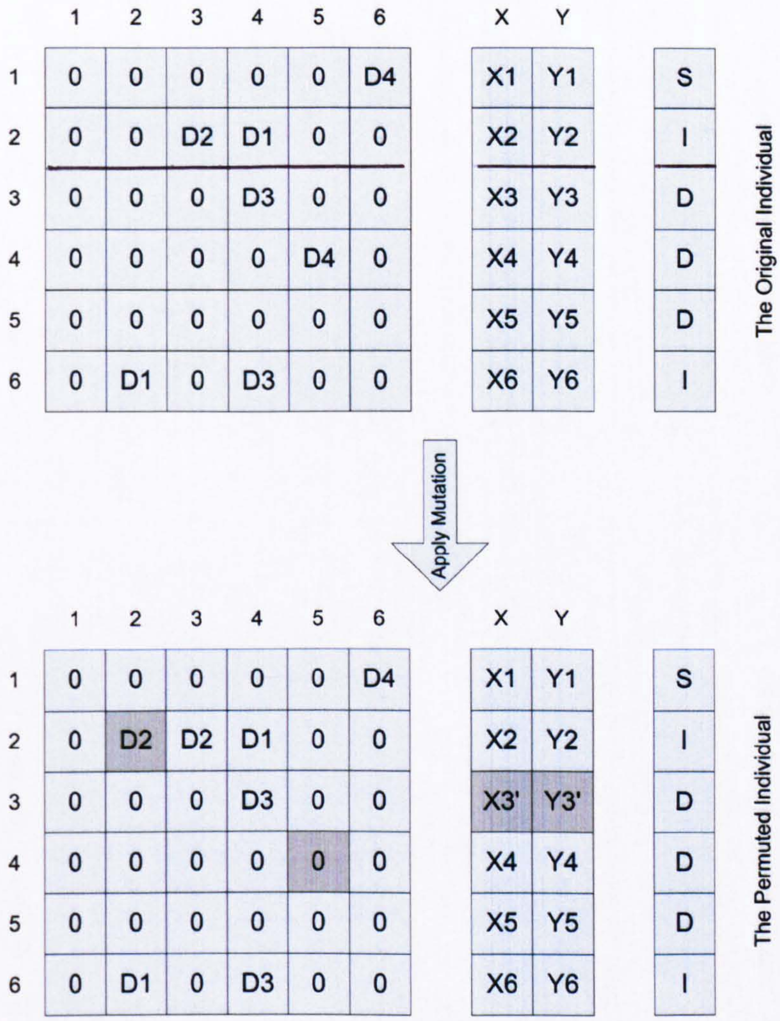


Figure 4.8: Mutation Operator

In the above mutation example, two random cells are selected in the connectivity matrix and their values are changed to a randomly selected pipe size from the list of available pipe diameters. In addition, two events occurred: (1) a link is removed from the network as in cell (4, 5) and (2) a link is added to the network as in cell (2, 2).

If a node is selected for mutation, all of its *(x, y)* coordinates are permuted within the boundaries of the world.

4.7 Initial GA Tests and Results

A computer program has been developed to carry out this procedure. This section demonstrates the initial experimental works in programming and developing GAs. In addition, it shows how GAs are used to develop a water distribution system in a designated world.

The basic GA is implemented in this program by means of selecting a number of individuals from the population; recombining them in some fashion to produce a number of offspring; introducing some mutation factor; evaluating the resultant offspring with respect to their fitness as solutions for the problem at hand and finally reintroducing (or replacing) the organisms into the population. It is also necessary to consider at what stage the operation of the algorithm should be terminated.

To demonstrate the initial performance of the program, an example of a town of a population size of $65,000\ to\ 100,000$ is selected. According to the Office for National Statistics (ONS) in the UK, the average area of a town would be $20.25\,km^2$ (Office for National Statistics (ONS), 2005). Since the world is always assumed to be rectangular, the average dimensions of the world will therefore be approximately $4.5 \times 4.5\,km$.

The grid size determines the minimum dimensions of features that can be modelled. A proposed town model, the whole area of the grid being termed the 'world', for use is shown in figure 4.9.

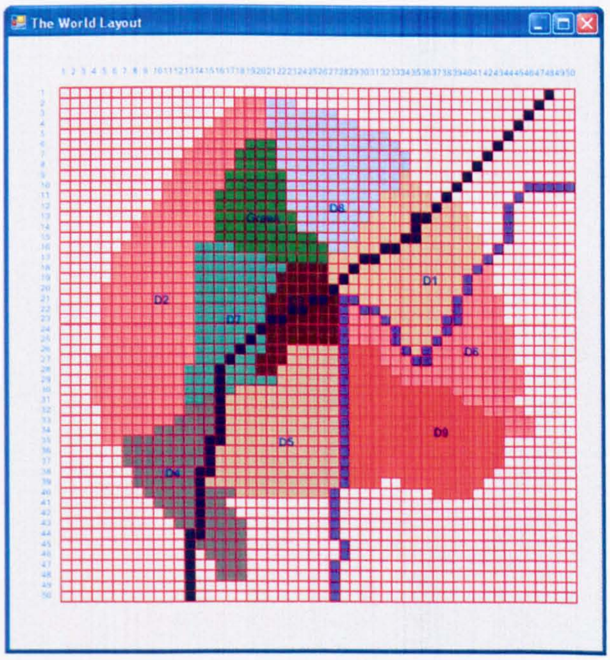


Figure 4.9: The Proposed World Model

This town is based on an existing town in the UK but its infrastructure has been removed and only the geometry retained. The dimensions of the world are 5 $\times$ 5 km. The selected grid size is 100 m. The town includes nine demand areas (D1 to D9), a green area and a city centre. A rail and a river pass the city. This world, which includes all the necessary features, can be used in the design of the WDNs.

The values for some necessary GA parameters must be determined before the algorithm can be used. These parameters are:

- Population size = 200
- Number of nodes = 12
- Number of generations = 3,000
- Mutation rate applied to connectivity matrix = 10%
- Mutation rate applied to node coordinates = 30%
- Crossover point is random for each generation

In this work, the speed of processing is not a concern, although every effort is made to run the GA algorithm efficiently. As the speed of new computers is increasing rapidly, it seems reasonable to let GA perform longer runs. Nevertheless, in cases when fitness function calculation is very time consuming or extremely complex, an approximate function may be used to drive the GA search process.
For the above parameters, the time of processing was approximately 50 minutes using a PC with Intel Quad Core processor @ 2.40 GHz and 4 GB RAM under Windows XP service pack 3 platform. The processing time for similar parameters using 25 meter grid size was approximately 15 minutes.

The initial model, which still need to be improved, is useful for generating a design of a WDN. The pipeline is connected to the source and used to deliver water to the demand areas where a demand node exists. The future improvements could be a decision mechanism which reflects the importance of each demand area. The decision mechanism controls the amount of water flow to a demand area. Figure 4.10 plots the GA convergence for one of the tests.

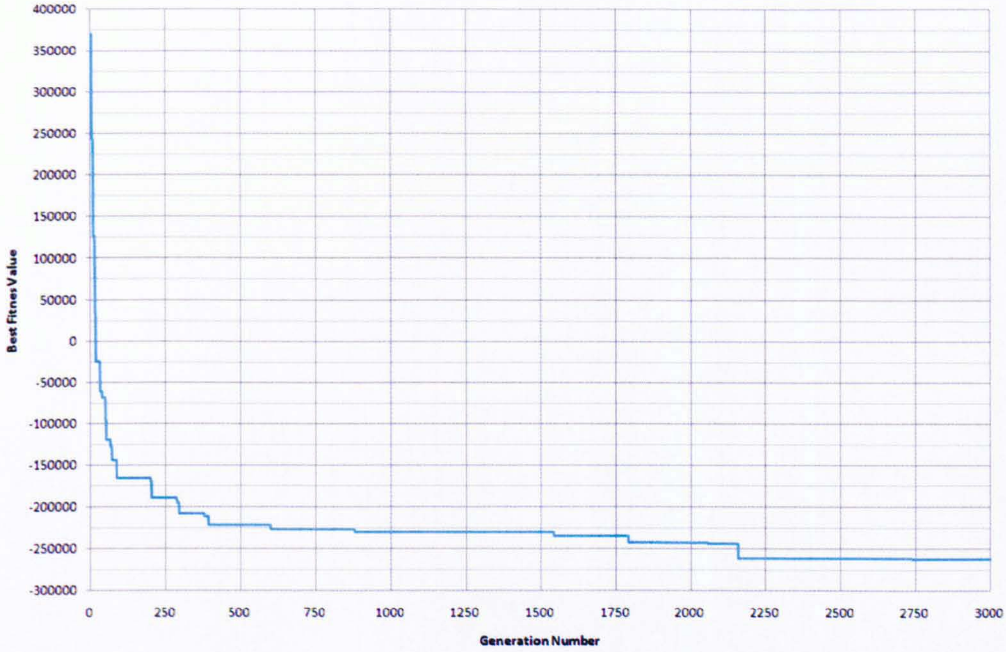


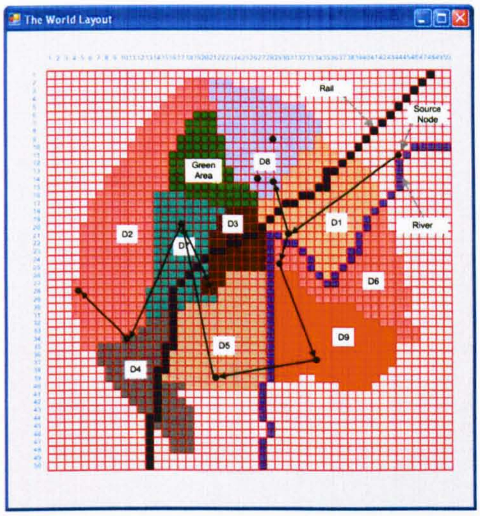
Figure 4.10: GA Convergence

Figure 4.10 shows that after rapid initial convergence; the solution is generally stable after generation 500 - 750. Other results from testing the model with same input parameters generally produce the same solution.

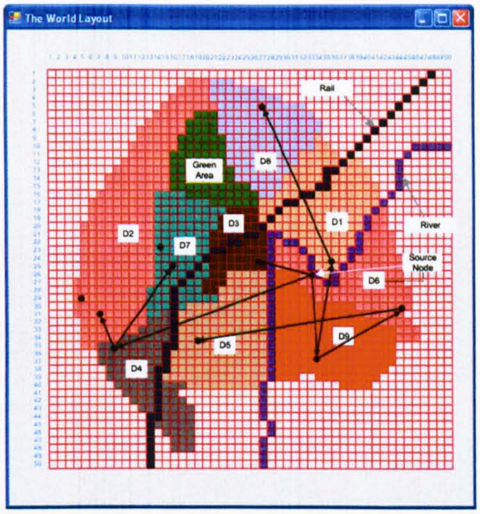
This problem had been 'solved' using a GA. A typical solution to the problem is shown in figure 4.11(a).

Decision-makers may change some network features such as source location. These decisions could be effective in generating WDN which may eventually produce optimal solutions with regard to cost or other design problems.

The actual estimation obtained will vary depending on decisions which can be made by the designers. For example, figure 4.11(b) shows a solution for the same problem but with a changed location for the source node. In addition, the network produced depends on the parameters used in the world definition and the fitness function. Evaluating such alternatives adds to the power of the process as part of a decision-support system.



(a) The water source is outside the city



(b) The water source is inside the city

Figure 4.11: Different Water Source Locations

The algorithm is a non-deterministic one. Hence, the results produced from one execution of the computer program will not necessarily be the same as another execution, given the same parameters. There is no guarantee with non-deterministic algorithms that any set of parameters will produce an optimal WDN during a single execution.

It follows, therefore, that very good results are possible even with a set of parameters that often produced bad results and, similarly, bad results are possible from a set of parameters that often produced good results. The objective in configuring parameters is to obtain the best overall average performance. The best breeding features need to be investigated.

4.8 Conclusion

This chapter presented the initial formulation of the problem and introduced model parameters. It discussed a possible design form of the fitness function. The chapter demonstrated the work done in terms of developing the world data and the chromosome representation problem. It also summarised the use of GA operators and demonstrated that GAs are potentially useful tools for building an optimisation based solution to a water planning problem.

The main points that arise from this chapter are:

- The use of a rectangular grid structure to represent a geographical area or a world is beneficial.
- The grids can have many properties which can be used to evaluate the fitness of the proposed solution and to impose constraints upon it.
- The structure of the chromosome is very important.
- The size of a chromosome made up of all the cells prohibits its practical application.
- The fitness function can take many forms and can be made up of terms.
- The encoding method presented and the operators used gave satisfactory solutions although the examples were limited.

Despite some weaknesses in the practicality of the initial formulation, it allowed for the successful testing of GAs and demonstrated the potential for GA use in addressing the problem.

The next chapter identifies and proposes specific developments which need to be introduced, with a rationale for so doing in terms of the development of initial problem formulation. In addition, the future improvements to the problem formulation is a decision mechanism which highlights the importance of each demand area. This mechanism controls the amount of water flow to a demand. A decision rule can be applied to how flow can be supplied from any node which reflects both physics and management policy.

Chapter 5

Model Development

5.1 An Introduction to the Chapter

Chapter 4 involved the steps to produce an initial prototype model which can be used to determine the necessary features of the 'final' model. This chapter discusses the steps involved in the development of the initial prototype model.

After a general description of the model, this chapter outlines the development of the GA formulation and details the components and operation of the current formulation. It explains the process of creating a decision mechanism tool that may reflect real decision-making in the design of WDNs.

A penalty method, which assigns a single numerical figure to each coded solution, is devised to formulate the optimal design of a pipe WDN. The fitness of a network design is taken as some function of the total network cost. Optimisation aims to minimise this function.

The chapter also describes a new chromosome representation of the candidate solutions.

The development of genetic operators such as crossover and mutation are explained in details. Other GA elements such as population size, age operator, and termination criteria are presented.

5.2 Model Description

The model tends to minimise the total costs required to transfer water from single or multiple sources to demand areas. The model consists of three main parts (1) the world; (2) WDN components; (3) GA components.

The world represents the environment in which the model will be built. The world is a graphical representation of selected features from the real-world (service area) in terms of land-use (e.g. rivers, parks, retail) and topographic data (e.g. valleys, mountains). Each feature has a unique ID within the boundary of the world. The boundary of the world is an arbitrary line determining the limits or extent of the service area.

WDN modelling provides an efficient way of predicting the network behaviour, calculating pipe flows, head losses, pressures and heads, reservoir inflows and outflows and operating costs. When dealing with large water systems, it is typically hard to model an entire water delivery system at its finest level of detail (e.g. including every existing pipe, valve and pump) (Walski et al., 2000; Vasan and Simonovic, 2010). It is not practical in large water systems to model every single component in the WDN. Certain simplifications are made to allow model implementation and an efficient GA testing in a timely manner.
In this thesis, the system is typically generalised, and only the larger pipes are included in the model. The demand points (nodes) of the model do not represent individual services (single sink or user), but collections of services (groups of sinks or users). These collections will be termed as "Demand Area" throughout this thesis. However, when a greater detail is necessary, portions of the model are removed and replaced with a greater detail.

The model use GA technique to find a near-optimal design of the WDN. The utilisation of GAs provides the opportunity to implement a decision model that enables a variety of factors to be explicitly included in the design of WDNs. For more details about optimisation using GAs and using GAs to assist in the design of WDNs, see section 3.4 and section 3.5.

A decision mechanism is developed in many stages. The design of the decision mechanism starts with investigating the requirements of such strategy. This includes a description of the elements required to enable the GA to be implemented. In this instance, the relevant data for the design of WDN are introduced. This leads to the introduction of the model formulation, which will be optimised by the GA. The decision mechanism is formulated for the WDN ensuring that the specific data requirements to implement the model are investigated. The data required for a simulation run are summarised as:

1. Service area components

    - Topology requirements: unique identifiers are required for each component in the service area (tied to each spatial feature by the GIS).
    - Boundary of the world and grid size.
    - Demand area components such as demand type (e.g. residential, retail), importance (see section 5.3.2 for details), and amount.
    - The amount of water source.

2. WDN components

    - Constant network components: number of nodes and discrete pipe diameter are values which remain constant for each individual over a simulation run.
    - Variable network components: number of pipes may be different for each individual in a GA iteration.
    - Hydraulic components (e.g. pipe roughness coefficients, pumping efficiency and maximum allowable nodal head).
    - Input driving datasets (e.g., price lists and curves).

3. GA components

    - GA operators: crossover type and mutation type with their associated rates.
    - Population size.
    - Termination criteria

4. Other external data

    - Input driving datasets (e.g. digital terrain file (DTF) and land access cost).

The world features and a WDN are objects, which can be referenced or related to a specific location in space. These objects have their own particular set of characteristics or descriptive attributes. These non-spatial alphanumeric data plus location information are stored and managed for all spatial features of interest. The spatial locations of service area features are stored in a GIS database in this work. It is also convenient to store the (x, y) coordinates of the point features (such as WDN nodes) in data tables. The model uses two primary model element types, namely the world and the WDN. These two types are represented by specific model elements. The representation of these features is detailed in section 4.4.

The model also is a GIS based tool with integrated database capability that can be used to model the flow, demand and pumping power in an optimal WDN. It contains pipe network hydraulic equations, real-world features, and a database (e.g. land access costs, terrain data). In addition, the model is a GA based model, which estimates an optimal WDN based on the pipeline features, topography, and amount of distributed water to demand areas due to the introduction of some decision mechanisms. This model evaluates the effects of design of WDN on the demand, flow, and pumping power and helps to find the best WDN design. Once the model has been constructed, the actual simulation run may be made. Results may be saved and printed (e.g. water pressure at selected nodes) or viewed graphically. A single water source is used and assumed to be sufficient to deliver water to all demand areas.

In the optimisation process of WDNs using GAs, the model is able to perform many sequential functions. These functions can be summarised as:

1. The first step is to represent the real-world features by an object defined in the computer software and prepare them for GA testing. Computer systems generally store numeric values. Therefore, numeric values are assigned to the characteristics of the real-world which are included in the GIS database. GIS data are set up in a text file which is read by the computer program. A typical example of this file is shown in figure 5.1.

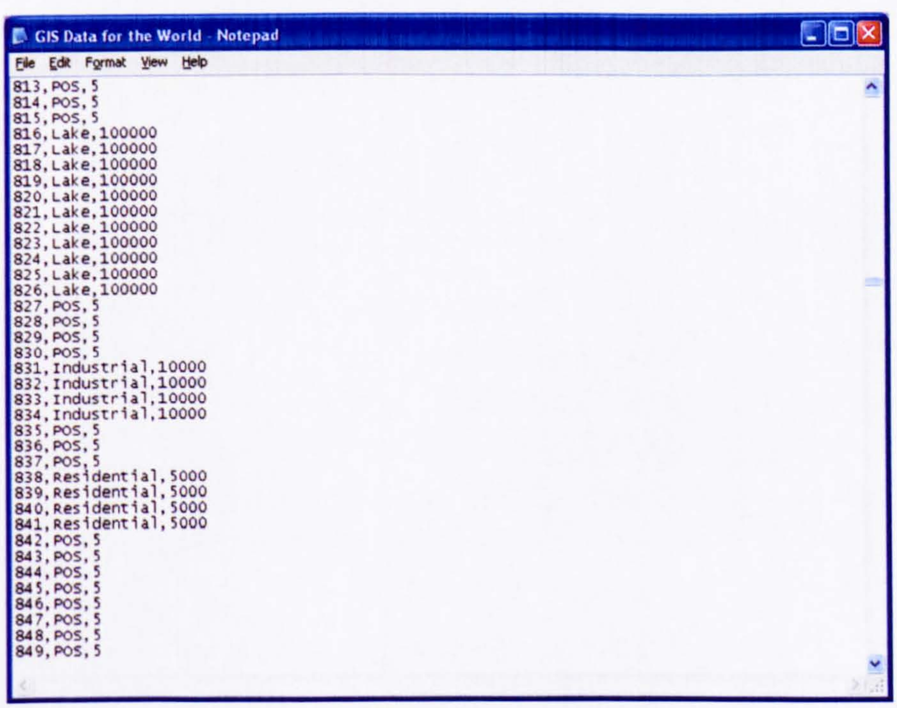


Figure 5.1: An Example of a GIS Data File

The information contained in the text file is stored in three columns separated by a comma. The information is; a cell number followed by a cell type then followed by the relevant land access cost. As an example from the above figure, cell number 816 in the sequence is a type of cell that represents a lake with a relevant access cost of 100,000; and cell number 838 in the sequence is a type of cell that represents a residential area with a relevant access cost of 5,000.

2. Optimise a WDN in terms of:
    (a) Maximise the supply to demand areas.
    (b) Minimise the power required for pumping the water to demand areas.
    (c) Minimise the wasted water.
    (d) Find the best network layout (best route with a minimum network length).
3. Provide an integrated environment for editing chromosome variables.
4. Provide a graphical view of the real-world objects along with the water network.
5. Plot the desired results.
6. Save the desired results in data files.

Network editing and checking

The network editing and checking tool is the user / decision-maker interaction with the solution during a GA run. It is used by the user when needed for further network analysis. A chromosome editing tool is devised to enable the user to edit the network using a graphical user interface. This tool provides the user the ability to edit some of the network properties, run a hydraulic / WDN analysis and view the results of the analysis. The editing tool is designed to change different WDN components that are grouped in one window tool. Once this window pops up, current chromosome values represent the current network configuration. Using this tool helps the user to better understand the behaviour of the model by changing some of the network properties. In addition, it enables the user to start the solution from a predefined or known solution (network). The editing tool window is shown in figure 5.2.

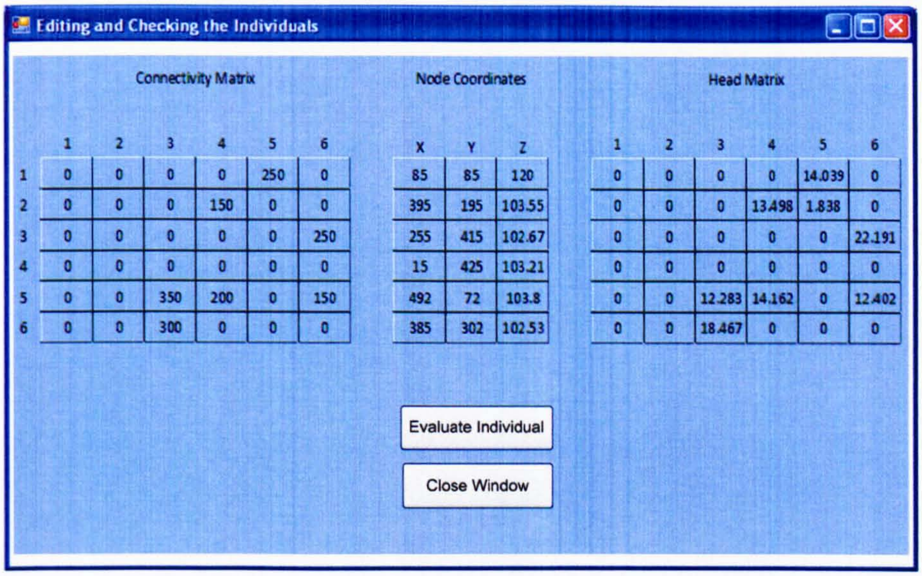


Figure 5.2: Editing Tool Window

Using the editing tool, the user can change the following features:

1. **Connectivity Matrix Values**

   Allows adding / removing pipes with different sizes from an admissible list to / from the model. This is also allows the possibility to add pumps (characterised by the links) and connect pipes to nodes.

2. **Node coordinates**

   Allows the three-dimensional (x, y, z) node coordinates to be changed within the boundary limits of the world. Editing (x, y) node coordinates moves nodes from current positions to different places and, consequently, will change the length of the pipes if exist. In addition, this process may convert a node from one type to another. For example, an intermediate node may be converted to a demand node if the edited node coordinates are placed within the boundary of one of the demand areas and vice versa. Editing the elevation (z coordinate) of the node changes node elevation above or under the ground level. As a result, the elevation of all pipe inverts connected to the node will be changed.

3. **Pumping head along the pipes**

   Allows to increase / decrease head values to / from the piping system. This allows adding / removing pumps (characterised by the links) to / from the WDN.

Additional tools have been developed to understand the behaviour of the model. These tools show many results that are generated by the running the

model such as the amount of the wasted water, total length of the network, and the total satisfied demand amount. In addition, they allow modelling different scenarios and specify multiple modelling alternatives on the same, or different, pipe network.
Finally, it is worth saying that the model can be 'paused' at any generation, which enables the user to view the results and perform network editing if desired.

5.3 The Decision Mechanism

The task of decision-making in the design of WDNs entails choosing between various alternatives. In this work, decision mechanism will be discussed under three headings:

1. Network layout geometry
2. Water allocation
3. Water distribution

5.3.1 Network Layout Geometry

Pipe network optimisation is a multidisciplinary task considering hydraulic, water quality, reliability, and network components requirements. Due to the complexity of the problem, most of the existing algorithms are restricted to consider the hydraulic and network components requirements (Dasić and Djordjevic, 2004; Afshar et al., 2005). The reliability requirement is usually addressed by considering branched or looped layout for the networks to be designed (Alperovits and Shamir, 1977; Quindry et al., 1981; Murphy et al., 1993; Dandy et al., 1996; Savic and Walters, 1997*b*; Cunha and Sousa, 1999; Boulos et al., 2001; Wu et al., 2001; Dasić and Djordjevic, 2004). Some researchers, on the other hand, have focused their attention on layout geometry optimisation neglecting the joint problem between component sizing and layout determination for pipe networks (Walters and Lohbeck, 1993; Davidson and Goulter, 1995; Davidson, 1999; Geem et al., 2002).

The joint problem of layout and other network components design of a WDN has been the subject of some investigations. Many studies in this theme have addressed the much easier problem of branched networks (Karmeli et al., 1968; Kessler and Shamir, 1989; Eiger et al., 1994; Alandi et al., 2007). Such systems, however, are not favoured in engineering practice, mostly because the optimisation models reduce the reliability in the design as they focus on minimising overall network cost (Mays, 2000). Looped systems are generally more desirable than branched systems, particularly in urban environments. This is because in branched networks any failure or scheduled maintenance of any of the pipes would lead to a part of the network being cut off from the source node(s).

The complexity features in the development of an algorithm capable of addressing this subject is the interrelation of layout, pipe sizes, nodal head, system hydraulics, demand amount, water quality, and reliability. In addition, to provide reliable systems, minimising cost in looped systems reduces the redundancy and hence the reliability of the system. This makes the optimal design of a looped WDN more complex than a branched WDN. A significant number of research worked on looped network optimisation (Alperovits and Shamir, 1977; Goulter, 1992; Simpson et al., 1994; Dandy et al., 1996; Walters et al., 1999; Cunha and Sousa, 2001; Eusuff and Lansey, 2003; Babayan et al., 2005).

In order to identify the nodes and keep track of the sequence of events, nodes are usually given a unique label. In this work, positive integers from 1 to N are assigned to the nodes and will be their names. These positive integers are called 'integer-labels' or simply 'labels'. The labels are not part of the basic structure of any network, and can be assigned arbitrarily or by some method. The interesting features of the network (such as whether it has a cycle, or is connected) exist independent of the node labels.
Number '1' is indicated by default as a "start" or "entry" node to start an examination or process the network. This is always considered as the node at the water source point. No node will receive *zero* label nor any negative number. The links do not need labels, because they can be identified by the labels on the terminal points (nodes).
Since a network is stored with node numberings, it is possible to create two representations of the same network which look quite different. For example, for a particular network, an identical network might exists with a different node labelling.

In this section, a decision mechanism is discussed for the network layout geometry in which cycles are not permitted in networks and should be removed. This is because the presence of cycles in networks violate some conditions in WDN analysis.

A path in a directed network that begins and ends at the same node is called a "cycle". The cycles allow the flow to be returned to the same node where the cycle started. A flow entering a node from the same node along the same path of pipes is not allowed in this work (see section 5.3.3 for more information about this decision). Hence, these infeasible cycles should be detected and removed from the GA solution.

Cycle identification is a control flow graph (CFG) analysis problem because it allows the basic blocks comprising each cycle to be identified (Tarjan, 1973; Sreedhar et al., 1996). A depth first search algorithm by Hopcroft and Tarjan (1973)(abbreviated as DFS) is used to determine whether or not a directed graph contains a cycle. DFS is a systematic way to find all the nodes reachable from a source node (this is explained in details in section 2.3.5). Once detected, these cycles need to be removed in this problem to produce a directed acyclic graph (network).

If cycles exist, flows in each of the loops must be corrected until the resulting error is within the allowable tolerance for the system. In this work, the flow is distributed into the system according to a decision mechanism and some user defined rules (see section 5.3.3 for more details).

After breaking cycles from the network, the optimisation algorithm is able to deal with both branched and looped networks and provide a certain level of reliability. The related network (graph) principles are presented in section 2.3 with a special emphasis to the cycles in networks (see section 2.3.4 for details). In addition, appendix B presents some related definitions to networks.

The problem of breaking cycles in a graph has been considered in many works. The main shortcoming of most of these researches is that they do not address in details the implementation details in cycle breaking. Orenstein et al. (1995) proposed an optimal algorithm for cycle breaking in directed digital circuit graphs. The proposed algorithm makes use of graph partitioning methods based on local properties of digital circuits. For example, it may not be necessary to break the easily testable cycles in the circuit.
Blanco et al. (2003) adopted a simple repairing operation to remove cycles. Once a cycle is detected in the network, one arc of the cycle is randomly deleted (this is repeated until a directed acyclic graph is achieved).
Levitin et al. (2010) proposed an algorithm, called the "Simple Cycle-Breaking Algorithm", that breaks all cycles in the graph that model communication networks. This is done by preventing certain pairs of edges from being used, sequentially, at some nodes in the graph when forwarding messages.

In this thesis, DFS is used to detect cycles in networks. A decision has been made to remove the last edge back to the starting node that formed the cycle. The process of detecting and breaking cycles will be repeated until acyclic graph is produced. Appendix C presents an example of the DFS process.

Figure 5.3 illustrates the decision process of deleting an edge using DFS algorithm.

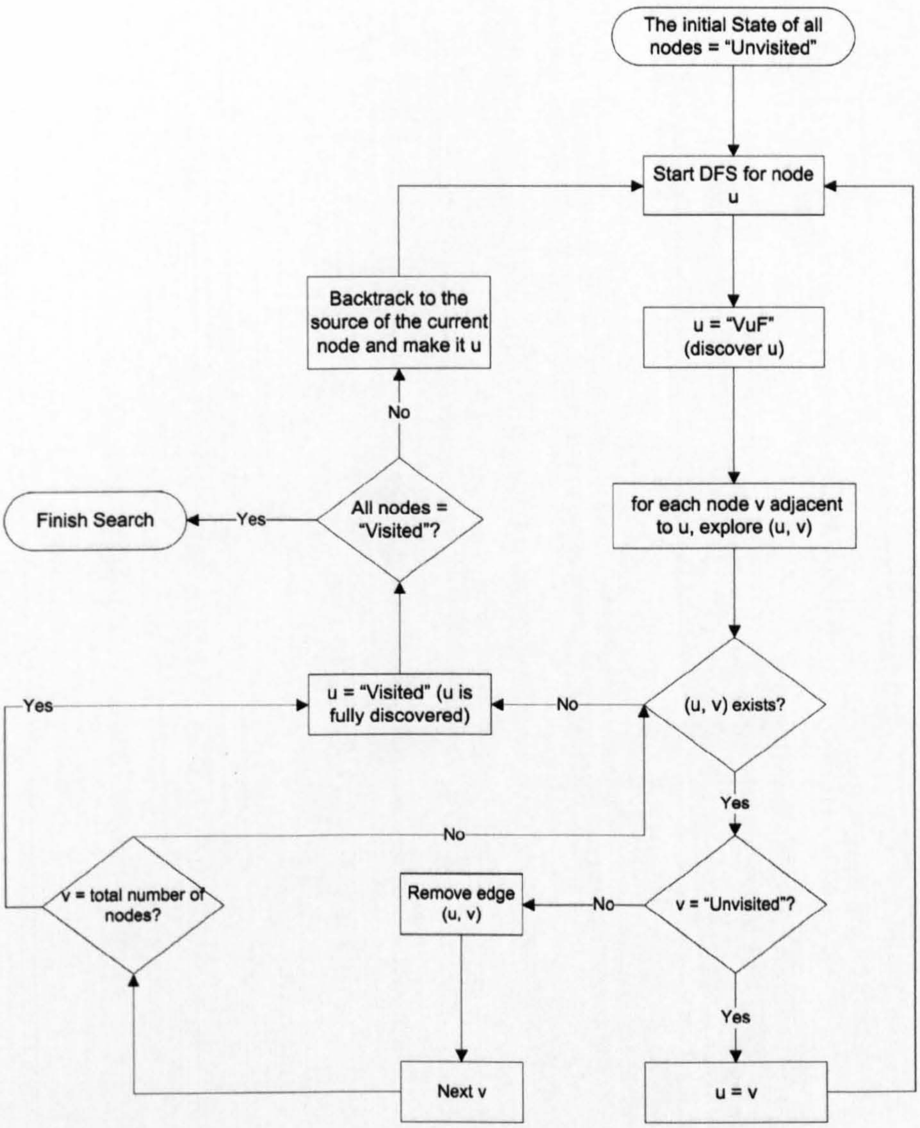


Figure 5.3: A Flowchart Illustrates Deleting Cycles Using DFS

The DFS concept and above steps are explained in details in section 2.3.5. However, in the above figure, one additional step is added to the DFS concept which is the decision of removing an edge that forms the cycle in the network.

A cycle starts from a node 'v' and travels through some set of nodes v_1, v_2, v_3, ..., v_k that then arrives back at 'v'. If the state of 'v' is "visited" then the last edge (i.e. (u, v)) is removed.

This decision is illustrated with an example of a network consists of cycles. The network has six nodes and seven edges as shown in figure 5.4.

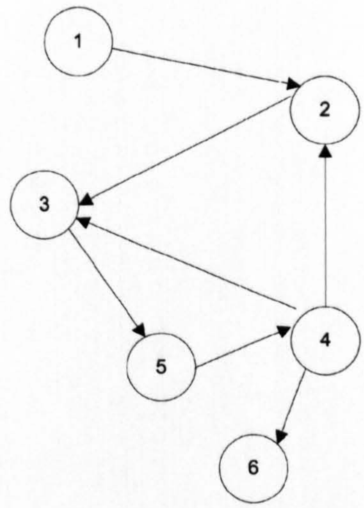


Figure 5.4: A Typical Example of a Network

The connections are presented in table 5.1, where node 1 is the source node:

Table 5.1: Connections Between Nodes in a Network Before Removing the Cycles in Example 5.4

Upstream Node	Downstream Node
1	2
2	3
3	5
5	4
4	2
4	3
4	6

DFS starts from the source node and the network is traversed by considering edge (1, 2) from the current node 1.

Node 2 is marked as "VuF" and will be the current node and the network is traversed by considering edge (2, 3).

The traverse process continues for nodes 3, 5 and 4 and these nodes are marked as "VuF".

At current node 4, the network is traversed by considering edge (4, 2) which will be deleted (see figure 5.5) because node 2 had been marked as "VuF" before (visited but not finished). The cycle is formed because a path travels through the following nodes: $2 \Rightarrow 3 \Rightarrow 5 \Rightarrow 4 \Rightarrow 2$

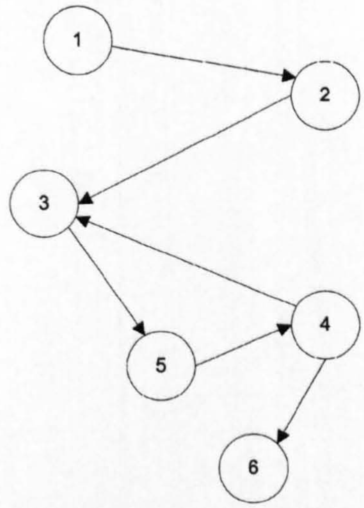


Figure 5.5: The Network After Removing Edge (4, 2) from the Original Network

The algorithm backtracks to node 4 and continues with the remaining nodes adjacent to node 4. Similarly, edge (4, 3) is deleted for the same reason. The network is shown in figure 5.6. Another cycle exists because a path travels through the following nodes: $3 \Rightarrow 5 \Rightarrow 4 \Rightarrow 3$

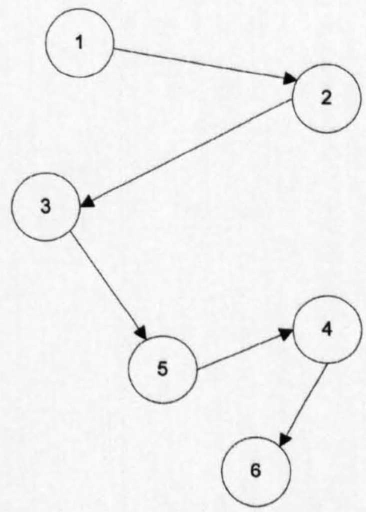


Figure 5.6: The Network After Removing Edge (4, 3)

The search proceeds to traverse edge (4, 6) and mark node 6 as current node ("VuF"). Nothing is connected to node 6 which is a dead end. Node 6 is marked as "Visited" and the algorithm backtracks along edge (4, 6) to node 4 again and makes it the current node. The algorithm traverses the network again at node 4 for any edge that missed earlier. If no edges exist or all edges are "Visited", node 4 is marked as "Visited" and the algorithm will backtrack along edge (5, 4) to node 5. DFS continues to backtrack the path that it has traced and deletes any cycles in the network until it has backtracked ("Visited") all the way back to the original source node 1. The final network will be the network shown in figure 5.6 and the connections are presented in table 5.2.

Table 5.2: Connections Between Nodes in a Network After Removing All Cycles From Example 5.4

Upstream Node	Downstream Node
1	2
2	3
3	5
5	4
4	6

5.3.2 Water Allocation (Water Rights)

Water allocation is the process of allocating available water to demand areas. Since countries and circumstances vary widely, there is no correct approach that can be replicated globally (Dinar et al., 1997; Quesne et al., 2007). The most effective means of any particular system of water allocation is influenced by local circumstances such as existing institutional and legal frameworks as well as the water resources infrastructure (Dinar et al., 1997). It is hard, therefore, to identify the solutions to water allocation problem.

Many forms of water allocation schemes attempt to combine both efficiency and equity principles. Through a detailed planning process and studies on water permit system, greater equity and efficiency in water allocation will be set in place (Quesne et al., 2007).

From the available water, it is possible to allocate water for ecosystem maintenance such as natural conservative parks and basic human needs such as domestic and retail. Other water allocation priorities can include strategic industries such as power stations and factories. Priority water allocations for social and environmental purposes are increasingly recognised in water law around the world (Quesne et al., 2007).

Water shortage happens when the total amount of source water is less than total amount of demanded water. With the increasing water scarcity (water shortage) in the world, there are two general approaches for mitigating the water scarcity problem, namely, supply enhancement and demand management (Griffin, 2006). Supply enhancement involves development of new water sources, while demand management means supply water within the current supply capacity. With the increasing cost of supply enhancement strategies and concerns on availability of additional water sources, the need for demand management has significantly increased (Griffin, 2006).

Meinzen-Dick and Mendoza (1996) and Dinar et al. (1997) defined several water allocation mechanisms such as marginal cost pricing, public allocation, water markets and user-based allocation. The water allocation decision mechanism in this work is based on public water allocation. Public allocation is seen in regions where the physical state, strength of local institutions, strategic importance and political influence decide what water resources can be used by the region as a whole, and allocates water within different parts of the region. The decision mechanism of water allocation in this work involves:

1. The amount of water demanded by each demand area.
2. The priority level (strategic importance) of each demand area.

Accounting for supply under conditions of availability is a key part of a successful water allocation system (Dinar et al., 1997). Water allocation to demand areas depends on the amount of available water at the source subject to carrier capacity constraints which affects payment to the supplier in the fitness function (see section 5.4.1.7).

Within any system of water allocation, the importance of certain categories of land-use such as residential, industrial and agricultural is defined. In addition, priorities may be different under the same demand category. For instance, residential areas with hospitals and schools are probably more important than areas where these facilities do not exist. In addition, in the domestic water sector, rights for the provision of drinking water should be guaranteed, while water for irrigation may be more conditional on availability. Furthermore, the state's interest in water allocation mechanism relates to their strategic importance, for example, because of their role in increasing food security or public health. This method is useful because it can protect the poor, sustain environmental needs, and provide a given level of water to meet minimal needs in the receiving sector. Allocation rules in this case can be based on statistical measures, on equal shares in available water volumes, on individual requirements, or based on political pressure.

The priority is chosen to be on an integer scale of 1 to 5 where 1 being the less important and 5 is the most important. It is conceptually true that a very big set of weights, described by Paelinck (1976) theorem, exists for any specific ordinal ranking of criteria. In this instance, the priority scale is a user defined order of preference measure and could be set to any desired number. In addition, the basis of categorisation is subjective and depends on the problem (Alfares and Duffuaa, 2009) in which any number of categories is possible. However, according to Alfares and Duffuaa (2009), too many categories are difficult to be intuitively appreciated where the distance between categories is not informative, and the criteria are not rigorous.

Water is allocated only to demand areas that are connected to the source of water with a pipe or a series of pipes. The connectivity in this work is defined as: node 'v' is connected to node 'u' of a network if there is a path from node 'u' to node 'v'.
To explain the concept of connectivity in this work, consider a typical network shown in figure 5.7.

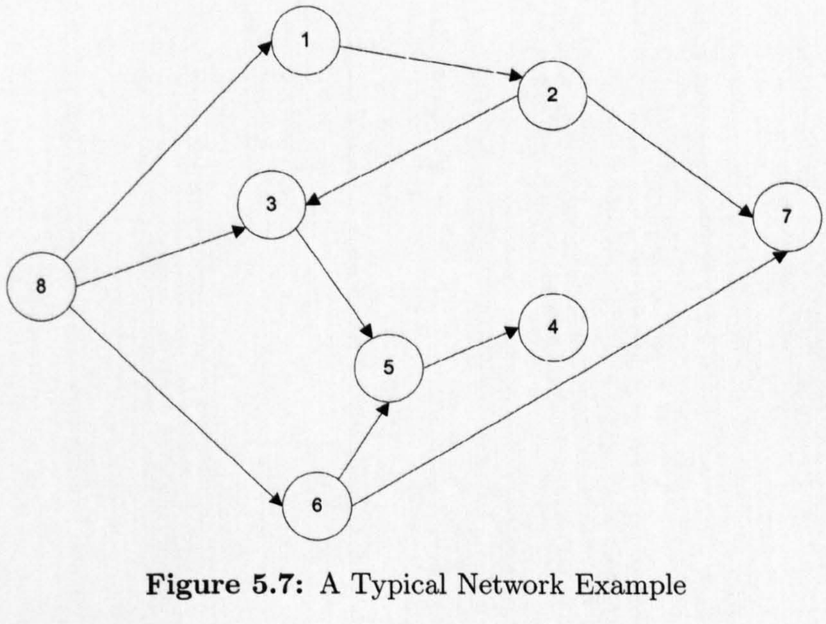


Figure 5.7: A Typical Network Example

The network consists of eight nodes and 10 links. Consider node 1 as the source node, the connectivity of the other nodes to source is presented in table 5.3.

Table 5.3: Connectivity Between Nodes and the Source Node in Example 5.7

Node Number	Connected to Source?	Path
2	Yes	$1 \Rightarrow 2$
3	Yes	$1 \Rightarrow 2 \Rightarrow 3$
4	Yes	$1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 5 \Rightarrow 4$
5	Yes	$1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 5$
6	No	
7	Yes	$1 \Rightarrow 2 \Rightarrow 7$
8	No	

Water allocation decision in this work is also depends on water availability. The availability of water amount is subject to:

1. No water shortage: if the amount of the water at source(s) is equal or larger than the total amount of demanded water, the full amount of demanded water is allocated to each demand area. The allocated water amount to each demand area is the maximum water right for that area.

2. Water shortage: if the amount of source water is less than the amount of total demanded water, a decision is needed to distribute the water according to demand amount and priority of each demand area.

Based on these states, equation 5.1 is used as decision mechanism to allocate water to demand areas.

$$Water\,allocation_j = \frac{Source_k}{\sum_{i=1}^{n} D_i\, I_i'}\, D_j\, I_j' \tag{5.1}$$

Where:

j = number of demand areas that are connected to source.

$Source_k$ = source of water.

k = number of sources.

D_j = amount of demand for area j.

I_j = importance of area j.

n = number of 'connected to source' demand areas.

The importance ratio (I_j') is represented as the importance of the demand area (I_j) divided by the upper limit of the importance scale (= 5), $I_j' = \frac{I_j}{5}$.

The term $\sum_{i=1}^{n} D_i\ I_i'$ is the summation of importance ratio multiplied by area demand.

Figure 5.8 illustrates the process of water allocation.

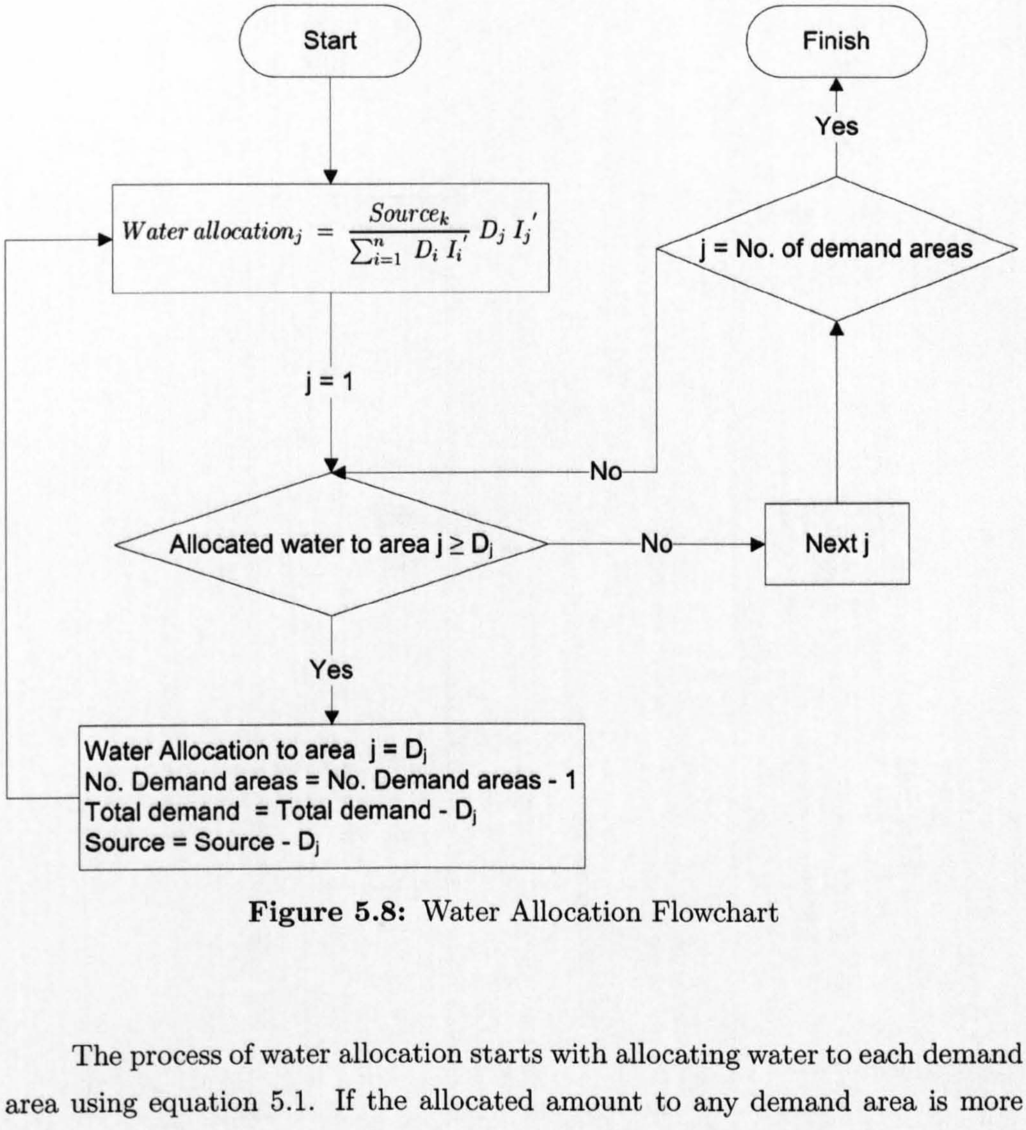


Figure 5.8: Water Allocation Flowchart

The process of water allocation starts with allocating water to each demand area using equation 5.1. If the allocated amount to any demand area is more than the required demand, the current area will have its maximum water right and the current demand will be excluded from the process of water allocation in the next iterations. In addition, the surplus of allocated water to that area will be reallocated again to other demands using the same decision. This process continues for all remaining demand areas and terminates when the allocated water for each demand area is less than its water right.

Although not required, the same formula works if the water amount at source is larger than the total water demand. The process of water allocation ends when all water rights (maximum required amount) had been allocated.

To explain the process of water allocation, table 5.4 presents the demand amount and importance of an example network consists of four demand areas. The total demand is 27,000 and the source water is assumed to be capable to provide 25,000.

Table 5.4: An Example of Demand Area Information

Area ID	Demand Amount	Importance
1	4,000	5
2	9,000	3
3	8,500	3
4	5,500	4

The solution starts using water allocation equation 5.1. The results are presented in table 5.5. This is the first iteration of the solution.

Table 5.5: Iteration 1 of Water Allocation Example

Area ID	Demand	Importance	$\frac{Importance}{5}$	Demand $\times$ $\frac{Importance}{5}$	Amount allocated
1	4,000	5	1.0	4,000	5,291.0
2	9,000	3	0.6	5,400	7,142.8
3	8,500	3	0.6	5,100	6,746.0
4	5,500	4	0.8	4,400	5,820.1
Total	27,000			18,900	

In iteration 1, area 2 and 3 had been allocated amounts less than their desired demand. Area 1 and 4 had been allocated amounts larger than their desired demands. The surplus allocated water from area 1 and 4 need to be reallocated. Further iterations are required to reallocate the surplus allocated water.

Area 1 has been allocated an amount of 1,291 more than its demand. Therefore, it can only have 4,000 and will be excluded from the process and the total demand will be 23,000 and the water source will 21,000.

Iteration 2 of the allocation process is shown in table 5.6.

Table 5.6: Iteration 2 of Water Allocation Example

Area ID	Demand	Importance	$\frac{Importance}{5}$	Demand $\times$ $\frac{Importance}{5}$	Amount allocated
1	0	0	0	0	0
2	9,000	3	0.6	5,400	7,610.7
3	8,500	3	0.6	5,100	7,187.9
4	5,500	4	0.8	4,400	6,201.3
Total	23,000			14,900	

In iteration 2, area 2 and 3 had been allocated amounts less than their desired demand. Area 4 has been allocated an amount larger than its desired demand. Therefore, the surplus water from area 4 needs to be reallocated to other demand areas. Thus, further iterations are required.

Area 4 received 701.3 more than its maximum demand. This amount will be reallocated to other demand areas.

In iteration 3, water source = 15,500 which will be allocated to two demand areas with a total of 17,500. Iteration 3 of the solution is shown in table 5.7.

Table 5.7: Iteration 3 of Water Allocation Example

Area ID	Demand	Importance	$\frac{Importance}{5}$	Demand $\times$ $\frac{Importance}{5}$	Amount allocated
1	0	0	0	0	0
2	9,000	3	0.6	5,400	7,971.4
3	8,500	3	0.6	5,100	7,528.6
4	0	0	0	0	0
Total	17,500			10,500	

Since all water amount at source had been allocated to demand areas and the amounts that are allocated to areas 2 and 3 are less than their maximum water rights, no further iterations are required and the process of water allocation terminates at this stage. Table 5.8 shows the final results of water allocation.

Table 5.8: Final Water Allocation

Area ID	Water Right
1	4,000.0
2	7,971.4
3	7,528.6
4	5,500.0

5.3.3 Water Distribution (Flow-rate Distribution)

A fundamental problem in the optimal design of WDN occurs when assigning the flows in the individual pipes (Goulter et al., 1986). In branched systems, a given demand pattern defines the flows in the pipes uniquely. In a looped system there are an infinite number of distributions of flow in the network that can meet a specified demand pattern.

The distribution of flows assumed in the pipes affects the lowest cost that can be achieved for a network. The flow-rate through each pipe can be calculated based on the hydraulic network model (Bhave, 2003).

Many researchers approached this issue. Alperovits and Shamir (1977) employed a gradient search approach to identify the flow pattern that permits the minimum overall cost for the distribution system to be obtained.

Goulter et al. (1986) addressed the effects of varying the paths used to ensure adequate pressure throughout a WDN. The path choice showed how the initial flows are changed, i.e., increased or decreased, by the gradient expressions in the flow modification step. They concluded that the amount of flow in a particular link in the final solution appears to be dependent on how often that link is included in the pressure defining constraints.

Lund and Israel (1995) presented an optimisation method for a preliminary estimation of the least-cost integration of several water supplies. The study tackled the issue of system capacity constraints in which the sum of all water supply types must be equal or exceeds the total demand.

Meier and Barkdoll (2000) described the use of a genetic algorithm to optimise a network model for a small town. Matching the optimal solutions produced by complete enumeration in a series of validation tests. The objective was to find a set of calibration test locations that, when analysed collectively, produces non-negligible flow in the pipe network.

5.3.3.1 The Decision

The design optimisation of WDN in the aforementioned studies had loops. The loops are created by connections between branches that are composed of minimum specified pipe sizes. These minimum pipe diameters do not provide sufficient capacity or ability to provide flow by alternative supply paths.

The GA in this work tends to reduce costs by reducing the diameter of, or completely eliminating, pipes. It also tends to reduce the length of the pipes and reduce the power required to pump the water to demand areas; thus, leaving the system with insufficient capacity to carry water to demand areas. It was, therefore, necessary to consider the maximum flow of water that a particular pipe can carry.

The decision of water through the pipes is taken at each node in the network. This decision affects the network layout, and the amount of water delivered to the demand areas. With this decision, the model will implement a GA method to optimise the cost and the maximum capacity of the network.

At each node in the system, water distribution decision is mainly divided into two sub-decisions:

1. The decision into how much water should flow through the outgoing pipes from a node.
2. The decision into how much water should be supplied to the local pipes (demand area) if the node type is a demand node.

A wide range of operating rules can be modelled by utilising the functional attributes of various node types and pipes that are used to configure the water supply system. For instance, the nodes are connected by pipe carriers, and these pipes can model minimum flows and maximum capacities according their sizes (pipe diameter).

5.3.3.2 Decision Considerations

1. System Capacity

   The Hazen-Williams equation, explained in section 2.4.7.6, is used to determine the maximum capacity of a pipe. The discharge amount is considered as the maximum water amount flow in which a pipe can carry.

   If the capacity of an outgoing pipe from a node is not enough to carry the distributed water according to the water distribution decision, only the maximum amount will be distributed through the pipe and the remaining water amount will be returned to the upstream node. This will be explained in details in section 5.3.3.3.

2. Water Distribution Decision

   Water distribution decision is the decision of how much water should flow through each outgoing pipe from a node. This decision is taken at each node in the system and depends on two aspects:

   (a) The cross sectional area of the pipe.

   (b) The hydraulic slope of the pipe.

At a certain node in the system, the expression of the decision can be written as:

$$Water\ Distribution_j = \frac{A_i\ S_i}{\sum_{i=1}^{n} A_i\ S_i} \times W_j \tag{5.2}$$

Where:
A_i is the cross sectional area of the i^{th} outgoing pipe from node j.
S_i is the hydraulic slope of the i^{th} outgoing pipe from node j.
W_j is the amount of available water at node j.
n is the total number of outgoing pipes from node j.
i is the number of the outgoing pipe from node j.
j is the number of nodes.

The water distribution decision is subject to many rules that govern the water distribution to the system. These rules are explained in the next section.

5.3.3.3 The Decision Rules

Figure 5.9 shows an example of a water network that supply water to a service area. This example will be used to explain the decision rules.

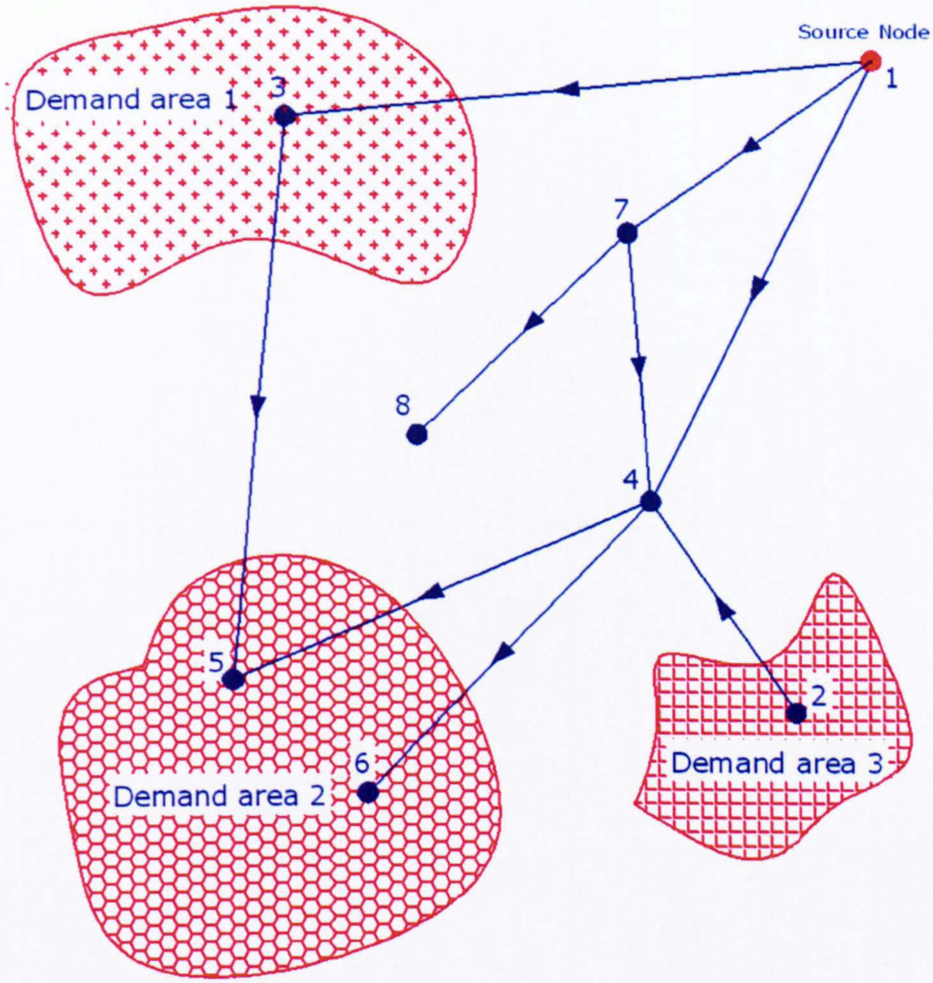


Figure 5.9: Example of a Water Network and Demand Areas

The above network configuration might be generated at any stage during the evolution process. The example network consists of eight nodes and nine links. The service area example consists of three demand areas that vary in size. This configuration is selected from one of the GA runs to demonstrate the decision rules.

The description of these nodes is presented in table 5.9.

Table 5.9: Nodes Description for Example 5.9

Node ID	Node Type	Connected to Source?	Location
1	Source	Source node	Source point
2	Demand	No	Demand area 3
3	Demand	Yes	Demand area 1
4	Intermediate	Yes	Public open space
5	Demand	Yes	Demand area 2
6	Demand	Yes	Demand area 2
7	Intermediate	Yes	Public open space
8	Intermediate	Yes	Public open space

The key rules to this decision are:

Rule 1

Unconnected to source nodes are not considered in the procedure. In the above example, node 2 is not connected to the source. As a result, demand area 3 is also not connected to the source node. Therefore, node 2 and demand area 3 will not be considered in the decisions.

Rule 2

The nodes may be classified according to their location in the world into source, intermediate or demand.

A source node is the node that is located at a water source point (e.g. node 1).

A demand node is the node that is located at a demand area such as node 2, 3, 5, and 6.

An intermediate node is neither source nor a demand node and located outside source and demand areas such as node 4, 7, and node 8. The area that is neither demand nor source is called 'public open space' (POS).

Rule 3

At source node, there may be a net gain of flow into the network. At an intermediate node, there may be a net gain of flow into the node. There is no net loss of flow out of the network at intermediate and source nodes.

Therefore, the demand at source and intermediate nodes = *zero*. The water at these nodes will be distributed to the outgoing pipes form them.

Rule 4

At any node in the network, if the distributed water amount to a pipe using the distribution decision is larger than the pipe capacity, the allowed amount of the distributed water will be the maximum pipe capacity.
The extra amount remains at the current node to be distributed to other pipes if exist. If no further distribution pipes exist, this amount will remain at the current node and considered as waste (the waste is described in section 5.3.3.4). To exemplify this rule, consider node 4 in figure 5.9. The associated pipes with flow amounts are shown in figure 5.10.

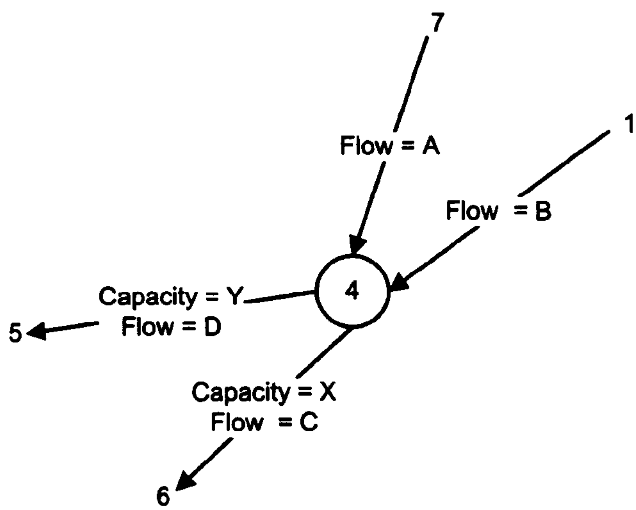


Figure 5.10: A Typical Node With Incoming and Outgoing Pipes

Node 4 has four associated pipes in which two are carrying the incoming flow from node 7 and node 1; and two will carry the distributed outgoing flow to node 5 and node 6.
The available flow at node 4 is the total amount of all incoming flow into node 4 (= $A + B$). This amount will be distributed into the outgoing pipes from node 4 to node 5 and 6.

The following flowchart illustrates this rule.

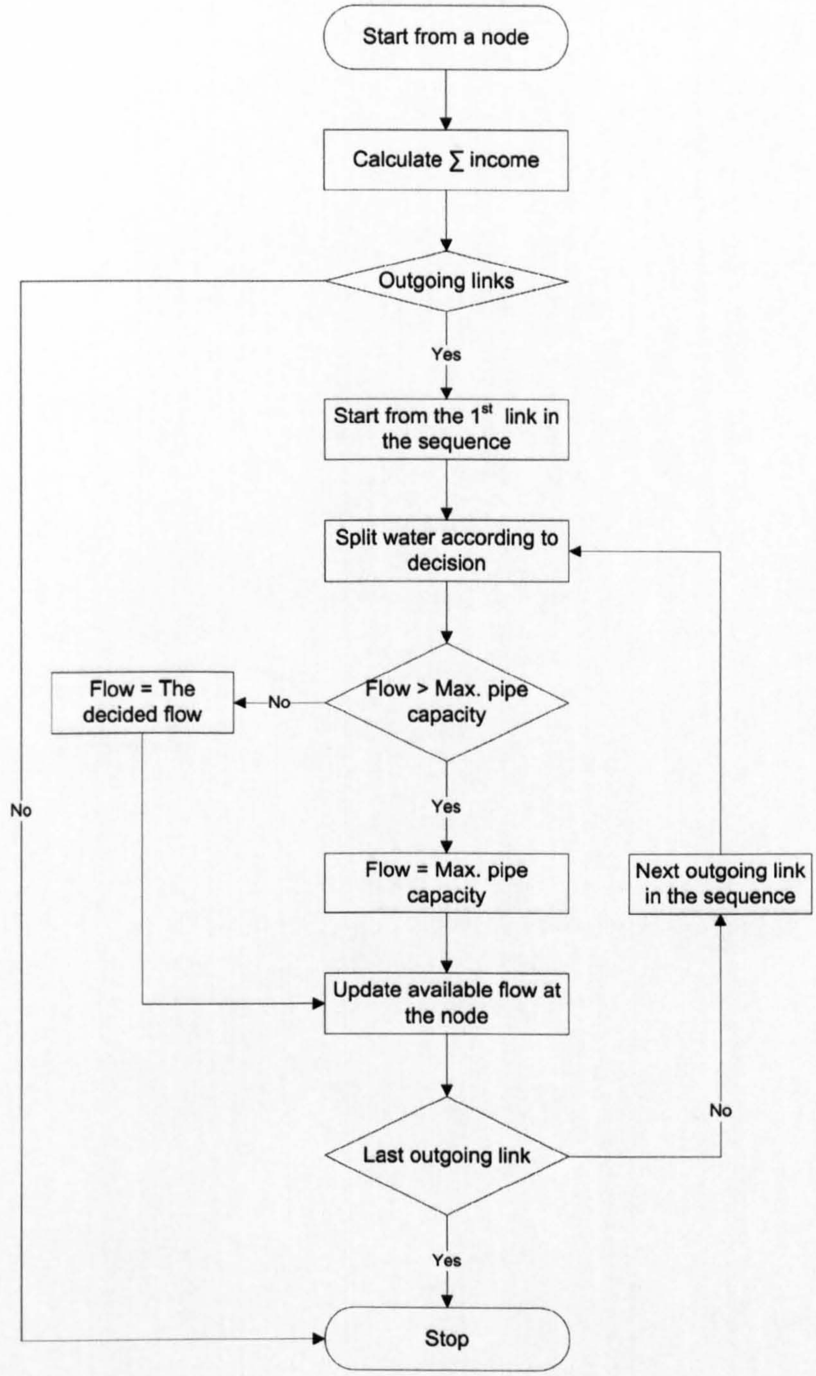


Figure 5.11: A Flowchart Illustrating Rule 4

The distribution decision involves the following steps:

I. Consider the first outgoing pipe in the sequence which connects node 5 to node 4 and calculate the amount of water that will flow into the pipe using the distribution decision (= 'D').

II. If 'D' > 'Y', then the available flow at node 4 = A + B - Y. If not, the available flow at node 4 = A + B - D.

III. The current amount at node 4 will be distributed into pipe 4-6 using the distribution decision (= 'C').

IV. If 'C' > 'X', the remaining flow at node 4 = $A + B - D - X$ or = $A + B - X - Y$. If not, the remaining flow at node 4 = $A + B - C - D$ or $A + B - Y - C$.

The remaining amount at node 4 is waste because the system does not make use of this amount. If $(A + B) > (C + D)$, then the waste is positive and the GA model will tend to reduce this amount by approaching the near-optimal system components.

Rule 5

At a demand node, the decision also involves the distribution of water to local pipes (demand area) according to a user defined factor of the demand amount. This factor is a percentage of the demand amount where the current node is located and shall be called "Demand Benefit Factor" and abbreviated as 'DBF'. For example, if the DBF is chosen to be 50%, then half of the current demand will be satisfied and the remaining water amount at the current node will be distributed to the outgoing pipes according to the distribution decision.

The DBF can be changed according to the user preferences through the program interface.

The flowchart shown in figure 5.12 illustrates this rule.

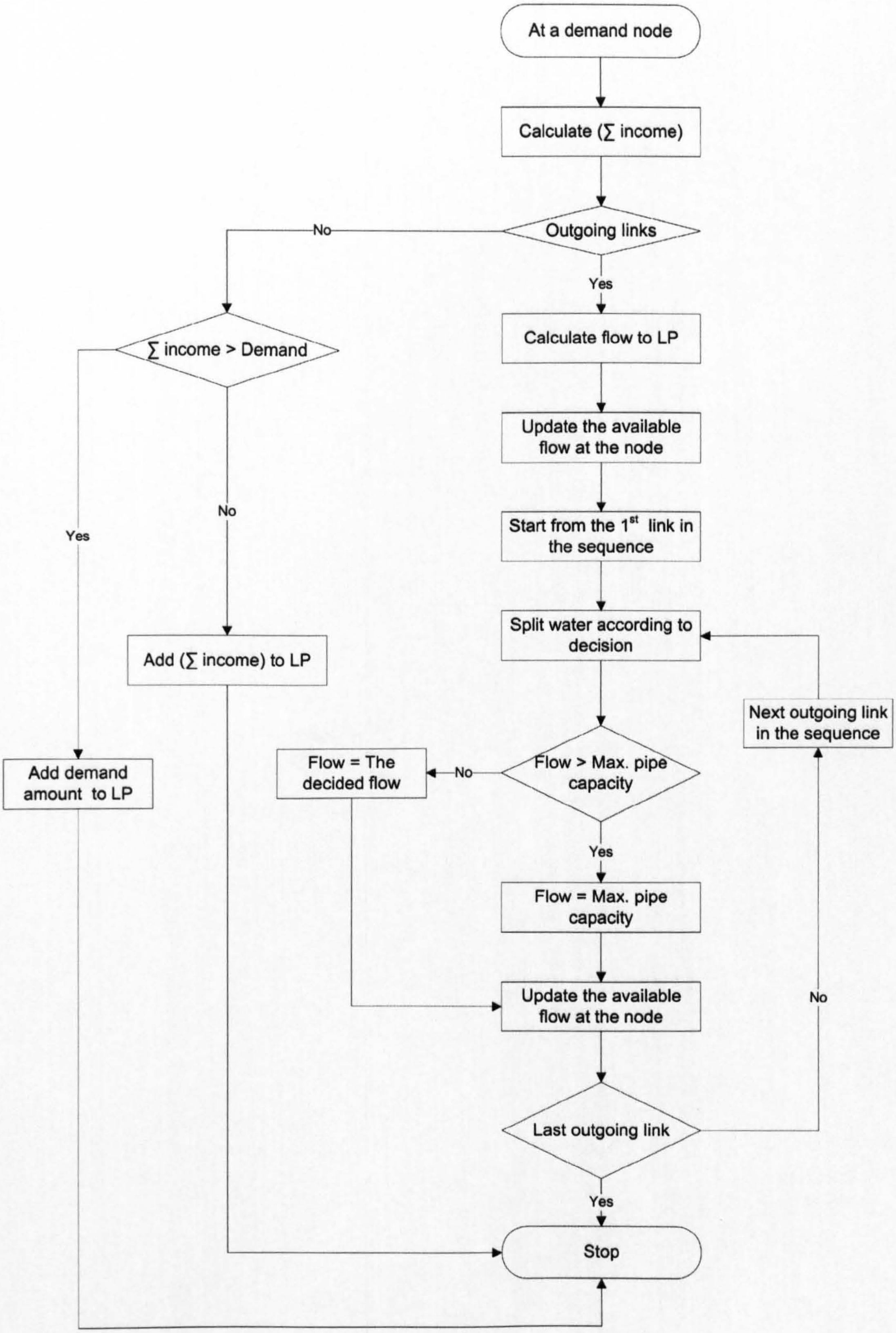


Figure 5.12: A Flowchart Illustrating Rule 5

The water distribution decision at a demand node involves two cases:

Case 1: The available flow at a demand node $\geqslant$ demand $\times$ DBF

The local pipes (demand area) will have an amount of (demand $\times$ DBF) from the entire node. The node will distribute the remaining water to the outgoing pipe(s) according to water distribution decision.

No water is distributed out of the demand node if there is no distribution pipes and the available water at the node will be given to the local pipes. To clarify this case, node '*3*' from the network shown in figure 5.9 will be considered.

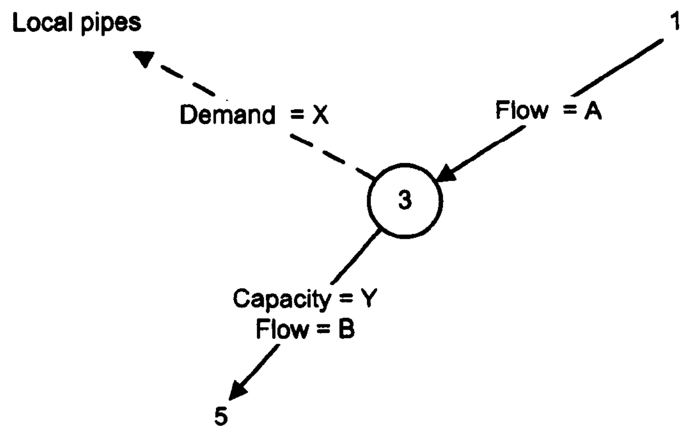


Figure 5.13: A Node With Incoming and Outgoing Pipes

Node 3 is a demand node. The node receives flow '*A*' through the incoming pipe from node 1. The node sends the flow '*B*' into an outgoing pipe to node 5 and also distributes some flow into the local pipes to supply the demand areas.

The steps involved in this case are:

I. Calculate the total amount of available flow at the demand node from all incoming pipes. In this example, the available amount of flow at node 3 = '*A*'. This amount is supplied by one pipe (link 1-3).

II. Calculate (X $\times$ DBF) and supply this amount to the local pipes. The unsatisfied demand = X - (X $\times$ DBF).

III. The remaining amount at node 3 = A - (X $\times$ DBF)

IV. If A - *(X* $\times$ *DBF)* > capacity '*Y*', then an amount of 'Y' will be distributed through 'link 3-5'. The remaining amount at node '*3*' = A - (X $\times$ DBF) - Y which will be considered as waste.

If A - *(X* $\times$ *DBF)* < capacity '*Y*', then A - *(X* $\times$ *DBF)* will be send to node 5 and the wasted water does not exist at node 3 in this case.

Case 2: The available flow at a demand node < demand × DBF

In this case, the local pipes will have the full amount of incoming water to the entire node. Therefore, no water is distributed out of the entire node because all incoming amount of water has been taken by the local pipes. In addition, the wasted water does not exist.

5.3.3.4 The Wasted Water

In a decision-making process, the wasted water can be defined in many forms. In this work, the wasted water is defined as the remaining water amount at a node. It represents the amount that has been sent to the node but has not been used by the local pipes and not distributed to the outgoing pipes from the node.

The water amount at the source node is not considered as waste. The source node is rather considered as a water storage point such as water tank.

This amount of water might stay at a node because:

1. The capacity of the outgoing pipes is limited and therefore, they are unable to carry and distribute the remaining water from the upstream node.
2. The full demand is satisfied if the node is a demand node.
3. The node is a deadlock node (e.g. node 8 in the example shown in figure 5.9).

5.4 Model Development

5.4.1 Fitness Function Design

The ability of an optimisation method to converge efficiently to a near-optimal solution depends on the nature of the objective function. The design of a WDN is a complex problem and involves decisions on a broad range of concerns. Cost is likely to be the primary emphasis which includes the costs of construction, operation, and maintenance. The designer's problem is to determine the minimum cost system while satisfying the demands at the required pressure heads.

The cost of the system includes the initial investment for system components, such as pipes, tanks, valves, and pumps, and the operating cost for pumping the water. The main constraints are that the desired demands are supplied with adequate pressure heads at withdrawal locations. Also, the flow of water in a distribution network and the pressure heads at nodes must satisfy the governing laws of conservation of mass and energy. In this instance, the WDN design optimisation problem can be stated as:

Minimise

Initial capital cost + operation cost

Subject to:

1. Conservation laws of mass and energy
2. Water demand constraints
3. Meeting nodal head requirements

In practice, additional complexities have also to be included for a robust model. The other factors that can be included are network layout, operation, uncertainty, reliability, and water quality. Details about the impact of these factors on the optimisation of WDN were discussed in chapter 3.

The main objective of the design of WDNs is to supply water in sufficient quantities at a reasonable cost to existing and future consumers. Each solution is evaluated using a fitness function that is specific to the problem being solved.

The fitness function described in this chapter is more advanced than in the initial problem formulation, taking on board four additional aspects.

Several fitness functions have been tested and developed in this work. An initial form of the fitness function is presented and tested in chapter 4. In the initial problem formulation, the initial fitness function was in a simplified form which included: the cost of pipes and their installation and the benefit from the supply of water to demand areas.

More complex fitness function is devised in this chapter. As an example, a fitness function might consist of the cost of pipes and their installation, pumping power cost, structural cost, costs incurred of the wasted water, and the benefit from the supply of water. This fitness function is used to demonstrate the work.

5.4.1.1 Pipe Installation Cost

Pipes are water transmission lines between two points. Therefore, they have starting and ending point with known coordinates. The length of a straight line can be found if the starting and ending coordinates are known. For more details about this component, see section 4.2.

The cost of pipe installation includes the cost of excavation of pipe trench, placing the pipe in the trench, backfilling the trench, and other miscellaneous cost items.

Pipe installation is expressed as a cost per unit length for each pipe size from an admissible list.

$$C_I = \sum_{i=1}^{n} L_i \times C_I(D_i) \tag{5.3}$$

Where C_I is the installation cost of the pipes in the network, L_i is the length of i^{th} pipeline, $C_I(D_i)$ is the installation cost of the diameter D_i per each unit length of the pipeline i, D is the internal diameter of the pipe which is selected from the permissible pipe set, and n is the number of the pipes in the network.

Equation 5.3 describes the total installation cost of the pipes in the network subject to:

1- Hydraulic constraints:

$$\sum_{i \in (in)k} Qi - \sum_{i \in (out)k} Qi = Q_k , \qquad k = 1, \ldots, J$$

$$\sum_{i \in l} J_i = 0 , \qquad l = 1, \ldots, L$$

$$Q_i = K\, C_{HW}\, D_i^{\alpha} \left(\frac{hl_i}{L_i}\right)^{\beta}$$

Where J and L number of existing nodes and loops in the network, respectively; Q_i is the flow rate in pipe i; Q_k is the required demand at consumption node k; hl_i is the head loss in the i^{th} pipe; L_i is the length of the i^{th} pipe; C_{HW} is the Hazen−Williams coefficient; $\alpha = 2.63$; $\beta = 0.54$; $K = 0.285$ for Q in (m^3/sec), and D_i is the pipe diameter in meters for the i^{th} pipe.

These constraints, therefore, describe the flow continuity at nodes, head loss balance in loops, and the Hazen−Williams equation.

2- Head and velocity constraints:

$$H_{min} \leqslant H_k \leqslant H_{max} , \qquad k = 1, \ldots, J$$

$$V_{min} \leqslant V_i \leqslant V_{max} , \qquad i = 1, \ldots, N$$

Where H_k is the nodal head; H_{min} and H_{max} are the minimum and maximum allowable hydraulic head respectively; V_i is the velocity of the flow in pipe i; V_{min} and V_{max} are the minimum and maximum allowable flow velocity respectively.

The second set of constraints refers to the minimum and maximum nodal head and flow velocity requirements.

3- Pipe size availability constraints:

$$D_{min} \leqslant D_i \leqslant D_{max}\,, \qquad i = 1, \dots, N$$

Where D_{min} and D_{max} are the minimum and maximum commercially available pipe diameters respectively; and N is the number of commercially available pipe diameters.

The optimal pipe diameters should be between the maximum and minimum commercially available pipe diameters.

5.4.1.2 Pipe Material Cost

The cost includes the cost of pipe manufacturing and the cost of transportation to the construction site. This cost is similar in pattern to the pipe installation cost.

$$C_M = \sum_{i=1}^{n} L_i \times C_M(D_i) \tag{5.4}$$

Where C_M is the material cost of the pipes in the network, L_i is the length of i^{th} pipeline; $C_M(D_i)$ is the materials cost of the diameter D_i per each unit length of the pipeline i; D is the internal diameter of the pipe which are selected from the permissible pipe set, and n is the number of the pipes in the network.

The same set of constraints mentioned in pipe installation cost (section 5.4.1.1) is applicable to the pipe material cost.

The installation and material costs are set by the user. Figure 5.14 shows a window interface that consists of the admissible list of pipe sizes with different installation and material costs.

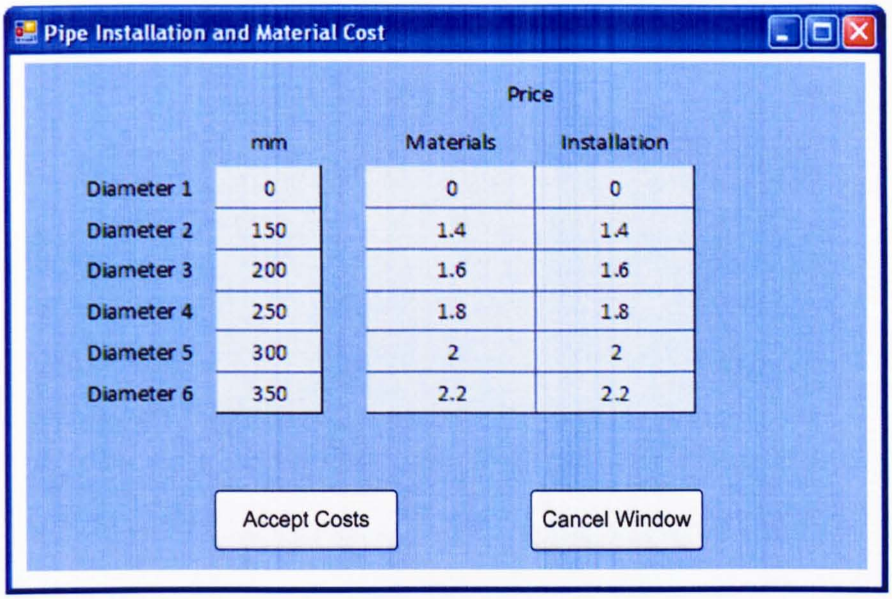


Figure 5.14: Pipe Installation and Material Cost Program Interface

5.4.1.3 Land Access Cost

This cost is incurred according to the disruption levels to the real-world features caused by the construction and maintenance works of the WDN. Each feature in the world has a different access cost. The features could be topography, slope, geology, soil, land-use, road, rail, forest, water bodies and streams. The land access cost is explained in more details in section 7.4.

The construction of a water distribution system may cause environmental impacts, which can be transferred to cost. It may degrade environmental amenities and agricultural production directly by construction and demolition or clearing land, indirectly by encouraging increased development and sprawl. In addition, land access cost may depend on the population density in an area. The denser the area is, the more land access cost.

Different access cost figures are assigned to these features to reflect some of the real-world costs and routing criteria. For instance, rough terrain access cost is higher than flat terrain and access cost of city centres is different from city outskirts.

The land access cost is assigned to each cell in the raster grid where each cell represents the world features in that part. Land access cost is the cost of crossing a specific area which can be expressed by:

$$C_{LA} = \sum_{i=1}^{n} \sum_{j=1}^{m} LS_{ij} \times C_{ij} \tag{5.5}$$

Where C_{LA} is the land access cost; LS_{ij} is the length of the j^{th} pipe segment in a cell of the i^{th} pipeline ; C_{ij} is the access cost of the cell where the pipe segment passes; n and m are the number of lines and number of segments in i^{th} line respectively. C_{ij} is explained in details in section 4.2.

5.4.1.4 Pumping Power

Pump head costs represent the costs of lifting the water to higher elevations. Pumping energy cost is given by:

$$C_{pumping} = \sum_{i=1}^{n} \frac{\rho \, g \, Q_i \, H_i}{3.6 \times 10^6 \, \eta} \times C_{power} \tag{5.6}$$

Where $C_{pumping}$ is the pumping cost; ρ is the mass density of water (= 1000 kg/m^3 at 4 °C), g is the acceleration of gravity (= 9.81 m/Sec^2); Q_i is the flow discharge ($m^3/hour$) in the i^{th} pipe; H_i is the pumping head (m) of the i^{th} pipe; η is the pump efficiency (%), and C_{power} is the unit cost of power energy (unit cost/Kilo-Watt).

Some of parameters required for this equation can be input in the program interface (figure 5.15).

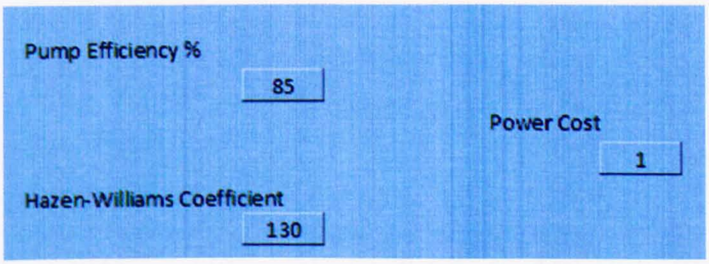


Figure 5.15: Program Interface For Pumping Power Equation Parameters

5.4.1.5 The Wasted Water Cost

The waste water is explained in section 5.3.3.4. The wasted water should be minimised and the system should only carry that water amounts that are going to be used by the demand areas. The expression of this cost is:

$$C_{WW} = \sum_{k=1}^{m} (Q_{(in)k} - Q_{(out)k}) \times C_{waste} \quad (5.7)$$

Where C_{WW} is the penalty cost from the wasted water; $Q_{(in)k}$ supply amount into node k; $Q_{(out)k}$ is the distribution amount out of node k; C_{waste} is the cost per wasted unit of water; and m is the number of nodes in the distribution system.

5.4.1.6 Extra Work Cost

It is a penalty cost incurred to nodes whose elevation is outside the desired pipe level. The desired pipe level is set to a certain depth under the ground level.

To demonstrate this cost, it was necessary to generate an elevation model for 3-D representation of a terrain's surface in the form of contour lines. This model includes a ground level at each grid cell. The process of generating the ground level starts with a random grid cell within the world. The contour lines are evenly spaced to indicate gentle slope. The contour interval is the difference in elevation between successive contour lines. Figure 5.16 shows a typical model used in this work for ground elevations at each cell.

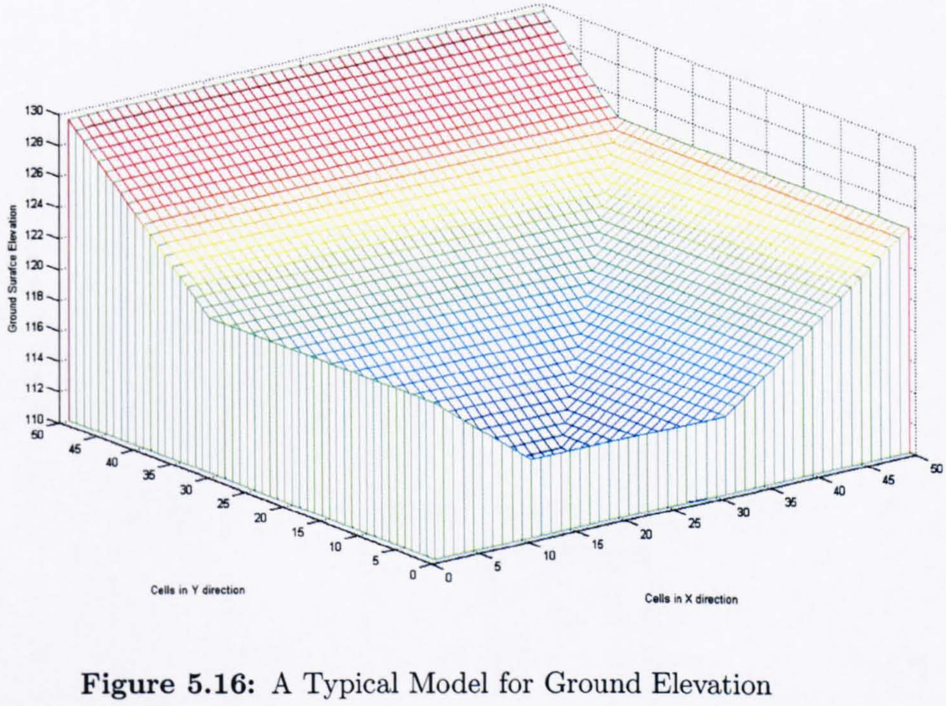


Figure 5.16: A Typical Model for Ground Elevation

Figure 5.17 shows the ground level and the ideal pipe level.

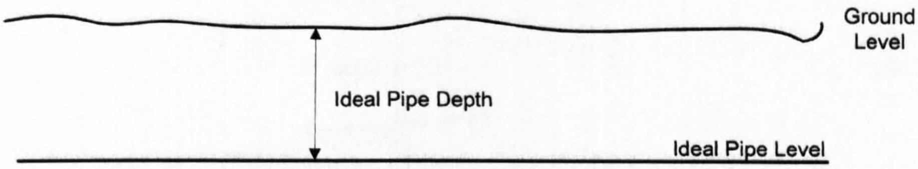


Figure 5.17: The Ground Level and the Ideal Pipe Level

In this instance, a node may have one of the three cases:

Case 1: Node elevation > Ground level.
If the node is located above the ground level, extra infrastructure needs to be built for the piping system. Therefore, a cost is incurred for the extra work needed to build the infrastructure.

Case 2: (Ground level - Ideal pipe depth) ⩽ Node elevation ⩽ Ground level.
If the node is located at the desired region, a cost is incurred to the depth of the pipe invert from the ideal pipe level. This cost decreases when the difference between the node elevation and the ground level increases.

Case 3: (Ground level - Ideal pipe depth) > Node elevation.
If the node is below the desired pipe level, extra excavations are required which adds to the cost of the WDN.

The expression for the extra work cost is one of the following:

$$
\begin{aligned}
C_{EW} &= \sum_{k=1}^{m} (E_k - E_{ground}) \times C_A && for\ case\ 1 \\
or &: \\
C_{EW} &= \sum_{k=1}^{m} (E_k - E_{ground} + I_{depth}) \times C_{PD} && for\ case\ 2 \\
or &: \\
C_{EW} &= \sum_{k=1}^{m} (E_{ground} - E_k - I_{depth}) \times C_U && for\ case\ 3
\end{aligned} \tag{5.8}
$$

Where C_{EW} is the cost of extra work; E_k is the elevation of node k; E_{ground} is the ground level; I_{depth} is the ideal pipe depth; C_A is the cost of building extra infrastructure above the ground level; C_{PD} is the cost of the pipe invert depth from the ideal pipe level; C_U is the cost of extra excavations under the desired pipe level; and m is the number of nodes in the system.

The extra work costs and the ideal pipe depth can be changed by the user to the desired values. The interface is shown in figure 5.18.

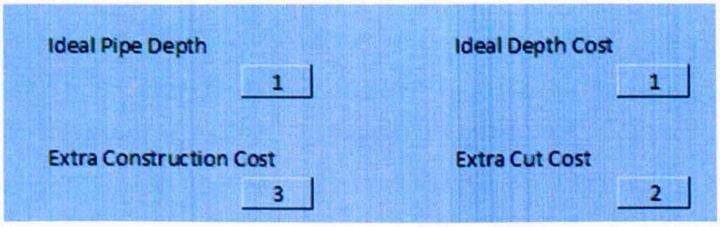


Figure 5.18: The Computer Interface of Extra Work Costs

5.4.1.7 The Payment to the Supplier

All water service providers in the world charge tariffs to recover their costs. The payment to the water supplier is the amount of water delivered, usually in cubic meters, to a demand area multiplied by a unitary cost. The amount of supplied water depends on capacity of the source, the required demand, and the capability of the pipe system to deliver the water.

The benefits from payment to supplier can be expressed as:

$$P(DW) = \sum_{d=1}^{l} DW_d \times C_{water} \tag{5.9}$$

Where $P(DW)$ is the payment to the supplier; DW_d is amount of water delivered to demand area d; C_{water} is the unit cost per each cubic meter of delivered water; and l is the number of demand areas in the system.

The fitness function (ff) of an individual solution is defined in terms of its 'cost'. This cost is not a true financial cost but one made up of factors from a variety of sources which can be balanced to suit the decision-maker.

From objective functions 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, and 5.9, the fitness function can be formulated as:

$$\begin{aligned} ff &= \alpha_1 \times C_I + \alpha_2 \times C_M + \alpha_3 \times C_{LA} + \alpha_4 \times C_{pumping} + \\ & \quad \alpha_5 \times C_{WW} + \alpha_6 \times C_{EW} - \alpha_7 \times P(DW) \end{aligned} \tag{5.10}$$

Where α_1, α_2, α_3, α_4, α_5, α_6, and α_7 are the balancing coefficients in the fitness function which are set by the user. These values can be entered to the computer program interface.

Fitness Function Balancing Coefficients

Material	Installation	Access Cost	Pump Power	Payment	Water Waste	Extra Work
1	1	0.25	1	1	1	10

Figure 5.19: Fitness Function Coefficients Interface

It is possible, by making some coefficients equal to *zero*, to consider subsets of the general problem and to examine the effects of each of the aspects of the fitness function. In addition, these coefficients are used in the sensitivity analysis to test how the solution is sensitive to each fitness function part.

In order to calculate the fitness, some cost figures are required. These figures can be obtained from industry such as material cost and construction cost for discrete pipe diameters where different diameter sizes have different material and installation costs. The proposed costs in this work are not intended to represent real fiscal costs.

The optimal solution is one which minimises the fitness function. It is, of course, the object to minimise the costs and accordingly, the lower the cost, the fitter the solution.

5.4.2 Chromosome Representation

Following the successful testing of the initial formulation, an improved formulation is developed which allowed for greater complexity and more realistic representation of the water network.

In order to effectively move water from the supply source to consumption points with sufficient pressure, the pipeline needs to account for head loss (friction and local losses) and the hydraulic grade line along the length of the system.

It was necessary to develop the chromosome representation defined in section 4.3 and introduce another component. This development has incorporated the necessary elements of the earlier representation.

The initial representation of the chromosome presented in section 4.3 is developed to include the head required to pump the water to demand areas.

This component is the head along each pipe connection between two terminal nodes.

The new chromosome component is called "Head Matrix" and shall be used in the rest of the thesis. Node type is removed from the original representation. A typical chromosome representation is shown in figure 5.20.

To Node

From Node	1	2	3	4	5	6
1	0	0	0	0	0	D4
2	0	0	D2	D1	0	0
3	0	0	0	D3	0	0
4	0	0	0	0	D4	0
5	0	0	0	0	0	0
6	0	D1	0	D3	0	0

Connectivity Matrix

X	Y	Z
X1	Y1	Z1
X2	Y2	Z2
X3	Y3	Z3
X4	Y4	Z4
X5	Y5	Z5
X6	Y6	Z6

Node Coordinates

1	2	3	4	5	6
0	0	0	0	0	H1
0	0	H2	H3	0	0
0	0	0	H4	0	0
0	0	0	0	H5	0
0	0	0	0	0	0
0	H6	0	H7	0	0

Head Matrix

Figure 5.20: A Typical Chromosome Representation

The above chromosome represents an example of a water network consists of six nodes and seven pipe connections. Head matrix is added to the chromosome, where H1, H2, H3, H4, H5, H6, and H7 represent real numbers of the head along pipe connections.

5.4.3 Selection

Selection operator is the step that selects individuals from a population for reproduction. Two parents are selected according to their fitness values to generate the new members of the next generation. It selects the best chromosomes (as determined by fitness value). If there are two or more chromosomes with the same best fitness, one of them is chosen randomly (Microsoft Corporation, 2011*a*).

5.4.4 Crossover (Recombination)

Once the chromosomes are selected from the population, they can be recombined. Recombination of chromosomes in the GA is done by the application of a crossover operator. A crossover operator is an operator that, typically, takes two selected individuals of the population at generation i (usually referred to as parents) and then produces two new chromosomes (usually referred to as children) that are potential members of the next generation population (i + 1). Each child is likely to be different from its parents, unless its parents are identical, and they will each retain a number of characteristics from their parents.

Several crossover types are developed and tested in this work. Clearly, it is not possible to include all these types here. In the GA model presented here, the following crossover operators are used: (1) Crossover type 1: chooses a location on the chromosome and swap the remaining genes from one parent to the other; (2) Crossover type 2: uses randomly generated mask to exchange genetic material; and (3) Crossover type 3: selects components in the chromosomes to exchange genetic materials;

Each crossover type will be explained in more details in this section. Each crossover type can be selected from the model interface. This interface is shown in figure 5.21.

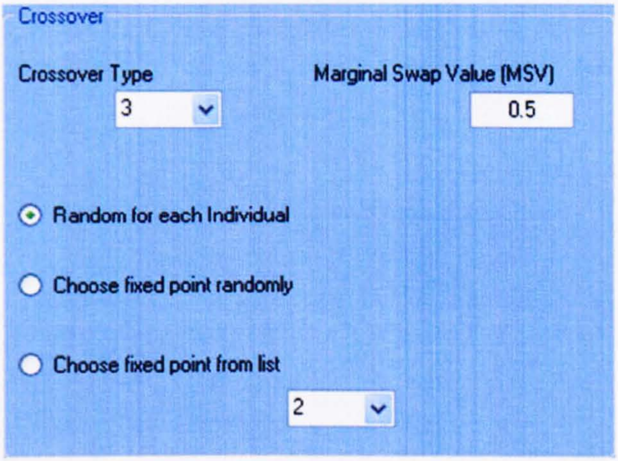


Figure 5.21: Program Interface for Crossover

5.4.4.1 Crossover Type 1

This crossover type is similar to the one that is conventionally called 'single point crossover'. Several types of single point crossover are investigated. Two crossover implementations will be discussed in this section.

1. Lateral crossover point is applied to all components (connectivity matrix, node coordinates, and head matrix) of the chromosome as shown in figure 5.22.

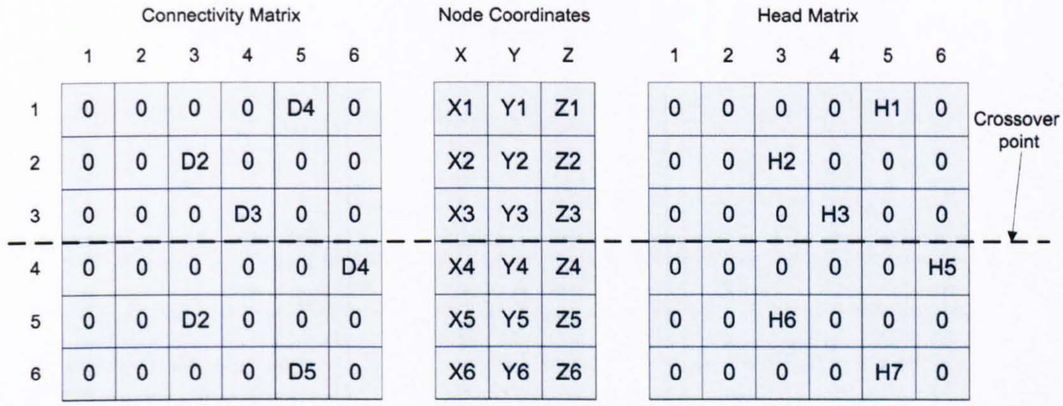


Figure 5.22: Crossover Type 1 Applied to All Chromosome Components

In this type, the information are exchanged for all chromosome components with respect to the crossover point at a node. The crossover operator works when parent 1 passes the upper part of its genetic materials to form the upper part of child 1 and parent 2 to form the upper part of child 2. The genetic materials of parent 1 below the crossover point are passed to child 2 and parent 2 passes its code to child 1.

2. Lateral crossover point is applied to some chromosome components (e.g. node coordinates and head matrix) as shown in figure 5.23.

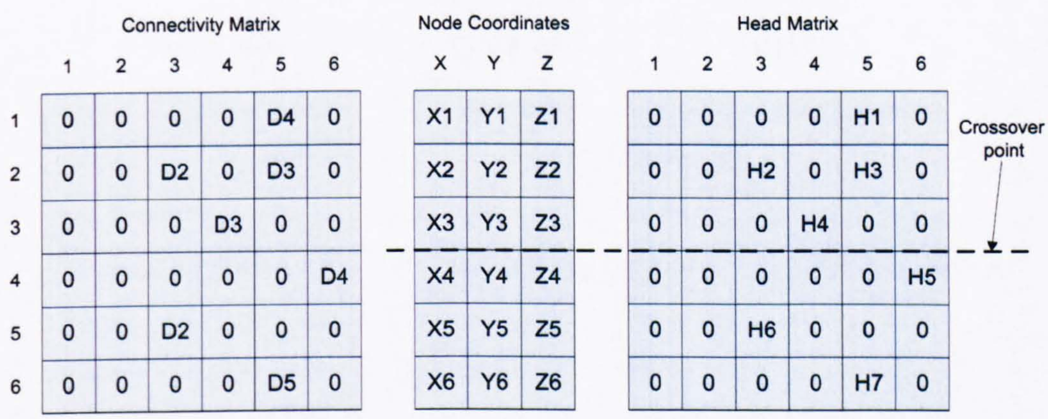


Figure 5.23: Crossover Type 1 Applied to Specific Chromosome Components

The crossover operator could be applied to one component or more at the same time. In this type, the crossover point is specifically applied laterally only to node coordinates and head matrix. The connectivity matrix in this example is unaffected by the crossover operator. The concept behind the implementation of this crossover type is similar to the one applied to all chromosome components except the fact that the children (child 1 and child 2) keep the excluded chromosome components (connectivity matrix in this instance) from parents (parent 1 and parent 2). In this example (figure 5.23), the connectivity matrix of child 1 and child 2 is the same as the connectivity matrix of parent 1 and parent 2 respectively.

In terms of the location of crossover point, two types are introduced:

i. Fixed crossover point

Crossover point is fixed for all individuals in the population and for all generations. The fixed crossover point may be selected randomly to occur in some locations of the chromosome or selected manually.
The crossover point is chosen such that, $1 \leqslant CP \leqslant L-1$, where CP is the crossover point and L is the number of nodes in the network.

ii. Dynamic crossover point

The crossover point is randomly chosen to occur somewhere in the chromosome. This point is similar for all individuals in the population but different for each generation.
Similar to the fixed point crossover, a crossover point CP is randomly selected such that, $1 \leqslant CP \leqslant L-1$, where L is the number of nodes in the network.

5.4.4.2 Crossover Type 2

This type of crossover is similar to the conventional "uniform crossover" which is explained in section A.2.3.2.
The implementation of this type of crossover is distinct from the other crossover types where crossover points are taken at individual node positions. This type of crossover requires two types of values:

1. A randomly generated value which is uniformly distributed between 0 and 1. This value is called "Random Mask Value" and abbreviated as (RMV).
2. A predefined swapping value called "Marginal Swap Value" (MSV). This value is defined by the user as shown in figure 5.21.

The genes of child individual are copied from the parents according to the RMV. At node positions where $RMV \geqslant MSV$ in the mask, genes are carried from parent 1 and parent 2 to child 2 and child 1 respectively. At positions where $RMV < MSV$ in the mask, genes are carried from parent 1 and parent 2 to child 1 and child 2 respectively.

Figure 5.24 shows the possible crossover positions at node positions depending on the RMV and MSV.

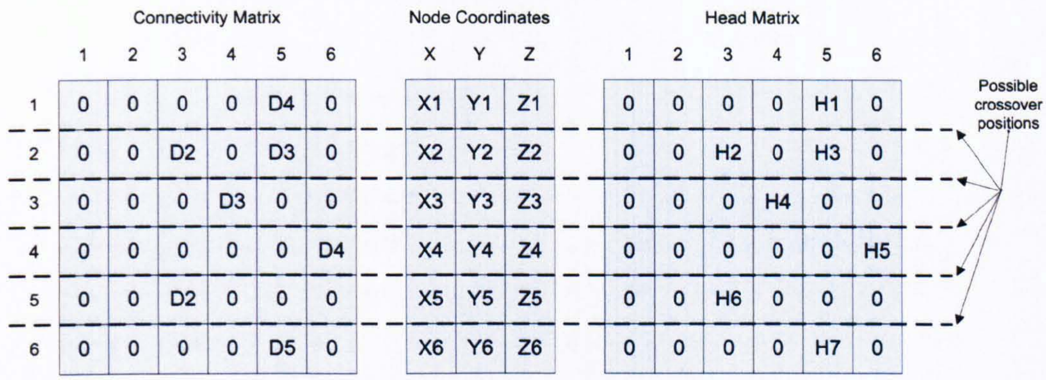


Figure 5.24: Crossover Type 2

5.4.4.3 Crossover Type 3

Some crossover types do not require the conventional crossover point such as crossover type 1. Crossover type 3 is implemented by restricting the location of crossover at different chromosome components. The GA uses a crossover operator that can split the chromosome at the locations between the partial chromosome solutions (chromosome components).

It is worth pointing that a similar approach was first suggested by Booker (1987) who termed it as "Reduced Surrogate Crossover". To reduce the chance of producing clones, Booker (1987) suggested examining the selected parents to define suitable crossover points. A reduced surrogate crossover operator reduces parent strings to a skeletal form in which only those bits that differ in two parents are represented.
This approach has also been independently investigated by Fairley (1991) who found it significantly improved the obtained results.

In this work, two types of this crossover are implemented:

1. Single-Component Crossover
   An example of this type of crossover is shown in figure 5.25.

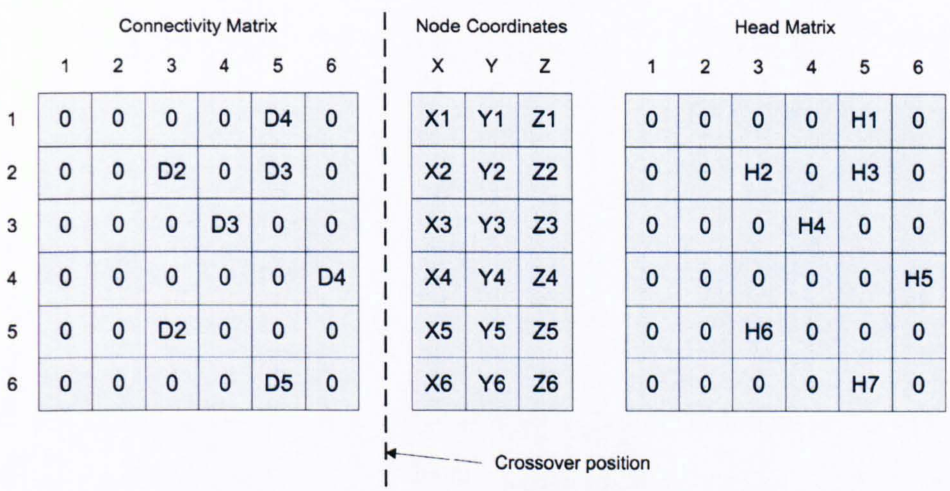


Figure 5.25: Single-Component Crossover

In this operator, the information are exchanged where chromosome components differ. The crossover operator works when parent 1 and parent 2 pass the connectivity matrix to form the connectivity part of child 1 and child 2 respectively. Other genetic materials to the right of crossover point (node coordinates and head matrix) of parent 1 is passed to node coordinates and head matrix part of child 2 and parent 2 passes its code to child 1.

2. Multiple-Component Crossover

   Figure 5.26 shows multiple-component crossover.

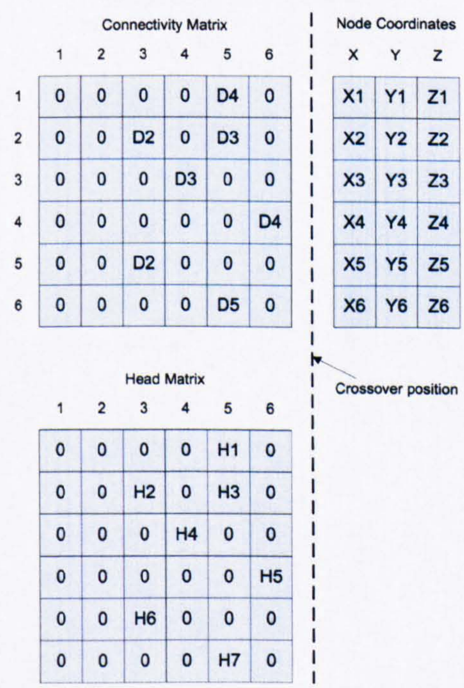


Figure 5.26: Multiple-Component Crossover

Multiple-component crossover works when connectivity matrix and head matrix of parent 1 and parent 2 are passed to child 1 and child 2 respectively. Node coordinates of parent 1 are passed to child 2 and parent 2 passes node coordinates to child 1.

Other combinations of double component crossover exist such as swapping the connectivity matrix and node coordinates between individuals.

5.4.5 Mutation

After crossover is applied to the population, some genetic materials may be lost. The mutation operator is used to restore the lost information. Mutation operator is also responsible for the exploration part of the search (investigation of new and unknown areas of the solution). Mutation adds to the diversity of a population and thereby increases the likelihood that the algorithm will generate individuals with better fitness values. The mutation operator applies random changes to individual parents to form children leaving the rest of the chromosome as it was in the parent.

Mutation causes the individual genetic representation to be changed according to some probabilistic rule. Three types of mutation probabilities are devised in this work. The mutation probability is assigned to the following:

1. The individuals:
   The mutation probability decides the average number of randomly chosen individuals for mutation. For example, suppose the population size is 'N' individuals and mutation rate is $x\%$, the average number of individuals to be randomly selected for mutations will be $N \times x\%$.

2. The connectivity matrix and head matrix:
   The average number of randomly selected cells in each matrix will be selected for mutation. For example, for 'M' number of nodes, there is $M \times M$ cells for connectivity matrix and the same for head matrix. If the mutation rate is $y\%$, the average number of genes that will be randomly selected for mutation will be $M \times M \times y\%$.

3. The node coordinates:
   The average number of randomly selected nodes is chosen for mutation. For instance, for 'M' number of nodes and a mutation rate of $k\%$, the average number of randomly selected node positions for mutation will be $M \times k\%$.

To implement each probability type, a random value which is uniformly distributed between 0 and 1 is generated. The mutation will take place if the generated value is equal or less than the probability of mutation.

As for crossover, there are many possible types of mutation, which can be applied to the current formulation. Several mutation types are developed and tested in this work. Some of these types will be presented.

Mutation types and mutation rates can be changed according to the user needs through the program interface. The program interface is shown in figure 5.27.

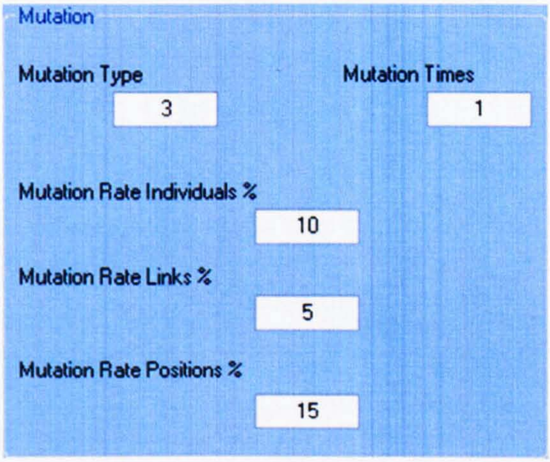


Figure 5.27: Mutation Program Interface

The above figure shows some text boxes where the user can enter the desired types of mutation and different mutation rates. Mutation times represent the number of times in which a specific mutation type can be repeated for the same individual.

Many types of mutation are developed. Only three types of mutation will be mentioned in this section.

5.4.5.1 Simple Mutation

Simple mutation replaces the value of a randomly chosen gene with a random value selected between user defined bounds. An example of this type is shown in figure 5.28.

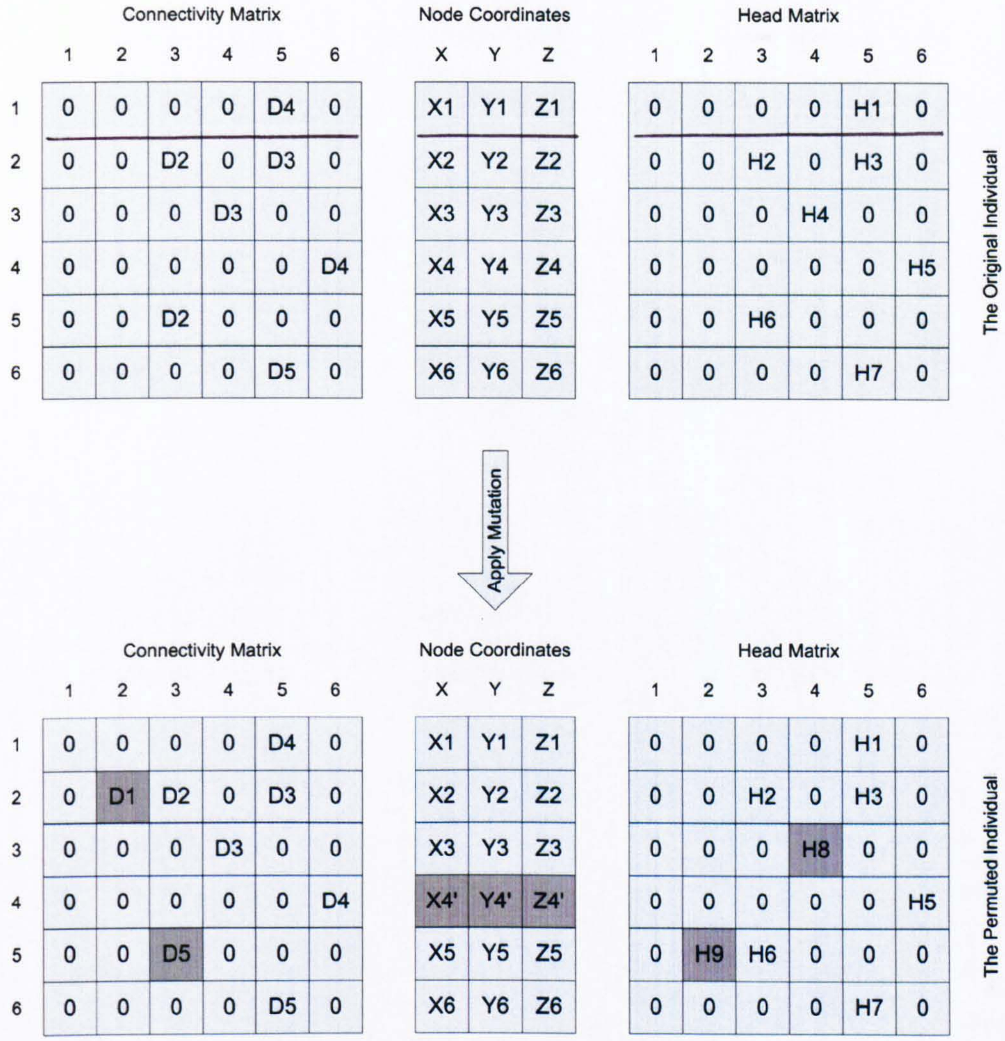


Figure 5.28: Simple Mutation

The simple mutation alters a random cell value in the connectivity matrix to a randomly selected pipe size from the list of available pipe diameters. This mutation also permutes the head values to randomly selected values between minimum and maximum bounds. In the connectivity matrix in figure 5.28, position (2, 2) is selected randomly and a link is added. This link formed a cycle and will be deleted in a later process.

If a node is selected for mutation, all of its (x, y, z) coordinates are permuted. The mutation changes the node coordinates within the boundaries of the world.

5.4.5.2 Swap Mutation

Swap mutation exchanges the values of two randomly chosen positions in each chromosome component. This type of mutation is applied only to the connectivity matrix and head matrix as shown in figure 5.29.

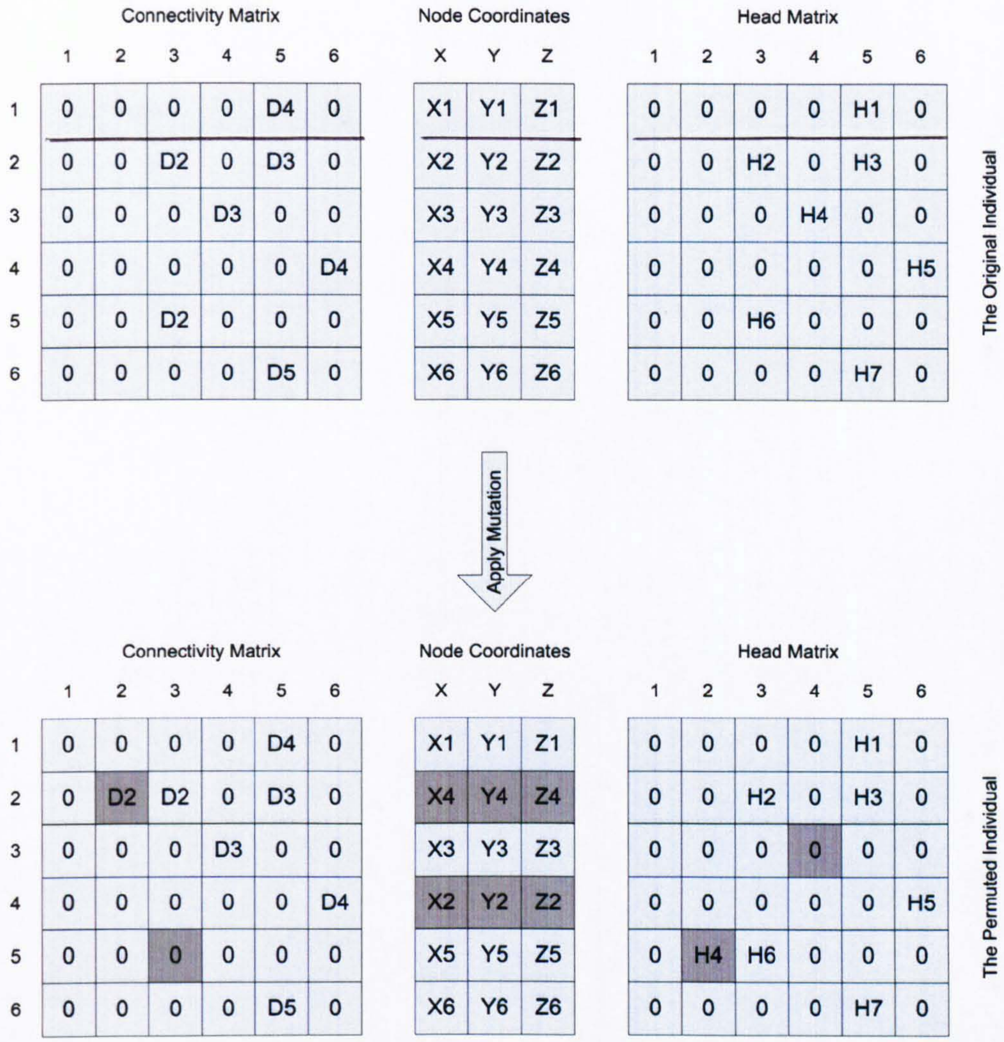


Figure 5.29: Swap Mutation

In the above figure, the value *zero* in location (2, 2) of the connectivity matrix is swapped with the value D2 in location (5, 3). Since cycles are not allowed, D2 in location (2, 2) will be deleted in one of the following step the of the process.

Similarly, the value H4 in location (3, 4) of the head matrix is exchanged with the value *zero*, where no head exists, in location (5, 2).

For node coordinates, the (x, y, z) coordinates of node 2 are swapped with coordinates of node 4.

In this type of mutation, similar mutation rates are applied to head and connectivity matrices but different cells may be swapped in each matrix. This is because both components share the mutation rate but do not share the same selected genes. The genes of each matrix are randomly chosen independently for mutation.

Other types of mutation may be emulated from this type. For example, a non-zero value in the matrix may be swapped with another non-zero value. This type of mutation is subject to: $N \geqslant 2$, where N is the number of non-zero values in the matrix.

5.4.5.3 Combined Mutation

Other types of mutation are developed by combining simple and swap mutation or part of them. Several types of this type of mutation are developed. An example of the combined mutation is shown in figure 5.30.

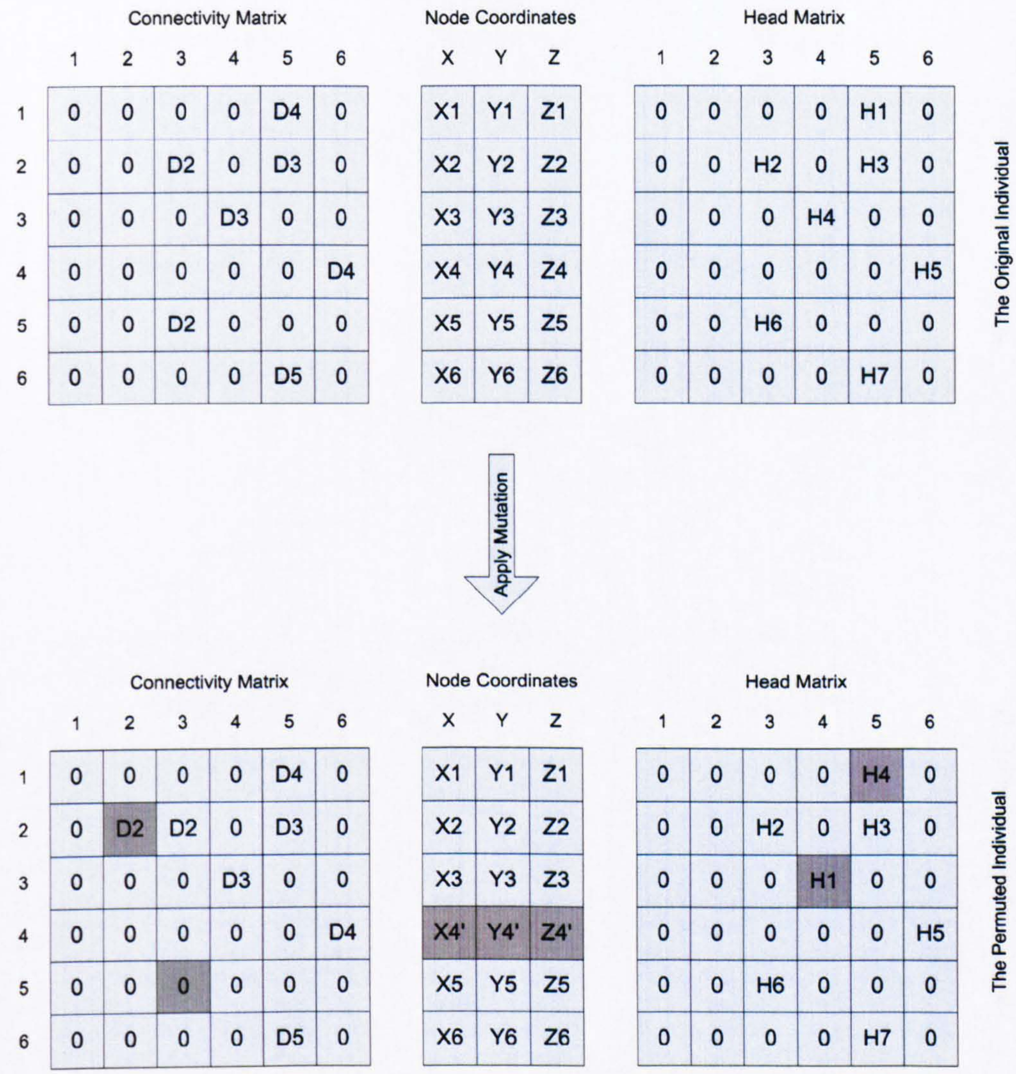


Figure 5.30: Combined Mutation

In the above example, a simple mutation is applied to the node coordinates. Swap mutation is applied to the connectivity matrix by exchanging the pipe size in location (2, 2) with the pipe size in location (5, 3). Two non-zero values in head matrix are exchanged. In this matrix, the value of H1 in cell (1, 5) is exchanged with H4 in cell (3, 4).

5.4.6 Population Size

One of the important decisions needed in the GAs is to set the number of individuals in the population. Population size is typically in the range of 20 1,000, but it can be smaller or much larger (Coley, 1999). If the population is too small, there might not be an adequate supply of genetic materials, and it will be difficult to identify good solutions. On the other hand, if the population is very large, the GA will waste time processing unnecessary individuals, and this may result in unacceptably slow performance.

The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. In order to develop a rational figure of population size, the expected number of nodes need to be estimated. The number of nodes depends on the layout and size of the world.

Empirical methods are used to determine the optimum population size. Optimum population size means the size that causes the quickest convergence for an instance of size N. The typical range of population size used for testing in this thesis is between 500 to 2,000 individuals (see chapter 6). This range is chosen after trying a range of population sizes (50 to 4,000) before settling on a range that seems to work best with the problem presented in this thesis.

5.4.7 Age Structure

Age structure considers removal of individuals by the ageing process regardless of the fitness value. A genetic algorithm with age structure can control selection process by the ageing process and maintain relatively high genetic diversity in a population.

The behaviour of the GA is tested in this work by assigning a parameter called "Maximum Age". In this method, each individual in the GA has a maximum lethal age. When an individual's age reaches the predefined maximum age, the individual is removed from a population by considering the fitness value. The eliminated individuals are more likely to disappear because the selection operation selects good individuals and at the same time eliminate bad individuals from population based on the evaluation of individual fitness.

The assigned age to the children once they are generated from parents by crossover and mutation is *zero*. The children's age as well as parent's age increases by one per each GA iteration. Once the individuals reach the maximum lethal age, it is removed from the population by changing its fitness value to represent a bad solution.

5.4.8 Termination Criteria

Termination is the criterion by which the genetic algorithm decides whether to continue searching or stop the search. This step checks if the user-specified termination criteria are satisfied. Several termination criteria are developed. The model include the following types of termination:

1. **Number of Generations**
   A termination method that stops the evolution when the user-specified maximum number of evolutions have been completed.

2. **Fitness Convergence**
   A termination method that stops the evolution when the fitness is deemed as converged. This can be achieved when user-specified maximum number of consecutive generations are finished without fitness improvement.

3. **Fitness Threshold**
   A termination method that stops the evolution when the best fitness in the current population becomes less than the user specified fitness threshold and the objective is set to minimise the fitness. This termination method also stops the evolution when the best fitness in the current population becomes greater than the user specified fitness threshold when the objective is to maximise the fitness.

The termination type can decided by the user. The program interface is shown in figure 5.31.

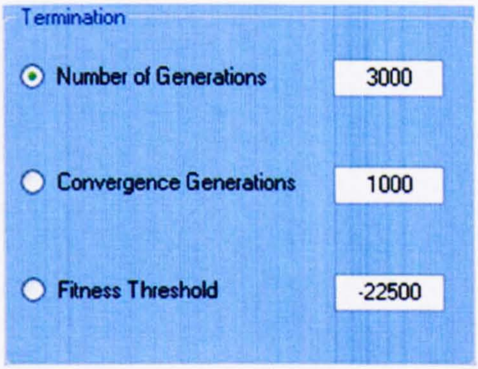


Figure 5.31: Termination Interface

5.5 Chapter Summary

The chapter presented a description of the 'final' model. Various decisions were incorporated within this model. These decisions were explained.

The adopted fitness function is presented. The representation of the chromosome is also presented in this chapter.

The developed GA operators such as crossover and mutation were discussed. Some important GA topics were presented such as population size and termination criteria.

The next chapter consists of detailed testing on the current formulation.

Chapter 6

Tests and Results

6.1 An Introduction to the Chapter

This chapter relates specifically to the experiments carried out with the current formulation. It describes the determination of the values of the various parameters of the GA which should be used to solve real problems. This provides the necessary basis with which to evaluate the use of the technique produced in real-life situations as described in chapter 7.

The chapter starts with a description of five 'manufactured' worlds which will be used in the experiments followed by a description of the experimental design. Following this, a discussion and analysis of these results leading to a set of recommendations to which parameters should be used. The parameters are: crossover and mutation, number of nodes, population size, mutation rate, and adjacency mutation operator.

At the end of this chapter, a special world layout is designed to demonstrate the ability of the model to avoid expensive passages to construct a WDN.

6.2 The Design of Experiments

Most optimisation algorithms have several parameters which may be tuned to adjust algorithm performance. This section describes the design of experiments and testing parameters. In order to determine the parameters, it is necessary to design experiments to evaluate the effect of the parameters on the ability of the model to find a solution.

There are seven main aspects to these parameters: world layout, crossover, mutation, population size, number of nodes, mutation rate, and adjacency mutation operator.
These are discussed individually in section 6.2.1 to 6.2.2.5 below. These have been designed to evaluate different measures.

The general approach in this chapter is to consider a range of parameter values and conduct a sensitivity analysis in order to determine which set of values to use.

6.2.1 The Design of the World

Five simplified 'manufactured' world layouts are used which allow several scenarios to be tested in respect of different GA parameters. The complexity of these worlds varies according to the number and the distribution of demand areas within the world. Each world consists of a source, an obstruction, and a number of demand areas randomly distributed within the boundary of the world.

The simplified area plans were depicted in graphical form as a square of 500×500 units, which was subdivided into 10×10 grid size. This results into a 50×50 cells (i.e. grid). The same grid system was adopted in the world model proposed for the initial testing presented in chapter 4.

(a) World 1

World 1 is the simplest geographical form used in these experiments. It consists of one water source, one demand area (e.g. residential), and one lake. The layout of world 1 is shown in figure 6.1.

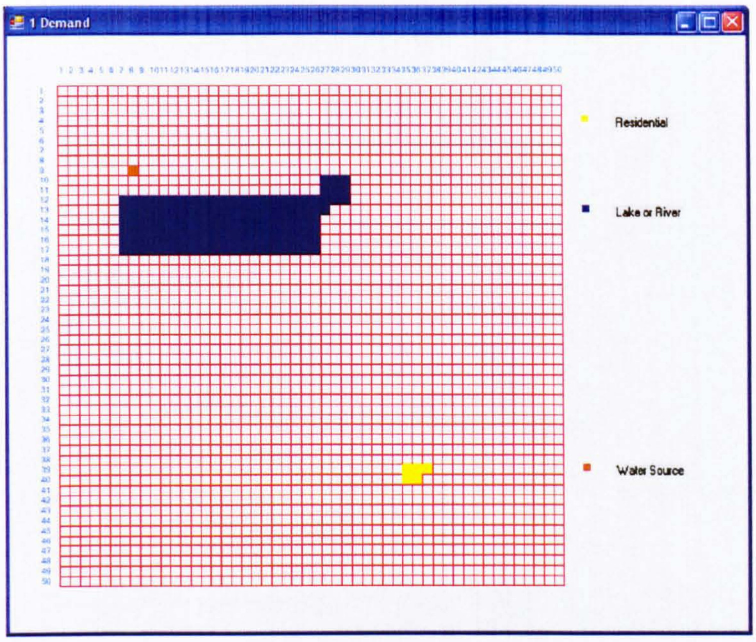


Figure 6.1: The Layout of World 1

(b) **World 2**

World 2 consists of one water source, three demand areas (one residential area, one industrial, and one retail), and a lake. The distribution of the demand areas and other world components is shown in figure 6.2.

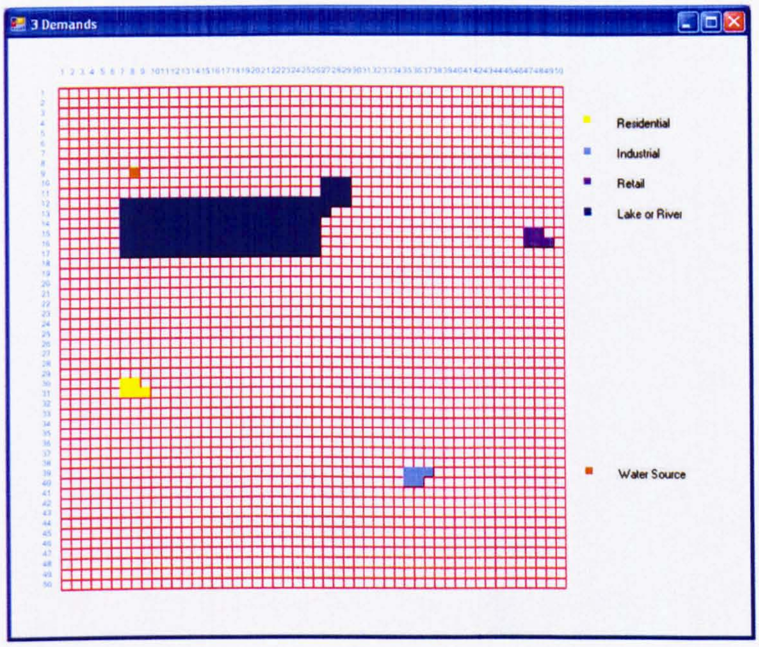


Figure 6.2: The Layout of World 2

(c) **World 3**

World 3 consists of one water source, five demand areas (two residential areas, two industrial, and one retail), and one lake. The distribution of the demand areas and other world components is shown in figure 6.3.

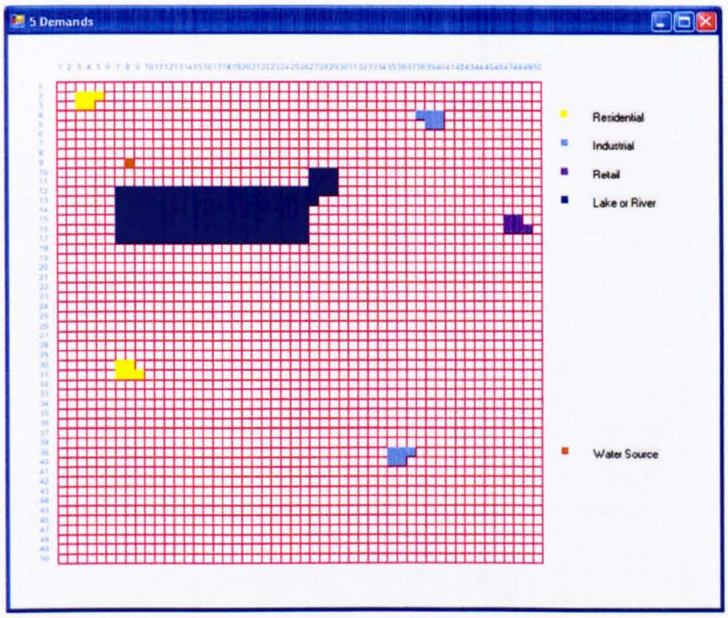


Figure 6.3: The Layout of World 3

(d) **World 4**

This world consists of one water source, seven demand areas (three residential, two industrial, and two retail), and one lake. The distribution of the demand areas and other world components is shown in figure 6.4.

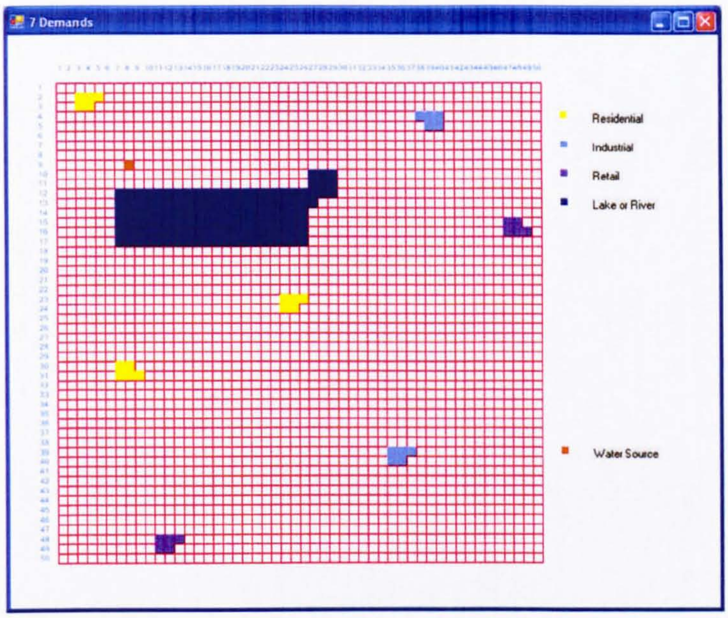


Figure 6.4: The Layout of World 4

(e) **World 5**

This world is the most complex geographical layout adopted in this chapter. It consists of one water source, nine demand areas (three residential, three industrial, and three retail), and one lake. The distribution of the demand areas and other components is shown in figure 6.5.

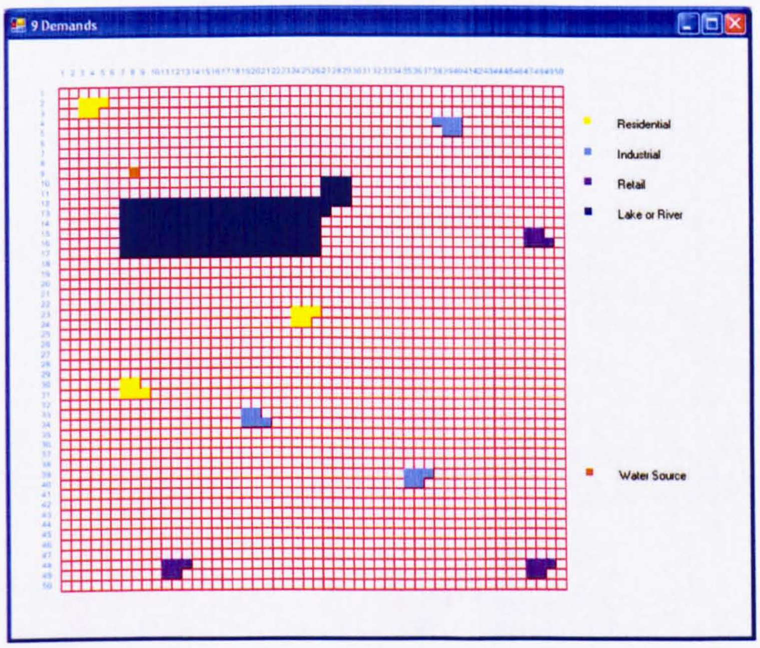


Figure 6.5: The Layout of World 5

6.2.2 Tests and Results

This section provides a testing regime to determine the effect of the complexity of the world on the selected GA parameters.

The success of the GA depends on the choice of operators and their associated parameters. Goldberg (1989) suggested that good GA performance requires the combination of the following: a high probability of crossover, a low probability of mutation, and a moderate population size. Different sets of GA parameter values are tested in this chapter.

It is traditional to compare the performance of GAs by executing different algorithms with similar GA parameters (Simpson et al., 1994; Savic and Walters, 1997*b*; Babayan et al., 2005). This allows results from different runs to be compared independently.
In each model run, GA converges to a certain optimal solution using the same model parameters. This is because each GA run starts with a different initial population seed. It is therefore a common approach to take the average of optimisation runs (Cantú-Paz and Goldberg, 2000; Han et al., 2004; Wang and Tai, 2005; Kadu et al., 2008). The optimal solutions obtained are chosen to be averaged over 10 runs for each set of parameters.

With many real-world problems, and especially those with large numbers of decision variables, certain decision variables will be more important than others will. In fact, in many problems, the majority of the fitness of the solution depends on a small number of the decision variables. Therefore, the most important model parameters that affect the decision variables are tested in this chapter.

Suh and Van Gucht (1987) and Passone et al. (2006) argued that problem specific knowledge can usefully be incorporated into the GA search. Domain knowledge may be used to prevent unfit chromosomes, or those which would violate problem constraints, from being produced in the first place. This avoids wasting time evaluating such individuals, and avoids introducing poor performers into the population. In addition, Goldberg (1989) described the techniques for adding knowledge-based crossover and mutation. He also discusses the hybridisation of GAs with other search techniques.
The knowledge obtained during the development of the model and from the initial experimentations are utilised to set the initial values of the GA model. This is useful to start testing model parameters and make each GA run more task specific.

The selection mechanism of the GA plays an important role. It drives the search towards better individuals and promotes the convergence. The selection operator selects the best chromosomes as determined by the fitness value. If there are two or more chromosomes with the same best fitness value, one of them is chosen randomly. This is because the 'Sort' command in Microsoft Visual Basic reorders a set of values ascending or descending according to user requirements in which the order of equal elements is effectively random (Microsoft Corporation, 2011*a*). The 'Sort' command in Microsoft Visual Basic, like the most built-in sorters, uses a quick-sort implementation in a helper class within the programming package.

The GA terminates when a maximum number of generations is reached. In this work, 3,000 generations have been found to be adequate in the experimentations and set as the default termination criteria. However, it is possible to amend this number as required.

The primary design of WDNs involves cost-effective specifications of a pipe network layout to satisfy expected consumer water demands within required pressure limits. In order to run the model, the demand amount per each demand area should be known. This amount is chosen to be 10,000 unless otherwise stated. However, this amount could be changed according to user preferences.

It is also assumed that the available water amount at the source node is enough to supply the entire world. For instance, if a world consists of three demand areas, the available water supply at the source node will be 30,000 (10,000 for each demand area). However, the amount of available water supply could be changed to reflect various real-life scenarios (e.g. water shortage).

At a demand node, demand benefit factor (DBF) (see section 5.3.3.3 for details) is decided to be 50% of the incoming flow to a demand node.

The crossover rate is the frequency with which the crossover operator is applied to the individuals. A crossover rate of 100% is used in all experiments.

In this chapter, three measures of solution quality are mainly employed. The first quality measure is the average of the best fitness values. This average is used as a measure because it provides an average measure of how individuals have performed in the problem domain.
If the GA parameters are correctly chosen, the population should evolve over successive generations so that the fitness of the best and the average fitness in each generation progress towards the global optimum.
It is reasonable to define convergence as the event that the percentage of improvement falls below a specified threshold. In this thesis, a threshold of 5% change in the average of the best fitness value is used.

The second measure is the Percentage of Total Demand Satisfied (abbreviated as 'PTDS'). The PTDS is said to be 'good' and the model performed well if 95% or more of the total demand is satisfied. The total demand satisfied is an element of the fitness function which effect the fitness value. It is therefore necessary to introduce another assessment measure to the problem solution.

The third measure is the number of GA runs which share at least 95% of the average solution for the same set of GA parameters. This measure demonstrates how well each GA solution is spread around the average.

To provide the reader with a structure on how GA parameters will be tested, table 6.1 presents the testing worlds, GA parameters and their associated values.

Table 6.1: The General Testing Regime

Chapter's Section	World	Testing Parameter							
		Crossover & Mutation	Population Size	Number of Nodes	Mutation Rates (%)			Adjacency Mutation	
					Population	Head & Link	Node Positions	Head	(x, y) Positions
6.2.2.2 Population Size 6.2.2.3 Number of Nodes	1	Crossover type 2 with mutation type 1	500, 1000, 1500 and 2000	10, 15, and 20	10	5	10	3	10
6.2.2.1 Crossover & Mutation 6.2.2.2 Population Size 6.2.2.3 Number of Nodes	2	Crossover type 1 and type 2 and Mutation type 1, 2, 3, and 4	500, 1000, 1500, and 2000	10, 15, and 20	10	5	10	3	10
6.2.2.2 Population Size 6.2.2.3 Number of Nodes 6.2.2.4 Mutation Rate 6.2.2.5 Adjacency Mutation	3	Crossover type 2 with mutation type 1	500, 1000, 1500 and 2000	10, 15, and 20	5, 10, and 15	1, 5, and 10	10, 15, and 20	3, 6, 9, 12, and 25	5, 10, 20, 30, 40, 50, and 500
6.2.2.1 Crossover & Mutation 6.2.2.2 Population Size 6.2.2.3 Number of Nodes	4	Crossover type 1 and type 2 and Mutation type 1, 2, 3, and 4	500, 1000, 1500, and 2000	10, 15, and 20	10	5	10	3	10
6.2.2.2 Population Size 6.2.2.3 Number of Nodes	5	Crossover type 2 with mutation type 1	500, 1000, 1500 and 2000	10, 15, and 20	10	5	10	3	10
6.3 A special World Layout	Special	Crossover type 2 with mutation type 1	1000	10	10	5	15	5	100

6.2.2.1 Crossover and Mutation

They are two basic operators in which the GA performance depends on them. The traditional view is that the crossover is primarily responsible for improvements in fitness, and the mutation serves a secondary role of reintroducing gene values that have been lost from the population.

To determine the effect of the combination of crossover and mutation on the model, the testing regime presented in figure 6.6 is employed.

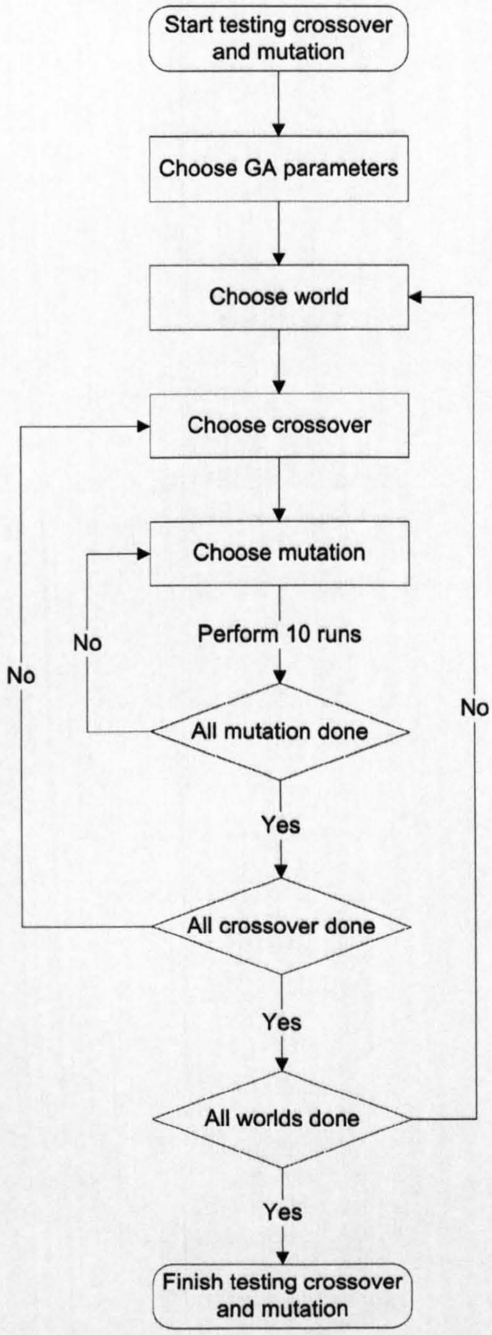


Figure 6.6: Testing Regime for Crossover and Mutation

Different crossover and mutation types are tested first to find the most effective operators for the GA search. Due to the large number of possible crossover and mutation combinations, some selected combinations are tested.

The crossover types that are used in these experimentations are crossover type 1 (explained in section 5.4.4.1) and crossover type 2 (explained in section 5.4.4.2). The crossover operator in these experimentations is applied to all components of the chromosome.

Several mutation types are derived and tested in this research (mutation types are defined in section 5.4.5).

Current mutation types used with crossover (type 1 and type 2) are explained in table 6.2.

Table 6.2: Mutation Types and Their Definitions

Mutation Type	Meaning	Parameter Definition
1	Uses "Simple Mutation" explained in section 5.4.5.1.	Mutation rates define the number of pipes sizes and head values to be changed and the location of randomly chosen nodes.
2	Uses "Swap Mutation" explained in section 5.4.5.2 for head and connectivity matrices (No mutation rate required). "Simple Mutation" is used to assign random (x, y, z) coordinates to a randomly chosen node.	For node coordinates, mutation rate defines the number of nodes to be changed.
3	Combines aspects from mutation type 1 and type 2. It uses "Simple Mutation" to permute the head and node coordinates; and "Swap Mutation" to swap the links.	Mutation rates define the number of head values to be changed and the location of randomly chosen nodes.
4	Similar to mutation type 3, it combines aspects from mutation type 1 and type 2. It uses "Simple Mutation" to permute the links and node coordinates; and "Swap Mutation" to swap the heads.	Mutation rates define the number of link values to be changed and the location of randomly chosen nodes.

Among the five worlds defined in section 6.2.1, two worlds are chosen for tests to select suitable crossover and mutation types. World two is chosen to represent a simple world layout and world four is chosen to represent a more complex layout.

Other parameters need to be selected to run the model with the aforementioned worlds. These parameters are: number of nodes, population size, number of generations, and mutation rate.

Eight nodes are selected for world two and ten nodes for world four. The selected number of nodes guarantees that the source and each demand area will have one node. The remaining nodes will work as intermediate nodes. Both worlds share the other selected GA parameters as shown in table 6.3.

Table 6.3: Selected GA Parameters for Crossover and Mutation Testing

Population	Mutation Rates		
Size	Individuals	Connectivity & Head	Node Coordinates
1,000	10%	5%	15%

The first experiment is for world two in which eight test sets are conducted. In this experiment, crossover type 1 is combined with mutation types 1, 2, 3, and 4; and crossover type 2 is combined with same set of mutation types. The convergence graph for these tests is shown in figure 6.7.

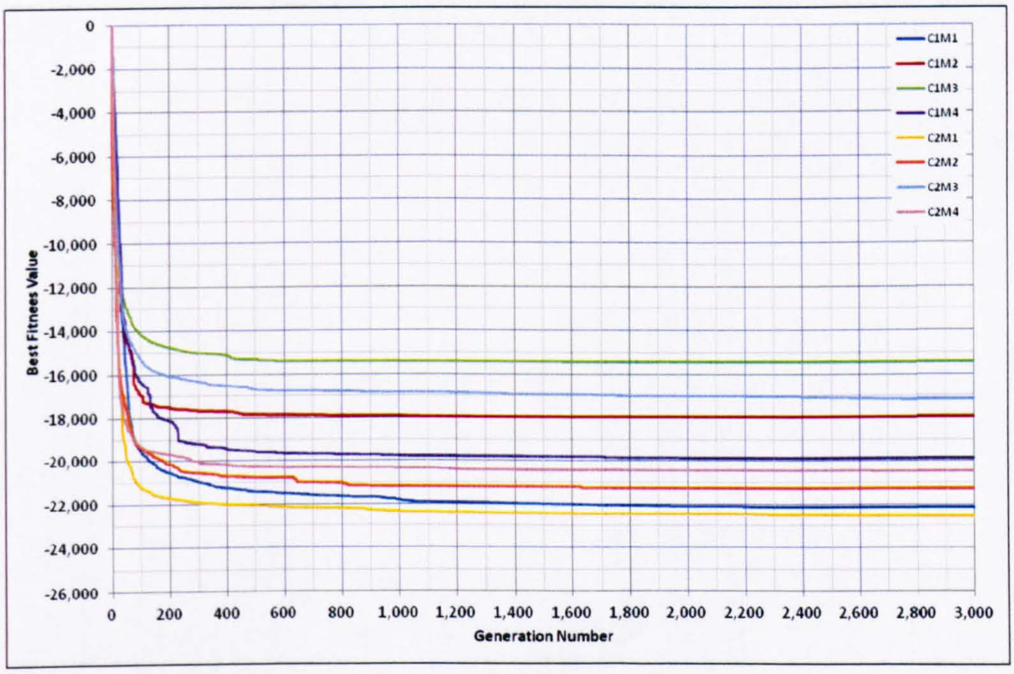


Figure 6.7: World 2 GA Convergence With Different Types of Crossover and Mutation

Where C1 and C2 denote crossover type 1 and crossover type 2 respectively; M1, M2, M3, and M4 denote the mutation types defined in table 6.2.

The convergence plots in figure 6.7 for each solution are similar in pattern. Due to scale and representation issues, the plots start from the third generation where all fitness values are less than *zero*. After a rapid initial improvement, solutions are generally stable after 200 to 500 generations. The best fitness value improves more slowly in later generations. Crossover tended to cause fitness values to decrease more rapidly on early generations while mutation provided a certain amount of a 'slow' random search and helped ensure new points of the search space are explored. This is in agreement with Beasley et al. (1993*b*), who points out that mutation becomes more productive but slow, and crossover less productive, as the population converges.

Best GA convergence is recorded in this experiment when crossover type 2 and mutation type 1 (C2M1) are used. A relatively 'good' GA convergence is recorded when mutation type 1 is used with crossover type 1 (C1M1). The difference in the fitness value between C2M1 and C1M1 is 466, this being less than 2%. This indicates that in both combinations, mutation type 1 successfully explored new regions of the search space and avoided premature convergence.

Conversely, crossover type 1 and crossover type 2 with mutation type 3 (C1M3 and C2M3) recorded poor performance compared to other combinations. This indicates that mutation type 3 is the least productive type.

The percentage of total demand satisfied (PTDS) is another measure to assess the best crossover and mutation combinations. The PTDS for each combination is presented in table 6.4.

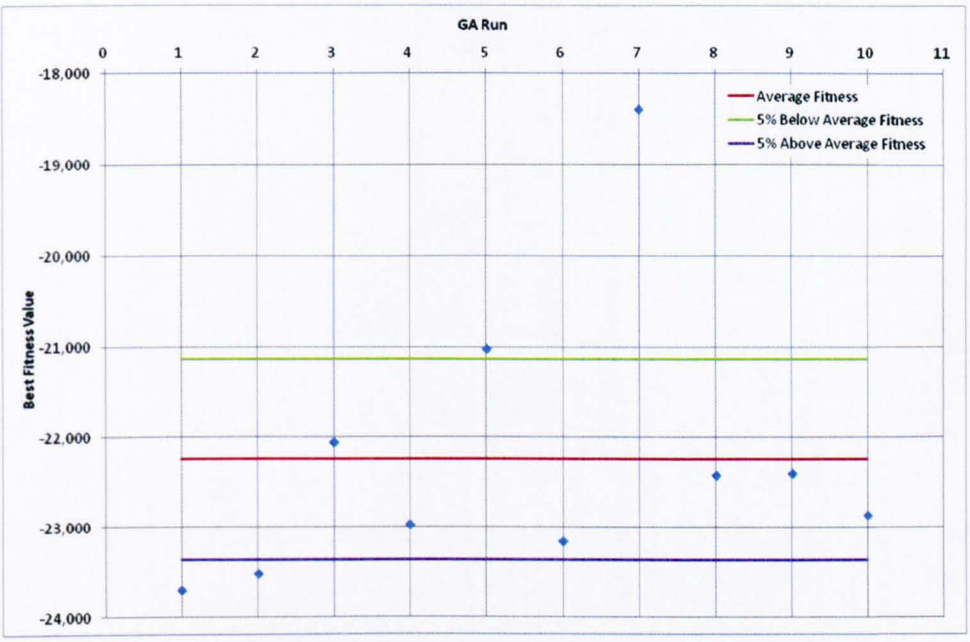
Table 6.4: World 2 PTDS With Different Types of Crossover and Mutation

Crossover and Mutation Types	C1M1	C1M2	C1M3	C1M4	C2M1	C2M2	C2M3	C2M4
Max. Demand Satisfied (%)	99.37	81.46	70.51	90.24	99.83	96.24	80.80	94.12

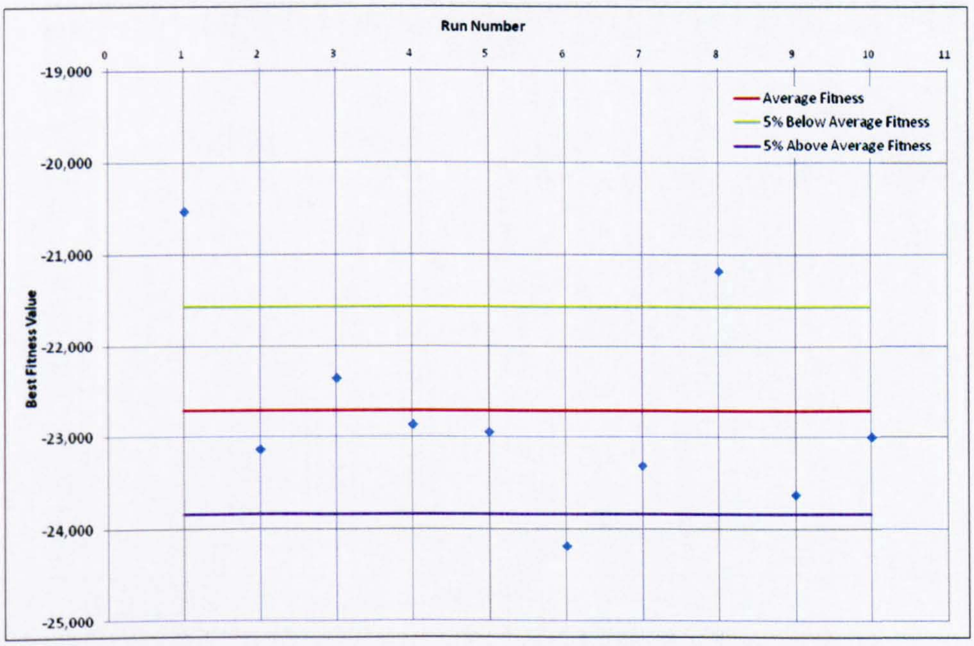
In this table, the highest values of PTDS are for C1M1 and C2M1 of more than 99%. Both combinations achieved an optimal solution in terms of the total amount of water delivered to demand areas. The performance of both combinations is considered similar because the difference in PTDS between them is marginal (= 0.46%). They proved to be the best among the other combinations.

The results of PTDS for C1M3 and C2M3 indicates poor performance. This performance is similar to the fitness value performance measure using the same combinations.

Only C1M1 and C2M1 will be taken into account for the third measure to assess how well these combinations represent a solution.
The number of the best fitness values for each GA run that recorded 95% or more of the average fitness value out of the 10 runs for C1M1 was 6 and for C2M1 was 7. This is illustrated in figure 6.8.



(a) Fitness Values for 10 Runs Using C1M1



(b) Fitness Values for 10 Runs Using C2M1

Figure 6.8: Fitness Values and Averages for C1M1 and C2M1 Used with World 2

In the above figures, the central line is the average fitness value for 10 GA runs, the top line is the 5% difference below the value of the average fitness, and the bottom line is the 5% difference above the average fitness value.

For C1M1, the best fitness value of 60% of the GA runs being 5% or less from the average fitness value. This percentage increases to 70% for C2M1. These percentages are reasonable as they represent more than 50% of the total GA runs.

Both crossover types produced 'good' results in terms of PTDS and GA convergence when used with mutation type 1. Both are efficient because they are able to exchange the required chromosome information to find the optimal or near-optimal solution using world 2. The combination of both crossover types did not prove to work well with other types of mutation.

In this world, a minimum of four links and five nodes are required to connect the demand areas to the source. Both crossover types were able to:

1. Exchange pipe information and have minimum number of links with suitable pipe sizes;
2. Exchange head information.

Although the crossover plays an important role in exchanging coordinates between individuals, mutation has a larger effect on node positions as it changes the position of a single node to adjacent values (adjacency mutation is explained in section 6.2.2.5.2).

It can be seen from GA convergence figure and PTDS table that crossover type 2 performs better than type 1 when used with different types of mutation. This performance is achieved because crossover type 2 uses a fixed mixing ratio between two parents. Unlike, crossover type 1, crossover type 2 enables the parent chromosomes to contribute small segment levels rather than big segment levels. If the mixing ratio is 50%, the offspring has approximately half of the genes from the first parent and the other half from the second parent, although crossover points are randomly chosen.

Mutation type 1 proved to be the best for this size of world when used with crossover type 1 or crossover type 2. The highest PTDS (more than 99%) are achieved using these combinations. As well as GA convergence plot, the values in table 6.4 shows that mutation type 3 proved to be the worst among other types of mutation when used with both crossover types.

Mutation type 1 permutes the genes in connectivity and head matrices and node coordinates. It randomly permutes genes in connectivity matrix to one of the available pipe diameters including zero diameter where no pipe exists. Proper pipe size may be allocated to the nodes which can carry the suitable amount of water. Bigger pipe sizes increase material and installation costs but they are able to carry the required amounts of water to supply demand areas.

Similar to the connectivity matrix, this mutation randomly selects genes at the head matrix and permutes the values within a predefined bound. This bound restricts the increment / decrement of the current gene value. This mutation technique is further explained in section 6.2.2.5.1.

A similar restriction but different in value is applied to node coordinates. If a node is selected for mutation, the movement of (x, y) node coordinates is restricted with a predefined bound. This mutation technique is explained in more details in section 6.2.2.5.2. The z coordinate is bounded by the minimum and maximum allowed node elevation.

These bounds restrict the values of chromosome components to float within a smaller search space.

Swapping genes only in the connectivity matrix proved to be ineffective as in mutation type 3. Although simple mutation is used to permute head and coordinate matrices, GA could not achieve near-optimal results. The mutation in connectivity matrix is unable to place suitable pipe sizes that are able to carry required water amounts from source to demand areas. A chromosome might not have a suitable pipe size to be exchanged with another size. If the chromosome has the suitable pipe size, the mutation may not be able to move this value to the desired position to match the required head to supply water to demand areas.

While 10 GA runs might not be enough to decide which combination is better in this context, further runs may prove applicability of each type.

Other factors may affect the GA convergence and PTDS, such as:

- The location of the demand areas, source and the obstacle.
- The size of the obstacle and the demand areas.
- The mutation rates.
- The maximum allowed head and node coordinates movement.
- The decision applied to the demand benefit.
- Cost parameters.

For this type of world, the layout is not complex. Therefore, GA and other model parameters proved to be able to find near-optimal solution and deliver sufficient water amounts to demand areas.

To test the similar parameters on a more complex world layout, world 4 is chosen and similar tests to world 2 were implemented. Same crossover, mutation and other GA parameters are used with this world.

The convergence graph for this world is shown in figure 6.9.

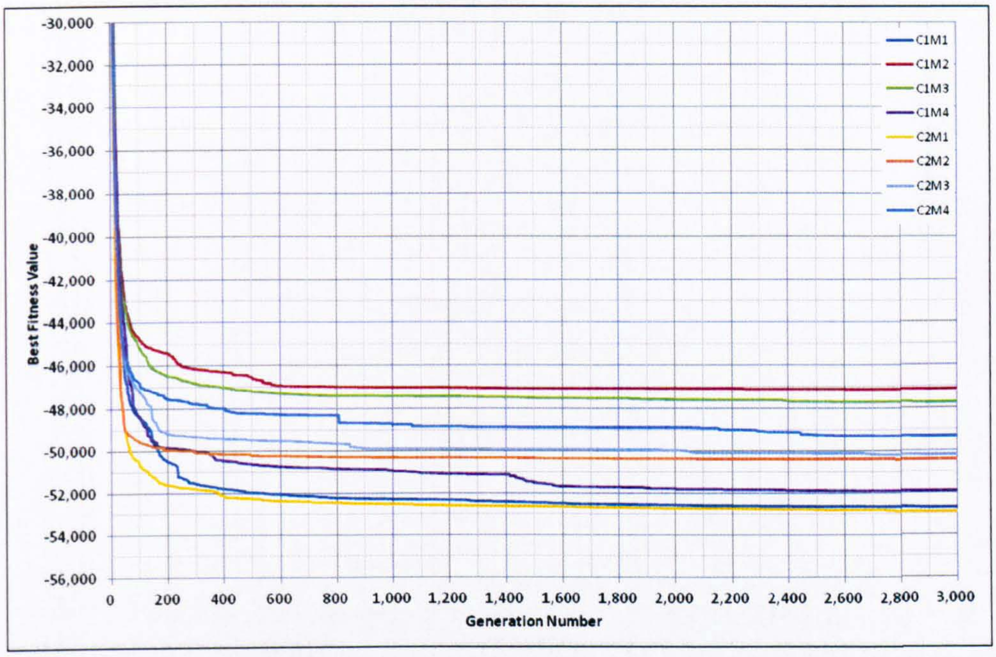


Figure 6.9: World 4 GA Convergence With Different Types of Crossover and Mutation

Most of the test results of this world are similar to test results obtained using different types of crossover and mutation with world 2.

Best GA convergence recorded in this experiment when crossover type 1 and mutation type 1 (C1M1) are used together. A very close best value to the previous one is recorded when mutation type 2 is used with crossover type 1 (C2M1). Both best fitness values in both experiments are very close in which the difference is 185, this being less than 0.5%. C1M2 and C1M3 recorded poor performance compared to other combinations.

In figure 6.9, the solution converges between generation 500 and 1,000, while the convergence was quicker using world 2 (see figure 6.7). This is because the GA finds the near-optimal solution relatively quickly for world 2 as it has only three demand areas to be supplied with water. The GA does not have to search through very large supply alternatives to find the optimum solution. The delay of GA convergence increases by increasing the number of demand areas.

In addition, from figure 6.9, the convergence plot for each solution is not similar in pattern. It can be seen that after rapid initial convergence, solutions become generally stable after 500 to 1000 generations for C1M1, C2M1, and C2M2. Other combinations recorded slower GA convergence.

By looking at the percentage of average demand satisfied (PTDS), table 6.5 presents different percentages for each test.

Table 6.5: World 4 PTDS With Different Types of Crossover and Mutation

Crossover and Mutation Types	C1M1	C1M2	C1M3	C1M4	C2M1	C2M2	C2M3	C2M4
Max. Demand Satisfied (%)	96.76	88.57	87.50	97.69	96.32	93.40	92.64	89.34

The difference between PTDS for C1M1 and C2M1 is marginal, this being less than 0.5%. The highest PTDS value is 97.69% for C1M4. Although this value is more than what is achieved using C1M1 and C2M1 with a small percentage, the best fitness plot using C1M4 is less than the best fitness plot using C1M1 and C2M1. In addition, the convergence plot using C1M4 is not as stable as the convergence plots for C1M1 and C2M1. The difference in PTDS between C1M4 and C1M1 is less than 1% and is 1.4% for C2M1 which are not significant. However, because C1M4 has the highest PTDS value, further analysis is required. Thus, it will be considered in the third performance measure.

Table 6.6 presents number of GA runs whose fitness values are at least 95% of the average fitness value for C1M1, C1M4, and C2M1.

Table 6.6: Number of GA Runs Achieved $\geqslant$ 95% of the Average Fitness Value for C1M1, C1M4, and C2M1 for World 4

Combination Type	Number of GA Runs That Achieved $\geqslant$ 95% of the Average Fitness Value
C1M1	7
C1M4	7
C2M1	7

The above table shows that the three combination types performed well in which the fitness value of 70% of the GA runs were 95% or more of the average fitness value. Although the convergence plot for C2M1 is better than those of C1M1 and C1M4, the spread of the number of best fitness values around the average is equal.

At this stage, it becomes harder to decide which combination among these performed better for world 4. It is therefore decided to look at the number of GA runs that achieved $\geqslant$ 95% of the best fitness value for C1M1, C1M4, and C2M1. This is an additional measure to find best performances. Table 6.8 presents number of GA runs that achieved $\geqslant$ 95% of the best fitness value for C1M1, C1M4, and C2M1.

Table 6.7: Number of GA Runs That Achieved $\geqslant$ 95% of the Average Fitness Value for C1M1, C1M4, and C2M1 For World 4

Combination Type	Number of GA Runs That Achieved $\geqslant$ 95% of the Average Fitness Value
C1M1	6
C1M4	2
C2M1	4

The above table shows that the best combination type is C1M1 in which 60% of GA runs that achieved $\geqslant$ 95% of the best fitness value. C1M4 was the worst among the other two types in which only 20% runs achieved the minimum threshold of convergence performance and therefore will be excluded from the considerations.

Although the GA convergence plot for C2M1 is better than C1M1, only 40% of the GA runs using C2M1 achieved $\geqslant$ 95% of the best fitness value.

Both crossover types proved to be effective when used with different types of mutation. In addition, most of PTDS values are greater than 92% which indicates that other combinations can be used but they are not recommended. As with world 2, mutation type 1 proved to be the best for world 4 when used with crossover type 1 or crossover type 2 (see figure 6.9).

Similar to world 2, both crossover types produced 'good' results with world 4. Both were successful in exchanging chromosome parameters to find the optimal or near-optimal solution with world 4. GA and other model parameters proved to be able to find the least-cost design solution and deliver sufficient water amounts to demand areas.

The PTDS achieved using C1M1 and C2M1 in world 2 is greater than 99%

while in world 4 is between 96% and 97%. The more complex the world layout, the less PTDS is achieved.

The number of GA runs using C1M1 and C2M1 in both worlds were very close. This is due to the nature of the GAs as they start from a population of randomly generated individuals and the course taken by the algorithm is determined by random numbers.

6.2.2.2 Population Size

Population size is a critical parameter with respect to the correct functioning of GAs (Herrera et al., 1998; Cantú-Paz and Goldberg, 2000). This is due to the requirement for sufficient genetic materials, allowing a the search to cover the search space adequately.

The success in applying the GA to a particular problem often depends on the adequacy of the population size and the difficulty of the problem. The population size depends on the problem difficulty, but typically contains several hundreds or thousands of possible solutions.

In order to implement the entire encoding successfully, it is important to recognise which aspects of the GA are affected by the choice of encoding. In particular, if an encoding is chosen which has poor expected chromosome coverage, the implementation of population size and/or mutation rate must be adjusted to compensate. Complex chromosome representations may require a large population size to achieve the desired expected coverage level, while computational time per generation increases drastically. In addition, increasing the population size may slow the overall population improvement rate (Goldberg, 1989).

Fitness evaluation is typically responsible for the majority of computation time in GA. If the population is too small, there might not be an adequate supply of genetic materials, and it will be difficult to identify good solutions. On the other hand, if the population is very large, the time of processing unnecessary individuals may result in unacceptably slow performance of the algorithm.

Understanding the role of the population during the search is likely to allow the reduction of population size and improved efficiency of the search.

The results of initial model tests with a population size up to 200 individuals were disappointing. This drawback can be overcome using larger population sizes. By doing so, computational time is expended to evaluate the fitness function.

The use of a large population has several advantages. First, this allows the GA to examine a large number of positions in the solution space simultaneously. Second, a large population is more resistant to the loss of diversity in the population.

The testing regime to determine the effect of the population size on the model is shown in figure 6.10.

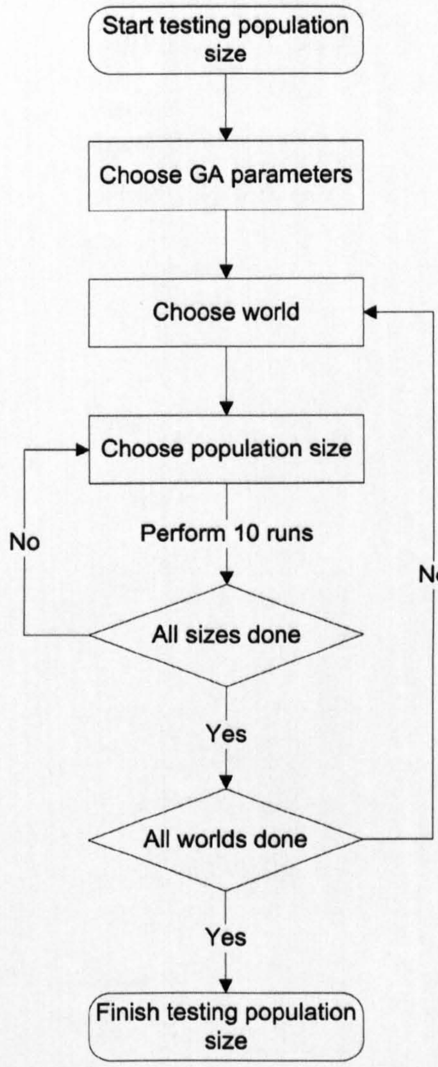


Figure 6.10: Testing Regime for Population Size

Different population sizes are tested in this work to find the suitable size for different worlds. GA tests are carried out using population sizes of 500, 1,000, 1,500, and 2,000 individuals for different worlds. All the five worlds are used in these tests. Crossover type 2 and mutation type 1 are used. Mutation rates are as follows: 10% applied to individuals, 5% applied to connectivity and head matrices, and 15% for node coordinates.

For World 1, which is the simplest geographical representation among other defined worlds, the GA convergence showed that best fitness values are achieved faster with smaller population sizes. Increasing population size does not yield better results for this world (see figure 6.11).

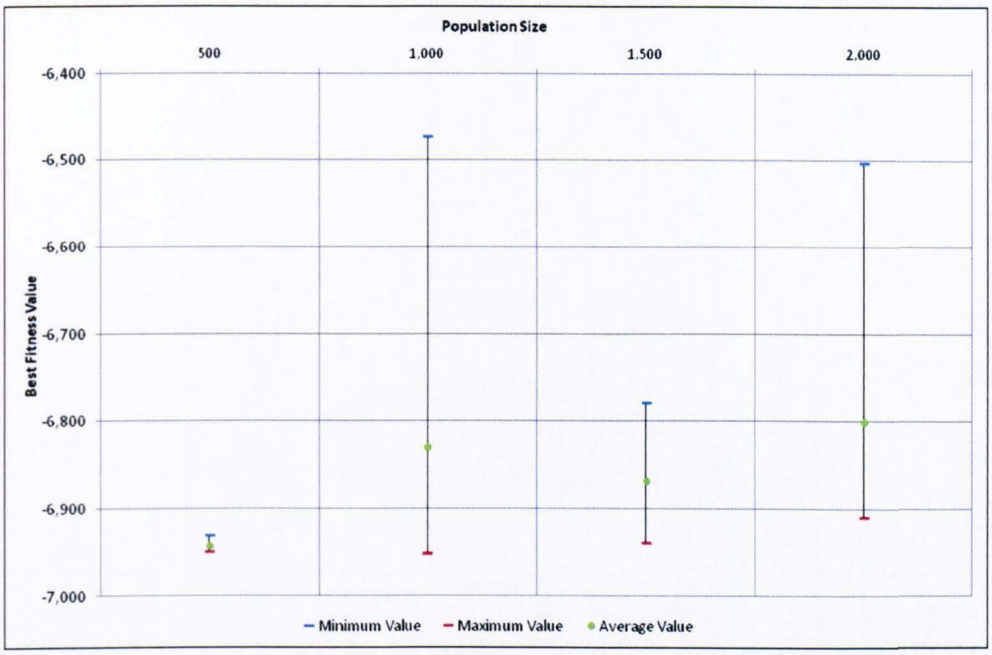


Figure 6.11: Best Fitness Values for Different Population Sizes Using World 1

The above figure shows that the best fitness is for a population size of 500 individuals. However, the best fitness values for all population sizes are close. The difference between the fitness values using 500 individuals (the best value) and 2,000 individuals is approximately 2%. It was also found that the performance actually is weakened beyond a population size of 1,000 individuals.

It is easy for the GA model to find a optimal solution for layouts as simple as world 1. The PTDS reaches 100% during the first 20 generations for all population sizes.

The same set of population sizes are used with world 2. The best fitness values using this world are plotted in figure 6.12.

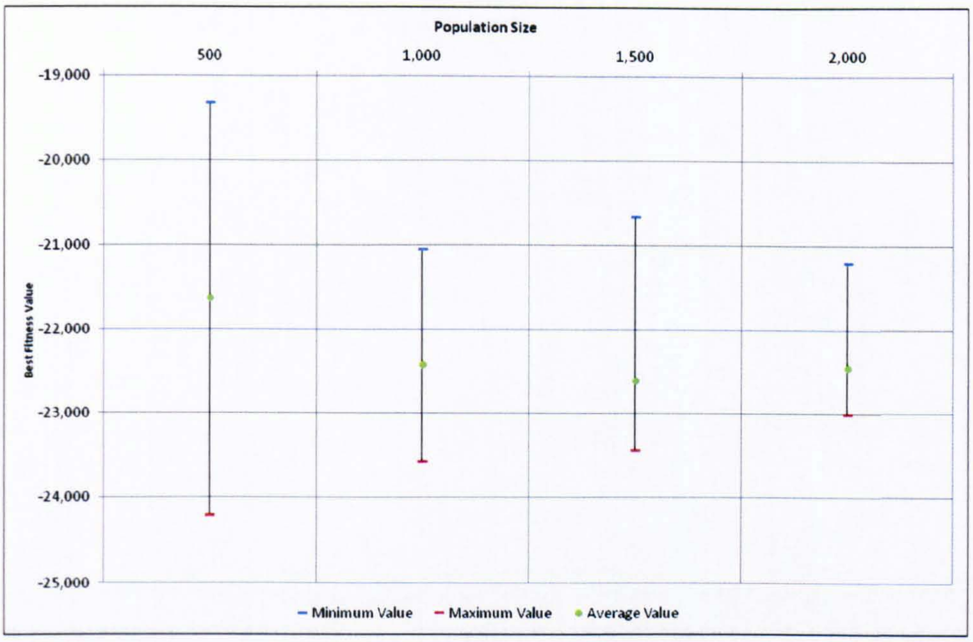


Figure 6.12: Best Fitness Values for Different Population Sizes Using World 2

In figure 6.12, a population size of 500 individuals performed worse by approximately 4.3% from the lowest fitness value (for 1,500 individuals). Larger population sizes performed better and produced very close solutions. The PTDS for all population sizes is more than 99.5%.

The gap in the performance between population size of 500 individuals and other population sizes increases in world 3 as shown in figure 6.13.

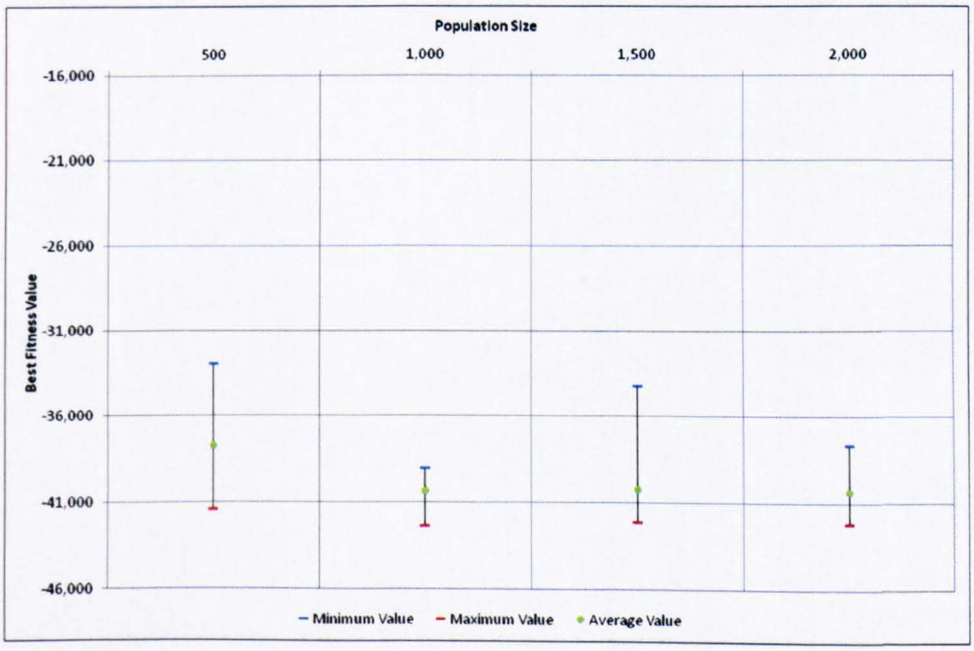


Figure 6.13: Best Fitness Values for Different Population Sizes Using World 3

Similar to world 2, figure 6.13 also shows that population sizes of 1,000, 1,500, and 2,000 individuals produced very close solutions. Among the three population sizes, the difference between the fitness values of 1,500 individuals and 2,000 individuals is marginal (less than 0.5%).

A population size of 500 individuals performed worse by 6.5% from the lowest fitness value (for 2,000 individuals). In addition, using 500 individuals in the model produces a PTDS of 95.5%. Although this percentage is acceptable in some instances, the model produced better PTDS (more than 99%) using larger population sizes. A population size of 500 individuals proved to be inefficient for worlds more than three demand areas.

The selected population sizes show a different behaviour with worlds containing more demand areas. Figure 6.14 shows world 4 GA convergence for different population sizes.

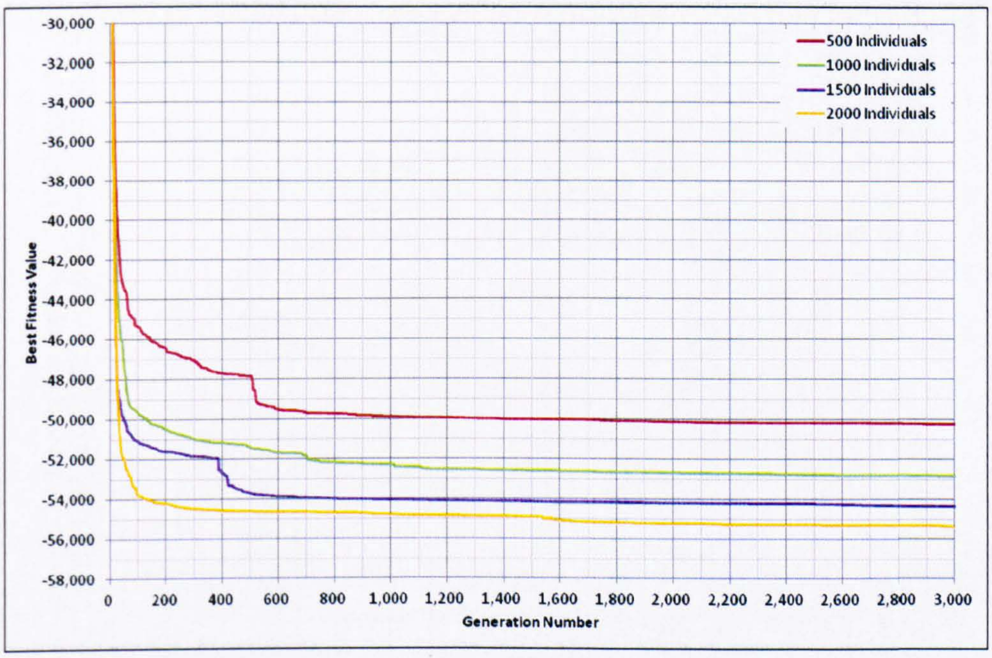


Figure 6.14: World 4 GA Convergence for Different Population Sizes

The above figure shows that using a population size of 2,000 individuals had recorded the best GA convergence among other population sizes. As population size grows, tendency of population diversity increases to generate chromosomes uniformly distributed in the search space. The above figure also shows that the worst GA convergence is recorded when a population size of 500 individuals is used in the model.

Table 6.8 presents the difference between the best fitness values for each population size and the fitness value using the population size of 2,000 individuals which is the lowest fitness value.

Table 6.8: Best Fitness Values and Their Differences from the Lowest Value for Different Population Sizes Using World 4

Population Size	Best Fitness Value	Difference From the Lowest Fitness Value
500	-50238.3	9.4%
1,000	-52825.8	4.7%
1,500	-54340.1	1.9%
2,000	-55428.5 (Lowest)	0.0%

The above table shows that the fitness value using 500 individuals is well below the lowest fitness value while the figures for larger population sizes performed better.

Using 500 individuals is not suitable for this world layout while population sizes of 1,000 and 1,500 individuals can be accepted and used for this world. They achieved ratios greater than 95% of the best fitness value obtained with 2,000 individuals.

The PTDS and their differences from the best PTDS value for different population sizes using world 4 are presented in table 6.9.

Table 6.9: PTDS and Their Differences From the Best PTDS Value for Different Population Sizes Using world 4

Population Size	PTDS	Difference From the Best PTDS
500	94.7	4.5%
1,000	97.2	2.0%
1,500	98.1	1.1%
2,000	99.2 (Best)	0.0%

It can be seen from the above table that the results for the PTDS performance measure are similar to the results from the best fitness value performance measure. In PTDS measure, using 500 individuals in the model did not perform well while using larger population sizes performed better and the total satisfied demand was more than 97%. The best PTDS recorded when 2,000 individuals are used in the model. The 1.1% and 2.0% difference from the best PTDS value are reasonable differences for their associated population sizes. The 1,000 and 1,500 individuals are, therefore, suitable population sizes that can be used in the model for this world layout.

Furthermore, table 6.10 presents the percentages of total number of GA runs whose fitness values are at least 95% of the average fitness value for each population size.

Table 6.10: Percentage of GA Runs of Fitness are $\geqslant$ 95% of the Average Fitness Value for Each Population Size for World 4

Population Size	GA runs That Achieved $\geqslant$ 95% of the Average Fitness value
500	40%
1,000	80%
1,500	80%
2,000	90%

The above table shows the best percentage is for 2,000 individuals followed by 1,000 and 1,500 by 10%. Less than 50% of the fitness values are $\geqslant$ 95% of the average fitness value when 500 individuals are used.

It can be noted from the previous measures that using 500 individuals in the model for world 4 does not produce 'good' results while using 1,500 and 2,000 always produce 'good' results. Using 1,000 individuals in the model is acceptable but not always recommended. Using such population sizes depends on the decision-maker and the nature of the problem .

The four population sizes are further tested with a world consists of nine demand areas. Figure 6.15 shows world 5 GA convergence for different population sizes.

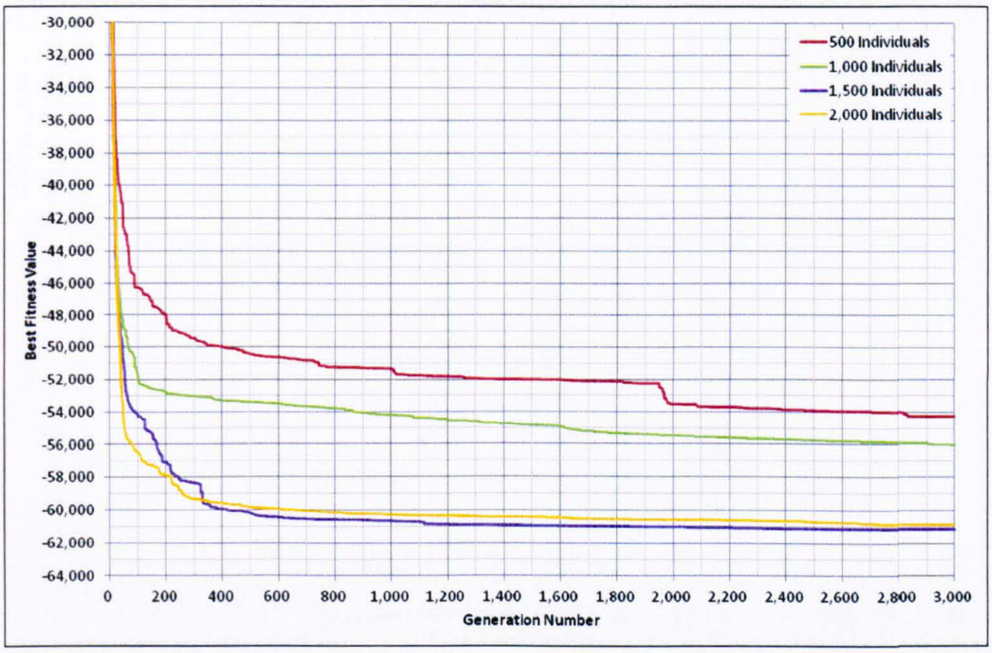


Figure 6.15: World 5 GA Convergence for Different Population Sizes

The above figure shows that the model performed poorly using population sizes of 500 and 1,000. These sizes were not able to add diversity to the population. Larger population sizes should be used for this world. Increasing the population size will increase the accuracy of the GA to find the near-optimal solution. A population size of 1,500 individuals had recorded the best GA convergence among other population sizes. However, the GA convergence using 1,500 and 2,000 individuals are very close where the difference in the best fitness values between them is 240.9, this being less than 0.5%.

In addition, table 6.11 presents the best fitness values for each population size and the difference between these values and the fitness value using the population size of 1,500 individuals which is the lowest fitness value.

Table 6.11: Best Fitness Values and Their Differences from the Lowest Value for Different Population Sizes Using World 5

Population Size	Best Fitness Value	Difference From the Lowest Fitness Value
500	-54304.9	11.2%
1,000	-56034.9	8.4%
1,500	-61169.6 (Lowest)	0.0%
2,000	-60928.7	0.4%

The above table shows that the fitness value using 500 and 1,000 individuals are well below the lowest fitness value while the larger than 1,000 individuals performed better in this measure.

The PTDS and their differences from the best PTDS value for different population sizes using world 5 are presented in table 6.12.

Table 6.12: PTDS and Their Differences From the Best PTDS Value for Different Population Sizes Using world 5

Population Size	PTDS	Difference From the Best PTDS
500	80.3	10.3%
1,000	83.5	6.7%
1,500	88.3	1.3%
2,000	89.5 (Best)	0.0%

It can be seen from the above table that the results for the PTDS performance measure are in agreement with the results from the best fitness value performance measure. In PTDS measure, using 500 and 1,000 individuals in the model did not perform well while using larger population sizes performed better. The best PTDS recorded when 2,000 individuals are used in the model. The 1.3% difference from the best PTDS value is a reasonable difference for 1,500 individuals to be used in the model for this world layout.

Due to the complexity of the world and in order to have a better idea about the spread of fitness values around the average value, the 95% threshold is extended to be 90% of the average fitness value.

Table 6.13 presents the percentages of total number of GA runs whose fitness values are 95% or more and 90% or more of the average fitness value for each population size.

Table 6.13: Percentages of GA Runs of Fitness are 95% and 90% of the Average Fitness Value for Each Population Size for World 5

Population Size	GA Runs of $\geqslant$ 95% of the Average Fitness Value	GA Runs of $\geqslant$ 90% of the Average Fitness Value
500	20%	50%
1,000	40%	50%
1,500	50%	60%
2,000	40%	80%

Most of the percentages of GA runs that with fitness values of 95% or more of the average fitness value were less than 50%. The 95% threshold of the average fitness value did not effectively represent the spread of the fitness values from the average due to the complexity of the world. It was necessary to consider a bigger range for this purpose, which was chosen to be 90%.

By doing so, the number of GA runs increased for each population size. The percentage was doubled from 40% to 80% for 2,000 individuals and became the highest percentage. The percentage increased by 10% for 1,000 and for 1,500 individuals. The worst percentage of 50% was for 500 and 1,000 individuals.

It can be noted from the previous measures that using 500 and 1,000 individuals in the model for world 5 does not produce 'good' results while using 1,500 and 2,000 always produce 'good' results.

Collectively, figure 6.16 shows the relationship between the number of demand areas and the PTDS for different population sizes.

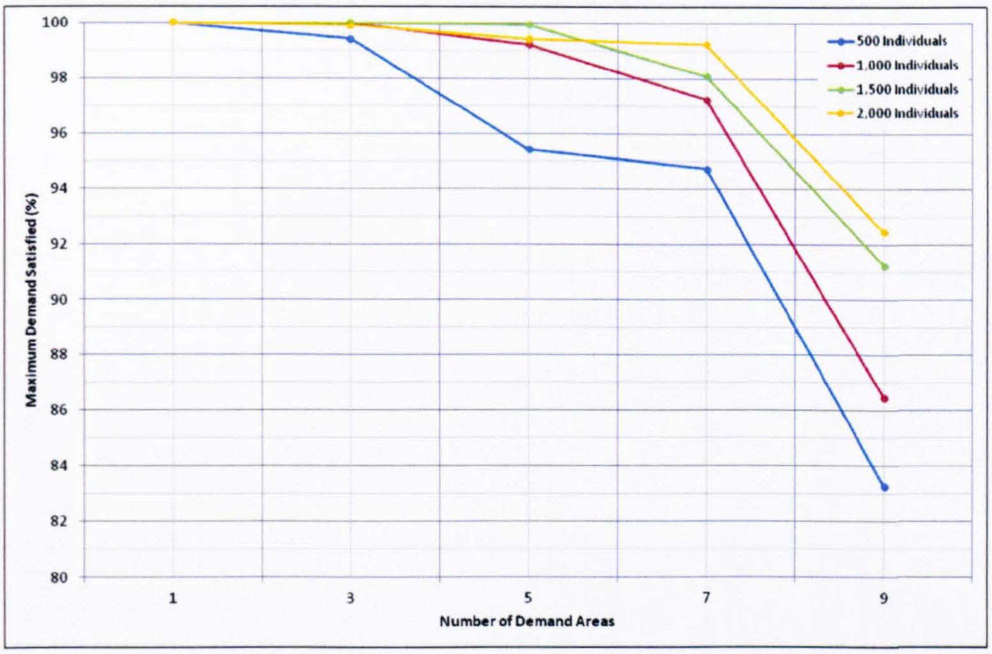


Figure 6.16: The Relationship Between the Number of Demand Areas and the PTDS for Different Population Sizes

It can be noted from the above figure that 500 individuals performed well for the world of one demand and the world consists of three demand areas. In addition, this population size can also be used for a world of five demand areas, although not recommended. Populations as small as 500 individuals are unsuitable for worlds consists of more than five demand areas because this size did not improve the probability of finding better solutions.

Larger population sizes as big as 1,000 and 1,500 individuals performed well for worlds up to seven demand areas. If more than seven demand areas exist in a world, larger population sizes are required to increase the performance and efficiency of the GA model by exploring the solution space and consistently find 'good' solutions.

The general conclusion is that increasing population size improves the probability of finding better solutions at an increased computational time.

It is found that the best population size is both problem dependent and related to the individual size. The effect of different population sizes on the same number of demand areas might change if the location of these areas changes within the boundaries of the world. It may also have an effect if the location of demand areas changes with respect to another feature in the world. For example, the effect of different population sizes might change if the location of the water source changes.

6.2.2.3 Number of Nodes

In water networks, the task of finding a path from source node to destination node is known as "routing". For a given network in this work, it may consists of more than one path. The chromosome is encoded as a list of nodes from source to destination node (demand areas).

The number of nodes determines the length of the chromosome which is fixed during the GA evolution process. The selection of the number of nodes depends on the size of the problem. This means that the real computation time increases as the network size becomes larger. Problem size may depend on the number, location, and geometrical shape of features within the world.

In testing the number of nodes, crossover type 2 and mutation type 1 are used as GA operators. The other GA parameters are presented in table 6.14.

Table 6.14: Selected GA Parameters for Number of Nodes Experiments

Population	Mutation Rates		
Size	Individuals	Connectivity & Head	Node Coordinates
1,000	10%	5%	15%

The five test worlds are used with different number of nodes. The tests are carried out for 10, 15, and 20 nodes. The effect of the amount of the total demand is also considered in this section. The testing regime for the number of nodes is shown in figure 6.17.

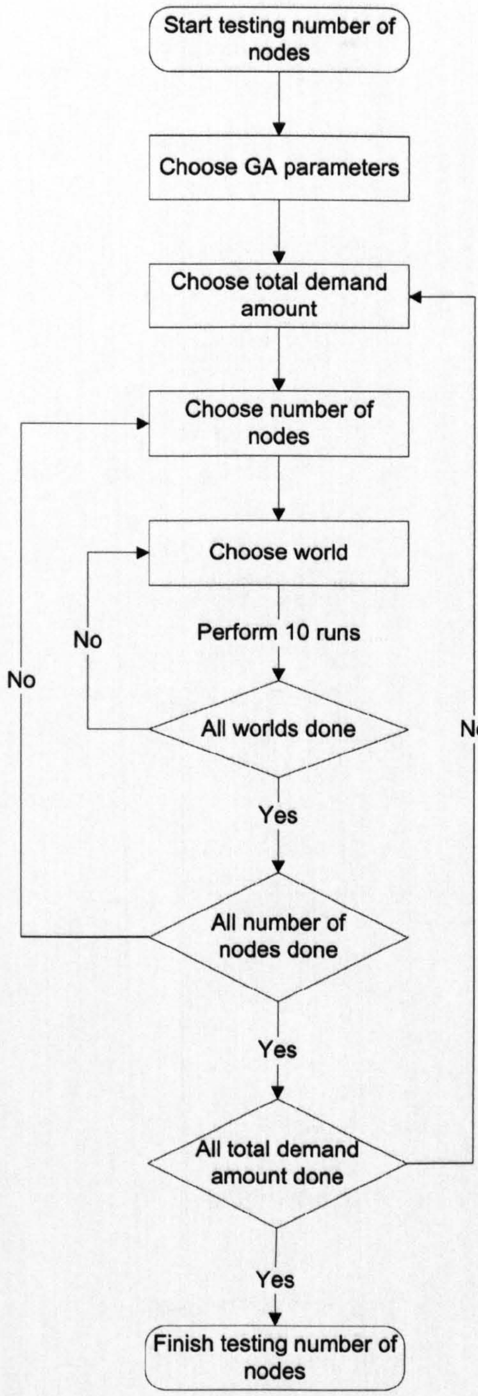


Figure 6.17: Testing Regime for Number of Nodes

The testing in this section is mainly divided into two parts according to the total demand amount for each world. Similar to experimentations in previous sections, the first part is based on a fixed demand amount for each demand area which uses an amount of 10,000 per each demand area. In this instance, each world will have a different total demand amount. For example, total demand amount for a world of three demand areas (e.g. world 2) is 30,000 and for a world of seven (e.g. world 4) demand areas is 70,000.

The second part of the experimentations will be implemented using a fixed demand amount for each world. The tests in this part use a total demand of 90,000 per each world. This amount is divided among the number of demand areas in the entire world. For instance, the demand amount per each demand area in world 2 will be 30,000 (90,000 / 3 demand areas) while the demand amount per each demand area in world 4 will be 12,857 (90,000 / 7 demand areas).

The primary purpose of a water distribution network is to deliver water to the consumer in the required quantity and at sufficient pressure. The results of these tests will be analysed and compared with respect to the PTDS.

6.2.2.3.1 The demand amount is fixed for each demand area

As stated earlier, the demand amount for each demand area is chosen to be 10,000. In this part, 20, 15, and 10 nodes will be tested with different worlds.

Table 6.15 presents the PTDS for the five worlds using 20 nodes.

Table 6.15: The PTDS Using 20 Nodes and 10,000 / Demand Area

World Number	1	2	3	4	5
PTDS	9.3	14.0	55.5	59.5	50.8

From the above table, it is clear that using 20 nodes in the model resulted in a very poor PTDS values. None of these percentages is close to the quality measure of 95%, and these values are well below the optimal solution.

The GA exhausted its search on unnecessary and unwanted chromosome genes. Long chromosome strings slow down the genetic features of the algorithm by increasing evolution times. Current mutation rates may be less than the required rates to increase the percentage of satisfied demand amount. Pipe size, head along the pipe, and node coordinates are chromosome decision variables. The mutation may be able to change the percentage of total satisfied demand amount if one or all of these variables are changed. In this test, none of these components is changed to improve the solution and derive the GA convergence to better solution.

Table 6.16 presents the PTDS for the five worlds using 15 nodes.

Table 6.16: The PTDS Using 15 Nodes and 10,000 / Demand Area

World Number	1	2	3	4	5
PTDS	43.8	73.3	86.4	78.0	74.6

The PTDS values in the above table show that the PTDS using 15 nodes increases the PTDS drastically compared to using 20 nodes. However, the PTDS values with 15 nodes are not 'good' enough to be considered. This is clear as the best PTDS is 86.4% for world 3 whilst the PTDS for other worlds are well below this value. The achieved PTDS using world 3 and 15 nodes is increased by approximately 31% using the same world but with 20 nodes. The minimum and maximum PTDS increments using 15 nodes are 18% for world 4 and 60% for world 2 respectively.

In this test, the PTDS for world 1 suppose to be the best one. However, the PTDS of this world, which the simplest world, was the lowest one among the other values when 20 and 15 nodes are used.

Although the number of nodes are decreased by five nodes, the GA model was not able to achieve optimal or near-optimal solutions. The current mutation rates and other GA parameters could not improve the GA performance and the GA search is exhausted in undesired search space.

A similar test is carried out with 10 nodes. Figure 6.18 shows the convergence graph of best PTDS values.

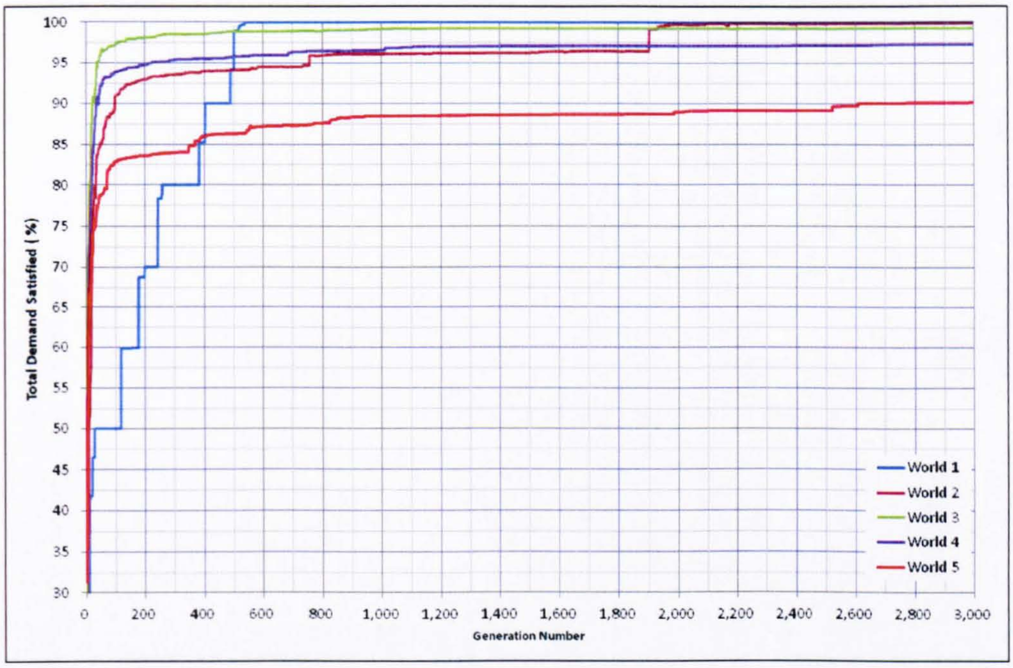


Figure 6.18: The PTDS Convergence Graph Using 10 Nodes and 10,000 / Demand Area

The above graph shows that the total demand satisfied was very rapid, where more than 70% of the total demand is satisfied in the first 200 generations and more than 85% is satisfied by 400 generations.

The maximum percentages of satisfied demand for this test are presented in table 6.17.

Table 6.17: The PTDS Using 10 Nodes and 10,000 / Demand Area

World Number	1	2	3	4	5
PTDS	100.0	99.9	99.4	97.3	90.1

The demand of one demand world is fully satisfied, and the optimal water supply is achieved. The 100% percentage decreases by a marginal amount of 0.1% for world 2 and 0.6% for world 3. For more complex worlds, this gap increases to be 2.7% for world 4 and 9.9% for world 5. Using 10 nodes with other GA parameters in the model for world layouts consisting of up to seven demand areas achieved more than 95% of the total demand. Less than 95% but more than 90% is satisfied using world 5 because the demand layout is more complex than other world layouts. This percentage can be accepted due to the fact that, in addition to the world complexity, it is hard for the optimisation algorithm to optimise all network components under current model configurations.

The percentages of total number of GA runs whose PTDS values are at least 95% of the average PTDS value for world layouts consists of up to seven demand areas is 100%. This percentage decreases drastically to 40% for world 5 which consists of nine demand areas, although satisfying more than 90% of the total demand. Using 10 nodes in the model achieved optimal or near-optimal solutions for other types of worlds. This is because the GA search focused on useful parts of the search space which improved the water supply.

Using 10 nodes in the model achieved significantly better results than those of 20 and 15 nodes (see figure 6.19).

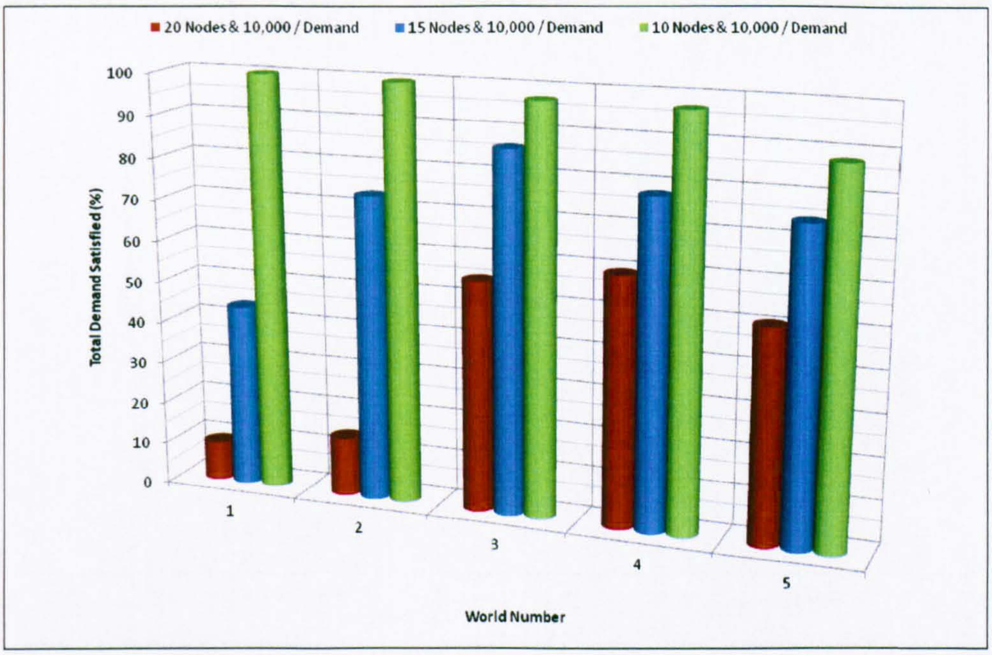


Figure 6.19: The PTDS Using 10, 15 & 20 Nodes With 10,000 / Demand Area

6.2.2.3.2 The demand amount is fixed for each world

A different theme of experiments is carried out with respect to the total demand amount. In these tests, this amount is fixed for each world regardless of the number of demand areas. In these experiments, 20, 15, and 10 nodes will be tested. The total demand amount for each world is selected to be 90,000 for each world.

Table 6.18 presents the PTDS for the five worlds using 20 nodes.

Table 6.18: The PTDS Using 20 Nodes and 90,000 / World

World Number	1	2	3	4	5
PTDS	17.6	41.2	68.0	49.4	50.8*

* This value is borrowed from the GA runs using 20 nodes and 10,000 / demand area because world 5 has nine demand areas in its layout in which the total demand equals to 90,000.

Similar to GA tests when 20 nodes with 10,000 / demand are used, the PTDS results in the above table are disappointing. Although they performed better for world 1, 2 and 3, it is clear that using 20 nodes in the model does not achieve 'good' results. In addition, the above values are still well below the 95% of the total demand due to the same reasons stated for 20 nodes with 10,000 / demand.

The next test uses the same model parameters with 15 nodes. The results of PTDS values for the five worlds are presented in table 6.19.

Table 6.19: The PTDS Using 15 Nodes and 90,000 / World

World Number	1	2	3	4	5
PTDS	68.7	62.9	86.5	81.5	74.6*

* This value is borrowed from the GA runs using 15 nodes and 10,000 / demand area because world 5 consists of nine demand areas in which the total demand equals to 90,000.

The PTDS values in the above table show that the PTDS using 15 nodes are improved drastically compared to those of 20 nodes. However, similar to tests using 20 nodes with 90,000 / world, the PTDS values with 15 nodes are not good enough to be considered. This is because 86.5% is the best value which is less than the quality measure of 95%.
The minimum and maximum PTDS increments using 15 nodes are 18.5% for world 3 and 51.1% for world 1 respectively. Interestingly, the PTDS in world 3 has a similar percentage to the test when 15 nodes and 10,000 / demand areas are used.

The last experiment in this section is for 10 nodes with 90,000 / world. Figure 6.20 shows the convergence graph for the best PTDS values.

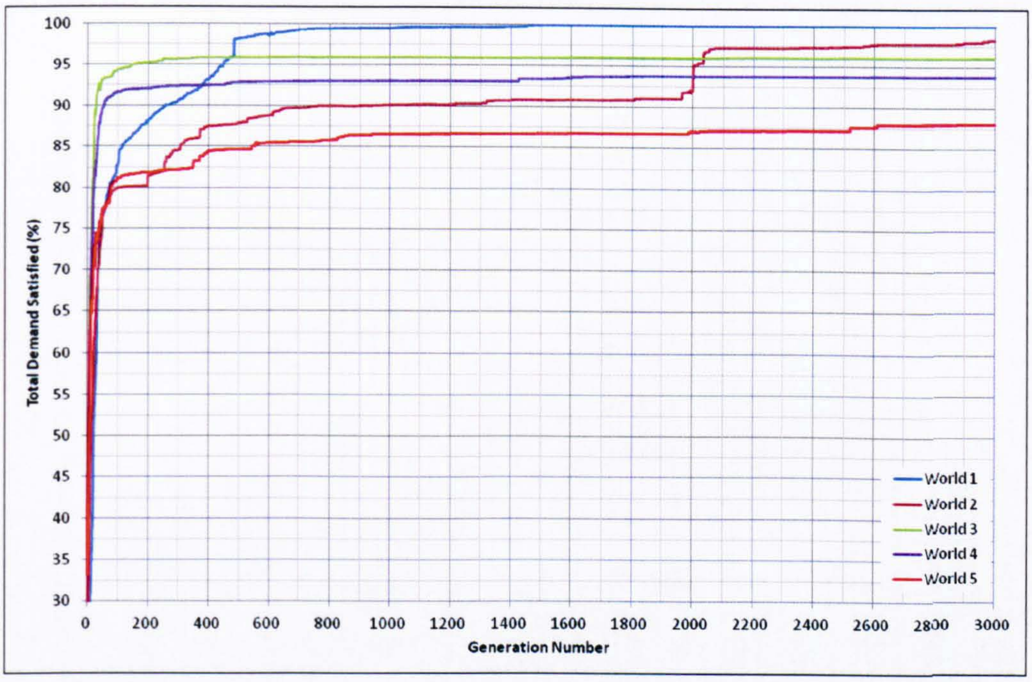


Figure 6.20: The PTDS Convergence Graph Using 10 Nodes and 90,000 / World

Similar to tests with 10 nodes and 10,000 / demand area, the above graph shows that the total demand satisfied was very rapid, where more than 70% of the total demand is satisfied in the first 200 generations. However, the PTDS convergence became slower in later generations.

The maximum PTDS values for this test are presented in table 6.20.

Table 6.20: The PTDS Using 10 Nodes and 90,000 / World

World Number	1	2	3	4	5
PTDS	100	98.2	96.1	93.8	90.1*

* As with 90,000 / world in the previous tests, this value is borrowed from the GA runs using 10 nodes and 10,000 / demand.

The optimal water demand is satisfied for world 1. The PTDS for other worlds decreased with increasing the number of demand areas. The 100% percentage decreased by a small amount of 1.8% for world 2 and 3.9% for world 3. This difference increases for more complex worlds. Less than 95% but more than 90% is satisfied with world 4 and world 5 because of the complexity of these worlds compared to other world layouts. All PTDS were more than 90%, which indicates a 'good' performance of the current model configurations.

The spread of the PTDS for each set around the average PTDS are also examined. Table 6.21 presents the percentages of total number of GA runs whose PTDS values are 95% or more of the average PTDS value for each world.

Table 6.21: GA runs With PTDS Values $\geqslant$ 95% of the Average PTDS Using 10 Nodes and 90,000 / World

World Number	1	2	3	4	5
GA Runs Achieved $\geqslant$ 95% of the Average PTDS	100%	90%	80%	70%	40%

The spread of the PTDS is 'good' for all worlds except the world 5. However, better PTDS values were achieved using 10 nodes and 10,000 / demand area. Using 10 nodes with a total demand of 90,000 performed better than using 15 and 20 nodes for all the proposed world layouts.

The performance of different tests with 90,000 total demand can be seen in figure 6.21.

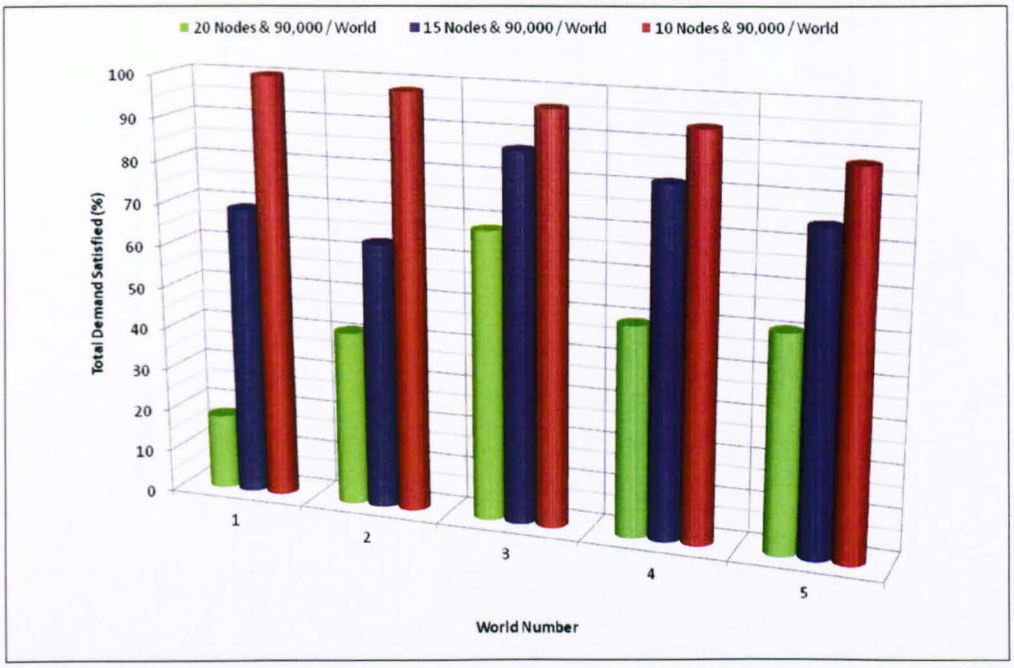


Figure 6.21: The PTDS Using 10, 15, & 20 Nodes With 90,000 / World

All of these tests can be compared to each other to conclude the best number of nodes. Figure 6.22 shows the overall result of the PTDS for all tests.

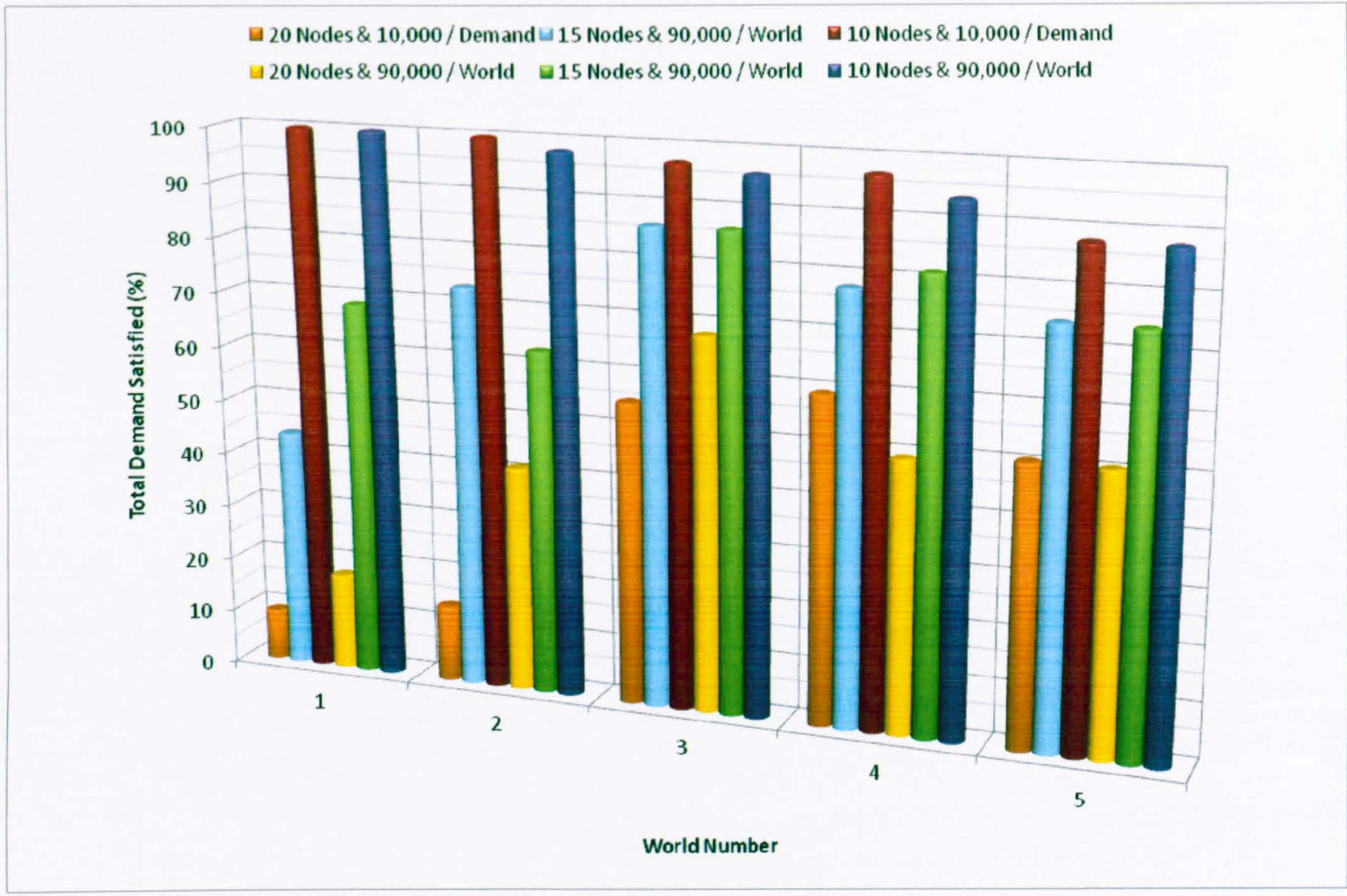


Figure 6.22: The PTDS Using 10, 15, & 20 Nodes With Different Demand Amounts / Demand Area

It can be seen from the above figure that the best performance is achieved using 10 nodes with 10,000 / demand area.

It is clear from the above figure that the worst PTDS are recorded when 20 nodes are used. Better results are produced using 15 nodes and the best are recorded for tests where 10 nodes are used. This implies that lesser number nodes for a specific number of demand areas produce better results.

In addition, the model could satisfy higher PTDS for a relatively low demand amounts / demand area. This is because the model is able to optimise the network to carry sufficient water amounts to demand areas with acceptable cost solutions.

6.2.2.4 Mutation Rate

Tate and Smith (1993) argued that the optimal mutation rate depends on the choice of encoding, and that problems with non-binary encodings may benefit from mutation rates higher than those generally used with binary encodings.

For the problems associated with poor expected gene coverages, it is required to adjust some of the GA parameters such as mutation probability or the population size. Increasing the population size may slow the overall population improvement rate. The alternative is therefore to increase the relative mutation rate.

It is part of the traditional genetic algorithms that low mutation rates lead to efficient search of the solution space, while high mutation rates result in diffusion of search effort. Each position of the chromosome in the population undergoes a random change according to a probability defined by a mutation rate (the mutation probability (p_m)).

In this work, three different types of mutation rates are applied to the problem:

1. Mutation rate applied to individuals. This rate is the percentage of randomly selected individuals for mutation.
2. Mutation rate applied to the three-dimensional coordinates of the node. The mutation permutes (x, y, z) coordinates of a number of randomly selected nodes in an individual.
3. Mutation rate applied to connectivity and head components of the chromosome. This rate represents the number of genes to be permuted in connectivity and head components of the chromosome.

In this section, testing the mutation rate will be presented in two parts. In the first part, mutation rates applied only to population and connectivity & head matrices will be tested.

The testing regime for mutation rates applied to population and connectivity & head is presented as a flowchart in figure 6.23.

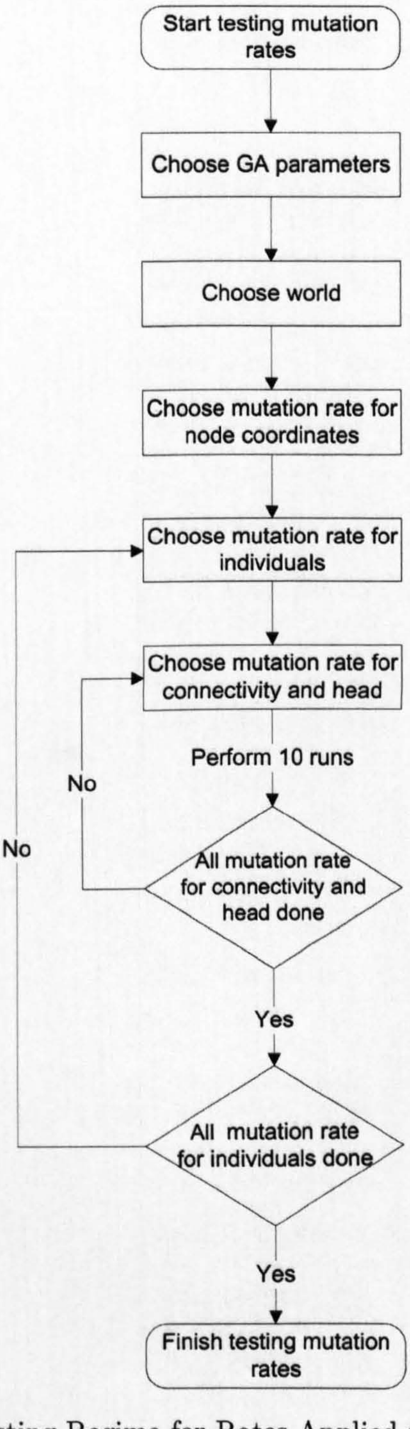


Figure 6.23: Testing Regime for Rates Applied to Population and Connectivity & Head

Several mutation rates are tested in this section. During the evolutionary process, it is assumed that the domain of variables is fixed and the probability of crossover (p_c) is constant (= 100%). The applied mutation rates are fixed for all generations and for all individuals in each test. Crossover type 2 and mutation type 1 are used for the breeding process.

Mutation rate also depends on the number of nodes. The number of nodes sets the chromosome length which is selected to be 10 nodes. The population size is chosen to be 1,000 individuals.

A large number of tests are required to test the selected mutation rates on all worlds and it is clearly not possible for all results to be shown in this chapter. One world (world 3) among the five worlds is selected to conduct all the testing in this section.

It is hypothesised that the knowledge obtained during the evolution could be utilised to update the mutation rates of node coordinates. The mutation rate of node coordinates is fixed at 15% which allows an average of two node coordinates, except the source node, to be permuted. If a node is selected for mutation, the (x, y, z) node coordinates will be permuted.

Due to the large number of possible mutation rates for each chromosome component, only selected mutation rates are tested. As shown at the testing regime, the mutation rate applied to the node coordinates will be fixed first. This will be tested later in this section. The experiments in this section will first use different mutation rates for population and for connectivity and head components of the chromosome.
Since the developed computer program consists of a number of text boxes, the users are able to change mutation rates and other GA parameters according to their preferences.

When a chromosome is chosen for mutation, a random change is made to the values of some locations in the chromosome. Each change to the chromosome values is made with a random uniform distribution. The distribution of the generated random numbers using Visual Basic programming language is uniform; the likelihood of returning each number is equal (Microsoft Corporation, 2011c).

A very small mutation rate may lead to genetic drift which can be reduced by increasing the mutation rate. However, a mutation rate that is too high may lead to loss of good solutions and the GA search becomes effectively random. It is therefore necessary to select suitable mutation rates to aid the GA convergence. It was decided to use mutation rates presented in table 6.22.

Table 6.22: Selected Mutation Rates in Tests With 15% Rate of Node Positions

Combination Number	Population (%)	Connectivity & Head (%)
1	5	1
2	5	5
3	5	10
4	10	1
5	10	5
6	10	10
7	15	1
8	15	5
9	15	10

The mutation rates applied to population are chosen to be 5%, 10%, and 15%. Therefore, for a population size of 1,000 individuals, the average number of individuals to be randomly selected for mutation is 50, 100, and 150 respectively.

The mutation rates of the connectivity and head components of the chromosome are chosen to be 1%, 5%, and 10%. The number of permuted genes depends on the number of nodes. As the connectivity and head components represent a square matrix in the chromosome, mutation rates are applied to these matrices. For 10 nodes, the dimensions of each matrix in the chromosome is 10×10 and the number of genes is 100. According to the selected mutation rates, an average of 1 gene, 5 genes, or 10 genes can be selected for mutation.

Figure 6.24 shows the best fitness value achieved using different combinations of the mutation rates.

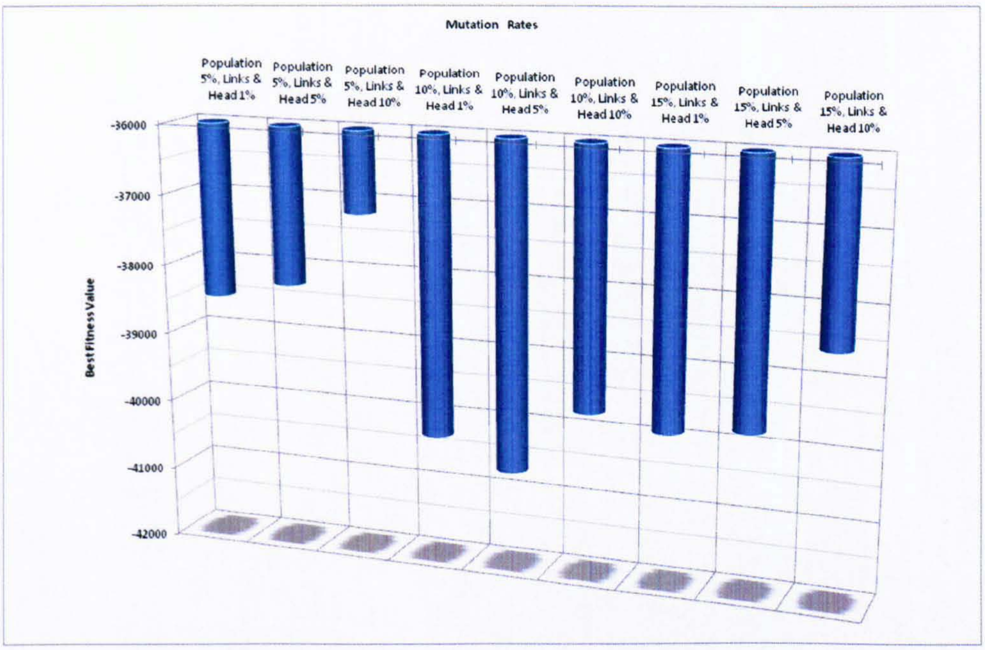


Figure 6.24: Best Fitness Values for Different Mutation Rates Used With World 3

Figure 6.24 showed that the best fitness value is achieved when mutation rate for individuals is 10% and 5% for connectivity and head components. These rates produced the best answers when used with a mutation rate of 15% applied to node positions. In addition, it is possible to achieve 'good' results when mutation rate for individuals is 10% and for connectivity and head is less than 5%.

The above figure also showed that mutation rates of 10% of connectivity and head did not generally perform well. It requires a lower mutation rate to operate effectively. Less than 10% should be used to reduce the randomness in the search for pipe sizes and the required head.

The mutation rate applied to population of more than 10% proved to produce 'good' results and was poor with 5% rate. The GA performed better when more individuals involved in the mutation process.

The spread of the fitness value using the above mutation rates is also examined. It was decided to choose the combinations that achieved $\geqslant 95\%$ of the best fitness value. These are shown in table 6.23.

Table 6.23: Selected Mutation Rates and Their Fitness Values

Combination Name	Best Fitness Value	Difference From The Best
Population 10%, Links & Head 5%	-40652.1 (Best)	0.0%
Population 10%, Links & Head 1%	-40253.2	1.0%
Population 15%, Links & Head 1%	-39880.8	1.9%
Population 15%, Links & Head 5%	-39766.0	2.2%
Population 10%, Links & Head 10%	-39702.6	2.3%

Table 6.24 presents the percentage of GA runs for selected mutation combinations that achieved 95% or more of the average fitness value.

Table 6.24: Selected Mutation Rates and GA Runs With Fitness $\geqslant$ 95% of the Average Fitness Value

Combination Name	GA Runs With Fitness $\geqslant$ 95% of the Average Fitness Value
Population 10%, Links & Head 5%	100%
Population 10%, Links & Head 1%	80%
Population 15%, Links & Head 1%	100%
Population 15%, Links & Head 5%	70%
Population 10%, Links & Head 10%	80%

Generally, all combinations in the above table performed well as all percentages were more than 50%. Since the percentage is 100% for mutation rates of population 10% and links & head 5% and the model achieved the best fitness, these rates will be used for testing different mutation rates for node coordinates.

Further experimentations were devised to test different mutation rates for node coordinates. The testing regime is presented as a flowchart in figure 6.25.

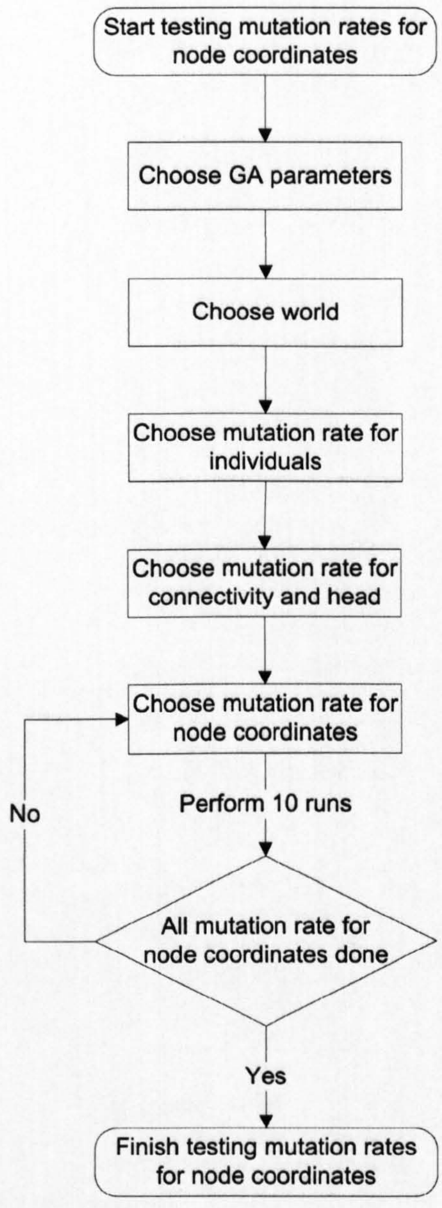


Figure 6.25: Testing Regime for Testing Rates Applied to Node Positions

Same GA parameters used in previous tests in this section will be used with these tests. Due to the large number of possible test scenarios and mutation rates, 10% and 20% of the nodes will be tested with world 3. The mutation rates are presented in table 6.25

Table 6.25: Mutation Rates for Testing Rates Applied to Node Positions

Combination Number	Population	Connectivity & Head	Node Positions
1	10%	5%	10%
2	10%	5%	15%
3	10%	5%	20%

Combination number 2 already exists and borrowed from previous experimentation for comparison purposes. The mutation rates for individuals and for connectivity and head are chosen due to their good performance in previous tests.

The mutation rate applied to population are chosen to be 10%. Therefore, for a population size of 1,000 individuals, the average number of individuals to be randomly selected for mutation is 100 individuals.
The mutation rate applied to connectivity and head matrices are chosen to be 5%. For 10 nodes, The average number of genes to be randomly selected for mutation is 5 genes.

For 10 nodes, the average number of node coordinates that will be randomly selected for mutation will be two nodes for a mutation rate of 20% and 15% and one node for a mutation rate of 10%. All node (x, y, z) coordinates will be permuted if selected for mutation. Figure 6.26 shows the GA convergence for the three combinations.

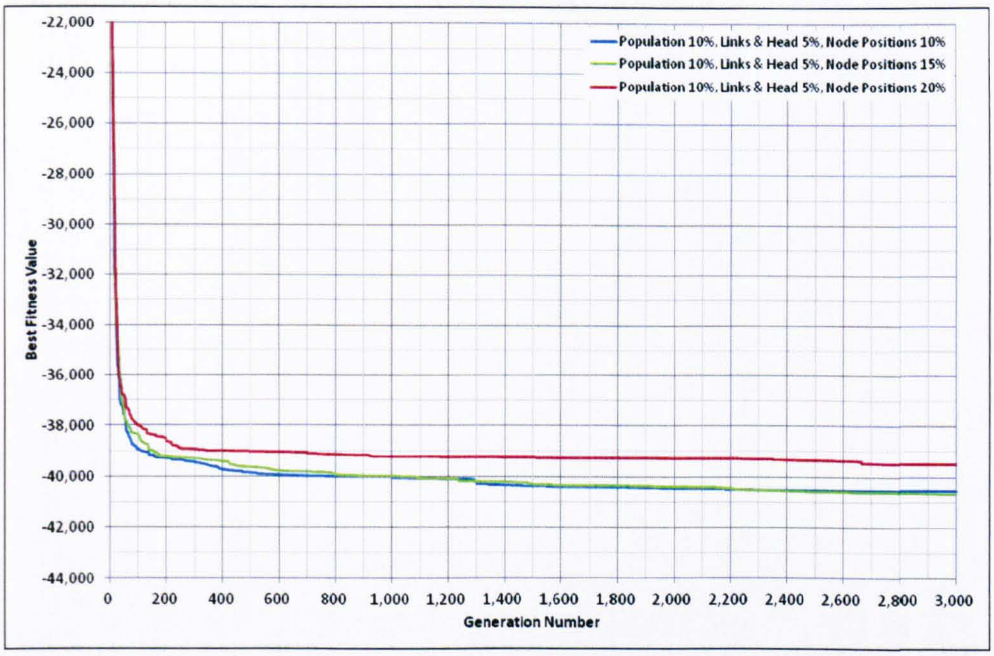


Figure 6.26: GA Convergence With Different Mutation Rates Used With World 3

From the above figure, the best GA performance is recorded when a mutation rate of equal or less than 15% is used for node coordinates with 10% for individuals and 5% for connectivity and head components of the chromosome. However, all solutions are very close in magnitude.

The spread of the best fitness value around 95% of the average fitness for each combination was equal or more than 90%. In addition, the achieved PTDS was more than 99% for all combinations.

It can be concluded from this section that the best suited mutation rates for worlds as big as world 3 using 10 nodes and 1,000 individuals is as follows: 10% for population, 5% for connectivity and head, and up to 20% for node coordinates. These mutation rates will be used for the rest of experimentation in this thesis.

It can be also concluded that running GA in this model without proper mutation rates often results in a population that quickly stagnates, converging to a local optimum owing to the lack of genetic diversity.

6.2.2.5 Adjacency Mutation Operator

During the progress of the optimisation, the freedom of the mutation to vary the values is constrained for head and node (x, y) coordinates. The mutation operator permutes individual genes into adjacent values, rather than the traditional GA approach of randomising gene values within the search space. This operator changes each chromosome in small increments rather than drastically changing a single chromosome.

The purpose of this mutation is to provide more insurance against the loss of genetic information and therefore, maintain diversity within the population.

To analyse the relative performance of adjacency operators that "nudge" gene values to adjacent values, several experiments are carried out for the maximum head and node (x, y) coordinate movements. This operator adds or subtracts some adjacent value ($\triangle$), called "Maximum Allowed Movement", to the value of the permuted gene.

As with previous experiments in this chapter, model parameters need to be selected. The experiments use 10 nodes with world 3 and crossover type 2 and mutation type 1 are used as GA operators. The other GA parameters are presented in table 6.26.

Table 6.26: Selected GA Parameters for Testing Adjacency Mutation Operator

Population	Mutation Rates		
Size	Individuals	Connectivity & Head	Node Coordinates
1,000	10%	5%	15%

6.2.2.5.1 Maximum Allowed Head Movement

The adjacency mutation proposed here allows for adjustment of head values upward or downward. The mutation replaces the initial head value h_{ij} with a new value h'_{ij} in the head matrix.

Head values in the initial population (at generation *zero*) ranges between 0 and 25 units. This range would be used for mutation if the adjacency operator is not used. The freedom of the mutation to vary the values between 0 and 25, that may or may not produce an adjacent value, is constrained to smaller ranges.

To test the effect of maximum allowed head movement on the model, a testing regime is proposed as presented in figure 6.27.

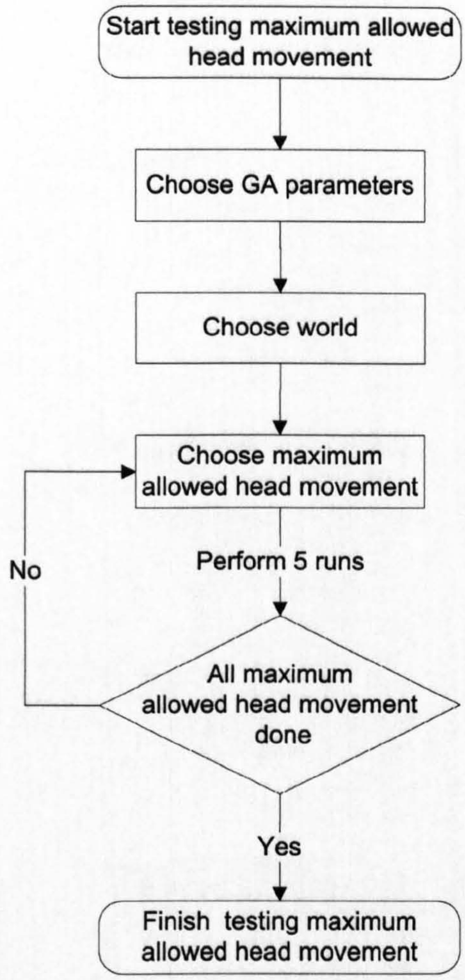


Figure 6.27: Testing Regime for Maximum Allowed Head Movement

The experiments tested a maximum allowed head movement of 3, 6, 9, and 12 units using world 3. The full range of 25 units is also used to analyse and compare the difference between using the maximum allowed head movement and the traditional mutation which uses the range of 0 to 25 units. The GA convergence of this test is shown in figure 6.28.

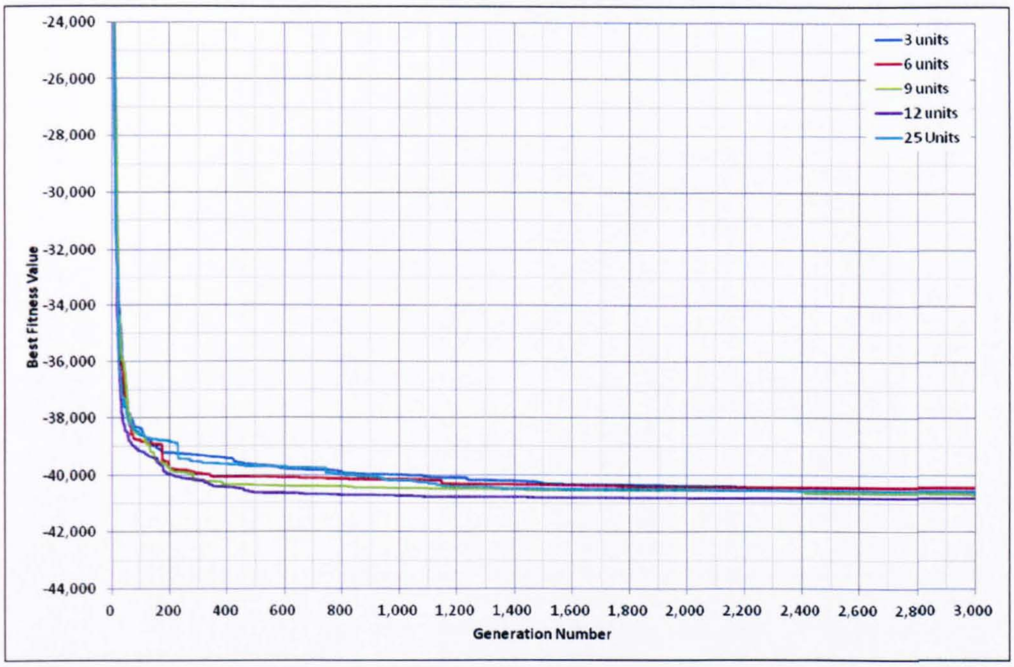


Figure 6.28: GA Convergence for Different Values of Maximum Allowed Head Movement

It can be seen from the above figure that maximum allowed head movement does not make significant differences to the GA convergence and the convergence speed is close for all tests. The difference between the maximum bes fitness and the best minimum fitness among these tests is less than 1%.

GA performance is not affected using the adjacency mutation operator for head values. Similar behaviour is recorded when head values are randomised within the search space.

6.2.2.5.2 Maximum Allowed Node (x, y) Coordinates Movement

Similar to the maximum allowed head movement, the adjacency mutation proposed here allows for movement of plane node coordinates upward or downward. The mutation replaces the initial value of x and y coordinates (x_i and y_i) with new values (x'_i and y'_i).

The maximum values that the plane node coordinates would have are constrained by the world boundaries. The plane boundaries of the world are 500×500 units. The range of 0 to 500 units for plane node coordinates are allowed in the initial population.

To test the effect of maximum allowed Node (x, y) Coordinates movement on the model, a testing regime is proposed. This testing regime is presented in figure 6.27.

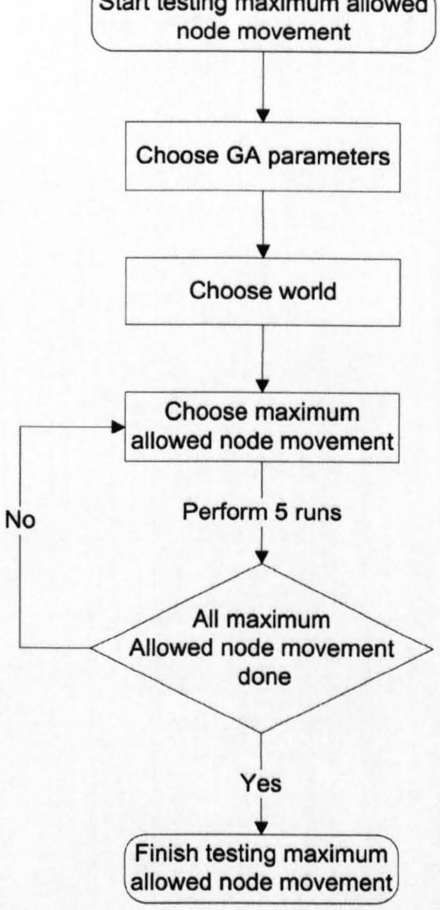


Figure 6.29: Testing Regime for Maximum Allowed Node Movement

The experiments used a maximum allowed (x, y) movement of 5, 10, 20, 30, 40, 50 and 500 units using world 3. Similar to head adjacency operator, the full range, 500 units, is allowed in order to analyse and compare the difference between using the maximum allowed head movement and the traditional mutation which uses the range of 0 to 500 units. Figure 6.30 graphs the GA convergence of this test.

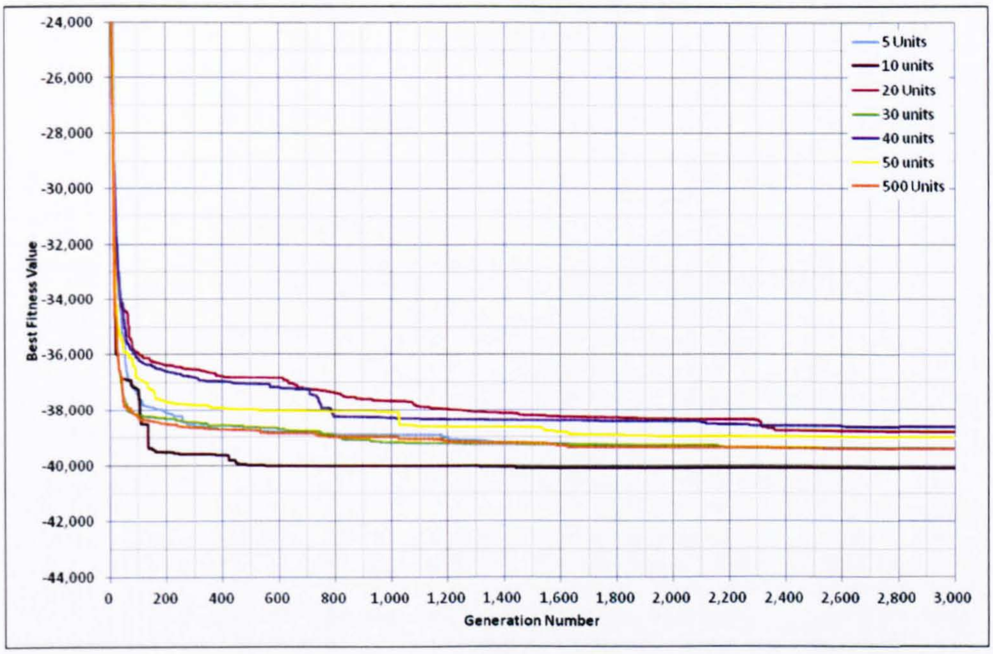


Figure 6.30: GA Convergence for Different Values of Maximum Allowed (x, y) Movement

The test results in this experiment are generally similar to those obtained obtained in section 6.2.2.5.1. All tested values of maximum allowed movement produced close performance in terms of GA convergence. The speed of the convergence showed some delays for 20 and 40 units. The difference between the maximum best fitness and the minimum fitness value is less than 4%. Closer values and convergence speed can be achieved by running more tests for each instance.

GA performance is not affected by using adjacency mutation operator for node plane coordinates. Similar behaviour is recorded when coordinate values are randomised within the boundaries of the world.

6.3 A Special World Layout

A special world layout is selected to demonstrate that the distribution network nodes can avoid high cost passages. The 'land access cost' is explained in section 5.4.1.3 and will be discussed in more details in chapter 7.

The aim of using this world is also to demonstrate that a WDN may have more than one path from the source node to demand areas.

The layout of the proposed world consists of one water source, one demand area, and two high cost passages (e.g. mountains). The world is shown in figure 6.31.

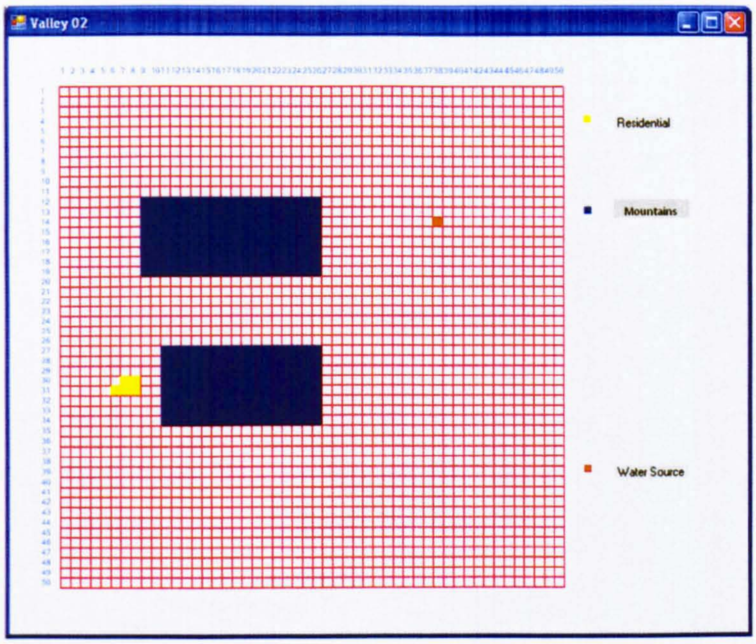


Figure 6.31: A Special World Layout

The design of this 'manufactured' world is similar to those defined in section 6.2.1. This world depicted in graphical form as a square of 500×500 units. This square is subdivided into 10×10 grid size which results into a 50 cells (i.e. grid). In addition, this world is designed to allow several paths from the water source to demand areas.

The relative values of access cost of mountains and other world components should not be marginal. The difference between these values should be big enough so that the algorithm is capable to easily recognise the phenomenon of different parts of the world.

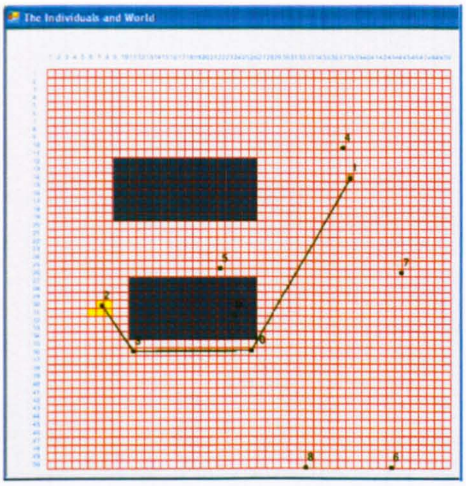
The assigned cost to mountains is much higher than the other parts of the world. The cost of crossing one unit length of the mountains is 100,000 units while the cost of crossing other parts of the world is one unit.

The best GA parameters obtained from the experimentations in this chapter will be used to in this test. Crossover type 2 and mutation type 1 are used in this test. The other GA and model parameters used with this world are presented in table 6.27.

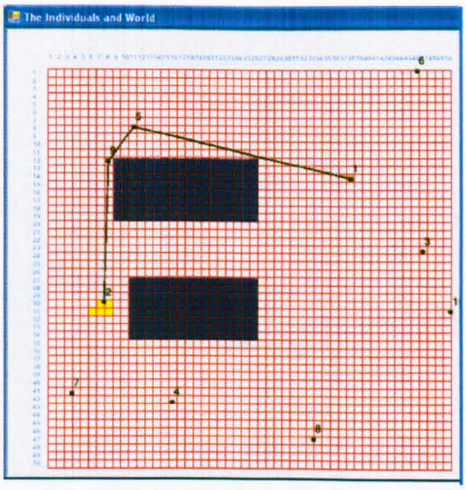
Table 6.27: GA and Model Parameters Used With the Special World Layout

Parameter	Value
Population Size	1,000
Number of Nodes	10
Number of Generations	3,000
Mutation Rate (Individuals)	10%
Mutation Rate (Links & Head)	5%
Mutation Rate (Node Positions)	15%
Total Demand Amount	10,000
Maximum Head Movement	5 units
Maximum Node Movement	100 units

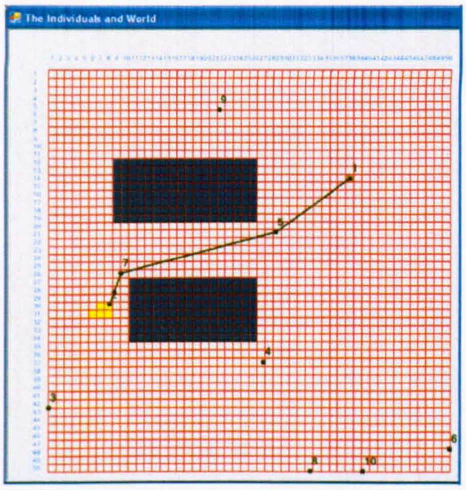
Several model runs using this world and the above parameters were implemented. Three different network layouts are produced as shown in figure 6.32.



(a) Typical Network Layout 1



(b) Typical Network Layout 2



(c) Typical Network Layout 3

Figure 6.32: Different Network Layouts

The above figures shows that networks are able to divert from expensive passages and find a cheaper path to supply water to demand areas. Other network layouts may be formed depending on the node positions.

The behaviour of the model using this world is similar to the model behaviour using the worlds defined in section 6.2.1. The average convergence speed is similar to the one of world 1. In addition, the full demand is satisfied within the first 30 generations.

6.4 Chapter Summary

This chapter presented a series of experiments and their associated results. These experiments were carried out on a set of five test worlds. These worlds have different layouts with different numbers of demand areas.

Due to the very large number of possible mutation and crossover combinations, various combinations were tested for two selected worlds. One of these represents a simple world layout (world 2) and the other represents a more complex world layout (world 4). Some combinations of the two operators seem to consistently perform significantly better than others.

The chapter presented tests on differently sized of population, because previous studies show that there is a crucial relationship between solution quality and population size. The population sizes required to solve the problems might vary from a few hundred to a few thousand. Results demonstrated that the population size required to solve the problem need to be increased with the increasing difficulty of the problem.

Various numbers of nodes were used for each world to find the suitable number of nodes for different world layouts.

Many variations on the mutation operator were proposed. Mutation rate is associated with population size and the number of nodes. The mutation rate for population is independent from the mutation rate of different chromosome components. Different combinations of these rates are tested.

The adjacency mutation operator was also tested to analyse the effect of freedom of mutation. This is tested individually on head component and plane coordinates of the node by setting a maximum allowable movement for each component.

A special world layout is introduced to demonstrate how a distribution network can be diverted around expensive passages.

The next chapter demonstrates the possibility of applying the current GA and other model parameters to a world layout which represent some real-world features.

Chapter 7

Application of the Model - An Example

7.1 An Introduction to the Chapter

In order to demonstrate the application of the formulation to a practical problem, a hypothetical town has been created which contains a representation of some necessary real-world features. The chapter starts with a description of the world followed by modelling of the town. This includes a discussion to the effect of the raster grid size on the features to be modelled.

Some access cost figures are assigned to the grid structure where different access cost figures are applied to each of the cells within the world. This means that the world features should be located and defined within the world with their relative costs. These are stored in a text file which is then read by the formulation and used for model calculations.

A demonstration of the current GA formulation on this world is also presented. The addition of a demand area to the current world layout is described and tests are carried out to show how the model will connect the new developed areas to the existing water network.

7.2 World Description

It is first necessary to build the geographic representation of the world. The world in this context is a city which represent a physical composition and the interaction between man-made and natural elements. Thus, a city may have public open spaces (e.g. parks), city centres with intimate and mixed uses, industrial, commercial and business parks with landscape greenbelts, wide roads and controlled traffic access points. In addition, the world may have preserved agriculture areas and natural features (natural topography, wetlands, floodplains, and water).

The geographic representation of the world will consist of some of these features and will be simple enough to demonstrate the work. However, this world has far more detail than the worlds defined in the previous chapter including a rail that effectively divides the world effectively into two halves. Figure 7.1 shows a realistic scale plan of a fictional town that shall be called "Kawar" which will be used in this chapter.

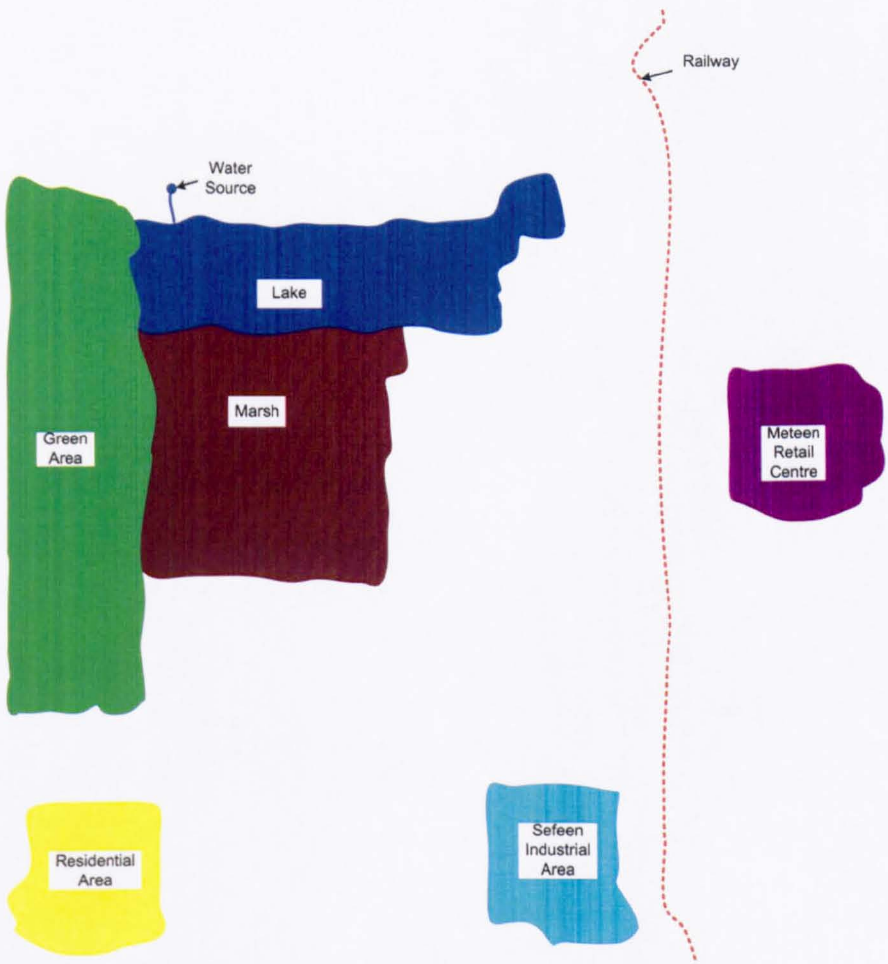


Figure 7.1: Kawar Town With Area Notation

This world has some necessary features that might exist in a world. The world consists of three demand areas (a residential, a retail, and an industrial), a lake, a green park and a marsh. A rail passes the world.

Land-use classifications are usually indicated by the use of symbols and/or in a color code. A simplified system like the proposed one for Kawar town should be adequate for small worlds. GIS maps generally use a different colour for each of the major land-use categories (Jeer, 1997). Many colors used on GIS maps have a relationship to the object or feature on the ground. For example, blue is often the color chosen for fresh water or ocean. These colours effectively illustrate land-use concepts by graphically displaying land-uses, rails, public infrastructure, and other real-world features. The suggested colors for Kawar town are shown on the following list of uses:

- Yellow for residential uses such as town houses.
- Purple for retail and commercial uses
- Blue for lakes and rivers
- Green for green parks and forests
- Aqua for industrial uses
- Orange for railways
- Maroon for a marsh

There are a number of points to make about the town of Kawar:

1. There are relatively big areas of straight lines. This may imply that the town is developed on a 'grid' system.
2. The majority of the town is west of the railway that crosses the town from north to south.
3. The town has natural features such as nature conservative park, a lake and a marsh.
4. The town has one residential area.
5. The town has access to a two-way rail, industrial, and retail.
6. The town has a single water source fed by an open channel from the lake.
7. Residential, retail and industrial areas can be considered demand areas.

8. The dimensions of this world are approximately $5\,km \times 5\,km$.

9. Town centre is removed from the plan

10. The road network are removed from the plan.

The town centre includes a full range of local shopping facilities, including a superstore, as well as local health and community facilities, local business offices, restaurants and cafés, and the opportunity for a worship centre.
The residential areas include a wide range of houses and flats with appropriate and integrated provision for public open space and schooling.
The commercial areas (retail and industrial) provide a wide range of uses including business and light industry, commercial leisure, and hotels.

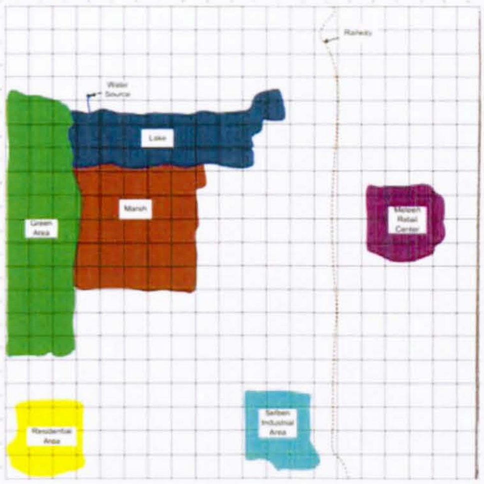
It is necessary to input all data and options which are shown on the world. This comprises existing demand areas and other features such as lakes, railways, green parks.

7.3 Modelling the Town

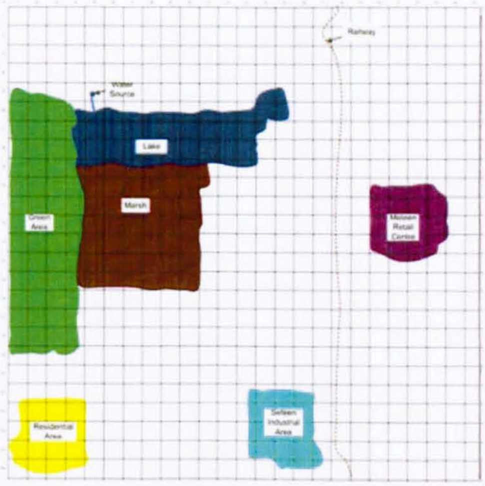
In vector GIS, polygons can represent the actual shape and area of a city. In raster GIS, a number of grid cells can show shape and size. Raster data, alternatively known as grid data, are pixel format information referenced to a coordinate system. The use of this grid is both common and beneficial as the grids can have many properties. The finer the pixel or grid size, the higher the spatial resolution of the data.

It is imperative that the scale, e.g. accuracy, of the data be known prior to conversion. The accuracy of the data, often referred to as the resolution, should determine the cell size of the output raster map during conversion.

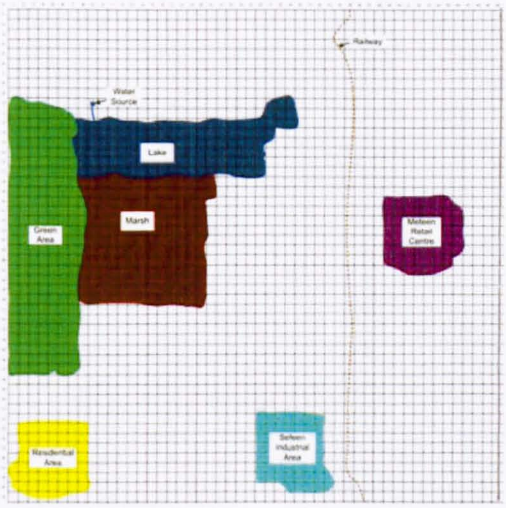
The problem of determining the proper grid cell size can be a concern. If one selects too coarse a grid size then data may be overly generalised. If one selects too fine a grid cell size then too many grid cells may be created resulting in a large data volume, slower processing time, and a more bulky dataset. Figure 7.2 shows different grid cell sizes superimposed over the town of Kawar.



(a) 250 m Grid Size



(b) 200 m Grid Size



(c) 100 m Grid Size

Figure 7.2: Different Grid Sizes Superimposed Over Kawar

The grid is superimposed over the town of Kawar so that the grid cells align with most land-use boundaries or parallel with the city features. This results in a raster grid using fewer cells than rotating the grid.

The level of detail (of features / phenomena) represented by a raster grid is often dependent on the grid cell size of the raster grid. The cell must be small enough to capture the required detail but large enough so computer storage and analysis can be performed efficiently. More features, smaller features, or a greater detail in the extents of features can be represented by a raster grid with a smaller cell size. However, more is not often better. Smaller cell sizes result in larger datasets to represent the entire world; therefore, there is a need for greater storage space, which often results in longer processing time.

Whilst there is no actual scale on the town, the selected grid cell size for this world is chosen to be $100\,m$ (grid cell size shown in figure 7.2(c)). The selected grid size would equate to 50×50 grid and $5\,km \times 5\,km$ town dimensions. The size of grid cells is selected on the basis of the data accuracy and the resolution needed by the user. A raster data structure is, in fact, a matrix where any coordinates can be quickly calculated if the origin point is known, and the size of the grid cells is known. Since grid cells can be handled as two-dimensional arrays in computer encoding, many analytical operations are easy to program.

Some observations can be made about the modelling process of the Kawar town. Firstly, the grid appears fine even with the limit in the model of a 50×50 maximum grid cells. Secondly, the issue of scale combined with the level of allocation, considering some examples around Nottingham, Melton Mowbray (population circa 27,000) is approximately $2.5\,km \times 3.5\,km$ whilst Loughborough (population circa 60,000) is some $5\,km \times 5\,km$. The population figures are according to The GeoNames Geographical Database (2011) and rounded up to the nearest thousand.
The town of Kawar could be an example of a town of a population size of $60,000$ to $90,000$. According to the Office for National Statistics (ONS) in the UK, the average area of a town would be $20.25\,km^2$ (Office for National Statistics (ONS), 2005). Since the world is always assumed to be rectangular, the average dimensions of the world will therefore be $4.5\,km \times 4.5\,km$.

Sometimes map features need transformation to adjust units of measurement, rotate a map, change the origin of a coordinate system, or remove a distortion. The orientation of the raster grid may be changed and placed over the world. The raster grid shown in figure 7.2(c) is rotated by 45 degrees over the town of Kawar as shown in figure 7.3.

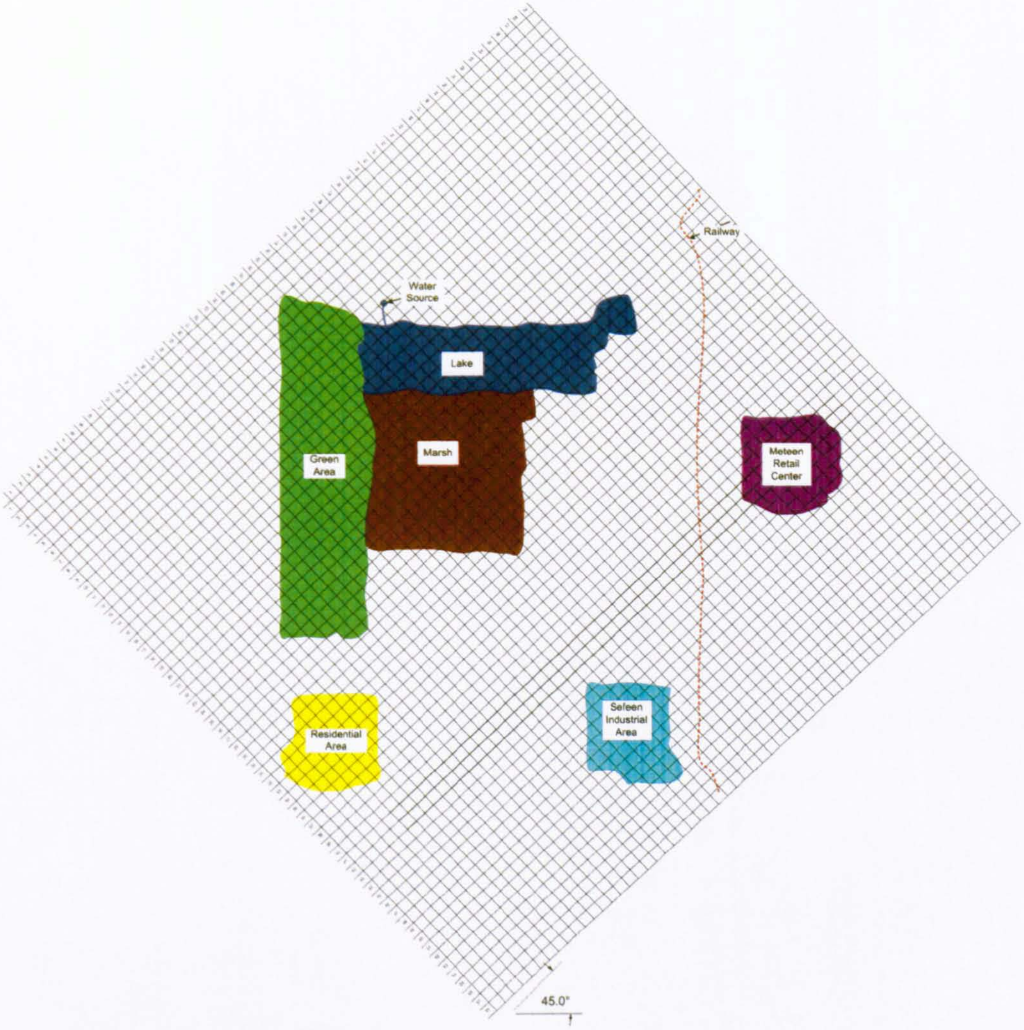


Figure 7.3: Kawar Town With 45 Degrees Rotation of the Rectangular Grid

With the orientation of raster grid at the above figure and may be with other rotations (e.g. 30 degrees and 60 degrees), input data and analytical operations become difficult to perform and pose difficulties in calculations of land access cost.
The grid should be oriented to minimise the number of cells that fall outside the world boundary. Even though these cells are outside the active grid, they occupy space in arrays and tie up computer memory. The other side effect of an orientation is that most grid cells are not aligned with city feature boundaries.

The chosen grid size and orientation of the raster grid shown in figure 7.2(c) is input into the GA model and the Kawar town model appears as shown in figure 7.4 where all the features of the town can be seen including the railway line.

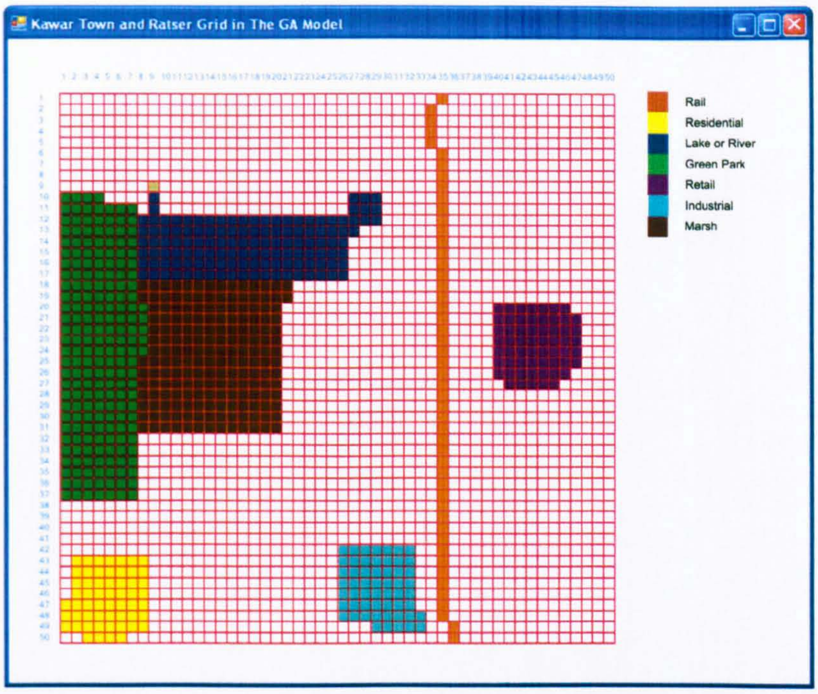


Figure 7.4: Kawar Town Being Input to the GA Model

Placing a raster grid over the town will generally results in straight lines on the raster dataset. Thus, squares on the raster grid are commonly changed into parallelograms of arbitrary scaling and angle orientation.

The display in figure 7.4 appears coarser with the effect of placing the raster grid. It is suggested that the GA model representation of Kawar can be recognised despite the effects of the raster grid. It is, therefore, accepted that:

- Lines are much more angular than in the original and some of the naturally curved lines are straight;
- The rail appears very wide because of the scale;
- In some occasions, an important feature may be on the edge of a cell and its location requires a decision to be made. An example of this is the edges of the lake.

The GA world as displayed can, therefore, only be taken as an approximation of the real-world.

The Kawar town has an existing water supply network. This network is superimposed over the Kawar town in figure 7.5. This network of pipes could be used as a possible solution to a problem. The representation in this figure could be useful as a chromosome. However, it is not ideal because it is difficult to implement.

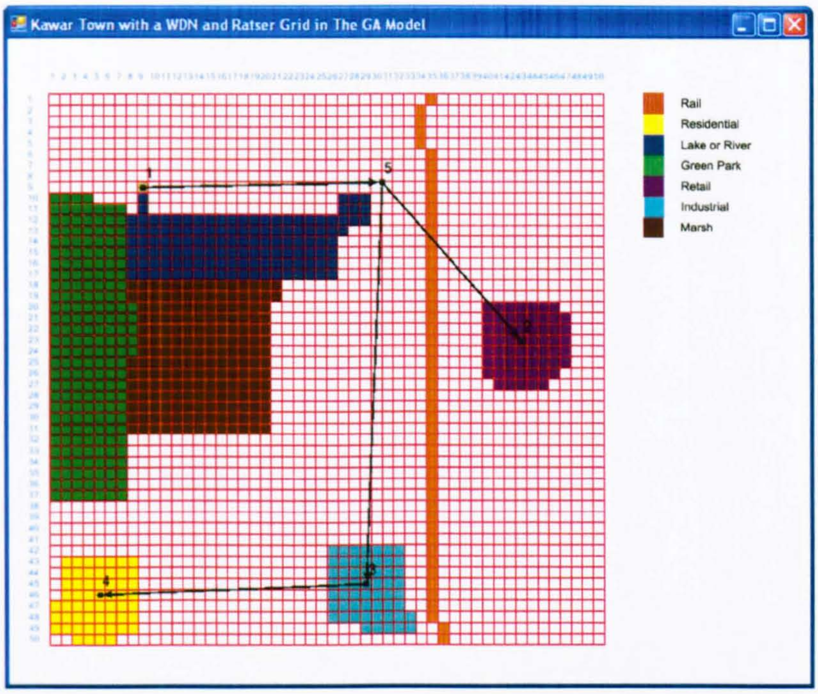


Figure 7.5: The Existing WDN Superimposed Over Kawar Town

There are a number of points to make about the existing WDN:

1. Pipes are straight lines between two terminal nodes.
2. Water distribution system components for this world, including pipelines, accessories, and storage facilities, are generally sized to provide adequate flow.
3. The water demand specified at any demand node of the network is the flow available at that location.
4. Demand requirements vary according to size of the area and the nature of the land-use.
5. The demand node is fixed at the center of the demand area.
6. A series of water transmission lines (local pipes) transfers the water from the distribution units to other parts of the demand area.
7. To connect the Mateen retail centre to the current network, it is required to take the pipe across the rail which adds an extra cost to the distribution network. This is because crossing cost of the rail is high (this is discussed in section 7.4).

7.4 Land Access Cost of Kawar

The land access cost should include all the costs related to construction and maintenance of the network. When assigning these, the user should consider aspects.

This cost element represents the cost of disruption caused by the construction and maintenance processes of the WDN at each cell within the world. This cost is not a real fiscal cost. It is relative to the fitness function, although that some costs are reflective of fiscal cost.

It difficult to mention all possible disruptions caused by pipeline construction, only some some will be discussed. The construction process of a WDN may impact:

- Residential, retail and industrial areas:

  Construction works and maintenance works might impact on several properties within these areas of the town. These projects might require widening of roads, changing junctions and building special access roads to the site. The additional problems for residents can be:

  - Pollution, noise, and vibration from construction activities.
  - Traffic disruption (e.g. public transport) caused by diversions and congestion.
  - Road safety issues caused by extra traffic.
  - Road widening schemes and diversions put in place for the duration of the project.

- National Pathways (e.g. roads and railways):

  Transportation is one of the critical networks required for moving material and people among locations distributed in a geographic area. Any disruption to these networks can have a major impact on economic productivity as well as making peoples daily lives more difficult.
  Constructing pipelines may cause damages to road edges caused by heavy goods vehicles traffic. In addition, railways may need to be moved / removed during the construction and maintenance durations.

- Landscape (e.g. public green parks):

  Public parks as representative of urban green areas have played a significant role against degradation of urban environment while keeping the rapid pace of urban growth (Iamtrakul et al., 2005). The construction of water networks might spoil the rural landscape with industrial infrastructure. Although pipelines are buried, establishing water networks in green spaces may disrupt the objectives of environmental protection, recreation, and to meet people's daily routines.

- Biodiversity and the ecosystem (e.g. marshes and farmlands):

  The destruction of habitat threatens rare species and result in losing many trees. Pipelines may cut through farmlands which is inconsistent with local authority requirement for sustainable agriculture.

The construction and maintenance processes might impact the other areas such as protected areas (e.g. Sites of Special Scientific Interest (SSSI)), military controlled zones (e.g. military bases), and historic areas (e.g. historic buildings).

Further possibilities may include the need to cross a lake or river (non allowed areas), in which case the cells which it is necessary to cross would be very expensive. There are many possibilities for travelling in non allowed areas and the user may set the appropriate costs to model the scenario.

The actual land access cost of a facility or land-use differs from place to place and changes over time periods.

In the worlds defined in section 6.2.1, there were only two different categories of access cost, low-cost areas and high-cost areas. The low-cost value is assigned to demand areas and to public open spaces (POS) whilst the high cost value is assigned to features defined as obstructions or have an expensive access cost such as lakes and rivers. The the town of Kawar requires a wider selection of access costs to be incorporated into this world. This is done directly onto a world grid.

Table 7.1 presents the land access costs in the fitness calculation.

Table 7.1: Typical Costs of Crossing Cells

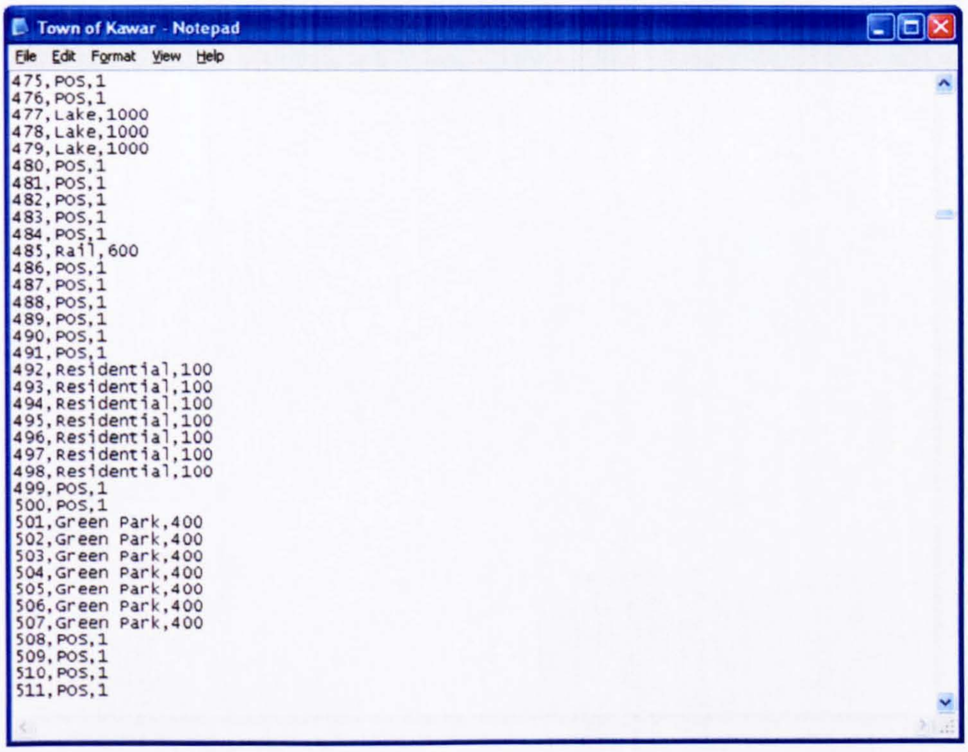
Type of Cell	Access Cost Per Cell
POS	1
Water Source	1
Residential	100
Industrial	200
Retail	300
Green Park	400
Marsh	500
Rail	600
Lake	1,000

Access costs are set for land of various types. For example, POS where movement and disruption caused by placing a pipeline is generally very cheap. A more expensive rate would be likely for other land-uses where the disruption caused by placing a water network affects the existing facilities (e.g residential, retail, and industrial) and/or degrade urban environments (e.g. green park). Further possibilities include the need to cross a railway or a lake, in which case the cells which it is necessary to cross would be very expensive.
A further option is that the land may be a military bases or a SSSI where both crossing and development is forbidden (high costs applied). A more difficult problem that the incorporation of additional land access costs potentially creates is the situation where there are two or more facilities competing for the same 'route'.

This valuation of these costs is entirely subjective and difficult to define numerically or financially or even from one person to another.
In addition, fitness function coefficients need to be scaled and adjusted to balance the fitness elements. It is difficult, however, to value these scaled elements against each other. As an example, the scaled values for material costs and installation costs can be argued as reasonable and logical within their own scale. However, the figures used for land access cost are generally higher than those used for other costs. It must be considered whether this difference in scale is appropriate

or not. It is for this reason that the fitness balancing coefficients are used.

Land access costs are set up in a file which includes land-use and the relative cost. This file is then read in by the formulation and applied in fitness calculations. Figure 7.6 shows a part of the typical land access cost file which is used in the experiments.


Town of Kawar - Notepad
File Edit Format View Help

```
475,POS,1
476,POS,1
477,Lake,1000
478,Lake,1000
479,Lake,1000
480,POS,1
481,POS,1
482,POS,1
483,POS,1
484,POS,1
485,Rail,600
486,POS,1
487,POS,1
488,POS,1
489,POS,1
490,POS,1
491,POS,1
492,Residential,100
493,Residential,100
494,Residential,100
495,Residential,100
496,Residential,100
497,Residential,100
498,Residential,100
499,POS,1
500,POS,1
501,Green Park,400
502,Green Park,400
503,Green Park,400
504,Green Park,400
505,Green Park,400
506,Green Park,400
507,Green Park,400
508,POS,1
509,POS,1
510,POS,1
511,POS,1
```

Figure 7.6: Typical Access Cost File

The information contained in the file is; a cell number, followed by a cell type, followed by the relevant access cost. For example, cell number 485 is a rail cell with a relevant access cost of 600 and cell number 498 is a residential cell with a relevant access cost of 100.

7.5 The Development of the Town of Kawar

There are many cases where people like to live and settle in new towns and cities, among them is the town of Kawar. As a result, there is increasing demand to build a new residential area within the boundaries of the town.

It is anticipated to develop the current world (town of Kawar) and add a residential area. To achieve this, the level of growth will require a sustainable approach to new development resulting in higher making of more efficient use of previously developed land-use and existing infrastructure. The world with the new area is shown in figure 7.7.

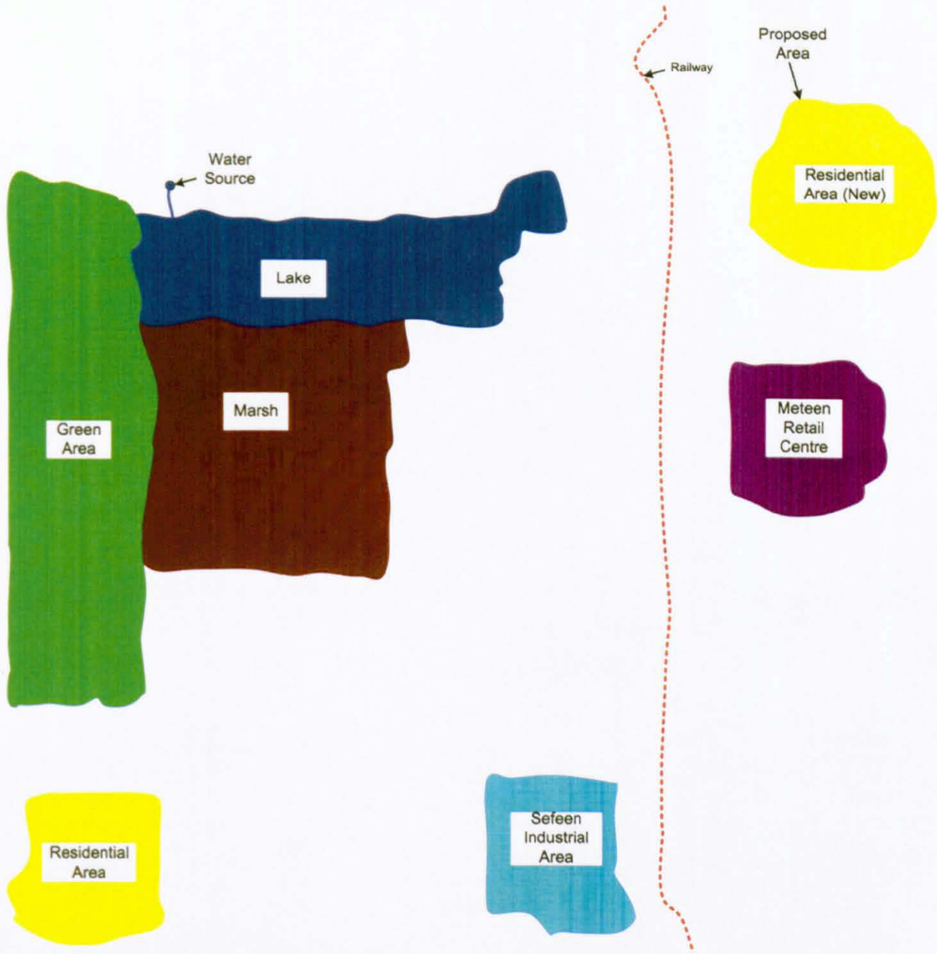


Figure 7.7: A Typical World Layout With Future Development

A feature in the 'real' world, whether existing or prospective, could be any shape or size and be positioned in any orientation.
While the location of the new residential area could be any where within the town boundaries, the criteria used for choosing the location of new residential projects inside the city may be the natural amenities and lower pollution levels, land price, the social characteristics of the neighbourhood (e.g. average crime rate), quick access to city center and commercial areas.

The main town features remain the same as shown in figure 7.1. There are some points about the new layout of the town of Kawar:

1. The new residential area in Kawar is located to the right of the railway and above Mateen retail centre.

2. By adding the new area, the town will have two residential areas.

3. The new area has an individual demand amount.

The prediction of future demand is uncertain and the demand for water resources is continuously increasing with the growth of population. When unplanned water needs arise, the original water network may fail to produce the expected level of benefits to the new demand areas. The main challenge is to better utilise available water sources by coordinated operation of WDN in order to meet the increasing demands under the existing pipe network.

The existing network with the new layout of Kawar is shown in figure 7.8.

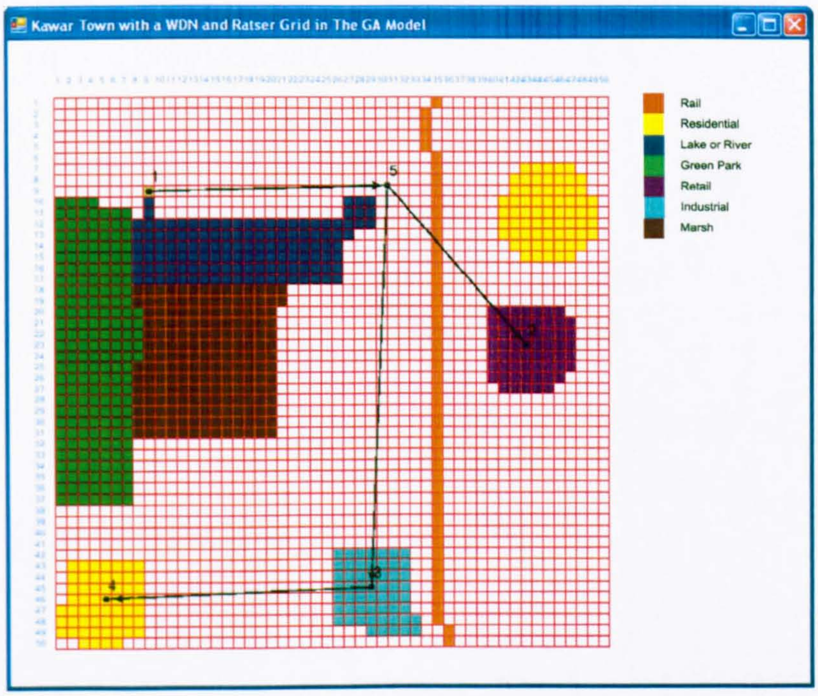


Figure 7.8: The Existing WDN Superimposed Over the New Layout of Kawar

The model in this work is designed to optimise both the network layout and hydraulic design of the existing WDN and the expansion of existing systems. The proposed world layout helps to evaluate the existing system as well as the expanded system to ensure that the WDN model copes with current and future demands. Additionally, it can be used to assist in decision-making and negotiations for real-world applications.

The design of WDN should cater for the requirement of surplus capacity for future growth. To meet this demand, the city's piped water network need to be extended to cover the new adjoining areas.

The first step in evaluating a water distribution system is to derive existing and future demands and to develop the hydraulic model. The model is then used to determine system improvements needed to correct existing and projected future deficiencies.

Walski et al. (2003) emphasised the need for models that will maximise the net benefits in addition to minimising the network cost. The biggest difficulty in water distribution system design is the prediction of future demand. A designer would like to provide excess head (beyond the required minimum head) at each node to overcome increased head losses under unexpected high demand.

Decision-makers make decisions by changing some network features such as source location or demand area locations (the location of the demand node). These decisions are very useful in generating WDNs which may eventually produce optimal solutions to cost or other design problems. As an example, the location of the source location may be changed in which the pipeline is connected to the source and is used to deliver water to the demand areas where a demand node exists.

7.6 A Demonstration of the GA on Kawar Town

In real-world applications, problems relating to WDNs may be generally divided into three categories:

1. Designing a new network
2. Modifying or expanding an existing network
3. Operating an existing network

At a broad level, the design of a new network would involve determining the network layout, pipe sizes, pump sizes, tank sizes and locations, valve locations and operating schedules of the pumps and valves in the system. A new WDN would have to be optimally designed to handle forecast demands at a desired level of service throughout its service life.

However, on account of the unprecedented growth and development that might happen in future. A network may not perform at the expected level of service and hence may have to be extended in order to improve its performance and supply water to satisfy the required demands. The extension and rehabilitation of an existing network would involve new nodes and pipes, replacement of existing pipes with new ones, cleaning and lining of existing pipes, duplicating the existing links and addition of new pumps and tanks.

The operation problem, on the other hand, involves determining the optimal pumping schedules for a given network so as to minimise the cost of operation, while ensuring adequacy of level of service under all possible loading conditions.

The experiments are carried out to show how well the GA model will evaluate prospective alternatives and connect the new parts of the city to the existing distribution network.

Domain knowledge can be used to design local improvement operators, which allow more efficient exploration of the search space around good points (Suh and Van Gucht, 1987). It can also be used to perform heuristic initialisation of the population, so that search begins with some reasonably good points, rather than a random set. Knowledge based operators and dynamic operator probabilities are probably going to help solve real-world problems.

As stated earlier, the network examined in this chapter consist of a single source node and multiple demand nodes. During the evolution process, demand nodes are fixed at the center of demand area and cannot move to the far boundaries of the demand area. It is possible for any node in the system to work as a source node for the new demand area.

The (x, y) coordinates of the five nodes in the network shown in figure 7.8 are given in table 7.2.

Table 7.2: (x, y) Node Coordinates for the Network Shown in figure 7.8

Node	Node Type	x	y
1	Source	850	850
2	Demand	4250	2250
3	Demand	2850	4450
4	Demand	4500	4550
5	Intermediate	3000	800

An additional demand node is required to be fixed at the center of the new demand area. This node is necessary to considered in the model in order to connect the new residential area. The specifications of the new node (node number 6) are shown in table 7.3.

Table 7.3: (x, y) Node Coordinates for the New Demand Area

Node	Node Type	x	y
6	Demand	4450	1050

It is impossible to include all possible tests here and where matters have already been considered as part of earlier testing. Therefore, the best model and GA parameters tested in chapter 6 are used in the tests.

The available water at the source node is 50,000 and the demand for each area is 10,000. Crossover type 2 and mutation type 1 are used. The other GA parameters used with the real-world are presented in table 7.4.

Table 7.4: GA Parameters Used in The Example World

Population Size	No. of Generations	Mutation Rate (Individuals)	Mutation Rate (Links)	Mutation Rate (Node Positions)
1,000	3,000	15%	5%	15%

The empirical results of this testing are averaged over several independent runs. The GA convergence of this world is shown in figure 7.9.

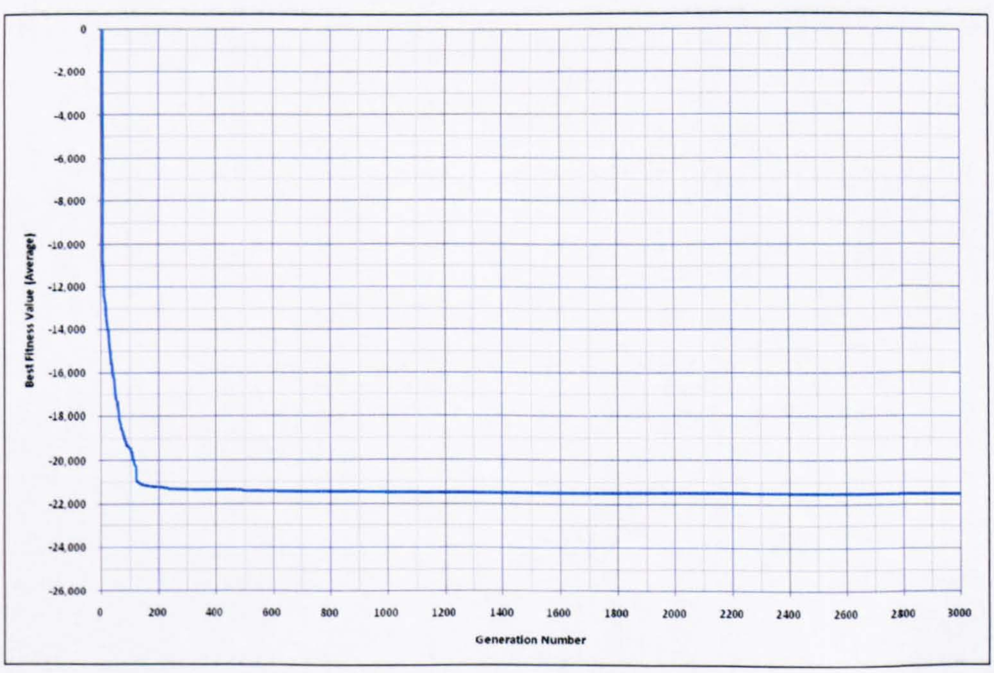


Figure 7.9: GA Convergence for The Typical Real-World

From the above figure, it may be seen that after rapid initial convergence within the first 100 generations, solutions are generally stable after 200 – 400 generations. It demonstrates that the convergence occurred at a similar rate with worlds used in chapter 6.

The model allows a rough optimisation run to be executed to determine areas of concern (new demand areas) in the pipe network design.

It is clearly not possible for all results to be shown here and, however, one solution is selected. The world with the new network layout is shown in figure 7.10.

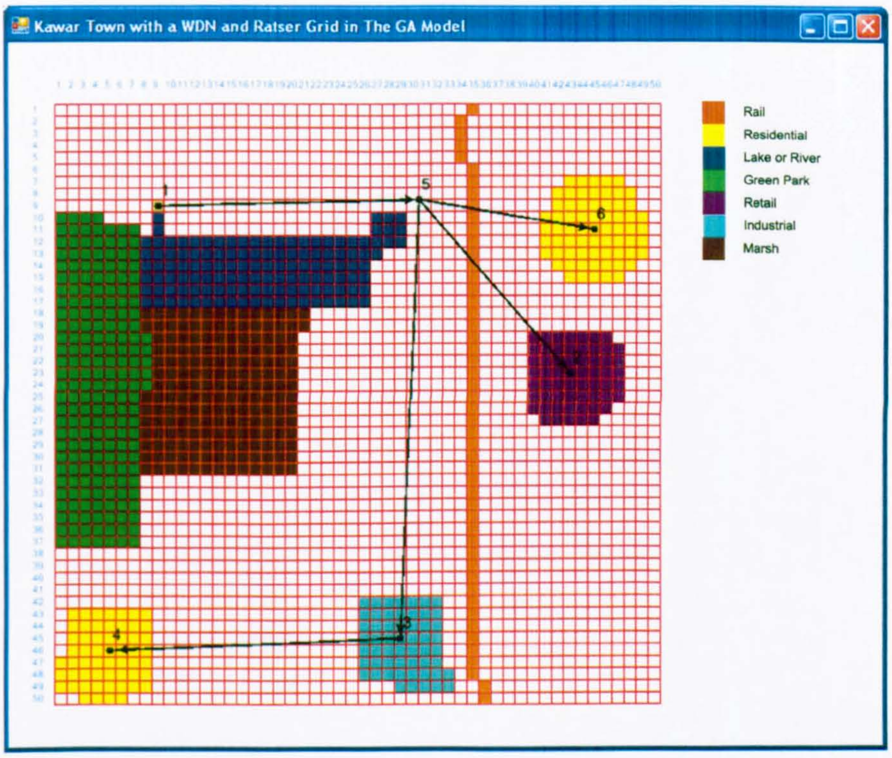


Figure 7.10: A World Layout With The New Water Network

In this figure, it can be seen that node 6 is connected to source via node 5. To supply water to the new residential area, it is most convenient for the GA model to connect node 6 to node 5 amongst other nodes in the system for many reasons. For example, the model would assign 'suitable' head and pipe size to one connection in the network (connection between 6 and 5), whilst it may require to assign 'suitable' values (head and pipe size) to more than one connection to connect 6 to the source and achieve the required benefits from this connection.

A number of additional observations can be made for the connection between node 5 and node 6:

1. Although the new pipeline passes the railway, it is the lowest cost route in terms of land access cost amongst the other possible routes.

2. The distance between node 6 and node 5 is not the shorter than any other distance between node 6 and the other network nodes. The distance between node 2 and node 6 is shorter.

7.7 Conclusion

The chapter described an example of a real-world. This world allowed a wider range of features to be modelled than worlds tested in earlier chapters.

The rectilinear grid approach to modelling the world did not prove to be a problem and indeed the speed of the algorithm allowed the grid to be very fine.

Different access costs are defined for each land-use increasing the complexity and the realism in the modelling of real-world features. This also meant that more specific requirements were applied to the GA.

The results of tests carried out on this world demonstrated that the model was successful to generate an optimal or a near-optimal water distribution network and supply sufficient water amounts to consumers. In addition, the potential exists for the use of this formulation in more complex and real-world scenarios.

Chapter 8

Conclusion and Further Work

This thesis is concerned with the topic of WDN design optimisation using GA and GIS towards the objectives of minimising cost and maximising delivered water to consumers.

The aims and objectives of the research set out in chapter 1 are presented in this chapter followed by a discussion on them. In addition, general conclusions are also discussed. The chapter finishes by a description of possible avenues for further research.

8.1 Aims and Objectives

As set out in section 1.2, the main aim of this research was to investigate the possibility of combining GAs and GIS in the WDN design optimisation. This is done with the aid of a decision mechanism which enables the model to reach a meaningful solution and provide a practical design technique for WDNs. The main objectives in order to reach its stated aim were:

1. To provide a background to what is required in planning with particular references to aspects which might affect or be affected by water supply.
2. To provide a background to the use of networks and theories related to networks which have been applied to WDNs and similar fields of WDNs.
3. To provide an understanding of WDNs including:
   - The planning process
   - WDN components
   - WDN features

- The types of WDNs
- The hydraulic theory

4. To provide wide literature review and critical analysis of work done by others to model and optimise WDNs.

5. From the important features identified through the achievement of previous objectives, to produce an initial prototype model for the design of a WDN which can be used to determine the necessary features of the 'final' model and how these might be included.

6. From objective 5, to produce the 'final' model.

7. To test the model to determine its practicality and the best values for all the parameters.

8. To demonstrate the applicability of the model on a hypothetical real-world layout.

In order to determine whether these objectives have been met, they will be considered in turn in the light of the preceding chapters of the thesis.

8.2 Conclusion

8.2.1 Conclusion Specific to Objectives

- **Objective 1**

  An understanding has been gained to some urban and regional planning aspects as described in section 2.2 and its subsections. Urban and regional planners are concerned with the interconnected layout standards of land development such as land-uses and water demands. For instance, a plan of a land-use development will determine the amount of water required.

  It was concluded that water supply is one of the urban problems which planners should consider when they implement urban development plans. These plans should cater for standards and rules of the physical environment and social development plans. In addition, studies should be conducted to avoid water supply development in unstable areas such as flood plains and earthquake faults.

  It was demonstrated in section 2.2.1 that planning provides decision-makers with the required information to make decisions, which provide a framework to develop economic, social, environmental, architectural, and political activities. The first step in the decision-making process is to define the problem then search for alternatives. Then, the selection criteria need to be identified to evaluate these alternatives. The last step is the selection of these alternatives. The primary focus of the planning of water supply is to make recommendations on a range of alternatives that should be evaluated to ensure the best possible service.

  When the data are organised, analysed and interpreted, they then become information and can be stored in GIS datasets. These data can be useful for effective decision-making. Therefore, it was demonstrated in section 2.2.3 that a new trend is emerging where GIS are combined with analytical, mathematical models to form spatial decision-support systems in urban and regional planning.

  A brief introduction to sustainable development is provided in section 2.2.2. The development is sustainable when it meets the present standards of living and quality of life without compromising the ability of future generations to meet their own needs. This concept is important in the design of WDNs.

- **Objective 2**

  Network models are an important category of mathematical programs that have numerous practical applications. A background on the theory of networks is provided in section 2.3 and its subsections. Directed and undirected graphs are introduced in 2.3.1. Other useful concepts in the networks such as adjacency matrix and adjacency list are also discussed. The knowledge from this section and its subsections is useful for the design of WDNs because they are represented as collections of nodes connected by links. In WDNs, a link might infer one-way connection from one node to another, which represents the direction of flow. In addition, a WDN may be more complex than this there may be more than one type of node or more than one type of link exists in a network. The connection between nodes in a WDN is represented by a non-zero value in a two-dimensional matrix, where the number of a row in the matrix represents the number of the upstream node, and the number of a column is the downstream node.

  More importantly for this work, cycles in networks are explained in 2.3.4 and the technique used to detect them is detailed in section 2.3.5 and its subsections. Depth first search is used to find cycles in a network and to find paths between two nodes or reporting that no such path exists. It is a systematic way to find all the nodes reachable from a given source node. This knowledge is partly used in the production of the 'final' model (objective 6).

- **Objective 3**

  An understanding has been gained of WDNs in section 2.4 and its subsections.

  The basic theory of WDN is presented in section 2.4.1 which provided a foundation to the necessary concepts and terminologies in WDN design.

  A WDN requires extensive planning in the design stage and maintenance during operations to ensure that good service is provided to the customers in the most reliable and economical way. For this, section 2.4.2 is devoted to water supply planning. The planning of WDNs should consider the future improvements to the current system. This includes upgrading the treatment plants, searching new water resources, and constructing new water tanks. The system should be able to supply sufficient water quantities and solve waste water system problems.

  Section 2.4.3 and section 2.4.4 briefly presented WDN components and features respectively. An understanding has been gained to how water utilities construct, operate, and maintain water supply systems. The components of the WDN may be different from a region to another according to the nature and the purpose of the system.

The types of WDNs are discussed in 2.4.5 which is a necessary topic to have the knowledge about the WDN types and their functions. Many authors classified WDNs according to their layout into three types: 1) serial, 2) Branched, and 3) looped. Most of urban water supply systems are combinations of both looped and branched networks. This combination adds the power to the system in terms of reliability and cost savings.

Section 2.4.7 constituted a review of essential concepts in fluid mechanics that is necessary for hydraulic network analysis. The topics include: pressure, flow-rate, head-loss, and friction losses. This should provide the reader with a foundation for understanding the concepts and terms used in this work.

- **Objective 4**

  Section 3.2 provided a general background on optimisation. A special attention and a broad introduction was given to the design optimisation of WDNs in section 3.3. This section discussed the WDN design features which may add more complexities to the design of WDN such as multiple loading conditions, uncertainty, network operation, water quality, and reliability.

  Mathematical models are required to analyse the WDNs. In addition, simulations are increasingly used to analyse many design problems in urban water systems. Simulation models in WDNs are introduced in section 2.4.9.

  A concise history of research into the WDN design problem was included, showing how the problem evolved over the years from simple pipe network design to complex design of all WDN components. This is done through a wide literature review on the WDN design optimisation as described in chapter 3 where section 3.3.1 presented mathematical optimisation techniques and section 3.3.2 reviewed meta-heuristic optimisation techniques. The literature reveals that a considerable work has been done in the design optimisation of WDNs. There are different optimisation techniques have been used. However, there is no single optimisation technique has been stated to be better than any other in the models described in the literature and there is no agreement amongst the researchers on the best hydraulic model to be used.

  Several mathematical optimisation techniques are applied to the design of WDN such as linear programming and dynamic programming. These techniques are used in the design of WDN for a least-cost implementation of a network. In this instance, the least-cost refers to the minimisation of the implementation cost of the network in terms of the cost of the pipes to be installed. The studies using these techniques were further developed to consider larger water systems and considered additional complexities. It was concluded by many researchers that the computational efficiency of mathematical programming methods is limited to continuous solutions which are not favoured from an engineering view. It is, therefore, necessary to use meta-heuristic techniques in situations where classical optimisation techniques would struggle to achieve good results with acceptable runtime.

Meta-heuristic algorithms are useful and efficient tools to solve combinatorial optimisation. These algorithms combine different concepts for exploring and exploiting the search space. For WDN design optimisation, researchers focused on heuristic based optimisation techniques such as Simulated Annealing (SA), Ant Colony Optimisation (ACO), and Particle Swarm Optimisation (PSO). These techniques were successful in obtaining lower cost solutions than the solutions previously reported in the literature.

Additional types of meta-heuristic algorithms used in the design of WDNs were listed in this section. The decision variables were pipe size, network layout, nodal head, supplied water, and pump.

Section 3.5 and its subsections provided a wide literature review on using GAs to assist in the design of WDNs. GAs have proven to be potentially useful tools for building an optimisation based solution to a WDN problem.

There is a need for a GA method which considers the following:

- Pipe size
- Network layout
- Head along the pipe
- The amount of delivered water to demand areas

An understanding has been gained on GIS applications in the design of WDNs as described in section 3.6. In addition, a framework in which GIS can be combined with GAs to assist in the design of WDN was presented in section 3.7. It was demonstrated in these sections that GIS-based world representation and data could be incorporated into the GA formulation. For example, GIS can be used to provide data about land-use and on water demands in any area. The spatial information in a GIS database can be filtered and prepared to be read by the GA model. This information is used to produce results by the formula of the objective function, which is subject to some constraints. The essence of GIS is to link different datasets and present them clearly in a variety of ways.

- **Objective 5**

  The formulation of the initial prototype includes:

  1. The initial design of fitness function (section 4.2).
  2. Initial chromosome representation (section 4.3).
  3. The design of the 'world' in which the model will be built (section 4.4).
  4. Initial GA operators (section 4.6).
  5. Initial GA tests (section 4.7).

  The fitness function determines how well the GA is able to solve the problem. The initial fitness function consists of three parameters: material and installation cost, land access cost, and the payment to the supplier. The design of the initial fitness function was simple enough to aid in the determination of GA use to address the problem.

  The important aspects in the initial chromosome representation for a WDN are the nodes and their positions and the links. The initial chromosome consists of: 1) the connectivity matrix, 2) 2-D node positions, and 3) node type. The connectivity matrix consists of the connection between nodes. These connections represent the pipes along which the water will flow. The nodes of the network must be positioned within the world. It is, therefore, necessary to include their (x, y) coordinates and their type.

  The world in this work was described in GIS in terms of models that define the concepts and procedures needed to translate real-world observations into data that performs meaningful analysis in GIS. The world is represented using three types of data: 1) world data, 2) raster grid, and 3) area information.

  Several types of crossover and mutation types are developed in chapter 4. These are in the testing of the initial prototype. These operators proved to be useful in the design of WDNs.

  The results of the initial prototype testing of the initial GA formulation are presented in chapter 4. In this chapter, the initial model formulation was useful to generate an optimal design for a WDN. The initial model focused primarily on the least-cost design of the WDN for which the pipeline is connected to the source and used to deliver water to the demand areas where a demand node exists. However, the initial model was developed to include the optimal WDN design problem with some additional complexities.

- **Objective 6**

  The 'final' model included changes to the fitness function, chromosome representation, and GA operators. These changes addressed more complexities to the problem.

  The initial GA formulation was developed and presented in more details in chapter 5. Additional objectives were added to the fitness function to represent more realistic real-world situation. These complexities considered the decision mechanism and hydraulic principles. The optimisation proceeds by considering alternative sizes for pipelines and other system elements and, for each network configuration, calculating the hydraulic properties of the network such as flow and pressure values based on some decision rules. The system has a feasible configuration (no cycles allowed) if the hydraulic properties satisfy the constraints set on them. A new chromosome representation was needed to be introduced as in section 5.4.2.

  Novel GA operators were developed for the real and integer coded chromosome components. These operators were presented in initial GA operators in section 4.6 and discussed and improved in more details in chapter 5 in section 5.4.4 and section 5.4.5.

  The optimisation model was able to produce an optimal or near-optimal pipe layout, sizes, and head. However, the objective of the least-cost design is an improper measure of effectiveness. The least-cost optimisation primarily reduces the size of the system components for reducing cost. In this work, the optimisation model was able to produce an optimal WDN that is able to deliver the required water amounts to demand areas.

  A decision mechanism is incorporated into the problem formulation. This decision is discussed in details in section 5.3 and its subsections. The decision-making criteria which are used in order to help identifying distribution amounts of water to demand areas, whilst seeking to provide the best design of the WDN.

  The decision also involved selecting one or more lines of action that will return the most perceived benefits to the consumers. The process of model development involved making decisions and making choices among alternative possible development paths. One of these decisions is highlighting the importance of specific parts in a world which relates to which activities are taking place in that part.

- **Objective 7**

  The detailed testing in chapter 6 was followed by the detailed GA formulation in chapter 5. The testing regime is presented in chapter 6 to test the best combinations of model parameters. This is done using five manufactured world layouts and one example which is hypothetical. The complexity of the five worlds varied according to the number and the distribution of demand areas within the world. Each world consisted of a water source, an obstruction, and a number of demand areas randomly distributed within the boundary of the world. The size and position of the water source as well as the obstruction are fixed for all worlds. The description of world is given in section 6.2.1.

The complete testing regime is proposed and presented in table 6.1. It was not possible to test all potential parameters and their combinations. Therefore, only selected GA parameters and values were used for testing. These parameters were: crossover and mutation (section 6.2.2.1), population size (section 6.2.2.2), number of nodes (section 6.2.2.3), and mutation rates (section 6.2.2.4). In addition, the adjacency mutation operator (section 6.2.2.5) did not prove to be effective to improve the solution and requires further investigation.

Several types of crossover and mutation were tested with two worlds. The best combinations of crossover and mutation observed were for crossover type 2 and mutation type 1 and crossover type 1 and mutation type 1.

Different population sizes were tested in this work of 500, 1,000, 1,500, and 2,000 individuals. All the five defined worlds were used in these tests. The suitability of each population size for different worlds is summarised in the below table.

World	Population Size			
Number	500	1,000	1,500	2,000
1	Yes	Yes	Yes	Yes
2	Yes	Yes	Yes	Yes
3	Yes*	Yes	Yes	Yes
4	No	Yes	Yes	Yes
5	No	No	Yes	Yes

* A population size of 500 individuals can be used for a world consists of five demand areas, but not recommended.

It was concluded that small population sizes can be used for simple geographical representations while larger sizes are required for more complex layouts.

The number of nodes was tested with the total demand amount for each world. Two sets of total demand amounts were tested. The first set used a fixed total amount for each world. This amount is divided evenly among the demand areas. The second set used a fixed demand amount for demand area. In this case, the total demand for each world is different. In this test, 10, 15, and 20 were tested with all worlds. The best performance was recorded when 10 nodes are used with a fixed demand amount for each demand area. The tests proved that using unnecessary nodes in the system affect the quality of solution and delay the GA convergence. The ability of the water network to deliver sufficient water quantities to demand areas depends on the required amount by each demand area. The results from the second set of experiments proved that sufficient amounts of water delivered to demand areas. It was also concluded that smaller total demands performed better.

Mutation rate in this work was applied to three aspects: 1) individuals, 2) node positions, and 3) connectivity and head components of the chromosome. In testing different mutation rates, world 3, 10 nodes, and 1,000 individuals were used. The best mutation rates were: 10% of individuals, 5% to connectivity and head components, and 20% of nodes for their positions.

Adjacency mutation was tested for head values and node coordinates. Different values were used in this test. All values in both tests performed in a similar manner.

The tests using the special world layout demonstrated that the water network is able to avoid expensive passages with different layouts.

- **Objective 8**

  The model is applied to a hypothetical real-world, Kawar Town, and found the current GA formulation copes with current and future city development. The real-world case study was conducted with the objectives of cost minimisation and water delivery maximisation. The model was shown to improve upon a preliminary engineered design for the current WDN in terms of cost and water delivery. Furthermore, the model was rapidly able to find feasible solutions to join developed area to the current network, and exhibited an acceptable running time, which warrants its use in practice.

8.2.2 General Conclusion

Besides the specific conclusion to objectives, additional conclusions can be extracted from this work. These conclusions are useful to researchers working in the field of WDNs and GAs. These conclusions can be summarised as:

1. An initial problem formulation and testing is recommended as it has been done in chapter 4. This is helpful in understanding the model behaviour on its initial stages in order to assist further model development.

2. For the complex problem presented in chapter 6, it was found that single-point crossover was not effective. This is because the chromosome has many components and single-point crossover is not helpful to exchange the required genetic materials for the population diversity.

3. Initial population: one weakness of the method is the effort required to generate feasible starting solutions on which the evolution program can operate.

4. Several crossover types were presented in section 5.4.4 and tested. The best crossover is one that allowed all chromosome components to crossover.

5. Several mutation types were presented in section 5.4.5 and tested. The best mutation type is one that allowed to permute all chromosome components.

6. One of the difficulties of designing the 'worlds' was setting the land access cost. Setting land access cost for the individual land-use of a world was achieved in a reasonable manner and scale. It was difficult, however, to value these scaled elements against each other. As an example, the access cost for public open space (POS) and residential detailed in section 7.4 for the initial set of experiments in chapter 6 can be argued as reasonable and logical within their own scale. However, the figure used for residential is generally of a higher order than the one used for other POS. It must be considered whether this difference in scale is appropriate.

7. The cells within the raster grid were of constant size, and its scale depended upon the size of the smallest world feature to be modelled. A broad grid size means that they can be inaccurate when considering fine world feature details.

8. One of the underlying problems of GA formulation was all individuals in the population becoming identical. To overcome this problem over successive generations, 'Age Structure' was introduced. The age structure is

explained in section 5.4.7 and found to be useful, to some extent, although not tested thoroughly.

9. The fitness function balancing coefficients are selected by the decision-maker such that different weights can be given to whichever criteria are used. These coefficients are useful when some of the criteria are more important than others such that more weight will be given to the desired criteria.

   By making some equal to zero, it was possible to consider sub-sets of the general problem and to examine the effects of each of the aspects of the fitness function.

8.3 Further Work

A number of avenues for further research are proposed to extend the efficiency and effectiveness of the optimisation methodologies and software components introduced and their application to the wider domain of WDN applications. Further work can be conducted on:

1. There are many ways in which the WDN design optimisation model may be extended to match the requirements of a real world WDN design more closely. It may be extended to include additional objectives such as:

   (a) Maximise water quality: Water quality could be included in the optimisation model considering water quality monitoring stations or calculate the minimum and maximum residual chlorine at each node in the system.

   (b) Maximise reliability: To achieve a degree of redundancy in the pipe network optimisation, the layout should have appropriate loops as they provide alternative flow paths in the case of pipe failure or for maintenance purposes.

2. The lack of an efficient process to generate initial population limits, to a certain extent, the practical utility of the approach, and further research should be conducted in this area.

3. To plan and design a WDN effectively, criteria should be developed and adopted by planners and designers against the adequacy of existing and planned systems. One of these criteria is the water source. In determining the adequacy of water source, it must be large enough to meet various demand conditions. It is a good practice to consider more than one source of supply in the model to provide reserve supply for emergency needs and to add an additional reliability to the system in case of malfunction or maintenance that may result in insufficient supply.

4. Water distribution works include common requirements in the industry such as distribution and equalising tanks, pipes, valves, and pumps. The WDN design optimisation in this thesis did not include the design of tanks, valves, or pumps. The model may be extended to include these design variables.

5. The 3-D surface model is still primitive (see section 5.4.1.6) and need to be improved for future model development. To evaluate the influence of different 3-D surface, surface maps with varying parameter, e.g. mountains

and valleys, values are essential. Considering the difficulty of 'manufacturing' part with desired surface topography variations, it is necessary to generate the surface with the desired 3-D parameters.

6. Knowledge-based operators and their rates are probably going to help solve real-world problems. This knowledge helps forming of and access to experience that may create new capabilities of the current GA model.

   The formation of adjacency operators still a problem to be improved to produce better results. The problem is how all the global optimum can be located, while avoiding the local optimum which may have only a slightly lower fitness. Ultimately, if the fitness function has many local optimums, no search technique is going to perform well on it. Better methods for designing fitness functions are needed, which can avoid such problems.

7. GA populations should be closely studied as they evolve. Measures should be taken to encourage population diversity.

8. Further research to heuristic optimisation methods should be conducted on hybrid methods, which combine the specific advantages of different optimisation approaches. The hybridisation will create a process capable of escaping the local optima and performing a robust search of a solution space (Glover et al., 1995).

9. There are many decision rules may be incorporated in WDN design optimisation model which may worth further investigation. These rules might include:

   (a) A decision to the amount of allocated water to demand areas with respect to their distances to the water source.

   (b) A decision to distribute the water into pipes according to the length of the pipe and the available head at upstream node.

Appendix A

An Introduction to Genetic Algorithms and GIS

A.1 An Introduction to the Appendix

This supplementary appendix introduces techniques and tools used to assist in the design of water distribution networks. It covers two main topics, genetic algorithms (GAs) and geographic information system (GIS).

The first part of the appendix starts with an introduction to genetic algorithms followed by GA encoding. GA operators and basic GA structure are presented.

The second part starts with an introduction to geographic information systems. An introduction to GIS models, which consist of raster and vector models are presented followed by an introduction to GIS data, which briefly introduces GIS data types. The second part also presents an overview of GIS applications followed by a summary to GIS.

A.2 Genetic Algorithms (GAs)

A.2.1 An Introduction to Genetic Algorithms

Genetic algorithms (GAs) are search techniques used numerically for optimisation to solve complex problems. They are based on the mechanics of natural selection and natural genetics. These adaptive approaches derive their inspiration from the natural process where biological organisms evolve.

The first GA model proposed by John Holland in 1975 (Holland, 1975). He developed the 'Schema Theorem', which is a template that identifies a subset of strings with similarities at certain string positions (See Steeb (2005) or De Jong (2006) for details about schema theorem). He also identified the mathematical basis for the operation of the algorithms in terms of schema theory and the basis for the selection and crossover of genetic materials (chromosomes) representing problem solutions.

The implementation of GAs in this work is described in more detail in chapter 5. In addition, definitions to some GA-related terms can be found in appendix B.

A GA encodes candidate solutions and starts with a population of randomly selected chromosomes. These population advances to better chromosomes during each successive generation by applying genetic operators (Selection, Crossover, and Mutation), details about these operators are presented in section A.2.3. GAs operate on a population of potential solutions by applying the principle of survival the fittest individual to produce better approximations to a solution. In addition, chromosomes are evaluated according to their ability to act as solutions to problems. Upon this evaluation, new chromosomes are formed to work as a new population. Fitness function must be considered for each problem by returning a single numerical fitness value (Herrera et al., 1998). The key is that the probability that an element is chosen to reproduce is based on its fitness. Eventually, unfit elements die from the population, to be replaced by successful solution offspring.

GAs are best suited to solve optimisation problems with large solution spaces. However, GAs generally require a high number of function evaluations compared with traditional optimisation methods. The most important advantage is that they are multiple solution algorithms. Other optimisation algorithms begin with one starting solution and iteratively change that solution to produce a single final solution. A genetic algorithm is unique because it begins with a group of solutions, and iteratively changes the entire group to produce a final generation of "good" solutions. A solution from the final generation might be excellent in one objective, yet not as good in the other objectives. The final generation gives decision-makers the opportunity to explore the trade-offs in the objectives before choosing a single solution.

GAs have proven to be successful and efficient in identifying optimal parameters in many applications. Section 3.4 discusses the optimisation Using GAs and demonstrates some GA application.

Computers can be used to quickly analyse thousands of alternatives in the optimisation process, otherwise the process may be implemented manually. The computers are used to automate the execution of GAs to search the design space for the best solutions.

A.2.2 GA Genotype Representation (GA encoding)

To solve a problem using GAs, suitable encoding and operators must be applied to ensure the robustness of the solution (Haupt and Haupt, 2004). Encoding is the core issue in GA concept, which is known to be very important in genetic search. GAs are able to solve real-world problems if they have been suitably encoded (Haupt and Haupt, 2004).

The distinction between genotype and phenotype has a physical meaning and is not, however, necessary for evolution. In genetic terms, genotype is a set of genes (chromosome) possessed by an individual organism. Phenotype (search space) is the observable properties of an organism. It is produced by the genotype in conjunction with the environment. Figure A.1 illustrates the relationship between the genotype and the phenotype.

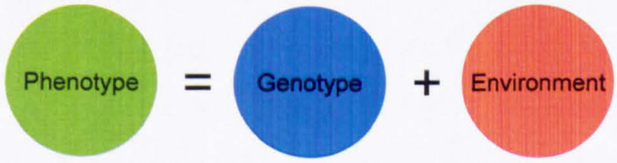


Figure A.1: The Relationship between Phenotype and Genotype

Encoding is a symbol structure that stands for a parameter in evolutionary computation methods. It means representing the potential solutions to contain a set of decision variables as strings of codes. Thus, the interpretation of a gene is a single character in the string coding. In addition, most of the recent research works on GAs still focus on encoding and representing the required solution as fixed length character strings (fixed chromosome length) (Coello and Lamont, 2005; De Jong, 2006).

The encoding mechanism depends on the problem variables. However, one of the preferable methods to represent GAs for optimisation problems is 'Binary String' representation method, which is a linear string representation. The reason behind using this method is that binary alphabet offers a maximum number of schemata (0 or 1) per bit if compared with the other coding techniques (Coley, 1999; Coello and Lamont, 2005; De Jong, 2006). Theoretical results showed them as the most appropriate representation for GA research for specific problems (Radcliffe, 1992; Coello and Lamont, 2005).
For example, if the problem is to optimise a function of three variables, *f(x, y, z)*, then each variable can be represented by a 10 bit binary number. The chromosome would therefore contain 30 binary of 0 or 1 digits. The chromosome length (L) of bit strings can be described as a 'schema'. the details of schema theorem can be found in Steeb (2005) and De Jong (2006). Figure A.2 shows an example of twenty bit strings to form a chromosome of five genes.

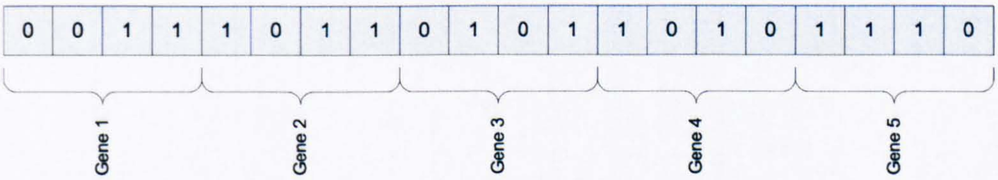


Figure A.2: Example of a Chromosome Using Binary Strings

There are 2^L possible bit strings in the length (L) for a population of (n) chromosomes. Thus, this representation will maximise 2^L and will attain maximum processing efficiency (Radcliffe, 1992).

However, binary representation is a problem dependent and encounters difficulties dealing with continuous search spaces with large dimensions and when big numerical precision is required. The problem rises when a variable has a finite number (which is not a power of 2) of discrete values. In addition, redundancy of binary codes in the solution may occur, and binary strings will not correspond with any actual orientation. Consequently, larger population sizes will be forced to reduce the efficiency of the solution. Thus, there are certain problems, which require other representations to be used.

On account to this, for other types of problems there are other types of representation can be used in GAs such as non-binary representation (see figure A.3 and A.4). Non-binary representation maybe a vector of integers or real numbers (figure A.3) so that each integer or real number represents a single parameter (Herrera et al., 1998; Skok et al., 2002; Mawdesley and Al-Jibouri, 2003).

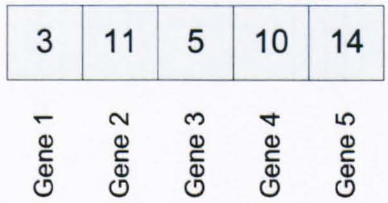


Figure A.3: Example of a Chromosome Using Integer Values

The gene values of the chromosome shown in the above figure is similar to the gene values shown in figure A.2.

To this end, the most suitable representation is the real number representation which works appropriately in optimising problems with variables in continuous search spaces (Herrera et al., 1998). Therefore, the chromosome representation will be a vector of floating point numbers (vector length) which is the solution to the problem.

Integer and real number chromosome representations are used in this work. For more details about this representation, see section 5.4.2.

To represent various GA problems depending on the problem's variables, there are other types of encoding such as alphabet, alphanumeric and Gray coding as shown in figure A.4.

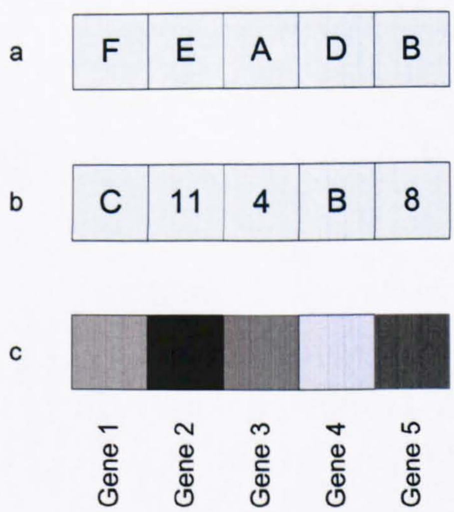


Figure A.4: (a)Alphabet Encoding (b) Alphanumeric Encoding (c) Gray Encoding

In practice, the only way to choose a representation over the other is to look at what other researchers have tried with similar problems and then choose a sensible approach that has the ability to code up and solve the problem efficiently (Coley, 1999; Haupt and Haupt, 2004).

A.2.3 GA operators

The typical form of GAs involves three steps of operators: selection, crossover, and mutation.

A.2.3.1 Selection

Selection operator is the step that selects candidates from a population for reproduction. It adopts the rule of 'survival of the best' by allocating more quotas of copies to fitter solutions.

There are different methods of selection such as roulette wheel selection (Mitchell, 1996; Coley, 1999; Haupt and Haupt, 2004), boltzmann selection (Maza and Tidor, 1993; Lee, 2003), tournament selection (Blickle and Thiele, 1995; Haupt and Haupt, 2004), and steady state selection (Whitley, 1989; Syswerda, 1991; Obitko, 1998).
The most common selection method for implementation is roulette wheel selection (fitness proportionate selection) (Mitchell, 1996; Herrera et al., 1998; Coley, 1999; Mawdesley et al., 2002).

In the fitness proportionate selection, the parents are selected according to their fitness. Thus, the fitter the candidate, the more times it is likely to be participated in the reproduction process. The concept of this method is to give each individual a slice of a circular roulette wheel. The area of each slice represents the value of the individual fitness. The wheel is then spun, and the ball comes to rest on one of the slices and thus to the fitness of the corresponding individual. This approach is illustrated in a typical example in figure A.5. This example present a population of n = four individuals (A, B, C and D).

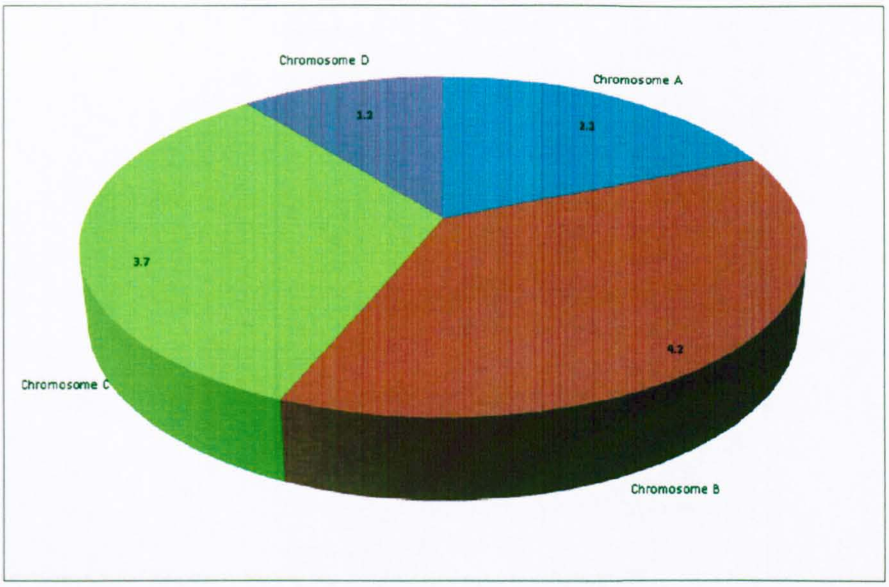


Figure A.5: Typical Example For Roulette Wheel Selection

The circumference of the wheel is given by the sum of the population fitness values and the ball represents a random number between *zero* and the sum of the population fitness values. The algorithm can be summarised as:

(Sum) Calculate the sum (S) of fitness of all population individuals.
(Randomise) Generate a random number between *zero* and S.
(Loop) Go through the population and add fitness of the population members (one at time) from *zero* to sum (S).
Notably, step 1 is performed only once for each population. For the example above ($n = 4$), the wheel will be spun four times. The first two spins would choose two chromosomes (might be B and C) to be parents; and the second two spins might choose B and D to be parents.

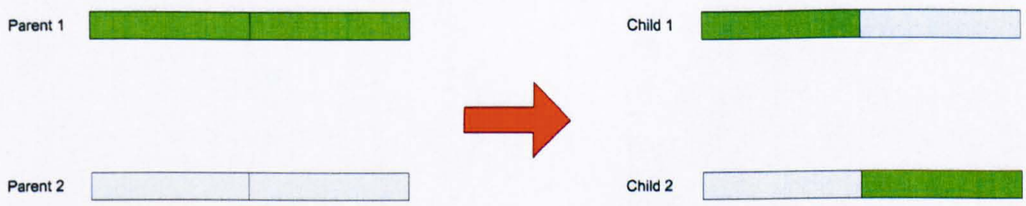
When a new population is created, there is a big chance to lose the best chromosome in which roulette wheel selection method does not guarantee the selection of any specific individual, including the fittest, in the population. This concept is called "Elitism" (Obitko, 1998; Coley, 1999). Elitism ensures that the chromosome with the best fitness value is passed to the next generation.

A.2.3.2 Crossover (Recombination)

Crossover or recombination operator chooses a location on each parent and exchange between them to produce two new offspring (children). First, the population after selection is divided into groups, each of which contains two chromosomes. Then, the two chromosomes (parents) in each group swap segments of their codes with each other to create two new chromosomes (children). The crossover operator introduces new chromosomes to the population, and hence the possibility of having fitter chromosomes in the population increase. This cannot be achieved by selection.

There are many crossover methodologies, such as single-point crossover, multi-point crossover and uniform crossover. The most commonly used crossover methodologies for standard GAs are single and multi-point crossovers (Mitchell, 1996; Coley, 1999).

Figure A.6 shows an example of the most common ones (single and multi-point crossover). Single-point crossover is the simplest form of this operator. A single-point crossover chooses a single locus (point) within the strings and exchanges the information between the two individuals as shown in figure A.6(a). Multi-point crossover (figure A.6(b) uses multiple loci (points) within the strings and swaps the information between the two individuals. As a result of this operator, the offspring contain code information of both parents.



a) Single-point crossover

b) Multi-point crossover

Figure A.6: Single and Multi-Point Crossover

It can be seen from figure A.6, crossover operator works when parent 1 passes its code strings to the left of that crossover point to child 1. Similarly, parent 2 passes its code strings to the left of the same crossover point of child 2. Then, the code to the right of the crossover point of parent 1 goes to child 2 and parent 2 passes its code to child 1. The process continues until all chromosomes locations are exchanged.

Some researchers used single-point crossover in their work (Adeli and Cheng, 1993; Hasancebi and Erbatur, 2000). Other researchers used multi-point crossover because it gives an improvement to GAs by performing a thorough search through the problem space (Beasley et al., 1993*a*; Glover et al., 1995; Haupt and Haupt, 2004). However, further crossover points reduce the performance of the GAs.

Syswerda (1989) introduced the uniform crossover in which the two children are produced by selecting at random, at each gene locus, from which parent the gene value should be copied. As with the other methodologies, the two resulting children retain all the genetic information from the parents it had been transferred in part to one child or the other.

A.2.3.3 Mutation

To ensure that the solutions in a population do not become stuck at a non-optimal solution, a randomisation element is introduced, which is known as 'Mutation'. Mutation operator randomly alters the value of a single bit within the individual strings. It forces the algorithm to search new areas and adds diversity to the population.

Mutation might create a better or worse chromosome, which will either thrive or diminish through next generations. Should it prove helpful, these children are more likely to survive to reproduce. Should it be harmful, these children are less likely to reproduce, so the bad trait will die with them.

In this operator, the new string is produced from a single old string. For example, a single-point mutation in an integer representation permutes a value in the chromosome as shown in figure A.7.

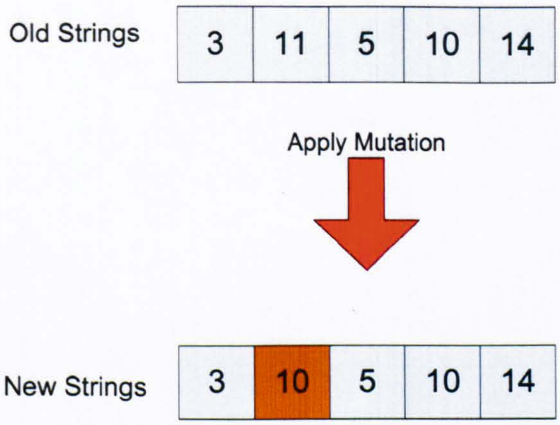


Figure A.7: An Example of Mutation Operator

Various forms of mutation are commonly used, varying from those which mutate individual bits of binary strings, randomise variables with their domain or adjacency operators that push gene values to adjacent values. Several mutation types are developed and tested in this work. These types are discussed in section 5.4.5.

A.2.4 Population Size

Selecting a GA population size is difficult due to many possible variations in the algorithm and the fitness function. The population size determines the amount of information the GA can exploit. Each member of the population is a solution, and each solution is a point in the search space.

A larger population means the GA's view of the search space is more detailed. In other words, larger populations are likely to give better final solutions. Many publications concluded that very small population sizes (e.g. below 20) have poor behaviour while larger population sizes are required to produce better solutions (Coley, 1999; Mawdesley et al., 2002; Reeves and Rowe, 2002; De Jong, 2006). On the other hand, maintaining a large population takes a lot of computational effort. A trade-off has to be made between the quality of final solutions, and the amount of computation one is willing to do.

Generally, two different methods are used to set the population size for a specific run of the GA (De Jong, 2006):

1. Reckoning: The user decides of how large the population size should be, simply based upon experience when using the GA (Haupt and Haupt, 2004; Beasley et al., 1993*b*). This is the most common method (van Dijk et al., 2002).

2. Analysis: An analysis may be carried out by modelling the dynamic interactions of the algorithm. The user can find a population-sizing equation which gives recommendations on the size of the population. Examples of this method are in studies by Goldberg et al. (1993) and Cantú-Paz and Goldberg (2000).

   While this seems to be the ideal solution, doing the analysis is very hard due to the complex behaviour of genetic algorithms. As a result, analysis has so far only been done of GAs for artificial problems with known properties.

A.2.5 Basic GA Structure

The basic genetic algorithm is implemented by selecting a number of individuals from a population; recombining them in some fashion to produce a number of offspring; introducing some mutation factor; evaluating the resultant offspring with respect to their fitness as solutions for the problem at hand and finally reintroducing (or replacing) the individuals into the base population.

The flowchart in figure A.8 illustrates the basic operation of a GAs through these repeated cycles of selection, recombination (crossover), mutation and replacement. It is also necessary to consider at what stage the operation of an algorithm should be terminated.

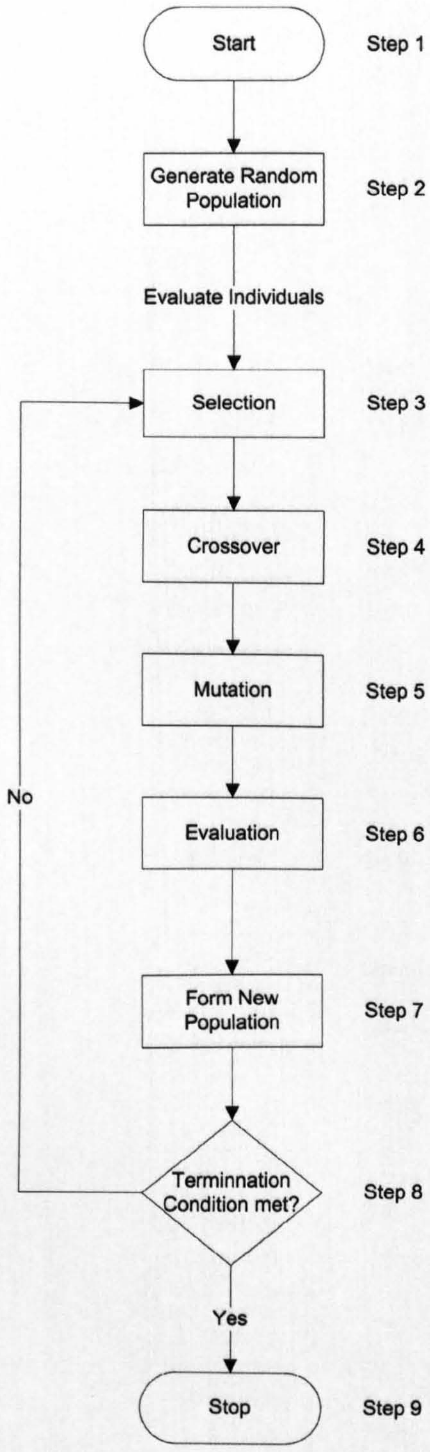


Figure A.8: Basic Genetic Algorithm Operation

In GAs, solutions are parametrically represented in strings of code (e.g. numeric), and a population of possible solutions is created (step 2). Each individual in this population is encoded into a chromosome to be manipulated by the genetic operators. After the individuals are created, they will be evaluated. Based on each individual's fitness, a selection mechanism (step 3) chooses pairs to be parents for the genetic manipulation process.

The manipulation process applies the crossover (step 4) and mutation (step 5) to produce a new population of individuals.
Next, each offspring candidate is evaluated according to its fitness value (step 6).
The generated offspring take the place of the older population to form a new population (step 7).
Step 3 to step 7 is repeated until a desired level of fitness is attained or a determined number of generations is reached (step 8).

Basic GA pseudo-code can be written as:

Table A.1: Basic GA Pseudo-Code

```
Basic Genetic Algorithm()
{
          Initialise population;
          Evaluate population;
          While termination criterion not reached
          {
                Select solutions for next population;
                Perform crossover and mutation;
                Evaluate population;
                Form new population;
          }
}
```

A.2.6 GAs Summary

The appendix introduced GAs and discussed the possible representations of GA variables. GA operators are also discussed.

This appendix also discussed the structure of a standard GA which starts with a random population of individuals. In every generation, the individuals in the current population encode a possible solution to a specific problem and evaluated according to some predefined quality criterion, referred to as the 'fitness function'. GA combines selection, crossover, and mutation operators in order to find the best solution to a problem. Finally, GA can handle problems with different degrees of complexity, different practical requirements, and user constraints.

The next part of this appendix will introduce geographic information systems (GIS). GIS is another tool used to assist in the design of water distribution networks.

A.3 Geographic Information Systems (GIS)

A.3.1 An Introduction to Geographic Information Systems

Geographic information systems (GIS) can be defined as the mapping information system that deals with geographically referenced spatial information. In addition, it integrates ideas and technology for a wide range of disciplines as well as science.

Computer applications can be used to store, retrieve, and analyse geographical information as computers can provide visual explanations to a project. Computers use a system of hardware, software, and procedures designed to work with spatially captured data to solve complex planning and management problems (DeMers, 2005).

In early 1960s, although computers and operating systems were primitive and required large physical computer resources to run, GIS started to develop when computer graphics were first founded and developed (Schuurman, 2004).

The concept of 'overlay' was introduced in the USA to become an essential methodology of GIS and become the basis of spatial analysis technique. In addition, computational methods to analyse large geographical dataset started to be explored.

GIS technology has been developed rapidly over the past two decades, and it is considered now as an essential tool for the effective use of geographical information. Conventional geographical maps show information related to areas such as emergency and disaster management, distribution of water networks and country boundaries. GIS uses geography to link information to location data, such as linking people to addresses and buildings to parcels (Heywood et al., 2006).

Geographical data are extracted and collected from aerial photos, visual descriptions, surveying, and statistical data. Although GIS data include a wide range of digital sources, data collection process is usually the same as those used for traditional cartography.

GIS is similar to an overhead scheme which consists of series of transparencies laid on one another so that different thematic information is stored by a set of independent layers as shown in figure A.9. For example, using layers of river streams, streets, and houses can produce a new GIS map layer.

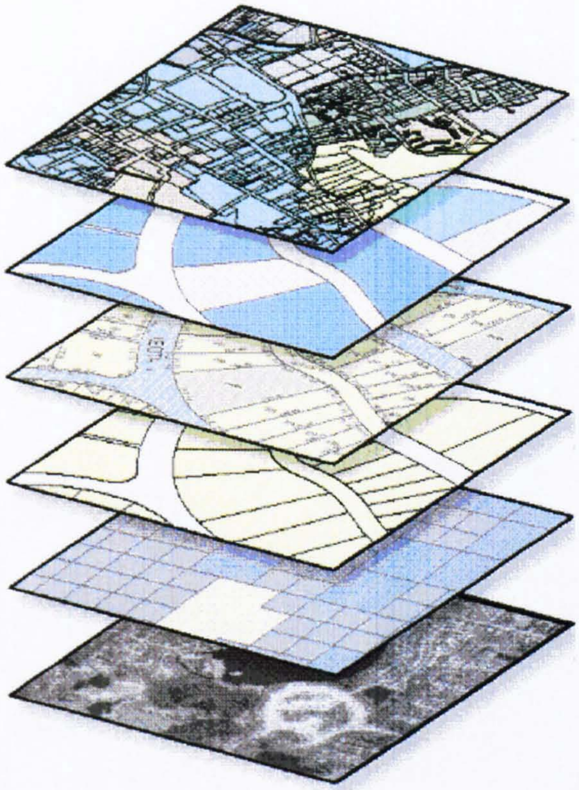


Figure A.9: GIS Layers

GIS information can be stored as layers and these layers can be on or off depending on the tasks need to be performed. To exemplify this, a theme of roads in a specific region may contain information about their locations, names, and lengths. Another theme may contain wetland areas in the same region and another could represent all the cities in the region. On account of this, each theme can be turned on or off making all the elements of the three themes appear or disappear with one mouse click or one touch on a keypad.

A.3.2 GIS Models

GIS has two different types of geographic data models, 'raster' and 'vector' models. These models are necessary to accommodate object interactions in the database and to link different data types.

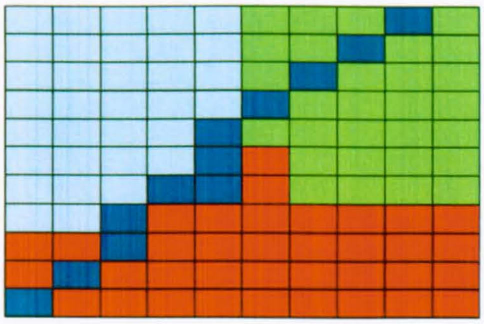
A.3.2.1 Raster Models

Raster models are scanned images, which represent the features of a continuous surface. Each cell in the continuous surface matrix represents a square on the ground. This means that data are characterised by a grid of picture pixels. Each pixel assigned a code representing some characteristic (colour or number), and the pixel position correlates to its geographic position (Chang, 2003; DeMers, 2005) (see figure A.10).

Raster model can be used to separate individual types of objects (e.g. water, parks) to identify their respective attributes. Examples of raster data used in GIS are satellite images and terrain surfaces with different heights.

4	4	4	4	4	2	2	2	1	2
4	4	4	4	4	2	2	1	2	2
4	4	4	4	4	2	1	2	2	2
4	4	4	4	4	1	2	2	2	2
4	4	4	4	1	3	2	2	2	2
4	4	4	4	1	3	2	2	2	2
4	4	4	1	1	3	3	3	3	3
4	4	1	3	3	3	3	3	3	3
3	3	1	3	3	3	3	3	3	3
3	1	3	3	3	3	3	3	3	3
1	3	3	3	3	3	3	3	3	3

a) Number raster representation



b) Colour raster representation

Figure A.10: An Example of Raster Model Representation

Figure A.10(a) represents some attributes characterised by numbers while figure A.10(b) represents the same attributes characterised by colours.

Raster models can be evolved to include direct links to existing database management systems. The data in table A.2 are attribute information extracted from a raster model shown in figure A.10. This table shows that number 1 refers to river cells (blue), number 2 (green cells) could represent a green park, number 3 (red colour) could represent a residential area and Cyan colour (number 4 cells) could be a retail area. In addition, the table may contain more information about the attributes such as the dimensions of a geographical area.

Table A.2: An Example of Raster Data File

Attribute number	Attribute colour	Cell count	Type
1	Blue	12	River
2	Green	29	Green parks
3	Red	36	Residential
4	Cyan	33	Retail

A.3.2.2 Vector Models

Vector models represent real-world objects as precisely as possible by storing points (nodes), lines (edges/arcs) or areas (faces/polygons) in a continuous coordinate space (Heywood et al., 2006; Chang, 2003; Schuurman, 2004).

The representation is described by feature codes, associated attributes and geographic (x, y) coordinates as shown in figure A.11. This figure and the table A.3 work as an example where the location of an electric pole, which is a point (entity A), can be described by a single (x, y) coordinates; roads and rivers which are linear features (entity B) can be stored as a collection of point coordinates and property boundaries can be stored as a closed loop of coordinates (entity C).

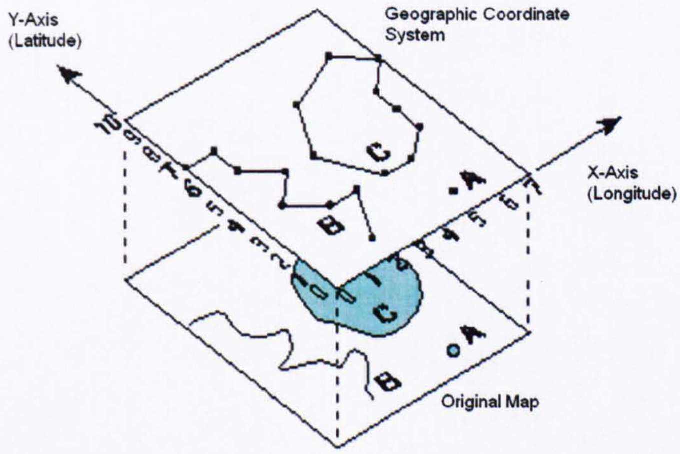


Figure A.11: An Example of a Vector Model

Table A.3: An Example of Vector Data File

Entity	(x, y) Coordinates	Name
A	5,1	Electric pole
B	0,7; 1,7; 1,6; 2,5; 1,4; 2,3; 3,3; 2,1	Stream
C	4,3; 5,3; 6,4; 6,5; 6,6; 7,7; 6,8; 4,7; 3,5	Property

The sequence of spatial data reconstruction can be realised within GIS so that satellite images can be converted to a vector structure by generating lines around all areas with the same classification. For example, GIS can analyse land-use (raster data) in conjunction with the property ownership information (vector data). Thus, an effective analysis can be performed using a mix of both raster and vector data models. Both vector and raster models for storing geographic data have unique advantages and disadvantages. Most common GIS software packages are able to handle both models, although the majority will concentrate on one of them. Figure A.12 presents raster and vector layers, which can be used as effective analysis tools.

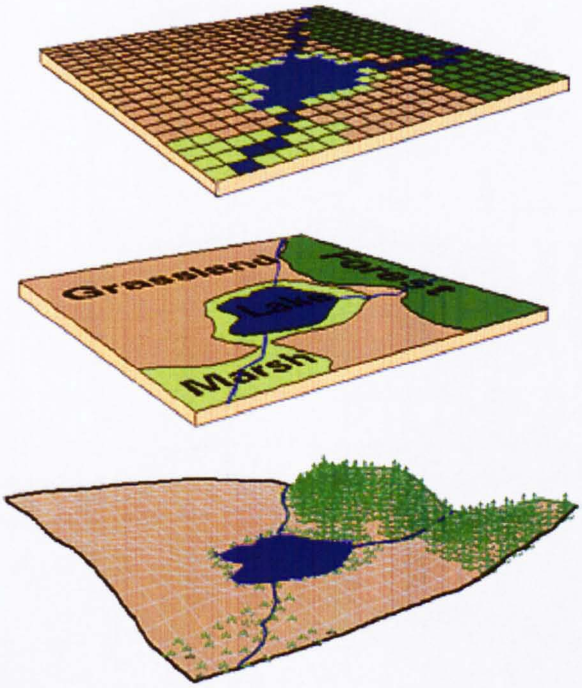


Figure A.12: GIS Models

A.3.3 GIS Data

Data are the most important part of GIS. More than 80% of a project total duration is spent on the data collection activities to establish a GIS project, especially large-scale projects (Chang, 2003; Heywood et al., 2006). GIS should be able to perform many functions regarding data such as data input, storage, management, analysis, and output.

The process of gaining data is costly and time consuming. This process starts with collecting the required data. The data should be converted from its existing form to one that can be used and stored by GIS (Heywood et al., 2006). For example, census data about population represent one form of data collection. Also, data can be collected using various measurement tools and techniques such as collecting precipitation data using rain gauges. In addition, spatial data may be extracted from ground measurements on the ground using global positioning system (GPS), satellite images or by digitising existing maps (Schuurman, 2004).

GIS data are stored in tables. These tables are stored and organised in databases according to the data type. GIS databases may have dozens of tables and hundreds of attributes. GIS results can be displayed all or part on screen, saved in digital formats, or printed, when necessary on different scale papers.

Conventional GIS reporting subsystems can be in the form of maps, charts, tabular data, and statistical reports.

GIS data can be divided into two types: 1) spatial data, and 2) attribute data (thematic data).

A.3.3.1 Spatial Data

Spatial data represent spatial features on the Earth's surface as 2-D or 3-D image features and form the spatial frame of a GIS (Chang, 2003).

These data can be captured by map scanning, analogue map digitisation, close range photogrammetry and remote sensing, geodetic approaches or other GPS techniques (e.g. total station) (Schuurman, 2004). In addition, they can be converted to vector data and vice versa.

Models of spatial features represented by raster and vector models in which satellite images can be converted to a vector model by generating lines around all objects with the same type.

Nowadays, spatial data analysis is not an issue because raster and vector data can be displayed simultaneously using layers.

A.3.3.2 Attribute Data (Thematic Data)

Attribute data describe the characteristics of spatial features and attached to the defined spatial locations in GIS. These data are stored in a GIS database as ordinary files and considered as records of spatial objects (Schuurman, 2004). For example, if a region is identified to be a spatial object, then the population, domestic water consumption, temperature and land-use percentages might be attributes.

One of the useful key aspects of GIS attribute data is consistency of data collection and reporting practices. Moreover, GIS database users can establish tabular reports and use relational databases (also called "Relational Database

Management Systems" (RDBMS)) which can be utilised to data search, data retrieve as well as data editing process (Chang, 2003). For example, captured spatial data for a GIS scheme (city centre for example) may include (Heywood et al., 2006):

- Geographic Coordinates and Geographic References: latitudes, longitudes and height as a geographic reference for selected points in a city.
- Connection Details: such as roads and rails that allow people to reach the city (Linear Geo-referencing).
- Non-Spatial Data (Attribute Data): wind and temperature. These data are attached to the spatial locations so that different features can be described (points, lines and polygons).

Finally, it is important to point that data quality control includes both spatial and thematic data. In addition, quality control of spatial datasets is easier than quantifying the qualities of thematic datasets (Chang, 2003; Heywood et al., 2006). Hence, meta-data are used as a quality index of a GIS for both spatial and attribute datasets.

A.3.4 GIS Applications

The applications of GIS have been spread very fast over the past few decades, and it is difficult to demonstrate all the possible uses of GIS. However, Federal Geographic Data Committee (FGDC) in USA adopted a wide array of GIS topic categories provided within ISO 19115 standards (See table A.4) (Federal Geographic Data Committee, 2007).

Table A.4: Examples of GIS application categories (ISO 19115)

Ocean	Intelligence / Military
Boundaries	Water
Meteorology / Atmosphere	Economy
Elevation	Environment
Geoscientific	Imagery / Earth Cover
Health	Planning / Cadastre
Transportation	Utilities / Communication

GIS is not only applicable to ground surface features of the earth, but also it is used for a wide range of oceanography and other marine applications such as bathymetry and data about coastlines.

In addition, GIS can be used in other disciplines outside geography such as urban and regional planning. Planning process may include collecting and analysing GIS data about social, economical, educational, and political disciplines in a specific region. For example, census data of population live in a specific region can be easily incorporated with GIS and may include very rich information about mean age, personal income, employment and other related information.

GIS can answer many generic questions about particular features, geographical patterns, socio-economic and other spatial implications. Many problems in environmental, economic and natural science are spatially related; therefore, a combination of GIS and planning methods can benefit the decision-making process. Most decision-making problems solved within GIS are related to a search for suitable sites, routes and land-uses (Jankowski, 1995). For example, water supply systems require suitable sites for treatment plants so that the power, and the pipeline routes can be minimised.

GIS applications have the potential to enhance many disciplines because GIS has the technology and the science power to achieve the following goals (Chang, 2003; Shamsi, 2005; DeMers, 2005):

1. Save time and money.
2. Provide the power of integration.
3. Offer a decision support framework.
4. Provide effective communication tools.

When the data are organised, analysed, and interpreted, they then becomes information. In this stage, raster and vector data can be useful for the decision-making and planning process (Yeh, 1991).

Decision support systems (DSS) offer a wide range of techniques for decision-makers to incorporate decision-making into a GIS environment. Therefore, this incorporation will assist them in selecting the best possible alternative from a set of feasible alternatives using user defined priorities.

Finally, in order to solve planning problems, GIS provide powerful functions that strength many fields, especially for those involving in complicated spatial and statistical problems. Section 3.6 demonstrates the use GIS to assist in the design of WDNs.

A.3.5 GIS Summary

GIS is a system of hardware and software used for storage, retrieval, mapping, and analysis of geographic data. It can be used for scientific investigations, resource management, and development planning.

GIS models consist of raster and vector models. Spatial features are stored in a coordinate system (e.g. latitude / longitude), which references a particular place on the earth. Descriptive attributes in tabular forms are associated with spatial features.

Spatial data and associated attributes in the same coordinate system can then be layered together for mapping and analysis. GIS has many applications.

GIS technology can be used for earth surface based on scientific investigations, asset management and location planning, archaeology, environmental impact studies, infrastructure assessment and development, urban and regional planning, marketing, population and demographic studies, location attributes applied statistical analysis, and other purposes. For example, GIS can be used by a company to site a new business location.

A.4 Appendix Summary

The appendix introduced genetic algorithms (GAs) and geographic information system (GIS) with an overview to their applications. Both are considered as useful tools in the design optimisation of WDNs.

The first part of the appendix started with an introduction to GAs. They have proven to be useful tools in the design and implementation of many applications. GA encoding and operators are discussed followed by a presentation of basic GA structure.

This second part of the appendix introduced GIS and demonstrated main GIS components. Raster and vector models are discussed as well as GIS data types. An overview of some practical GIS applications are presented.

Appendix B

Definitions

B.1 Evolutionary Algorithms

Allele	An allele is the different possible states of a gene (or genes). The simplest example taken from biology is that the gene that determines eye colour. The alleles of this gene are the different colours (Brown, Green, Blue ... etc).
Children (offspring)	are the new individuals which are usually produced by a pair of parents.
Chromosome (Individual)	is a set of genes which represents the individual. Chromosomes contain the solution to a problem in the form of genes.
Convergence	is the progression towards increasing uniformity. A gene is said to have converged when 95% of the population share the same value (De Jong, 1975). The population is said to have converged when all of the genes have converged.
Diversity	is the variety and abundance of organisms at a given place and time.
Elitism	it copies the best chromosomes from the current population to the new population without undergoing recombination ensuring that they are preserved from generation to generation. The rest is done in classical way.
Evolutionary Algorithm (EA)	is a collective term for all probabilistic optimisation and approximation algorithms that are inspired by Darwinian evolution.

Fitness Function	is a process which evaluates a member of a population and gives it a score or fitness which is the measure by which individuals are compared to each other to judge their relative performance for the optimisation problem.
Gene	is a subunit of a chromosome. Each gene represents a specific attribute that is encoded within the genome at a specific location known as a locus. This attribute normally represents a decision variable to be considered in the optimisation but can also convey other information specific to an organism.
Generation	is the time unit of the EAs. Generation refers to the interval during which the number of new individuals created is equal to the population size. Each iteration in GAs produces a new generation of possible solutions for a given problem.
Genetic Algorithms (GAs)	are sets of computational search algorithms that simulate the biological genetic process and can be used to optimise the solution of many problems in the real world.
Genetic Operator	is an operator in a genetic algorithm acts upon the chromosome to produce a new individual. Example operators are mutation and crossover.
Genome	is the set of all genes in an individual.
Locus	The locus of a gene is the position it occurs in a chromosome. The locus is usually fixed for a given gene. However, in certain types of GA with variable-length chromosomes the locus can vary.
Parents	are individuals (usually two) which have been selected from the population for reproduction due to their fitness.
Population	is a set of individuals exhibiting equal or similar genome structures, which allows the application of genetic operators.

Reproduction	is the production of child solutions from their parents ordinarily as a result of crossover and mutation or it may be used to mean an exact copy of the original parents.
Search Space	it is the space of all feasible solutions. Each point in the search space represent one feasible solution. Each feasible solution can be 'marked' by its fitness for the problem.

B.2 Water Distribution Networks

Flow Capacity	it is an upper, and sometimes lower, limit on the amount of flow rate in a pipe in a network.
Gravity Network	it a water network in which water is provided from a reservoir by gravity only to customers. Pressure in a water distribution system is typically maintained by a pressurised water tank serving an urban area. Gravity networks has a small pressure and need pumps to allow higher flows.
Link (pipe)	it is a component of a water distribution network that transports water from one place to another. Pipes has a uniform commercially available diameters.
Loop	it is a route in the distribution network that starts from a node and travelling only once along any of the connected links and ends at the same node.
Network Design	it is an optimisation problem to layout new pipes or extend an existing WDS ensuring that the new network or service meets the needs of the users and designers.
Network Calibration	it is an optimisation problem that compare the results predicted by the model with observations taken in the field. It is a two step process: 1) Comparison of pressures and flows; and 2) Adjustment of the input data for the model.
Network Rehabilitation	it is an optimisation problem that restores a WDS by rehabilitating portions of the system such as pipes (especially older, unlined, and metal pipes), pumps, valves, and reservoirs.
Node	it is a central or connecting point at which two or more links intersect or branch. It is a terminal point where a link begins and a link ends.
Pump Network	it a water network in which water is lifted from a reservoir by pumps to customers. The flow in pipelines is maintained by creating a pressure head by pumping.

Route	it is a course taken in getting from a starting node and termination at another node travelling only once through any link.

B.3 Networks (Graphs)

Complete Graph	it is a simple undirected graph. It has the feature that each pair of vertices has an edge connecting them.
Connected Graph	it is a graph where any two vertices are connected by some path.
Degree of a Vertex	is the number of edges connected to that vertex. The degree is not necessarily equal to the number of vertices adjacent to a vertex, since there may be more than one edge between any two vertices. A directed graph has both an in-degree and an out-degree for each vertex, which are the numbers of in-coming and out-going edges respectively.
Directed Graph	often called a 'digraph', is one in which each edge runs in one direction (such as a one-way road between two points), where $(v_0, v_1) \neq (v_1, v_0)$. Edges can have weights associated with them can be represented as arrows indicating their orientation.
Edge	it is drawn as a line connecting two vertices. Also called a bond (physics), a link (computer science), or a tie (sociology).
Graph (Network)	a graph consists of a set of objects, called vertices, with certain pairs of these objects connected by links called edges.
Graph Theory	it is the study of points and lines. In particular, it involves the ways in which sets of points, called vertices, can be connected by lines, called edges.
Multi-graph	it is a graph with multiple edges between the same vertices. A multi-graph may have several loops. However, not all authors allow them to have loops.

Path	is an ordered sequence of nodes in which there is one edge between consecutive nodes and its length is the number of edges contained in the sequence. A simple path is one with no repeated vertices and a cycle is a simple path except the last vertex is the same as the first vertex.
Planar Graph	it is a graph which can be drawn on the plane so that its edges intersect only at their end vertices.
Spanning tree	is a sub-graph that is a tree. normally a tree selected among the edges in a graph or network so that all of the nodes in the tree are connected.
Simple Graph	it is an undirected graph that has no loops and no more than one edge between any two different vertices. In a simple graph the edges of the graph form a set (rather than a multi-set) and each edge is a pair of distinct vertices.
Sub-graph	it is a subset of vertices and edges forming a graph. A connected component is a maximal connected sub-graph.
Tree	it is a connected acyclic simple graph.
Undirected Graph	it is one in which the pair of vertices in a edge is unordered and runs in both directions, where $(v_0, v_1) = (v_1, v_0)$. An undirected graph can be represented by a directed graph by having two edges between each pair of connected vertices, one in each direction.
Vertex	(plural: vertices) it is the fundamental unit of a network, also called a site (physics), a node (computer science), or an actor (sociology). A vertex is simply drawn as a node or a dot.
Weighted Network	It is a network in which the edges between two nodes has a specific value assigned to them. An example of this type of networks might be cities and their distances between them.

Appendix C

A Depth-First Search Example

This appendix illustrates how a depth-first search algorithm works by applying it on a simple graph. The following graph is an example of the search tree. It is a series of interconnected nodes that DFS will be searching through:

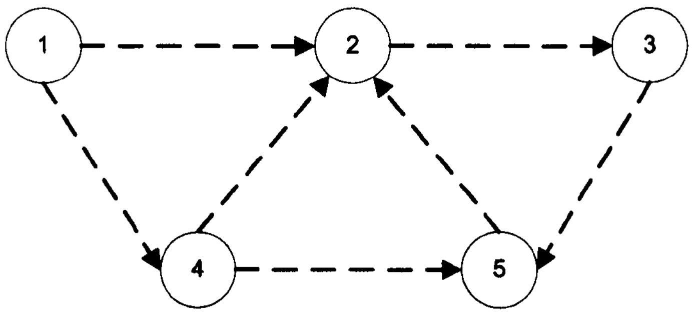


Figure C.1: Example Graph

There are a number of points can be made about the above graph:

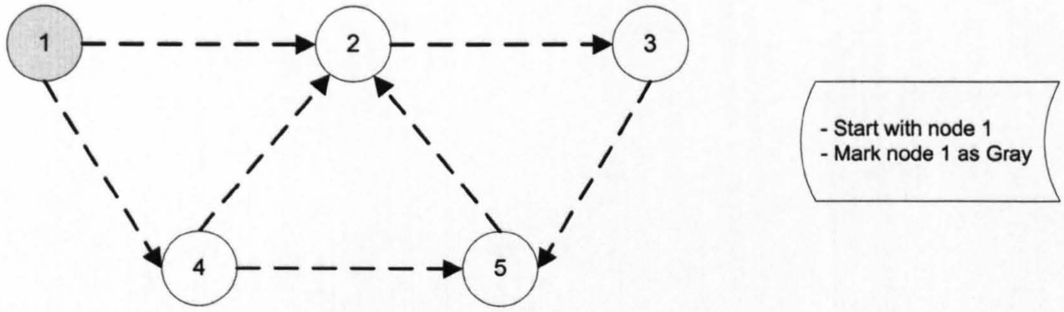
1. The path connections are one way (i.e. the above graph is a directed graph).
2. Each numbered circle in the graph is a node.
3. A node can be connected to another via an edge (dashed arrows), and those nodes that its connects to are called neighbours (e.g. node 2 and 4 are neighbours of node 1).

When possible, a depth first traversal chooses a node adjacent to the current node to visit next. If all adjacent nodes have already been discovered, or there are no adjacent nodes, then the algorithm backtracks to the last node that had undiscovered neighbours. Once all reachable nodes have been visited, the algorithm selects from any remaining undiscovered vertices and continues the traversal. The algorithm finishes when all vertices have been visited.

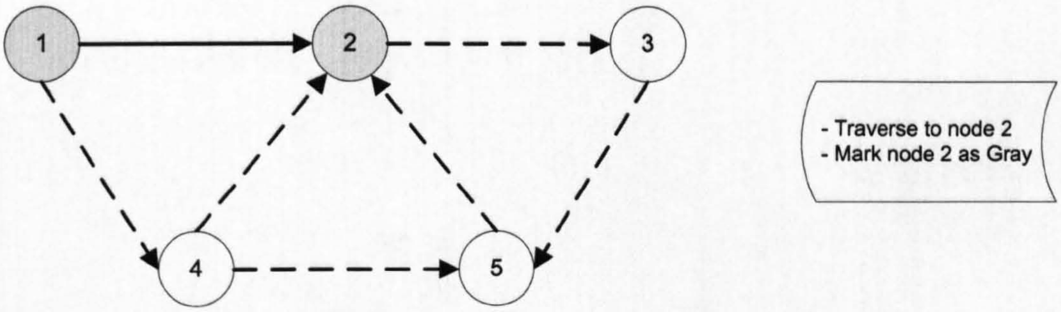
The state of a node is stored in a color marker as follows:

1. White - for the "Unvisited" state,
2. Gray - for the "discovered but not finished (VuF)" state, and
3. Red - for the "Visited" state.

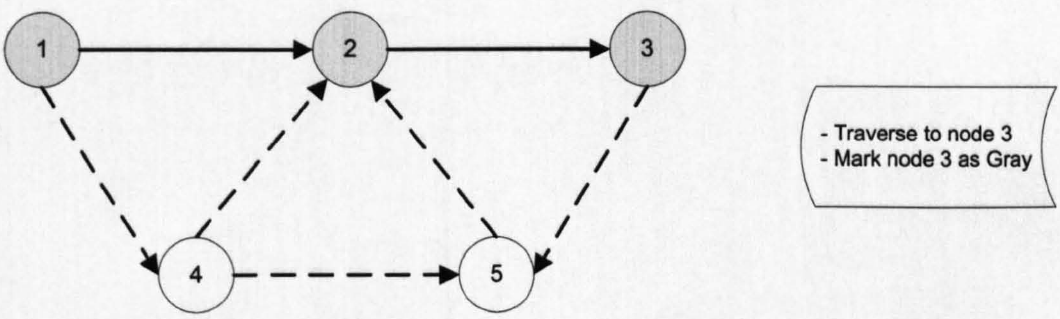
The graph shown in figure C.1 will be explored using the DFS algorithm, starting from the (arbitrarily chosen) node 1. The steps are depicted below:



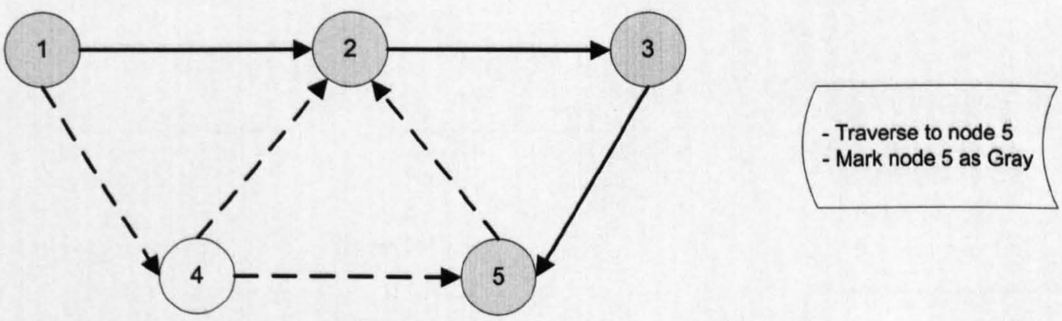
Step 1



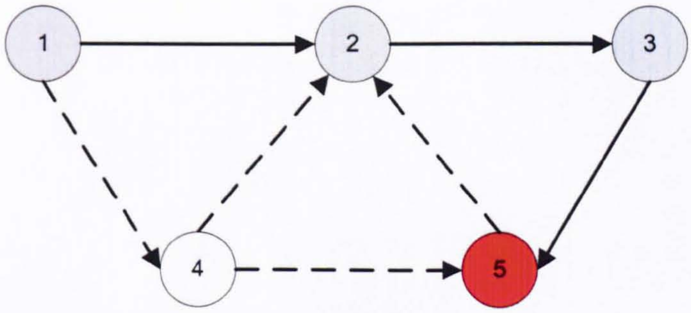
Step 2



Step 3

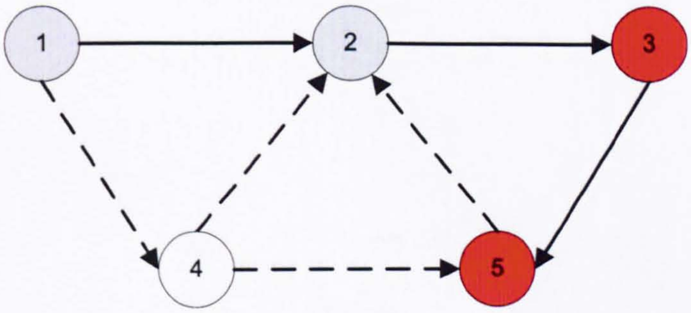


Step 4



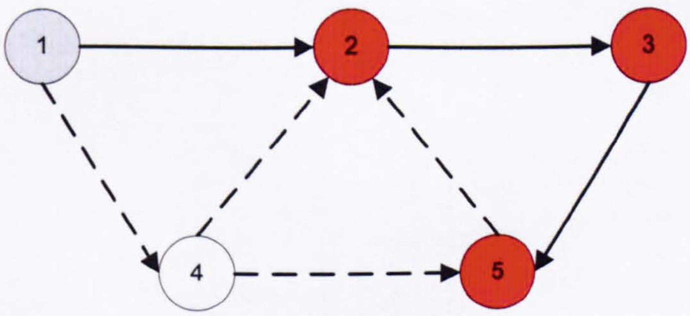
- No unvisited neighbours from 5
- Mark node 5 as Red
- Backtrack to node 3

Step 5



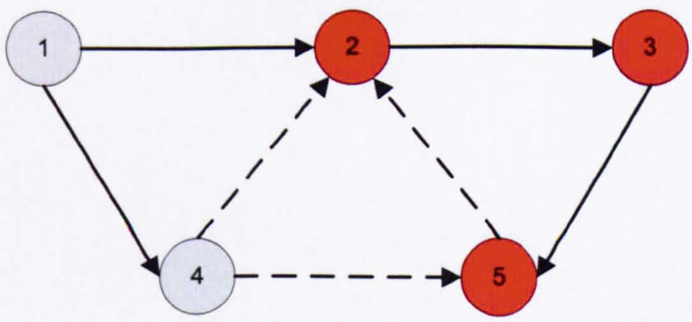
- No unvisited neighbours from 3
- Mark node 3 as Red
- Backtrack to node 2

Step 6



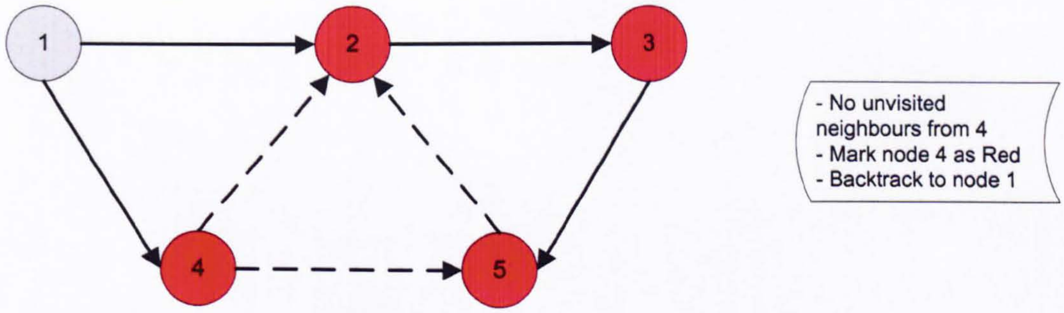
- No unvisited neighbours from 2
- Mark node 2 as Red
- Backtrack to node 1

Step 7

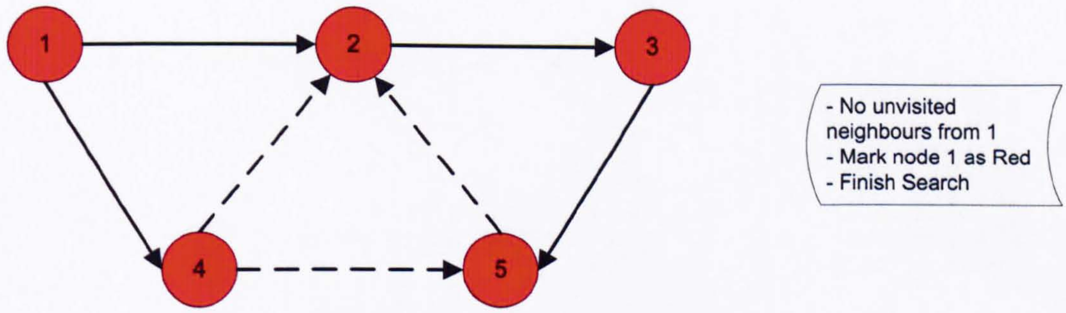


- Traverse to node 4
- Mark node 4 as Gray

Step 8



Step 9



Step 10

Figure C.2: DFS Example

The dashed arrows on the graphs are undiscovered paths.

Appendix D

The Computer Program

D.1 An Introduction to the Appendix

This appendix presents the main aspects of the computer model used to demonstrate the work carried out in this research.

Some of the model features and screen dumps specific to smaller models tasks were presented in chapter 5. The main prototype interfaces will be presented in this appendix. In addition, some of the important subroutines and functions will also be included and the relevance of these codes will be discussed.

D.2 An Introduction to the Model

No standard computer package is used to develop the GA model in this research. The final prototype program is constructed using Visual Basic .NET (VB.NET) programming language using Visual Studio 2010 (VS2010) developed by Microsoft Corporation.

VB.NET is fully object-oriented (OO) programming language (Microsoft Corporation, 2011*b*). Object-oriented programming is a type of programming in which programmers define not only the data type of a data structure, but also the types of operations (functions) that can be applied to the data structure. In this way, the data structure becomes an object that includes both data and functions. In addition, programmers can create relationships between one object and another. For example, objects can inherit characteristics from other objects. A specialized program known as a 'compiler' takes the instructions written in the programming language and converts them to machine language. This means that a Visual Basic programmer don't have to understand what the computer is doing or how it does it.

Microsoft Visual Studio 2010, which is a Windows development package, is a powerful Integrated Development Environment (IDE) that is used to develop console and Graphical User Interface (GUI) applications along with Windows forms applications, web sites, and web applications with managed code for all platforms supported by Microsoft Windows, Windows Mobile, .NET Framework, and other frameworks.

Coding in GUI environment is a transition to traditional and linear programming methods where the user is guided through a linear path of execution and is limited to small set of operations. In GUI environment, the number of options open to the user is much greater, allowing more freedom to the user and developer. Features such as easier comprehension, user-friendliness, faster application development and many other aspects such as introduction to ActiveX technology and Internet features make Visual Basic an interesting tool to work with.

Computer modelling means using a computer to 'model' situations to see how they are likely to work out if the user changes different model parameters. It is a simulation or model of a situation in the real-world or an imaginary world which has parameters which the user can alter. A detailed description about modelling water distribution networks is presented in section 2.4.9.

D.3 Model Description

Genetic algorithms provide computers with a method of problem-solving which is based upon implementations of evolutionary processes. The typical components of the developed prototype decision support system for the design of water distribution network and analysis is presented as a flow-chart in figure D.1.

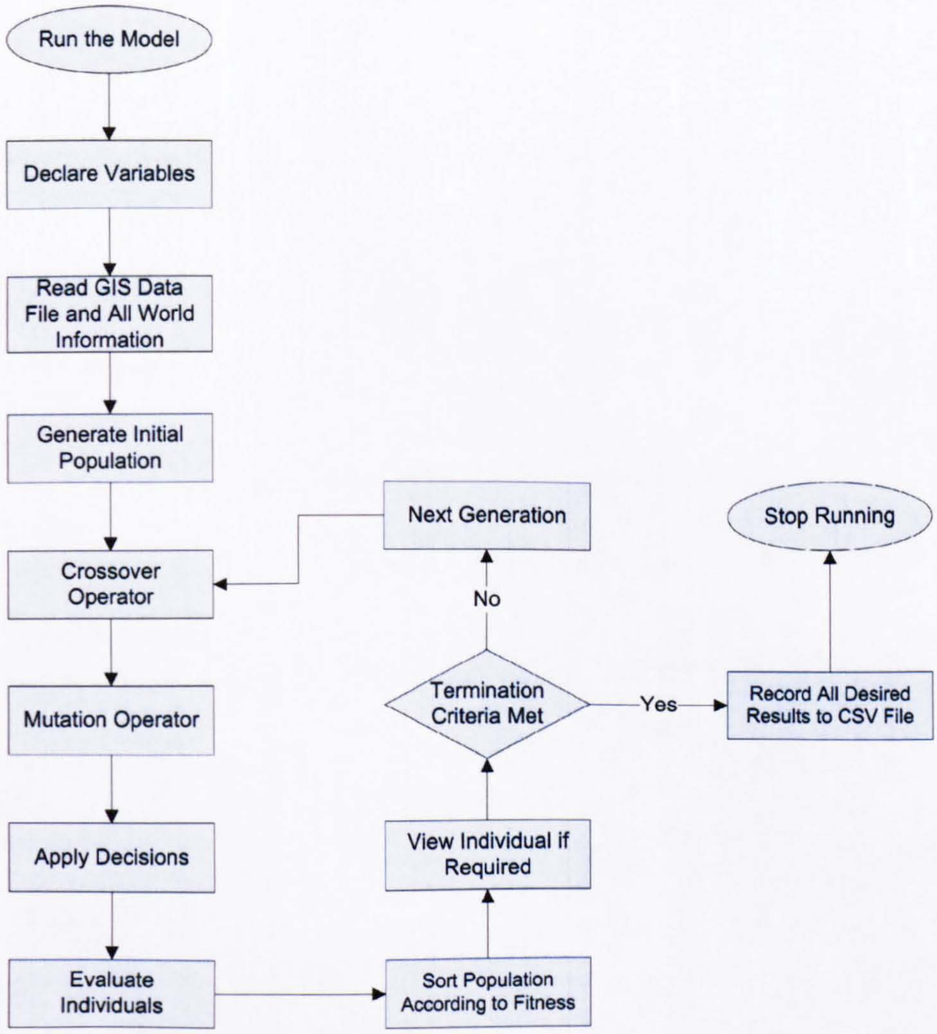


Figure D.1: The Prototype Flow-Chart

A key part of developing Visual Basic applications is ensuring that the code is well structured. This involves segmenting the code into projects, modules, functions, and subroutines so that it is easy to understand and maintain. A complete Visual Basic application is typically contained in a single project. In this model, the code is placed in several code files. Within each file, the VB code is further separated into self contained and re-usable procedures.

The variables are declared in modules and are declared once the program loads the main model window. Two types of procedures are used in this model:

1. Functions: Functions are procedures which perform a task and return a value when completed.

2. Subroutines: Subroutines are procedures which perform a task but return no value when completed.

Although subroutines are very important in the model, It is especially useful to be able to return values from functions. For example, the function may need to return the result of the task it performed (e.g. the result of the fitness calculation). A function might also return a True or False value to indicate when the task was performed successfully. The Visual Basic code which called the function then acts based on the returned value.

The main program routine is shown in figure D.2. This routine is included a number of subroutines which represent the core of the model. The subroutines may call other functions or subroutines during the run.

```
'Run the program
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As _
                          System.EventArgs) Handles Button1.Click

    ReadAreaInfo_Sub() ' Read area information
    ReadGridInfo_Sub() ' Read Grid information
    ReadGlobalInfo_Sub() ' Read other infor (e.g. Costs)
    ThreeD_Distribution_Sub() ' Assign elevations (3D) values to the grid
    InitialPopulation_Sub()

    For Generation_Main = 1 To NumberOfGenerations

        CrossOver_Sub()
        Mutation_Sub()
        CalculateAllFitness() ' Fitness Evaluation

        ' Sorts the current population
        Array.Sort(APopulation, New FitnessValueComparer)

        ' Vew Individual if required
        If Generation_Main = ShowSequence Then
            Form3.ShowDialog()
            ShowSequence += Sequence
        End If

    Next Generation_Main

    'Saving Data to CSV file
    Save_to_CSV()

    'Records the run time
    EndTime = CDate(DateTime.Now.ToLongTimeString)

End Sub
```

Figure D.2: The Main Model Code

The computer program begins with declaring a set of variables on loading and assigned values to some of the variables provided by text and combo boxes. Some of these variables internally resemble the chromosomes which store the genetic information. Each genome of these digital chromosomes represents a trait of the proposed data structure. This information are stored either a character string in which each character represents an integer value which describes the magnitude of a trait.

The developed model is an interactive windows form, tab driven, and point and click graphical user interface. This window form is responsible for automating input data preparation, execution of all associated models, checking the model, and display of results. This interface consists of several types of conventional controls provided by VS2010 toolbox. A brief description of some of these controls is provided below:

1. Tab Control: Manages and displays a related collection of tabs that contain controls and components.
2. Button: Raises an event when clicked.
3. Text Box: Enables the user to enter text and provides multi-line editing.
4. Check Box: Enables to select or clear the associated option.
5. Radio Button: Enables the user to select a dingle option from a group of choices when grouped with other radio buttons.
6. Combo Box: Displays an editable text box with a drop down list of permitted values.

The prototype consists of four tabs placed on the top of the main window. Each of these tabs will be addressed in the following sub-sections.

D.3.1 Main Interface (figure D.3)

This interface is the most important one in the prototype because the main prototype (such as demand benefit factor) and GA parameters are provided through this interface.

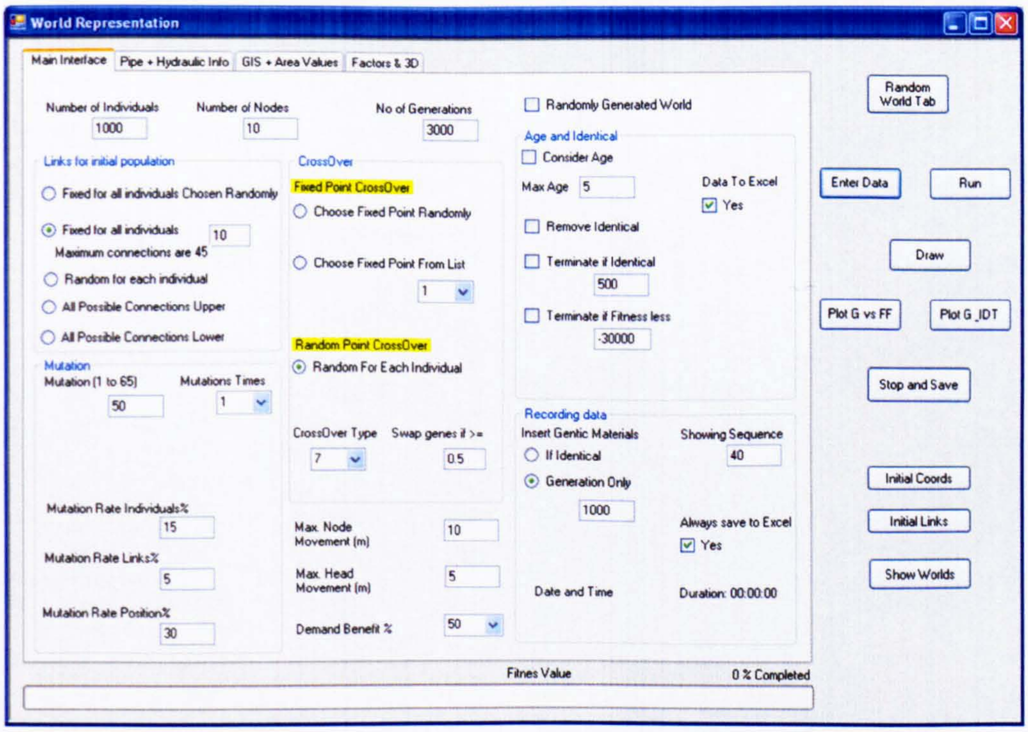


Figure D.3: Main Program Interface

D.3.2 Pipe + Hydraulic Info (figure D.4)

Pipe sizes and their associated prices are introduced in this tab. In addition, other information such as the necessary parameters for energy equation exists in this tab.

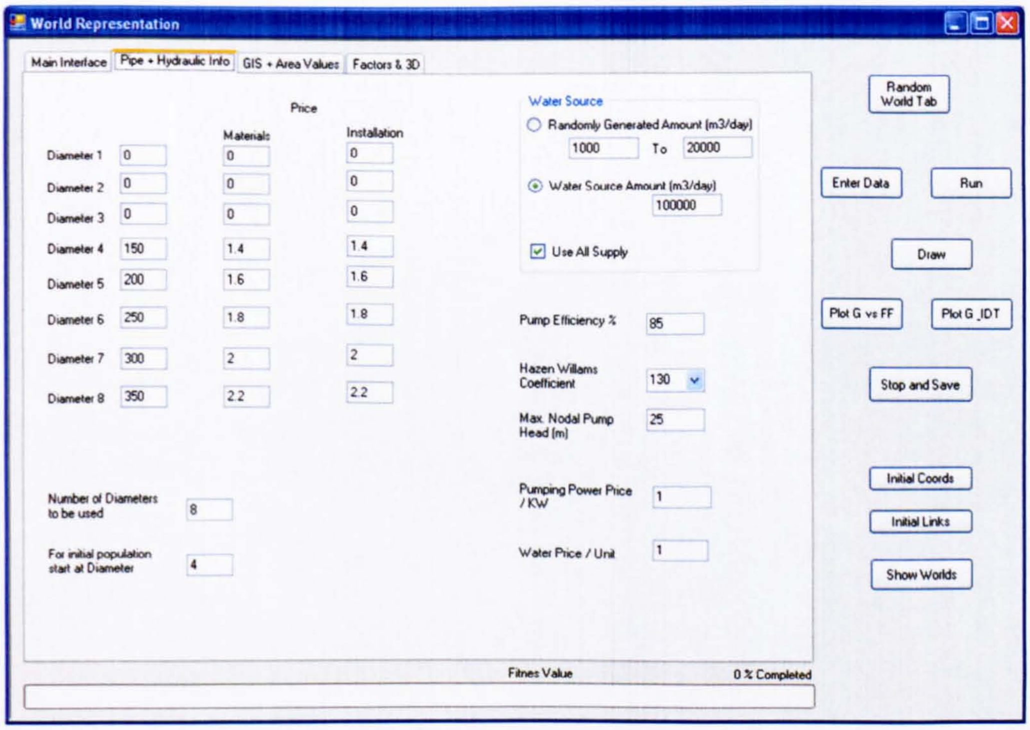


Figure D.4: Pipe and Hydraulic Information Interface

D.3.3 GIS + Area Values (figure D.5)

This tab is especially designed to produce different random worlds and save their data into text files. This information consists of the number of demand areas, the size demand, water demand, and the area access cost. In addition, the provides a button which provides the ability to view an existing world which provides the power to choose and compare different world layouts

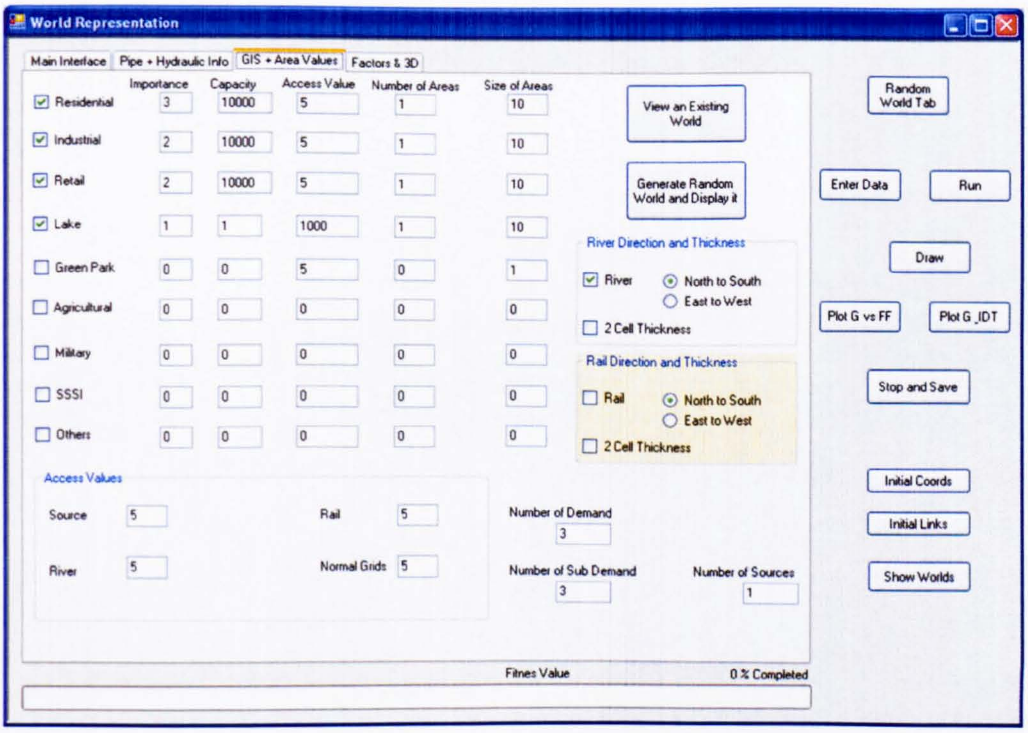


Figure D.5: GIS and Area Information Interface

D.3.4 Factors & 3-D (figure D.6)

Fitness function coefficients are introduced using this tab. In addition, the necessary parameters to generate the 3-D terrain model and calculate the fitness are presented in this tab.

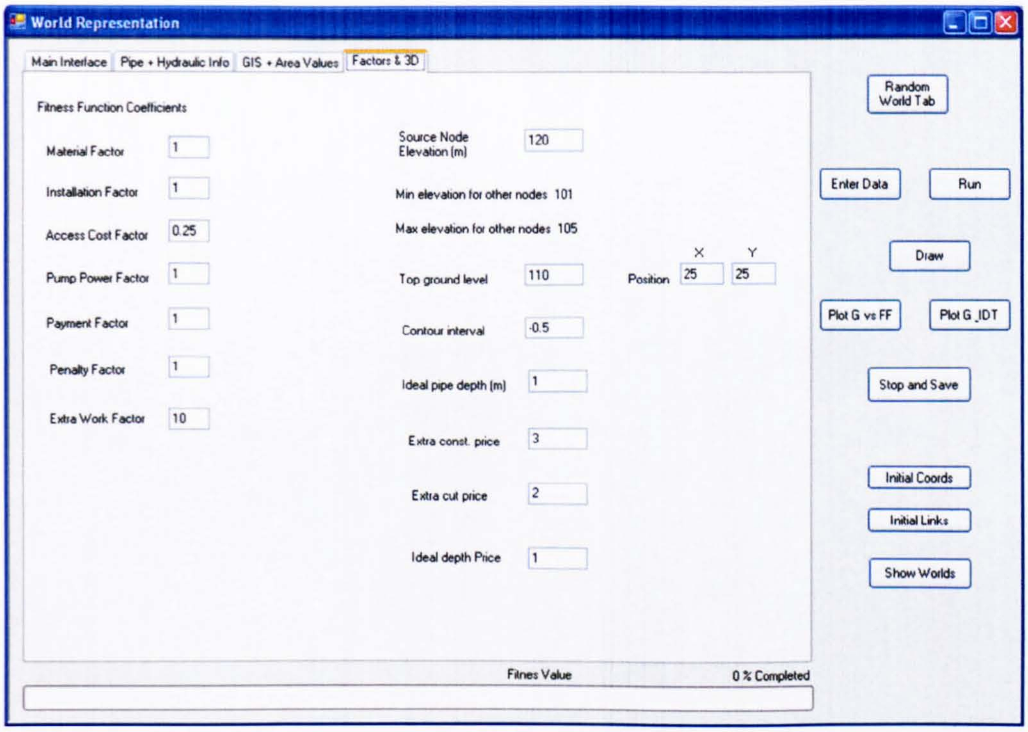


Figure D.6: Fitness Function Factors and 3-D Information Interface

After assigning and tuning the desired GA and other model parameters into the main model interface, the user should start running the model by clicking on 'Run' button. The world data can be loaded to the model when a dialogue box appears and the world data file is selected. All world data will be stored into predefined variables and other data structures (e.g. matrices).

The computer program then creates the initial population through stochastic (random) means, and then evaluates these individuals. The initial population then will be sorted descending according to their fitness values. This is a kind of genetic selection in which the computer model uses a procedure to test the fitness of the chromosome and assigns a sort of numeric value for its fitness in comparison to other chromosomes.

The computer program then takes the fittest chromosomes and creates another generation through the use of genetic operators (crossover and mutation). These two genetic operators can be used in different combinations, all of them producing different results.

The new generation of chromosomes is created using genetic recombination (crossover) where new offspring are created from the fittest chromosomes of the previous generation.

Crossover is followed by mutation operator which is analogous to the simulation of genetic mutation, in which the offspring are identical to their parents but have random, stochastic changes in their structure (and thus their characteristic are modified).

Before evaluating the population, the developed decision mechanism is employed in which the individuals' characteristics will be changed. For example, the layout of the network will be changed after removing the cycles if they exist. This change will affect the way that the individual will be evaluated. An example of the code for removing cycles in networks is shown in figure D.7.

```
Public Sub RemoveConnctedLoops(ByVal LoopIndividual As ASolution)

    For MarkWhite = 0 To NumberOfNodes - 1
        TheNodeList(MarkWhite) = "Not_Visited"
    Next MarkWhite
    Dim WhiteNodeCounter As Integer = 0
    Do
        If TheNodeList(WhiteNodeCounter) = "Not_Visited" Then
            Depth_First_Search(WhiteNodeCounter)
        End If
        WhiteNodeCounter += 1
    Loop Until TheNodeList.All(Function(VisitedNodes) VisitedNodes = "Visited")

End Sub

Public Function Depth_First_Search(ByRef FromNodeDFS As Integer) As Integer

    TheNodeList(FromNodeDFS) = "Grey"
    For U = 1 To NumberOfNodes - 1
        If APopulation(PassedIndividualNumber).TheConnections(FromNodeDFS, U) > 0 Then
            If TheNodeList(U) = "Grey" Then
                APopulation(PassedIndividualNumber).TheConnections(FromNodeDFS, U) = 0
                APopulation(PassedIndividualNumber).LinkPumpHead(FromNodeDFS, U) = 0
            ElseIf TheNodeList(U) = "Not_Visited" Then
                Depth_First_Search(U)
            End If
        End If
    Next U
    TheNodeList(FromNodeDFS) = "Visited"

End Function
```

Figure D.7: VB Code For Removing Cycles

After applying the decision, the population is now ready to be evaluated by the fitness function. The population then will be sorted to according to their fitness function and a new population will be formed for the next generations. At this stage, The user have the power to view the best individual (the default) or any other individual in the population. This process of testing for fitness and creating new generations is repeated until the user-specified maximum number of evolutions has been run.

References

Adeli, H. and Cheng, N.-T. (1993), 'Integrated genetic algorithm for optimization of space structures', *Journal of Aerospace Engineering* **6**(4), 315–328.

Afshar, M. H., Akbari, M. and Marino, M. A. (2005), 'Simultaneous layout and size optimization of water distribution networks: Engineering approach', *Journal of Infrastructure Systems* **11**(4), 221–230.

Afshar, M. and Marino, M. (2005), 'A convergent genetic algorithm for pipe network optimization', *Scientia Iranica* **12**(4), 392–401.

Alandi, P. P., lvarez, J. F. O. and Martn-Benito, J. M. T. (2007), 'Optimization of irrigation water distribution networks, layout included', *Agricultural Water Management* **88**(1-3), 110–118.

Alfares, H. K. and Duffuaa, S. O. (2009), 'Assigning cardinal weights in multi-criteria decision making based on ordinal ranking', *Journal of Multi-Criteria Decision Analysis* **15**(5-6), 125–133.

Alperovits, E. and Shamir, U. (1977), 'Design of optimal water distribution systems', *Water Resources Research* **13**(6), 885–900.

Amit, R. K. and Ramachandran, P. (2009), 'Optimal design of water distribution networks, a review', **The selected works of R K Amit. Available at: http://works.bepress.com/rkamit/7**.

Andrew, L., Mostaghim, S. and Randall, M. (2009), *Biologically-inspired Optimisation Methods: Parallel Algorithms, Systems and Methods*, Springer Publishing Company, Incorporated.

Atkinson, R., Morley, M., Walters, G. and Savic, D. (1998), GANET-the integration of GIS, network analysis and genetic algorithm optimization software for water network analysis, *in* 'Hydroinformatics' 98: proceedings of the third International Conference on Hydroinformatics, Copenhagen, Denmark, 24-26 August', pp. 357–362.

Babayan, A., Kapelan, Z., Savic, D. and Walters, G. (2005), 'Least-cost design of water distribution networks under demand uncertainty', *Journal of Water Resources Planning and Management* **131**(5), 375–382.

Balling, R. (2003), The maximum fitness function; multi-objective city and regional planning, *in* 'Evolutionary Multi-Criterion Optimization', Springer-Verlag GmbH, pp. 1–15.

Balling, R. J., Taber, J. T., Brown, M. R. and Day, K. (1999), 'Multi-objective urban planning using genetic algorithm', *Journal of Urban Planning and Development* **125**(2), 86–99.

Bartolin, H. and Martinez, F. (2003), Modelling and calibration of water distribution systems. a new GIS approach, *in* '23rd ESRI International User Conference'.

Beasley, D., Bull, D. R. and Martin, R. R. (1993*a*), 'An overview of genetic algorithms: Part 1, fundamentals', *University Computing, 1993,* **15**(2), 59–69.

Beasley, D., Bull, D. R. and Martin, R. R. (1993*b*), 'An overview of genetic algorithms: Part 2, research topics', *University Computing, 1993,* **15**(4), 170–181.

Bhave, P. (2003), *Optimal design of water distribution networks*, Alpha Science International Ltd.

Bhave, P. R. and Gupta, R. (2006), *Analysis of Water Distribution Networks*, Alpha Science International Ltd.

Birkin, M., Dew, P., Macfarland, O. and Hodrien, J. (2005), HYDRA: A prototype grid-enabled spatial decision support system, *in* 'First International Conference on e-Social Science'.

Blair, N., Berry, J. and McGreal, S. (2007), 'Regional spatial policy for economic growth: lessons from the deployment of collaborative planning in Northern Ireland', *Urban Studies* **44**(3), 439–455.

Blanco, R., Inza, I. and Larraaga, P. (2003), 'Learning bayesian networks in the space of structures by estimation of distribution algorithms', *International Journal of Intelligent Systems* **18**(2), 205–220.

Blickle, T. and Thiele, L. (1995), A mathematical analysis of tournament selection, *in* 'Proceedings of the Sixth International Conference on Genetic Algorithms', Morgan Kaufmann Publishers Inc., pp. 9–16.

Booker, L. (1987), Improving search in genetic algorithms, *in* L. Davis, ed., 'Genetic algorithms and simulated annealing', Morgan Kauffmann Publishers, pp. 61–73.

Boulos, P. F., Wu, Z., Orr, C. H., Moore, M., Hsiung, P. and Thomas, D. (2001), Optimal pump operation of water distribution systems using genetic algorithms, *in* 'Distribution System Symposium', Orlando, FL. USA.

Broad, D. R., Dandy, G. C. and Maier, H. R. (2005), 'Water distribution system optimization using metamodels', *Journal of Water Resources Planning and Management* **131**(3), 172–180.

Brooks, D. and Wolfe, S. (2007), 'Water demand management as governance: Lessons from the Middle East and South Africa', *Water Resresources in the Middle East: the Israeli-Palestinian Water Issues: from conflict to cooperation* **2**, 311–323.

Brundtland (1987), Our common future, Technical report, Oxford University Press, Oxford, GB.

Brunn, S., Williams, J. and Zeigler, D. (2003), *Cities of the world: World regional urban development*, Rowman & Littlefield Publishers.

Bui, T. X. (1999), Decision support systems for sustainable development: An overview, *in* G. E. Kersten, Z. Mikolajuk and A. G.-O. Yeh, eds, 'Decision Support Systems for Sustainable Development: A Resource Book of Methods and Applications', Kluwer Academic Publishers, Hingham, MA, USA.

Cambridge Advanced Learner's Dictionary (2011), 'Cambridge dictionaries online'. Viewed: August 2011.
URL: *dictionary.cambridge.org*

Canellas, B., Balle, S., Tintoré, J. and Orfila, A. (2010), 'Wave height prediction in the Western Mediterranean using genetic algorithms', *Ocean Engineering* **37**(8-9), 742–748.

Cantú-Paz, E. and Goldberg, D. E. (2000), 'Efficient parallel genetic algorithms: Theory and practice', *Computer Methods in Applied Mechanics and Engineering* **186**(2-4), 221–238.

Chang, K. T. (2003), *Introduction to geographic information systems*, McGraw Hill Companies, Inc.

Chong, E. and Żak, S. (2001), *An Introduction to Optimization*, 2nd edn, Wiley-interscience.

Coello, C. A. C. and Lamont, G. B. (2005), *Applications of multi-objective evolutionary algorithms*, World Scientific Publishing Company, Incorporated.

Coley, D. A. (1999), *An introduction to genetic algorithms for scientists and engineers*, World Scientific Publishing Co. Ltd.

Committee on Public Water Supply Distribution Systems (2006), *Drinking Water Distribution Systems*, National Academies Press.

Cova, T. J. and Church, R. L. (2000), 'Exploratory spatial optimization in site search: A neighborhood operator approach', *Computers, Environment and Urban Systems* **24**(5), 401–419.

Cui, L.-J. and Kuczera, G. (2003), 'Optimizing urban water supply headworks using probabilistic search methods', *Journal of Water Resources Planning and Management* **129**(5), 380–387.

Cunha, M. and Sousa, J. (1999), 'Water distribution network design optimization: Simulated annealing approach', *Journal of Water Resources Planning and Management* **125**(4), 215–221.

Cunha, M. and Sousa, J. (2001), 'Hydraulic infrastructures design using simulated annealing', *Journal of infrastructure systems* **7**(1), 32–39.

Dandy, G. C. and Engelhardt, M. (2001), 'Optimal scheduling of water pipe replacement using genetic algorithms', *Journal of Water Resources Planning and Management* **127**(4), 214–223.

Dandy, G. C. and Engelhardt, M. O. (2006), 'Multi-objective trade-offs between cost and reliability in the replacement of water mains', *Journal of Water Resources Planning and Management* **132**(2), 79–88.

Dandy, G. C., Simpson, A. R. and Murphy, L. J. (1996), 'An improved genetic algorithm for pipe network optimization', *Water Resources Research* **32**(2), 449–458.

Dantzig, G. (1998), *Linear programming and extensions*, Princeton Univ Press.

Dasić, T. and Djordjevic, B. (2004), Method for water distribution systems reliability evaluation, *in* C. Pahl-Wostl, S. Schmidt, A. E. Rizzoli and A. J. Jakeman, eds, 'International Environmental Modelling and Software Society iEMSs', Vol. 1, University of Osnabrück, Germany.

Davidson, J. (1999), 'Evolution program for layout geometry of rectilinear looped networks', *Journal of Computing in Civil Engineering* **13**(4), 246–253.

Davidson, J. and Goulter, I. (1991), 'Heuristic for layout design of rural gas systems', *Journal of Computing in Civil Engineering* **5**(3), 315–332.

Davidson, J. W. and Goulter, I. C. (1995), 'Evolution program for design of rectilinear branched networks', *Journal of Computing in Civil Engineering* **9**(2), 112–121.

De Jong, K. A. (1975), Analysis of the behaviour of a class of genetic adaptive systems, Phd thesis, The University of Michigan.

De Jong, K. A. (2006), *Evolutionary computation: A unified approach*, Massachusetts Institute of Technology (MIT) Press.

De Schaetzen, W. B. F. and Boulos, P. F. (2002), Optimal water distribution system management using ESRI MapObjects technology, *in* '22nd ESRI International User Conference'.

De Schaetzen, W. B. F., Walters, G. A. and Savic, D. A. (2000), 'Optimal sampling design for model calibration using shortest path, genetic and entropy algorithms', *Urban Water* **2**(2), 141 – 152.

DeMers, M. N. (2005), *Fundamentals of Geographic Information Systems*, John Wiley and Sons, Inc.

Dinar, A., Rosegrant, M. and Meinzen-Dick, R. (1997), *Water allocation mechanisms: Principles and examples*, World Bank, Agriculture and Natural Resources Dept., Sector Policy and Water Resources Division and International Food Policy Research Institute.

Dorigo, M., Birattari, M. and Stutzle, T. (2006), 'Ant colony optimization', *IEEE Computational Intelligence Magazine* **1**(4), 28–39.

Dorogovtsev, S. N. and Mendes, J. F. F. (2002), 'Evolution of networks', *Advances in Physics* **51**(4), 1079–1187.

Ebrahimipoor, A., Alimohamadi, A., Alsheikh, A. and Aghighi, H. (2009), 'Routing of water pipeline using GIS and genetic algorithm', *Journal of Applied Sciences* **9**(23), 4137–4145.

Eiger, G., Shamir, U. and Ben-Tal, A. (1994), 'Optimal design of water distribution networks', *Water Resources Research* **30**(9), 2637–2646.

ESRI (2011), 'Cell size of raster data'. Accessed at [26th February 2011].
URL: *webhelp.esri.com/arcgiSDEsktop/9.3/index.cfm?TopicName=Cell_size_of_raster_data*

Eusuff, M. M. and Lansey, K. E. (2003), 'Optimization of water distribution network design using the shuffled frog leaping algorithm', *Journal of Water Resources Planning and Management* **129**(3), 210–225.

Fairley, A. (1991), 'Comparison of methods of choosing the crossover point in the genetic crossover operation', *Dept. of Computer Science, University of Liverpool* .

Federal Geographic Data Committee (2007), 'Geospatial metadata standards'. Accessed at [25^{th} February 2010].
URL: *www.fgdc.gov/metadata/geospatial-metadata-standards*

Fedra, K. (1999), 'Urban environmental management: Monitoring, GIS, and modelling', *Computers, Environment and Urban Systems* **23**(6), 443–457.

Feeney, M. M.-E., Rajabifard, D. A. and Williamson, P. I. (2001), Spatial data infrastructure frameworks to support decision-making for sustainable development, *in* 'Proceedings of the 5th Global Spatial Data Infrastructure (GSDI) Conference', Cartagena, Colombia.

Fletcher, R. (2000), *Practical methods of optimization*, John Wiley & Sons.

Fonseca, C. M. and Fleming, P. J. (1993), Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization, *in* 'Proceedings of the fifth international conference on genetic algorithms', pp. 416–423.

Food and Agriculture Organization of the United Nations (1993), *Guidelines for land-use planning*, Food and Agriculture Organization.

Ford, M. (2007), A Genetic Algorithm Based Decision Support System For The Sustainable Location Of Development Allocations, Phd thesis, School of Civil Engineering, The University of Nottingham.

Friend, J. and Hickling, A. (2005), Foundation, *in* J. Friend and A. Hickling, eds, 'Planning Under Pressure: The Strategic Choice Approach', Butterworth-Heinemann, Oxford, pp. 1–22.

Fujiwara, O. and Khang, D. (1990), 'A two-phase decomposition method for optimal design of looped water distribution networks', *Water Resours* **26**(4), 539–549.

Gaham, M. and Bouzouia, B. (2009), Intelligent product-driven manufacturing control: A mixed genetic algorithms and machine learning approach to product intelligence synthesis, *in* 'Information, Communication and Automation Technologies, 2009. ICAT 2009. XXII International Symposium on', IEEE, pp. 1–8.

Garca-Pedrajas, N., Ortiz-Boyer, D. and Hervs-Martnez, C. (2006), 'An alternative approach for neural network evolution with a genetic algorithm: Crossover by combinatorial optimization', *Neural Networks* **19**(4), 514–528.

Geem, Z. (2006), 'Optimal cost design of water distribution networks using harmony search', *Engineering Optimization* **38**(3), 259–277.

Geem, Z., Kim, J. and Loganathan, G. (2002), 'Harmony search optimization: Application to pipe network design', *International journal of modelling & simulation* **22**(2), 125–133.

Geem, Z. W. (2009), 'Harmony search optimisation to the pump-included water distribution network design', *Civil Engineering and Environmental Systems* **26**(3), 211–221.

Gleick, P. H. (1998), 'Water in crisis: Paths to sustainable water use', *Ecological Applications* **8**(3), 571–579.

Glover, F. (1977), 'Heuristics for integer programming using surrogate constraints', *Decision Sciences* **8**(1), 156–166.

Glover, F., Kelly, J. P. and Laguna, M. (1995), 'Genetic algorithms and tabu search: Hybrids for optimization', *Computers & Operations Research* **22**(1), 111–134.

Goldberg, D. (1989), *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley Professional.

Goldberg, D. E., Deb, K., Kargupta, H. and Harik, G. (1993), Rapid accurate optimization of difficult problems using fast messy genetic algorithms, *in* 'Proceedings of the 5th International Conference on Genetic Algorithms', Morgan Kaufmann Publishers Inc., pp. 56–64.

Goldberg, D. E. and Kuo, C. H. (1987), 'Genetic algorithms in pipeline optimization', *Journal of Computing in Civil Engineering* **1**(2), 128–141.

Goulter, I. (1992), 'Systems analysis in water-distribution network design from theory to practice', *Journal of Water Resources Planning and Management* **118**(3), 238–248.

Goulter, I., Lussier, B. and Morgan, D. (1986), 'Implications of head loss path choice in the optimization of water distribution networks', *Water Resources Research* **22**(5), 819–822.

Griffin, R. C. (2006), *Water resource economics: the analysis of scarcity, policies, and projects*, Vol. 1, The MIT Press.

Gross, J. and Yellen, J. (2004), *Handbook of graph theory*, CRC Press.

Gupta, I., Gupta, A. and Khanna, P. (1999), 'Genetic algorithm for optimization of water distribution systems', *Environmental Modelling and Software* **14**(5), 437–446.

Halhal, D., Walters, G. A., Ouazar, D. and Savic, D. A. (1997), 'Water network rehabilitation with structured messy genetic algorithm', *Journal of Water Resources Planning and Management* **123**(3), 137–146.

Halhal, D., Walters, G., Savic, D. and Ouazar, D. (1999), 'Scheduling of water distribution system rehabilitation using structured messy genetic algorithms', *Evolutionary Computation* **7**(3), 311–329.

Han, M., Munemoto, J. and Matsushita, D. (2004), 'Optimizing parallel layout of MRHC to improve quality of views from dwelling units with genetic algorithm', *Journal of Asian Architecture and Building Engineering* **3**(2), 341–347.

Hasancebi, O. and Erbatur, F. (2000), 'Evaluation of crossover techniques in genetic algorithm based optimum structural design', *Computers & Structures* **78**(1-3), 435–448.

Haupt, R. L. and Haupt, S. E. (2004), *Practical genetic algorithms*, John Wiley & Sons, Inc.

Hayes-Roth, B. and Hayes-Roth, F. (1979), 'A cognitive model of planning', *Cognitive Science* **3**(4), 275–310.

Healey, P. (2006), *Collaborative planning: Sampling places in fragmented societies*, Palgrave Macmillan.

Herrera, F., Lozano, M. and Verdegay, J. L. (1998), 'Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis', *Artificial Intelligence Review* **12**(4), 265–319.

Heywood, I., Cornelius, S. and Carver, S. (2006), *An introduction to geographical information systems*, Pearson Education limited.

Ho, C.-I., Lin, M.-D. and Lo, S.-L. (2010), 'Use of a GIS-based hybrid artificial neural network to prioritize the order of pipe replacement in a water distribution network', *Environmental Monitoring and Assessment* **166**(1–4), 177–189.

Holland, J. (1975), *Adaption in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI.

Hopcroft, J. and Tarjan, R. (1973), 'Algorithm 447: efficient algorithms for graph manipulation', *Commun. ACM* **16**(6), 372–378.

Iamtrakul, P., Teknomo, K. and Hokao, K. (2005), Public park valuation using travel cost method, *in* 'Proceedings of the Eastern Asia Society for Transportation Studies', Vol. 5, pp. 1249–1264.

Janikow, C. and Michalewicz, Z. (1991), An experimental comparison of binary and floating point representations in genetic algorithms, *in* 'Proceedings of the Fourth International Conference on Genetic Algorithms', Vol. 31, California, USA, p. 36.

Jankowski, P. (1995), 'Integrating geographical information systems and multiple criteria decision-making methods', *International Journal of Geographical Information Science* **9**(3), 251–273.

Jeer, S. (1997), 'Traditional color coding for land uses', *American Planning Association* pp. 4–5.

Jones, R. M., Copas, C. V. and Edmonds, E. A. (1997), 'GIS support for distributed group-work in regional planning', *International Journal of Geographical Information Science* **11**(1), 53–71.

Kadu, M. S., Gupta, R. and Bhave, P. R. (2008), 'Optimal design of water networks using a modified genetic algorithm with reduction in search space', *Journal of Water Resources Planning and Management* **134**(2), 147–160.

Kapelan, Z., Savic, D. and Walters, G. (2003), Robust least cost design of water distribution systems using GAs, *in* 'Advances in water supply management: Proceedings of the International Conference on Computing and Control for the Water Industry, 15-17 September 2003, London, UK', Taylor & Francis, p. 147.

Kapelan, Z., Savic, D. and Walters, G. (2005), 'Multiobjective design of water distribution systems under uncertainty', *Water Resources Research* **41**(11), W11407.

Karmeli, D., Gadish, Y. and Meyers, S. (1968), 'Design of optimal water distribution networks', *Journal of Pipeline Division* **94**(10), 1–10.

Kennedy, J. and Eberhart, R. (1995), Particle swarm optimization, *in* 'Proceedings of the 1995 IEEE International Conference on Neural Networks', Vol. 4, IEEE, pp. 1942–1948.

Kessler, A. and Shamir, U. (1989), 'Analysis of the linear programming gradient method for optimal design of water supply networks', *Water Resources Research* **25**(7), 1469–1480.

Kirkpatrick, S. (1984), 'Optimization by simulated annealing: Quantitative studies', *Journal of Statistical Physics* **34**(5), 975–986.

Koza, J. (1990), *Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems*, Stanford University.

Lee, C.-Y. (2003), 'Entropy-boltzmann selection in the genetic algorithms', *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* **33**(1), 138–149.

Leng, G., McGinnity, T. and Prasad, G. (2006), 'Design for self-organizing fuzzy neural networks based on genetic algorithms', *IEEE Transactions on Fuzzy Systems* **14**(6), 755–766.

Levitin, L., Karpovsky, M. and Mustafa, M. (2010), 'Minimal sets of turns for breaking cycles in graphs modeling networks', *IEEE Transactions on Acoustics, Speech and Signal Processing. Parallel Distribution Systems* **21**, 1342–1353.
URL: *http://dx.doi.org/10.1109/TPDS.2009.174*

Levy, J. M. (2002), *Contemporary Urban Planning*, Prentice Hall, Englewood Cliffs, New Jersey.

Lippai, I., Heaney, J. P. and Laguna, M. (1999), 'Robust water system design with commercial intelligent search optimizers', *Journal of Computing in Civil Engineering* **13**(3), 135–143.

Loganathan, G., Greene, J. and Ahn, T. (1995), 'Design heuristic for globally minimum cost water-distribution systems', *Journal of Water Resources Planning and Management* **121**(2), 182–192.

Loucks, D. P., Beek, E. v., Stedinger, J. R. and Dijkman, J. P. (2005), *Water resources systems planning and management*, UNESCO Publishing.

Lowry, M. B. (2004), Coordinated regional and city planning using a genetic algorithm, PhD thesis, Brigham Young University.

Lund, J. R. and Israel, M. (1995), 'Optimization of transfers in urban water supply planning', *Journal of Water Resources Planning and Management* **121**(1), 41–48.

Madsen, H., Krenk, S. and Lind, N. (2009), *Methods of Structural Safety*, Dover Publications Inc.

Maier, H., Simpson, A., Zecchin, A., Foong, W., Phang, K., Seah, H. and Tan, C. (2003), 'Ant colony optimization for design of water distribution systems', *Journal of Water Resources Planning and Management* **129**(3), 200–209.

Makropoulos, C. and Butler, D. (2003), 'Fuzzy logic spatial decision support system for urban water management', *Journal of Water Resources Planning and Management* **129**(1), 69–77.

Malczewski, J. (1999), *GIS and multicriteria decision analysis*, John Wiley and Sons, Inc, New York.

Matthews, K. B. (2001), Applying genetic algorithms to multi-objective land-use planning, PhD thesis, The Robert Gordon University, Aberdeen.

Mawdesley, M. J. and Al-Jibouri, S. H. (2003), 'Proposed genetic algorithms for construction site layout', *Engineering Applications of Artificial Intelligence* **16**(5-6), 501–509.

Mawdesley, M. J., Al-jibouri, S. H. and Yang, H. (2002), 'Genetic algorithms for construction site layout in project planning', *Journal of Construction Engineering and Management* **128**(5), 418–426.

Mays, L. W. (2000), *Water Distribution Systems Handbook*, McGraw-Hill.

Maza, M. d. l. and Tidor, B. (1993), An analysis of selection procedures with particular attention paid to proportional and Boltzmann selection, *in* 'Proceedings of the 5th International Conference on Genetic Algorithms', Morgan Kaufmann Publishers, Inc., pp. 124–131.

McKinney, D. C. and Cai, X. (2002), 'Linking GIS and water resources management models: An object-oriented method', *Environmental Modelling & Software* **17**(5), 413–425.

Meier, R. W. and Barkdoll, B. D. (2000), 'Sampling design for network model calibration using genetic algorithms', *Journal of Water Resources Planning and Management* **126**(4), 245–250.

Meinzen-Dick, R. and Mendoza, M. (1996), 'Alternative water allocation mechanisms: Indian and international experiences', *Economic and political Weekly* **31**(13), 25–30.

Michalewicz, Z. (1999), *Genetic algorithms + data structures = evolution programs*, Springer, Berlin, Germany.

Microsoft Corporation (2011*a*), 'Array.Sort(Of T) Method (T())'. Accesssed at [8^{th} August 2011].
URL: *msdn.microsoft.com/en-us/library/kwx6zbd4.aspx*

Microsoft Corporation (2011*b*), 'Object-Oriented Programming in Visual Basic .NET'. Accesssed at [5^{th} December 2011].
URL: *http://msdn.microsoft.com/en-us/library/aa289512(v=vs.71).aspx*

Microsoft Corporation (2011c), 'Random constructor'. Accesssed at [9^{th} August 2010].
URL: *msdn.microsoft.com/en-us/library/aa329889(v=vs.71).aspx*

Midwest Regional Planning Organization (2001), Principles for regional planning, Technical report, Midwest Regional Planning Organization.

Mikolajuk, Z. and Yeh, A. G.-O. (1999), Sustainable development and decision support systems, *in* G. E. Kersten, Z. Mikolajuk and A. G.-O. Yeh, eds, 'Decision Support Systems for Sustainable Development : A Resource Book of Methods and Applications', Kluwer Academic Publishers, Hingham, MA, USA.

Mira, J., lvarez, J., Valle, C., Gasca, R., Toro, M. and Camacho, E. (2003), A genetic algorithm for assembly sequence planning, *in* 'Artificial Neural Nets Problem Solving Methods', Vol. 2687, Springer Berlin / Heidelberg, pp. 1044–1044.

Mitchell, M. (1996), *An introduction to genetic algorithms*, Massachusetts Institute of Technology.

Mohan, S. and Babu, K. (2010), 'Optimal water distribution network design with Honey-Bee mating optimization', *Journal of Computing in Civil Engineering* **24**(1), 117–126.

Montalvo, I., Izquierdo, J., Pérez, R. and Iglesias, P. L. (2008), 'A diversity-enriched variant of discrete PSO applied to the design of Water Distribution Networks', *Engineering Optimization* **40**(07), 655–668.
URL: *http://peer.ccsd.cnrs.fr/peer-00545357/en/*

Morgan, D. and Goulter, I. (1985), 'Optimal urban water distribution design', *Water Resources Research* **21**(5), 642–652.

Morley, M., Makropoulos, C., Savic, D. and Butler, D. (2004), Decision-support system workbench for sustainable water management problems, *in* 'International Environmental Modelling And Software Society (iEMSs)', 14 -17 June 2004, University of Osnabrck, Germany.

Morley, M. S., Atkinson, R. M., Savic, D. A. and Walters, G. A. (2001), 'GAnet: genetic algorithm platform for pipe network optimisation', *Advances in Engineering Software* **32**(6), 467–475.

Munavalli, G. R. and Kumar, M. S. M. (2003), 'Optimal scheduling of multiple chlorine sources in water distribution systems', *Journal of Water Resources Planning and Management* **129**(6), 493–504.

Murphy, L., Simpson, A. and Dandy, G. (1993), 'Design of a pipe network using genetic algorithms', *water melbourne then artarmon* **20**(4), 40–42.

Newman, M. E. J. (2003), 'The structure and function of complex networks', *SIAM Review* **45**(2), pp. 167–256.

Obitko, M. (1998), 'Outline of the basic genetic algorithm'. Accssed at [15th January 2009].
URL: *www.obitko.com/tutorials/genetic-algorithms/ga-basic-description.php*

Office for National Statistics (ONS) (2005), 'Ks01 usual resident population: Census 2001, key statistics for urban areas'. Accessed at [22^{nd} February 2010].
URL: *www.statistics.gov.uk/statbase/ssdataset.asp?vlnk=8271&More=Y*

Orenstein, T., Kohavi, Z. and Pomeranz, I. (1995), 'An optimal algorithm for cycle breaking in directed graphs', *Journal of Electronic Testing* **7**(1), 71–81.

Ortiz-Boyer, D., Hervás-Martínez, C. and García-Pedrajas, N. (2007), 'Improving crossover operator for real-coded genetic algorithms using virtual parents', *Journal of Heuristics* **13**(3), 265–314.

Ostfeld, A. and Shamir, U. (1996), 'Design of optimal reliable multiquality water-supply systems', *Journal of Water Resources Planning and Management* **122**(5), 322–333.

Paelinck, J. H. P. (1976), 'Qualitative multiple criteria analysis, environmental protection and multiregional development', *Papers in Regional Science* **36**(1), 59–76.

Passone, S., Chung, P. and Nassehi, V. (2006), 'Incorporating domain-specific knowledge into a genetic algorithm to implement case-based reasoning adaptation', *Knowledge-Based Systems* **19**(3), 192 – 201.

Pelikan, M. (2005), *Hierarchical Bayesian Optimization Algorithm*, Vol. 170 / 2005, Springer Berlin / Heidelberg.

Possiel, W. J., Saunier, R. E. and Meganck, R. A. (1995), In-situ conservation of biodiversity, *in* R. E. Saunier and R. A. Meganck, eds, 'Conservation of Biodiversity and the New Regional Planning', Organization of American States and the IUCN–The World Conservation Union.

Prasad, T. D. (2010), 'Design of pumped water distribution networks with storage', *Journal of Water Resources Planning and Management* **136**(1), 129–132.

Prasad, T. D. and Park, N.-S. (2004), 'Multiobjective genetic algorithms for design of water distribution networks', *Journal of Water Resources Planning and Management* **130**(1), 73–82.

Quesne, T. L., Pegram, G. and Heyden, C. V. D. (2007), Allocating scarce water: A WWF primer on water allocation, water rights and water markets, Technical report, World Water Forum.

Quindry, G., Liebman, J. and Brill, E. (1981), 'Optimization of looped water distribution systems', *Journal of the Environmental Engineering Division* **107**(4), 665–679.

Radcliffe, N. J. (1992), Non-linear genetic representations, *in* Männer and B. Manderick, eds, 'Parallel Problem Solving from Nature, 2', Vol. 2, Elsevier Science Publishers, Amsterdam, pp. 259–268.

Reca, J. and Martínez, J. (2006), 'Genetic algorithms for the design of looped irrigation water distribution networks', *Water Resources Research* **42**(5), W05416.

Reeves, C. R. and Rowe, J. E. (2002), *Genetic algorithms: Principles and perspectives: A guide to GA theory*, Vol. 20, Kluwer Academic Publishers.

Samani, H. and Mottaghi, A. (2006), 'Optimization of water distribution networks using integer linear programming', *Journal Of Hydraulic Engineering* **132**(5), 501–509.

Samuels, P., Woods, B., Hutchings, C., Felgate, J., Mobbs, P., Elliott, C. and Brook, D. (2006), *Sustainable water management in land use planning*, CIRIA, London.

Sanchez, T. and Wolf, J. (2007), 'Regional Transportation Planning and Metropolitan Governance', *Public Administration Review* **67**(1), 171–174.

Savic, A. A. and Walters, G. A. (1999), 'Hydroinformatics, data mining and maintenance of UK water networks', *Anti-Corrosion Methods and Materials* **46**(6), 415–425.

Savic, D. A. and Walters, G. A. (1997*a*), 'Evolving sustainable water networks', *Hydrological Sciences Journal* **42**(4), pp. 549–564.

Savic, D. A. and Walters, G. A. (1997*b*), 'Genetic algorithms for least-cost design of water distribution networks', *Journal of Water Resources Planning and Management* **123**(2), 67–77.

Savic, D., Walters, G. and Schwab, M. (1997), Multiobjective genetic algorithms for pump scheduling in water supply, *in* D. Corne and J. Shapiro, eds, 'Evolutionary Computing', Vol. 1305 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 227–235.

Schaake, J. and Lai, D. (1969), *Linear programming and dynamic programming application to water distribution network design*, MIT Hydrodynamics Laboratory.

Schütze, M., Campisano, A., Colas, H., Vanrolleghem, P. and Schilling, W. (2003), Real-time control of urban water systems, *in* 'International Conference on Pumps, Electromechanical Devices and Systems Applied to Urban Water Management', Valencia, Spain, pp. 1–19.

Schuurman, N. (2004), *GIS: A short introduction*, Blackwell Publishing.

Sendich, E. (2006), *Planning and urban design standards*, John Wiley and Sons, Inc, Hoboken, New Jersey.

Shamsi, U. M. (2005), *Gis applications: for water, wastewater, and storm water systems*, CRC Press.

Shihu, S., Dong, Z., Suiqing, L., Mingqun, M., Ming, Z., Yixing, Y., Jinliang, G. and Hongbin, Z. (2010), Decision support system of water distribution network expansion, *in* '2010 International Conference on E-Business and E-Government (ICEE)', pp. 1520 –1523.

Simpson, A. R., Dandy, G. C. and Murphy, L. J. (1994), 'Genetic algorithms compared to other techniques for pipe optimization', *Journal of Water Resources Planning and Management* **120**(4), 423–443.

Sivanandam, S. N. and Deepa, S. N. (2008), *Introduction to Genetic Algorithms*, Springer-Verlag Berlin Heidelberg.

Skok, M., Skrlec, D. and Krajcar, S. (2002), Genetic algorithm and GIS enhanced long term planning of large link structured distribution systems, *in* D. Skrlec, ed., 'Power Engineering 2002 Large Engineering Systems Conference on LESCOPE 02', pp. 55–60.

Sreedhar, V. C., Gao, G. R. and Lee, Y.-F. (1996), 'Identifying loops using dj graphs', *ACM Transactions on Programming Languages and Systems (TOPLAS)* **18**(6), 649–658.

Srinivas, M. and Patnaik, L. M. (1994), 'Genetic algorithms: A survey', *IEEE Computer Society* **27**(6), 17–26.

Steeb, W.-H. (2005), *The nonlinear workbook*, World Scientific Publishing Company, Incorporated.

Steger, C. W. (2002), Urban design, *in* J. M. Levy, ed., 'Contemporary Urban Planning', Prentice Hall, Englewood Cliffs, New Jersey, pp. 127–149.

Steiss, A. W. (2002), Capital facilities planning, *in* J. M. Levy, ed., 'Contemporary Urban Planning', Prentice Hall, Englewood Cliffs, New Jersey, pp. 151–168. Capital Facilities Planning.

Stewart, T. J., Janssen, R. and van Herwijnen, M. (2004), 'A genetic algorithm approach to multi-objective land use planning', *Computers & Operations Research* **31**(14), 2293–2313.

Suh, J. Y. and Van Gucht, D. (1987), Incorporating heuristic information into genetic search, *in* 'Proceedings of the Second International Conference on Genetic Algorithms and their application', L. Erlbaum Associates Inc., Hillsdale, NJ, USA, pp. 100–107.

Swamee, P. K. and Sharma, A. K. (2008), *Design of water supply pipe networks*, John Wiley & Sons, Inc.

Syswerda, G. (1989), Uniform crossover in genetic algorithms, *in* 'Proceedings of the third international conference on Genetic algorithms', Morgan Kaufmann Publishers Inc., George Mason University, United States, pp. 2–9.

Syswerda, G. (1991), 'A study of reproduction in generational and steady-state genetic algorithms', *Foundation of Genetic Algorithms* pp. 94–101.

Taher, S. A. and Labadie, J. W. (1996), 'Optimal design of water-distribution networks with GIS', *Journal of Water Resources Planning and Management* **122**(4), 301–311.

Tarjan, R. (1973), Testing flow graph reducibility, ACM, Austin, Texas, United States, pp. 96–107.

Tate, D. and Smith, A. (1993), Expected allele coverage and the role of mutation in genetic algorithms, *in* 'Proceedings of the Fifth International Conference on Genetic Algorithms', Morgan Kaufmann Publishers Inc., pp. 31–37.

The GeoNames Geographical Database (2011). Accessed at [15^{th} July 2011].
URL: *www.geonames.org*

Tolson, B. A., Maier, H. R., Simpson, A. R., M.Asce, Lence, B. J. and A.M.Asce (2004), 'Genetic algorithms for reliability-based optimization of water distribution systems', *Journal of Water Resources Planning and Management* **130**(1), 63–72.

Vairavamoorthy, K. and Shen, Y. (2004), Least cost design of water distribution network using particle swarm optimisation, *in* P. . B. Liong, ed., 'Proceedings of the 6th International Conference on Hydroinformatics: Singapore, 21-24 June 2004', Vol. 1, World Scientific Publishing.

van Dijk, S., Thierens, D. and de Berg, M. (2002), 'Using Genetic Algorithms for Solving Hard Problems in GIS', *GeoInformatica* **6**(4), 381–413.

Van Zyl, J. E., Savic, D. A. and Walters, G. A. (2004), 'Operational optimization of water distribution systems using a hybrid genetic algorithm', *Journal of Water Resources Planning and Management* **130**(2), 160–170.

Vasan, A. and Simonovic, S. P. (2010), 'Optimization of water distribution network design using differential evolution', *Journal of Water Resources Planning and Management* **136**(2), 279–287.

Vitkovsky, J. and Simpson, A. (1997), 'Calibration and leak detection in pipe networks using inverse transient analysis and genetic algorithms', *Department of Civil and Environmental Engineering, University of Adelaide: Adelaide, Australia* .

Walski, T., Chase, D., Savic, D., Grayman, W., Beckwith, S. and Koelle, E. (2003), *Advanced water distribution modeling and management*, Haestad Press.

Walski, T. M., Brill, J. E. D., Gessler, J., Goulter, I. C., Jeppson, R. M., Lansey, K., Lee, H.-L., Liebman, J. C., Mays, L., Morgan, D. R. and Ormsbee, L. (1987), 'Battle of the network models: epilogue', *Journal of Water Resources Planning and Management, ASCE* **113**(2), 191–203.

Walski, T. M., Chase, D. V. and Savic, D. A. (2001), *Water distribution modelling*, Haestad Methods, Inc.

Walski, T., Youshock, M. and Rhee, H. (2000), Use of modeling in decision making for water distribution master planning, *in* 'Building Partnerships - 2000 Joint Conference on Water Resource Engineering and Water Resources Planning & Management'.

Walters, G. A., Halhal, D., Savic, D. and Ouazar, D. (1999), 'Improved design of "anytown" distribution network using structured messy genetic algorithms', *Urban Water* **1**(1), 23–38.

Walters, G. and Lohbeck, T. (1993), 'Optimal layout of tree networks using genetic algorithms', *Engineering Optimization* **22**(1), 27–48.

Wang, S. Y. and Tai, K. (2005), 'Bar-system representation for topology optimization using genetic algorithms', *Engineering Computations: International Journal for Computer-Aided Engineering* **22**(2), 206–231.

Wang, Y. and de Silva, C. (2008), 'A machine-learning approach to multi-robot coordination', *Engineering Applications of Artificial Intelligence* **21**(3), 470–484.

Ward, S. V. (2004), *Planning and urban change*, Sage Publications Ltd.

Weise, T. (2009), *Global optimization algorithms: Theory and application*, Free Software Foundation.

Wheeler, S. (1998), Planning sustainable and livable cities, *in* R. LeGates and F. Stout, eds, 'The City Reader', London: Routledge, pp. 434–445.

Whitley, D. (1989), The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best, *in* 'Proceedings of the third international conference on Genetic algorithms', Vol. 1, Citeseer, pp. 116–121.

Wu, Y., Lee, G., Fu, X., Soh, H. and Hung, T. (2009), 'Mining weather information in dengue outbreak: Predicting future cases based on wavelet, SVM and GA', *Advances in Electrical Engineering and Computational Science* **39**(1), 483–494.

Wu, Z., Boulos, P., de Schaetzen, W., Orr, C.-H. and Moore, M. (2001), Using genetic algorithms for water distribution system optimization, *in* 'Bridging the Gap: Meeting the Worlds Water and Environmental Resources Challenges', Vol. 111, ASCE, Orlando, Florida, USA, pp. 409–409.

Xu, C. and Goulter, I. C. (1999), 'Reliability-based optimal design of water distribution networks', *Journal of Water Resources Planning and Management* **125**(6), 352–362.

Yaakup, A. and Sulaiman, S. (2002), GIS as tools for monitoring the urban development in metropolitan region: A case of Klang valley region, Peninsular Malaysia, *in* 'Proceedings 6th Seminar on GIS for Developing Countries (GISDECO 2002)', ITC, Enschede, Netherlands.

Yeh, A. G.-O. (1991), 'The development and applications of geographic information systems for urban and regional planning in the developing countries', *International Journal of Geographical Information Science* **5**(1), 5–27.

Zecchin, A., Maier, H., Simpson, A., Leonard, M. and Nixon, J. (2007), 'Ant colony optimization applied to water distribution system design: comparative study of five algorithms', *Journal of Water Resources Planning and Management* **133**(1), 87–92.

Zecchin, A., Simpson, A., Maier, H. and Nixon, J. (2005), 'Parametric study for an ant algorithm applied to water distribution system optimization', *IEEE Transactions on Evolutionary Computation* **9**(2), 175–191.

Zegras, C., Sussman, J. and Conklin, C. (2004), 'Scenario planning for strategic regional transportation planning', *Journal of Urban Planning and Development* **130**(1), 2–13.