Artificial Intelligence Techniques for Flood Risk Management in Urban Environments

Submitted by

William Keith Paul Sayers

To the University of Exeter as a Thesis for the degree of

Doctor of Engineering in Water Engineering

November 2015

I certify that all material in this thesis which is not my own work has been identified and that no material has previously been submitted and approved for the award of a degree by this or any other university.

……………………………..

Abstract

Flooding is an important concern for the UK, as evidenced by the many extreme flooding events in the last decade. Improved flood risk intervention strategies are therefore highly desirable. The application of hydroinformatics tools, and optimisation algorithms in particular, which could provide guidance towards improved intervention strategies, is hindered by the necessity of performing flood modelling in the process of evaluating solutions. Flood modelling is a computationally demanding task; reducing its impact upon the optimisation process would therefore be a significant achievement and of considerable benefit to this research area. In this thesis sophisticated multi-objective optimisation algorithms have been utilised in combination with cutting-edge flood-risk assessment models to identify least-cost and most-benefit flood risk interventions that can be made on a drainage network. Software analysis and optimisation has improved the flood risk model performance. Additionally, artificial neural networks used as feature detectors have been employed as part of a novel development of an optimisation algorithm. This has alleviated the computational time-demands caused by using extremely complex models. The results from testing indicate that the developed algorithm with feature detectors outperforms (given limited computational resources available) a base multi-objective genetic algorithm. It does so in terms of both dominated hypervolume and a modified convergence metric, at each iteration. This indicates both that a shorter run of the algorithm produces a more optimal result than a similar length run of a chosen base algorithm, and also that a full run to complete convergence takes fewer iterations (and therefore less time) with the new algorithm.

Acknowledgements

First, I would like to thank my academic supervisors, Dragan Savic and Zoran Kapelan, and my industrial supervisor, Richard Kellagher, for their help, guidance, and hard work on my behalf throughout the course of my EngD. I could not have achieved this without their assistance.

I would also like to thank my colleagues at HR Wallingford, and particularly the members of the floods and water groups. The floods group welcomed me into their team and included me as one of their own, and the water group provided a large amount of essential advice and assistance.

Additionally, I am very grateful to the members of staff and fellow researchers at the University of Exeter and particularly the Centre for Water Systems, who have aided me and helped me to get to where I am today.

I wish to thank the STREAM-IDC programme and the EPSRC for making this project possible, as well as all the events and courses they have arranged and/or funded as part of my EngD experience.

Finally, I wish to thank my parents, family and friends for their support and encouragement, and particularly my partner Louise for her love, support, encouragement and faith in me throughout this project.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	**Definition**
ADAPT	Advanced drainage analysis and planning tool
ANN	Artificial neural network
BAK	BakRyan network
BIN	Balerma irrigation network
BLA	Blacksburg network
Capex	Capital Expenditure
CPU	Central Processing Unit
CSO	Combined Sewer Overflow
DTI-SAM	Department of trade and industry – system-based analysis and management
EAD	Expected annual damage

Abbreviation	Definition
EXN	Exeter network
FOS	Fossolo network
GBAS	Graph based ant system
GOY	GoYang network
HAN	Hanoi network
LEM	Learning evolution model
LEMMO	Learning evolution model for multiple-objective optimisation
LoS	Level of Service
MOD	Modena network
MOGA	Multi objective genetic algorithm
MOGA-ANN	Multi-objective genetic algorithm with adaptive neural networks

Abbreviation	Definition
NP	Non-deterministic polynomial-time set
NP-Hard	Non-deterministic polynomial-time hard
NSGA-II	Non dominated sorting genetic algorithm II
NYT	New York tunnel network
PAES	Pareto archived evolution strategy
PES	Pescara network
RAM	Random Access Memory
RFSM	Rapid flood spreading model
RPM	Revolutions per minute
RPROP	Resilient back propagation algorithm
SAM-Risk	System based analysis and management risk tool
SAM-Risk II	System based analysis and management risk tool II

Abbreviation	Definition
SAM-UMC	System based analysis and management – Urban model control
SPEA	Strength pareto evolutionary algorithm
SPEA2	Strength pareto evolutionary algorithm 2
TLN	Two-loop network
TRN	Two-reservoir network
VEGA	Vector evaluated genetic algorithm
WDS	Water Distribution System

List of Terms

Term	**Definition**
Algorithm	An algorithm is a given set of repeatable steps that can be performed to achieve a desired outcome.
Convergence	The point at which an optimisation algorithm has identified the most optimal point/set of points that it will manage to for a given run.
Dominated Hypervolume / S-Metric	The “dominated hypervolume” or “S-Metric” is a measure of the volume or hypervolume dominated by the estimated Pareto front.
Heuristic	A heuristic is a method of obtaining a solution to a given problem that will not be optimal, but should be “good enough” to be used in place of having the known-correct answer.
Hypervolume	A hypervolume is a region defined by more than three dimensions.

Term	Definition
Impact Zone	RFSM in pre-processing identifies a number of impact zones. These represent topographical depressions where water will collect in case of flooding.
Non-deterministic polynomial-time hard	Non-deterministic polynomial-time hard (NP-hard) problems, are problems that are at least as hard as the hardest problems in the set NP.
Non-deterministic polynomial-time set	Usually referred to as NP, this is the set of computational problems for which a given solution can be verified as a solution in polynomial time by a Turing machine.
Objective Function	This term described the function within an optimisation algorithm that evaluates how well a given set of variables solve a given problem.
Optimisation	Optimisation in computing means either the process of modifying an existing algorithm to improve its performance, or describes an algorithm that is designed to produce a solution

Term	Definition
	to a given problem by altering variables describing the problem to improve the answer to an “objective” function.
Pareto front / set / frontier	The Pareto front is the sub-set of solutions that produce the optimum values for all objective functions, satisfying given constraints.

1. Introduction

1.1 General Introduction

Flooding is a natural disaster that can have extreme consequences on communities and the people within them. In the UK, there is a risk of flooding from rivers, the sea, groundwater, reservoirs and surface water (Environment Agency, 2009), which can all have a devastating impact on individuals and communities. The UK has experienced widespread flooding in recent years, most notably 2007, 2009, 2012 and 2014 (Centre for Ecology & Hydrology, 2014; JBA Risk Management Limited and Met Office UK, 2012; Marsh and Hannaford, 2007; Smythe, 2013).

Climate change, ageing flood-prevention infrastructure and methodologies, and socio-economic circumstances have all played a part in these events and how they have impacted the UK. Additionally, the majority of the UK drainage networks are still combined systems, handling both waste water and storm water (Marsalek et al., 1998). This leads to massive additional pressure on drainage systems during rainfall events. Looking to the future, and taking into account the trend across the time-scale of this research, it seems likely that more extreme weather events will become a fact of life and the UK, therefore, must adopt a pro-active approach to improving the way in which it manages the risks of flooding.

One of the key areas of flood risk management is the identification of intervention strategies that can be applied to drainage networks, in order to reduce the flood risk associated with those networks. Inevitably, the bodies concerned with applying these intervention strategies to the networks will be interested in ensuring that they maximise the return on their investment or in other words, achieve the maximum reduction of risk of flooding for the amount of money they are investing. Additionally, they will be keen to ensure that they make informed and justifiable choices in terms of identifying the optimal investment point to target.

The main method of achieving these goals to date has been with the aid of human engineers who invest a great deal of time into providing information to decision makers and making sure that they can have confidence in the information they have provided. The problem solving approach can involve computational modelling (Marsalek et al., 1998), examination of previous approaches used in situations that are related in some way, and a considerable amount of engineering experience and knowledge being applied. This thesis aims to advance the development of flood risk intervention strategies for urban drainage networks.

As computers lack human intuition for what makes a “good” or “bad” solution, the only way to truly solve a problem for computers is to perform an exhaustive analysis of every possible solution. In situations where this is not possible (due to time or computing power constraints), heuristic algorithms are generally used, which are aimed at producing a “good enough” solutions. One such type of heuristic algorithm is a “genetic algorithm” (see section 2.3.2).

As an example of the computing power/time issue, if a genetic algorithm were applied to the problem in some way, assuming a run of twenty thousand iterations, and a population of two hundred possible solutions, there would be four million total analyses involved. If each of those analyses took an average of only two minutes, the complete run time would be in excess of fifteen years.

In order to solve these problems, this thesis has utilised multi-objective optimisation techniques. These techniques develop a Pareto front allowing an expert user to give guidance, as to prime investment points, and the improvement in (reduction of) flood risk potentially available at each investment point. In order to allow these techniques to complete within a reasonable period, it has been necessary to investigate heuristic methods of decreasing the computational cost of the necessary objective functions. A problem with these methods is the reduction in accuracy that is necessarily a part of the way they operate. Care must be taken in their application to ensure that as little as possible accuracy is lost in the process.

In order to gain the benefits that these technologies can supply, this thesis describes and demonstrates the application of an object oriented software solution utilising these methods, which could guide engineers in the process of developing investment options and advising their clients/decision-makers as to the appropriate investment points available to them.

It should be noted that all experimentation carried out as part of this thesis was performed on a modern desktop computer system, comprising an Intel i5 750 four core central processing unit (CPU) with 8 gigabytes of RAM and a 7500 RPM hard disk drive. Any time estimates, where given, are based on this computer system.

1.2 Aims and Objectives of Research

The main aim of this thesis is to describe the development of a new methodology for drainage system flood risk management, which will aid in the identification of optimal investment points, as well as the maximum potential reduction in risk for a given investment. Broken down into a list of individual objectives this comprise:

1. Identification of a multi-objective optimisation algorithm to use as a starting and comparison point for the process of developing an improved optimisation algorithm targeted at complex flood risk analysis objectives. The algorithm and the objectives for the algorithm were to be custom-developed rather than used from a library, as they had to be suitable for usage within other applications in the wider HR Wallingford software domain. Algorithms from a library could have had licensing issues, and would not have been as well integrated as a custom developed algorithm. Due to this custom development, it would have been prohibitively time-consuming to try several as part of this thesis, so literature on similar optimisation problems has been examined to identify a suitable base algorithm.
2. The development and testing of a benchmark multi-objective algorithm to ensure best-performance on extremely complex, NP-hard problems. Due to the interactions of surface-flow and drainage system flow, as well as the complexities of drainage system flow analysis, drainage system flood risk problems fit within this category of problem. Additionally, this had to be implemented through an object-oriented structured software engineering

approach with a suitable user-interface, as one of the requirements for this EngD is software that can be further utilised in practice.

3. To formulate the overall optimisation problem for the multi-objective optimisation algorithm. Including, objectives that best described the drainage system flood risk management problem based on expected annual damage and capital cost of intervention strategy; constraints that sensibly limited proposed solutions; and the decision variables that made up these proposed solutions.
4. To improve the computational efficiency of the optimisation process. Initially this involved testing the performance of the multi-objective genetic algorithm and investigating methods for reducing the computational burden of the optimisation. Heuristic methods, such as classifier-based meta-models, for improving the computational efficiency of an objective function within an optimisation were then investigated. One or more of these heuristic methods was then to be developed and tested thoroughly with regard to computational efficiency, and the impact on accuracy of results.
5. To test and verify computational efficiency and effectiveness of the new methodology on a real case study involving drainage system flood risk optimisation.

1.3 Structure of Thesis

This thesis contains seven chapters including this introductory chapter. This chapter identifies the purpose of the research being undertaken and gives a brief overview of the nature of the problems being investigated and the structure of the rest of the thesis.

In chapter two, a review of literature is undertaken. This review covers flood risk management techniques, optimisation algorithms, multiple-objective versions of optimisation algorithms and finally machine learning techniques.

In chapter three, a pre-existing flood risk analysis system is described. This system will form the foundations of the flood risk analysis software solution being developed for this thesis. Optimisations can be performed to the original implementation of this flood risk analysis system. Testing of said optimisations to ensure improved efficiency with no drop in accuracy will then be described.

In chapter four, the implementation of optimisation algorithms and machine learning techniques for the purposes of this thesis is examined. Improvements made to the existing tool set are identified, and the performance gains examined. The structure of the optimisation tool set and the manner in which different segments of it are constructed is then examined. Finally, the methodology behind the optimisation process, including the selection of a reduced rainfall set to decrease expected annual damage (EAD) calculation time is examined and discussed.

In chapter five, several test cases are examined. These are water distribution system problems on which several optimisation algorithms have been run. The results from these optimisation algorithms have been combined and analysed to give estimated Pareto fronts (Wang et al., 2014). These Pareto fronts are intended for use as reference fronts for the purposes of analysing and comparing optimisation algorithm performance. These test cases were selected as they are fairly similar to drainage design problems in terms of variables required and suitable measurements of system performance, less computationally intensive and therefore considerably less time consuming to optimise upon.

In chapter six, a fully functioning drainage system model of a real-world catchment is analysed utilising the techniques and methodologies that have been so far identified, implemented and tested. The performance of the techniques and methodologies developed during this thesis can therefore be shown to have suitability to real-world problems and models.

In chapter seven, the findings of this thesis are identified and discussed, any conclusions that can be drawn from those findings are likewise identified and examined, and any future work that could be carried out to further progress knowledge in this area is identified.

2. Literature Review

2.1 Introduction

Urban flooding is estimated to cost two hundred and seventy million pounds per year in England and Wales combined, with eighty thousand homes at risk (Parliamentary Office of Science and Technology, 2007). It is usually caused by rainfall overwhelming combined drainage systems, rivers overflowing due to excessive surface run off, or a combination of the two. In addition to these monetary costs, flooding presents various health risks (Fewtrell and Kay, 2008), which strongly affect the quality of life of individuals even after the flooding event has itself passed. Looking at these facts, it is clear that there is a need to develop improved methods for identifying the most suitable intervention strategies for flood risk reduction in given urban areas.

The modelling of urban flood risk is a highly computationally intensive problem, due to the complex nature of urban flood plains and hydrology. Any intervention strategy identified must be accurate in terms of the benefits it proposes. At the same time, any algorithm that purports to develop these intervention strategies to reduce flood risk must be efficient enough to allow for risk analysis and options appraisal within a reasonable time frame.

HR Wallingford has previously developed the System-based Analysis and Management of urban flood risks software (DTI-SAM) in partnership with the Department for Business, Enterprise, and Regulatory Reform (previously the Department of Trade and Industry) along with several other interested parties and completed in 2009 (Kellagher et al., 2009). This project's goal was to develop a risk-based methodology and tool set for assessing the average flood damage per year that is likely to be incurred within a given area in terms of Expected Annual Damage (EAD), which is a cost measure, based on pounds sterling in this project. Additionally, this tool set and methodology would identify what proportion of this EAD value was due to each drainage asset within the catchment area. As of project completion, these goals were successfully achieved.

This thesis builds upon this previous project by refining, optimising and improving the developed tool set, using optimisation techniques to identify a range of Pareto optimal intervention strategies based on the estimated cost (in terms of implementing the changes from the original drainage network) and the EAD values of the assessed intervention strategies.

2.2 Flood Risk Management

2.2.1 Introduction

As previously mentioned (see section 2.1) flooding has a substantial economic impact (Parliamentary Office of Science and Technology, 2007) and is associated with extensive health risks (Fewtrell and Kay, 2008). Flood risk assessment serves as an important tool for ensuring actions taken minimise flood risk insofar as it is possible to do so, within the constraints applicable to a given situation.

In England and Wales, flood risk assessment is highly important for insurance purposes for buildings, developments and for maintenance planning. Wherever a planning application is submitted, a flood risk assessment is also required if the land in question is within a zone of flood risk (flood zones 2 or 3) and/or is greater than one hectare of land (Department for Communities and Local Government, 2010).

Flood risk assessments are usually carried out utilising a design storm approach (Hromadka and Whitley, 1988), which means that the drainage systems are designed to have the capacity to successfully stand up to the largest rainfall event expected within a given number of years, without incurring flooding events. This approach has limitations in that the method of assessment of performance involves no analysis of how well the drainage system performs, but is a test of whether or not the system meets a certain criterion, and thus does not lend itself to being utilised as part of an optimisation process. The latter requires an objective function that can identify how good a solution is, as opposed to simply whether or not it meets certain criteria.

In recent years, the focus has moved towards risk-based flood risk assessment (Kellagher et al., 2008; Sayers et al., 2002). The key elements of a risk-based approach are as follows:

- Under a risk-based approach, the system being analysed is evaluated in terms of the consequences of the failure of that system, rather than in terms of the severity of the rainfall event on which failure is likely to start occurring.
- A risk-based approach allows comparison of intervention options based upon how well those options mitigate the consequences of flooding, should flooding occur. Merely optimising on whether flooding has

occurred is not an easily differentiable problem, i.e. there is no way to tell if there is a difference between two solutions which both cause flooding. This is a problem as a solution that results in a few inches of flooding and minimal damage would clearly be preferable to one that results in six foot of water and extensive harm to infrastructure.

- When utilising a risk-based approach, all events can be analysed regardless of the likelihood of their occurrence. In a design storm approach, the system is analysed with regard to one specific system load.

2.2.2 Urban Flood Risk Analysis

Urban flooding is a particularly challenging problem to analyse, due to the complex nature of urban flood plains and rainfall. Urban flood plains are effectively a series of highly complex, interconnected channels at varying levels, with the impermeable nature of various elements of urban terrain and buildings further complicating analysis. A huge amount of research has been performed in this field in recent years (Ashley et al., 2008; Bach et al., 2014; Chen et al., 2012a, 2012b; Djordjević et al., 1999; Fedeski and Gwilliam, 2007; Garcìa et al., 2015; Ghimire et al., 2013; Gires et al., 2012; Kubal et al., 2009; Leandro et al., 2011; Maksimović and Prodanović, 2001; Rodríguez et al., 2013) and knowledge of the area is rapidly improving although there remains much research to be done.

A conceptually similar project to the one undertaken in this thesis was recently completed (Woodward, 2012; Woodward et al., 2013a, 2013b). The techniques proved extremely useful when applied to the problems within this thesis. Particularly, the identification of a reduced rainfall set in order to improve the computational efficiency of an objective being utilised in a multi-objective optimisation algorithm.

As mentioned in section 2.1 the work in this thesis is based upon a previous project, which succeeded in its goal of producing a methodology and supporting computer tool set to perform risk-based flood risk assessment on large catchment areas (Kellagher et al., 2009). A substantial amount of tool development was involved in this thesis, built partially upon this tool set, as the developed tool set was extremely computationally intensive to run and unsuitable for inclusion within a multi-objective optimisation algorithm. Additionally, the multi-objective algorithm itself had to be developed and build, plus additional components on top of that to improve efficiency and maintain accuracy. Finally, this software all had to be developed in such a way that it was suitable for inclusions in other projects, could support decision making, and was generally fit for purpose.

The finalised tool set for DTI-SAM included, SAM-Risk 1 and 2 (Kellagher et al., 2009), SAM-UMC (Wills, 2013), Rapid Flood Spreading Model (RFSM) (Lhomme et al., 2008), Infoworks CS, and a set of design-storm or time-series based rainfall data that allows for drainage system simulations to be run within Infoworks (Kellagher et al., 2009). This software formed the basis of core aspects of the software developed during this thesis.

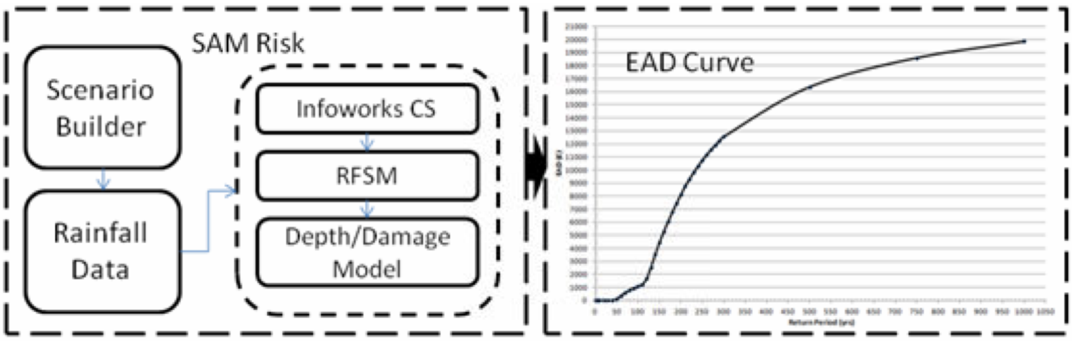


Figure 1 - DTI SAM toolset structure (Kellagher et al., 2009)

SAM Risk (see Figure 1) 1 and 2 are the user-interface and controller for the tool set, building the scenarios for SAM-UMC to process. SAM Risk 1 assumes full system functionality (no blocked or collapsed drains), whereas, SAM Risk 2 allows for the possibility of parts of the drainage system collapsing, or becoming blocked. The chance of a drain collapsing, or blocking, is calculated using an empirical equation (Long, 2008) for calculating pipe-failure probability. SAM-UMC then functions as an interface to the Infoworks CS COM object (Innovyze, 2007), which allows for the loading and running of simulations, from which the results can then be exported and collected. The rapid flood-spreading model distributes the excess water at each manhole (identified during the Infoworks CS run) over the terrain model.

This project uses the SAM Risk 1 model, without taking into account the potential for drainage system collapse and blockage. SAM Risk 2 has a far larger demand on computational time, but produces very similar results to SAM Risk 1 on most networks (Kellagher, 2010). Further information on the function of the DTI-SAM tool set and calculation of risk can be found in chapter 3.

2.2.3 Flood Risk Management Summary

In summary, risk-based methods of assessing flood risk are becoming more mainstream, and allow for a consequence-based evaluation of a flooding event, which gives a useful scale to measure how good a given drainage system is, compared to another. The DTI-SAM project previously completed at HR Wallingford has produced a useful tool set and methodology, which this project can build upon and incorporate in order to optimise flood-risk interventions.

2.3 Optimisation Algorithms

2.3.1 Introduction

There are many kinds of optimisation algorithm, including: linear programming (Schrijver, 1998), integer programming (Schrijver, 1998), non-linear programming (Bertsekas, 1999), gradient descent algorithms (Baldi, 1995), evolutionary algorithms (Back, 1996) and swarm algorithms (Brownlee, 2012). In this chapter search algorithms are the main focus, which encompass some of the aforementioned types of algorithm. Search algorithms are designed to identify a desirable item, or items, amongst a superset of items.

There are two main forms of search algorithm, deterministic algorithms, which are effective at solving simple problems and non-deterministic algorithms which are less efficient on simple problems. Non-deterministic algorithms can, however, solve considerably more complex problems that deterministic algorithms would either not converge on, or would not converge within a reasonable time. Deterministic algorithms encompass hill climbing or gradient descent algorithms

(Burges et al., 2005), the A* search algorithm (Liu and Gong, 2011) and TABU search (de Werra and Hertz, 1989; Gendreau and Potvin, 2005; Glover et al., 1993; Glover and Laguna, 1997; Hertz and de Werra, 1991; Soriano and Gendreau, 1996) amongst others. Non-deterministic algorithms encompass genetic algorithms (De Jong, 1975; Holland, 1975, 1962; Schaffer, 1985), simulated annealing (Kirkpatrick et al., 1983), and ant-colony optimisation (Dorigo et al., 1997; Dorigo and Blum, 2005; Dorigo and Stutzle, 2004; Gutjahr, 2000) amongst, again, a multitude of others.

The problem this project is attempting to solve is highly complex and based on literature consulted, it is likely that deterministic algorithms would fail to converge to a good answer on the problem which this thesis solves in a reasonable time. This is because that problem is a non-linear, combinatorial, NP-hard problem, which are all characteristics of problems which deterministic algorithms do not solve well (Garey and Johnson, 1979). Therefore, deterministic algorithms will be ignored entirely, and only a brief overview of single-objective non-deterministic algorithms follows to set the scene, followed by an examination of the development and application of the powerful multi-objective techniques that would have a chance of obtaining a reasonably good answer to this project's problem within a sensible time frame.

2.3.2 Genetic Algorithms

Genetic algorithms are based loosely upon Darwin's theory of natural selection (Darwin, 1859), which suggests (to simplify greatly) that organisms evolve based on the more useful elements of an organism enabling that organism to breed more, thus passing on those elements to some or all of its offspring, who will, again, breed more. Eventually the organisms with the useful traits will either supplant or exist alongside the original organisms. These useful elements would initially be produced by a mutation of existing elements, before being passed down parent to child in this manner. Genetic algorithms are immensely popular optimisation algorithms due to their suitability for non-linear, non-convex, multi-modal and discrete problems with which traditional gradient descent derived algorithms may perform poorly in comparison (Nicklow et al., 2010).

The process of a genetic algorithm can be separated into four distinct sub-processes: generation, selection, crossover and mutation. Generation involves building an initial population of potential solutions either by random creation or some other method. Selection is where the population is evaluated, and then a subset of that population is selected by one of many possible selection algorithms (e.g., fitness proportionate (Back, 1996), stochastic universal sampling (Ghimire et al., 2013), tournament selection (Miller and Shaw, 1996; Nicklow et al., 2010) etc.) for the generation of a child population via the next stages of crossover and mutation. Most modern implementations use either truncation or tournament selection (Nicklow et al., 2010) as these are scaling invariant and inherently elitist, which has been shown (Bayer and Finkel, 2004; Reed et al., 2000; Yoon and Shoemaker, 2001) to enhance the effectiveness of the genetic algorithm. Crossover is the process of generating new chromosomes by combining aspects from previous solutions chosen by the selection algorithm via one of several possible crossover algorithms (single-point, multi-point, uniform, partially mapped crossover, etc.) in the hope of producing a “child” chromosome more fit then either of its "parent" chromosomes. Mutation involves introducing a chance of making random changes to the chromosomes, which helps to prevent premature convergence and allow a fuller exploration of the search space by including genes that were not present in the initial random population. A final process that is not essential to the function of the algorithm, but which vastly improves its efficiency and effectiveness, is called "elitism". Elitism ensures that the best scoring chromosomes from each population make the transition from parent to child population intact. This ensures that promising search areas are not lost to the algorithm part way through iteration.

Genetic algorithms have been around since the 1960's (Holland, 1962). However, they only began to gain wider acceptance as an effective and efficient optimisation strategy in 1975. This was due to both the publication of "Adaptation in Natural and Artificial Systems" (Holland, 1975) and the thesis entitled "An analysis of the behaviour of a class of genetic adaptive systems" (De Jong, 1975).

Holland (1975) presented the concept of adaptive algorithms utilising the concepts of mutation, selection and crossover, and De Jong (1975) showed that genetic algorithms could perform exceptionally well on discontinuous and noisy data that is challenging for many other optimisation techniques. Genetic algorithms have been utilised on many and varied problems since their development (Goldberg and Wang, 1997; Huang et al., 2009; Montana and Davis, 1989; Santarelli et al., 2006; Scully and Brown, 2009) with good success rates.

2.3.3 Simulated Annealing

The annealing process in metalworking inspired the simulated annealing algorithm. Annealing is the process of heat-treating metal to achieve desired properties within the material by heating it up and then allowing it to cool very slowly. Annealing occurs because over time, the atoms within the metal align themselves towards the equilibrium state when the bonds between atoms have been broken (hence the heat).

Simulated annealing (Kirkpatrick et al., 1983) is a computational emulation of this process, in order to apply it to optimisation problems. A temperature is tracked within the algorithm, beginning at a high level and gradually decreasing throughout the execution of the algorithm. The algorithm usually halts when the temperature reaches a pre-determined level. Initially, one solution to the problem in question is generated and the score obtained through the objective function represents the “energy” of that particular state. At each cycle of the algorithm, this state is altered to generate a new state. This new state is then evaluated and if its energy is lower, it replaces the current state. If the energy of the new state is higher, then it still may replace the current state, but that is based upon chance influenced by the current temperature and the difference in energy. As the temperature lowers, the chance of inferior solutions replacing the main solution drops swiftly (Smith and Savić, 2006).

It has been proven (Geman and Geman, 1984) that with a sufficiently drawn out cooling schedule the simulated annealing algorithm will always converge to the best possible solution. Most implementations of simulated annealing are, however, on far faster cooling schedules in order to be of use in providing an answer within a reasonable time frame. Simulated annealing does generally perform well over shorter intervals, however, and has proven to be an extremely effective solution for single objective optimisation problems.

2.3.4 Ant-Colony Optimisation

Ant-colony optimisation is a relative newcomer to the field of optimisation algorithms, having been first introduced in the early 1990's by M. Dorigo and his colleagues in Italy (Dorigo, 1992; Dorigo et al., 1997, 1991) as an algorithm for solving combinatorial optimisation problems. Like evolutionary algorithms, ant-colony optimisation is a meta-heuristic algorithm inspired by nature. In this case, by the methods that ants in the natural world use to guide other members of their colony to discovered food sources, i.e., pheromone trails.

When ants are exploring an area around their nest for food, they initially explore in a wholly random manner. When an individual ant discovers a food source, it evaluates the quantity and quality of this food source, and then carries a portion back to the colony's nest leaving a pheromone trail behind it. The pheromone trail varies depending upon the quantity and quality of the food source. Other ants are attracted to follow this trail and will then discover the food and leave their own pheromone trail. The pheromones laid to mark these trails evaporate over time, so over time longer trails will become weaker than shorter ones and attract fewer ants in consequence. In this manner, although no direct communication has taken place, greater quantities of ants will be drawn towards the best food sources, the shortest distance from the nest (Dorigo and Blum, 2005).

Ant-colony optimisation has been applied to various problems successfully (Dorigo and Stutzle, 2004) since it was developed. A proof of convergence focusing on a particular implementation of ant-colony optimisation, called “Graph Based Ant System (GBAS)” to an optimum solution was published in 2000 (Gutjahr, 2000). This was followed by a more generalised proof of convergence to any optimal solution in 2002 (Gutjahr, 2002). Practical applications of GBAS have, however, been rare (Dorigo and Blum, 2005) and work continued on proving convergence of more commonly used ant-colony algorithms with an included positive lower bound. This has finally been completed with a proof for convergence in value and solution (Dorigo and Stutzle, 2004, 2002).

Ant-colony optimisations main advantage over evolutionary algorithms, or other optimisation techniques, lies in its ability to be run on-line and swiftly compensate for live alterations to the problem being solved. It can be used with great effect for route planning, network planning, and similar problems, due to these capabilities. Evolutionary algorithms do, however, have a longer record of accomplishment and are considered a safer option, particularly where the problem has no element of volatility and does not have to be solved on-line.

2.3.5 Multi-Objective Optimisation

The vast majority of optimisation algorithms are designed around the idea of a single objective; therefore, there is a proliferation of highly efficient, accurate algorithms to deal with single-objective problems that are highly documented (De Jong, 1975; Dorigo, 1992; Dorigo et al., 1997; Holland, 1975, 1962; Kirkpatrick et al., 1983). The field of multi-objective optimisation is more challenging and more useful. Most real-world problems do not involve simply trying to find the most optimal approach to a problem to achieve one fixed solution. They are a matter of weighing different options against each other, based on several separate criteria, and attempting to select the approach with the most usable balance.

Multi-objective optimisation algorithms are designed to work with more than one objective function. The algorithm works to minimise or maximise each of the objectives simultaneously. The objective functions are usually in conflict with each other, as if there was no conflict between the two objective functions it would be more efficient and possibly more accurate to develop a number of single-objective optimisation algorithms and find the optimum value for each objective in this way (Coello, 1999).

The issue of multi-objective optimisation is defined by (Coello, 1999) as finding the vector described in equation 1 where 2 and 3 are satisfied, the vector function in 4 is optimised, and the decision variable vector is as shown in 5.

$$\bar{X} = \left[x^*_1, x^*_2, \dots, x^*_n\right] \tag{1}$$

$$\begin{gathered} g_i(\bar{x}) \geq 0 \\ i = 1,2,3, \dots, m \end{gathered} \tag{2}$$

$$h_i \bar{x} \geq 0$$
$$i = 1,2,3, \ldots, p \quad (3)$$

$$\bar{f}(\bar{x}) = [f_1(\bar{x}), f_2(\bar{x}), \ldots, f_k(\bar{x})]^T \quad (4)$$

$$\bar{x} = [x_1, x_2, \ldots, x_n]^T \quad (5)$$

In words, multi-objective optimisation is the problem of finding from a given set, which satisfies the constraints listed in 2 and 3, the sub-set that is composed of the optimum values of all objective functions. This set is known as a “Pareto set” (Pareto, 1896), the non-inferior or non-dominated sets, which contain Pareto optimal solutions. A point is considered Pareto optimal if no vector exists which would improve the score of one criterion without causing a simultaneous deterioration in some other criterion.

2.3.6 Multi-Objective Evolutionary Algorithms

The first mention of the concept of a truly functional multi-objective genetic algorithm (i.e., a genetic algorithm that could handle multiple objectives without resorting to objective function aggregation) dates back to the 1960’s (Rosenberg, 1967). However, no multi-objective genetic algorithm was developed at that time. An attempt was made in 1983 (Ito et al., 1983) to develop a multi-objective genetic algorithm, but usually credit is given to Schaffer with his Vector Evaluated Genetic Algorithm (VEGA) for developing the first fully functioning multi-objective genetic algorithm (Schaffer, 1985, 1984). VEGA offered a credible multi-objective genetic algorithm, but it failed to include a mechanism for multi-objective elitism. This dramatically affects the speed at which an algorithm converges to good solutions, as promising solutions may be lost throughout the process.

After the development of VEGA, the most popular approaches for multi-objective genetic algorithms were aggregating functions. The most commonly used versions of these were the weighted-sum approach (Coello, 1999; Jones et al., 1993; Liu et al., 1998; Syswerda and Palmucci, 1991; Wilson and Macleod, 1993; Yang and Gen, 1994), goal programming (Charnes and Cooper, 1961; Coello, 1999; Ijirii, 1965; Sandgren, 1994; Wienke et al., 1992), goal attainment (Coello, 1999; Wilson and Macleod, 1993), s-constraint (Coello, 1999; Quagliarella and Vicini, 1997; Ranjithan et al., 1992; Ritzel et al., 1994).

These aggregating functions had several common problems, including a difficulty in working well on non-convex search spaces. Furthermore, where weights were used within the algorithm a very good knowledge of the objective functions in question was required for the values of those weights to be decided. So the need for improvement in the field was still very obvious.

The initial ideas of including the concept of pareto-optimality in multi-objective algorithms arose in Goldberg's book in 1989 (Goldberg, 1989). Whilst criticising

VEGA, he suggested that the use of non-dominated ranking of solutions with selection could move a population towards the Pareto front. There was no implementation of this idea for an algorithm supplied, but the majority of multi-objective algorithms developed after the publication of this book drew in a large part upon his ideas and suggestions (Coello, 2005), most notable the non-dominated sorting genetic algorithm (NSGA) (Srinivas and Deb, 1994), the niched Pareto genetic algorithm (Horn et al., 1994), and the multi-objective genetic algorithm (MOGA) (Fonseca and Fleming, 1993). Additionally, in 1992 a method was developed (Tanaka and Tanino, 1992) to incorporate user preferences into a multi-objective evolutionary algorithm.

From this point onwards, a focus shift occurred. The problem of building effective algorithms had been solved, and the goal was now to produce ever more effective and efficient algorithms (Coello, 2005). One of the main initial steps moving towards efficiency and effectiveness was the introduction of elitism to the multi-objective evolutionary algorithm playing field. Elitism involves artificially preserving the most optimal chromosomes produced at each point where chromosomes may be lost from the algorithm process, to ensure that promising solutions are not lost. Although early studies hinted at the possibility of application of elitism to multi-objective evolutionary algorithms, the formal introduction of this concept to the subject is usually credited to Echart Zitzler (Zitzler and Thiele, 1999) and his strength Pareto evolutionary algorithm (SPEA). After the publication of Zitzler's paper the majority of multi-objective evolutionary algorithms implemented some form of elitism (Coello, 2005). The most common form of elitism within a multi-objective evolutionary algorithm involves an external population comprised of all generated non-dominated solutions. Every solution entered into the external population must be non-dominated with regard to that population, and replaces any solution that it dominates within it.

The most popular current multi-objective evolutionary algorithms are SPEA (Zitzler and Thiele, 1999, 1998), SPEA2 (the second iteration of SPEA) (Zitzler et al., 2002), the pareto-archived evolution strategy (PAES) (Knowles and Corne, 2000), and NSGA II (the second iteration of NSGA) (Deb et al., 2002, 2000).

2.3.7 Multi-Objective Simulated Annealing

Although attempts have been made to convert simulated annealing to multiple objective optimisation due to its effectiveness as a single objective algorithm (Bandyopadhyay et al., 2008; Smith and Savić, 2006), it does not lend itself to the concept in the same way as evolutionary algorithms do, with their large population based approach. Attempts have generally revolved around objective function aggregation, similar to earlier multiple-objective evolutionary algorithm attempts (Smith and Savić, 2006).

2.3.8 Multi-Objective Ant-Colony Optimisation

Similarly, to multi-objective simulated annealing, attempts have been made to develop multi-objective ant-colony optimisation algorithms (López-Ibáñez et al., 2004; López-Ibáñez and Stützle, 2012; López-Ibáñez and Stutzle, 2010). Ant-colony optimisation does not, however, lend itself so conveniently to multi-objective optimisation and finding Pareto sets as evolutionary algorithms do. Additionally, ant-colony optimisation is a fairly recent algorithm and does not have a comparable body of published applications to some other algorithms to back up its effectiveness.

2.3.9 Optimisation Algorithms Summary

In summary, there are an extremely large amount of optimisation algorithms available, even when discussing multi-objective optimisation. Genetic algorithms in general have been used for many applications, and are particularly suited for adaptation to multi-objective optimisation, due to population-based manner in which they operate. Ant-colony optimisation methods have the same benefit but have fewer real-world successful applications to date. NSGA-II has a proven track-record of published applications to various problems (Behzadian et al., 2009; Bekele and Nicklow, 2007; Kannan et al., 2009). It has also been applied successfully to water-distribution problems (which have some similarities to drainage problems) as the base of promising heuristic optimisation algorithms (Behzadian et al., 2009; di Pierro et al., 2009; Fu and Kapelan, 2010; Jourdan et al., 2005, 2004).

2.4 Machine Learning

2.4.1 Introduction

Machine learning is a branch of artificial intelligence techniques dealing with algorithms that can learn from data. The most common usage is for data mining, and they can be used to great effect for classification of data (Kotsiantis, 2007). There are two main types of learning undertaken by machine learning algorithms, commonly referred to as “supervised” and “unsupervised” learning (Kotsiantis, 2007).

Data sets for training machine learning algorithms may be continuous, categorical, or binary. Where instances within the data set are provided with known labels (i.e. the correct outputs) the training process is known as a “supervised” process (Kotsiantis, 2007). Where there are no known labels, the process is known as “unsupervised” (Kotsiantis, 2007). Algorithms designed to undertake unsupervised learning generally work with clustering techniques such as Bayesian techniques (Neal, 1995). Clustering techniques are methods of identifying similarities between data instances. Those instances are then given (often varying degrees of) membership of “clusters” in an attempt to identify unknown but potentially useful classifications of data. These have been used on such diverse problems as road sign recognition (Prieto and Allen, 2009), water resources (Kalteh et al., 2008) and text detection with character recognition (Coates et al., 2011).

In this thesis the concentration is on classification algorithms, specifically artificial neural networks, and supervised training. This is because part of the work performed will be following on from previous work on developing neural network meta-models for multi-objective optimisation (Behzadian et al., 2009). Additionally, the second area where machine-learning techniques are utilised is within the LEMMO (Learning Evolution Model for Multiple-objective Optimisation) algorithm (Jourdan et al., 2005). The LEMMO algorithm is designed in the same way as the LEM algorithm that it was built upon (Michalski et al., 2000) to utilise any machine-learning algorithm. Artificial neural network (ANN) code was already implemented, so ANNs will be utilised for the machine learning part of this algorithm to minimise development time.

2.4.2 Artificial Neural Networks

The first neural network model which featured digital neurons was developed as early as 1943, although in this model no capability for learning was initially included (McCulloch and Pitts, 1943), limiting the usefulness of the model. In 1958 Frank Rosenblatt developed the “Perceptron” model (Rosenblatt, 1958), however Rosenblatt was unable to identify a reliable mathematically accurate mechanism for allowing multi-layer perceptrons to learn. The next major advance in artificial neural networks occurred in 1974, when Werbos (1974) succeeded in discovering the back-propagation algorithm, which was also independently re-discovered in 1982 (Parker, 1982). The application of neural networks to varied and complex problems is, today, a common occurrence (Behzadian et al., 2009; Biswajeet et al., 2010; Rowley et al., 1998).

There are two methods of training an artificial neural network – supervised and unsupervised (Kotsiantis, 2007). Supervised learning requires a set of training data that is pre-processed such that, along with each instance of data, there is an included expected output for the artificial neural network. The most common model for supervised-learning neural networks architecture is a feed forward network (see Figure 2). This is an arrangement of different layers of “nodes”, most commonly an input layer, a “hidden” layer, and an output layer. Each layer within this arrangement has connections to the outputs of nodes of the previous layer, and each of these connections has an associated weight (Lippman, 1987).

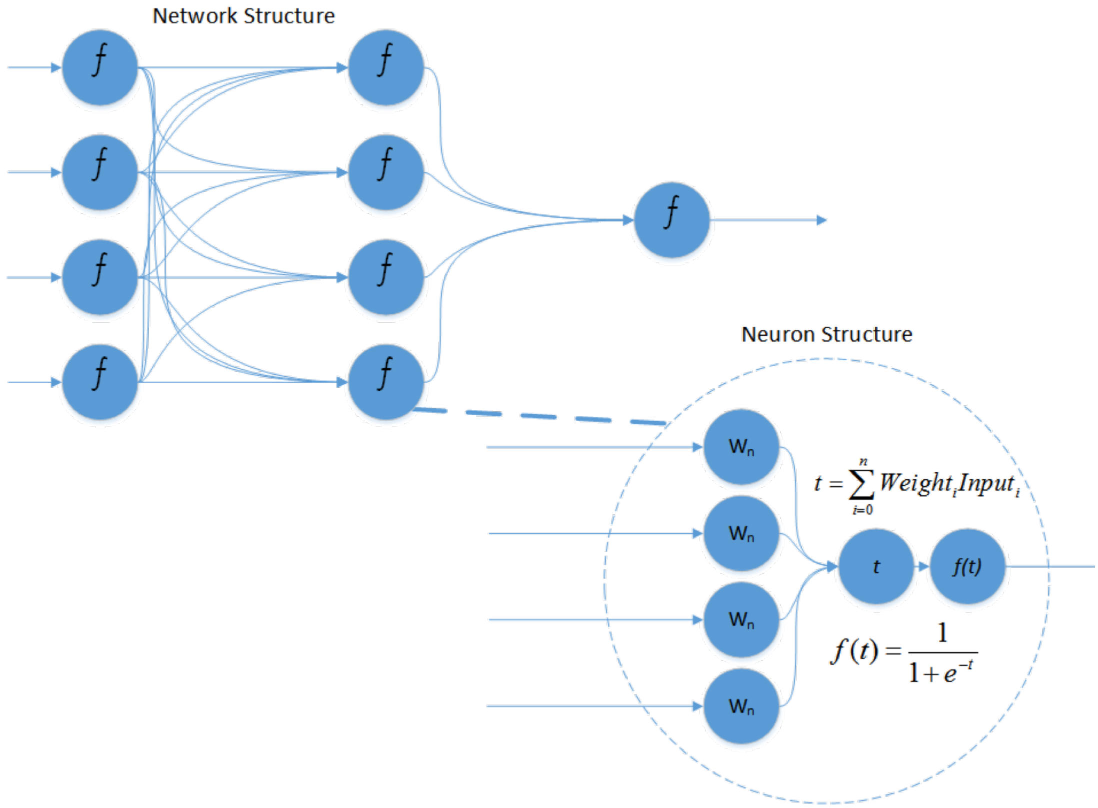


Figure 2 - Feed forward artificial neural network structure using a sigmoid activation function

Data then enters into the network at the "input" points (see Figure 2) and proceeds through the network node by node. At each connection it is multiplied by the value of the weight attached to that connection. At each node it is processed by a function – usually a differentiable function to facilitate training, of which the most popular are the logistic function (sigmoid function, see equation 6), and the Gaussian function (see equation 7). Artificial neural networks using sigmoid or Gaussian functions have been shown to be capable of approximating any arbitrary continuous function on a limited size domain, with varying accuracy depending on the number of neurons in the network (Cybenko, 1989; Hartman et al., 1990; Hornik et al., 1989; Park and Sandberg, 1991). Indeed, it has been shown (Hornik, 1991) that the choice of activation function is not as critical in allowing for the potential of universal approximation as the feed-forward architecture.

$$S(t) = \frac{1}{1 + e^{-t}} \qquad (6)$$

$$f(x) = ae - \frac{(x - b)^2}{2c^2} + d \qquad (7)$$

Training of feed forward artificial neural networks is accomplished by modifying the weights within the network to move them closer to achieving a desired output. The most commonly used algorithm to achieve this is the back-propagation algorithm (Parker, 1982; Rumelhart et al., 1986; Werbos, 1974). Back-propagation is a supervised learning technique that involves propagating error backwards through the network. It is a gradient descent method and because of this in its pure form it will be trapped at any localised optima that occur in the search space. A nearly ubiquitous addition to back-propagation in order to avoid this effect is “momentum” (Rumelhart et al., 1986). Momentum allows the back-propagation algorithm to be influenced by recent trends in the error surface, reducing the likelihood of, but not eliminating, the possibility of being stuck in local optima.

Back-propagation can be viewed as a simple gradient descent algorithm optimising the weights with the error from the artificial neural network performing the role of an objective function to be minimised. Taking this viewpoint makes it clear that any optimisation algorithm could be utilised to the same purpose. Genetic algorithms (see 2.3.2), simulated annealing (see 2.3.3), ant-colony optimisation (see 2.3.4) or any other optimisation algorithm can also be utilised, therefore, as part of the training process of an artificial neural network (Montana and Davis, 1989). This is also true for other versions of error propagation algorithms (Heaton, 2014; Igel and Hüsken, 2000).

2.4.3 Bayesian Belief Networks

Bayes' rule is a mathematical rule, which identifies the way in which existing beliefs should be altered, given a set of evidence previously unavailable. This can be simply illustrated with an example utilising white/black marbles as measures of belief. If it is imagined that a new-born baby sees the sun set, and wonders whether it will rise again. As the child has no prior knowledge, it assigns a fifty-fifty chance that the sun will rise again the next morning, and represents this by placing a white marble and a black marble into a bag. The following day, the sun does rise. The child, therefore, places another white marble into the bag to represent his increase in belief that the sun will rise. The probability that a random marble selected from the bag will be white (i.e. the child's belief that the sun will rise), has gone from 50% to 66.67%. As time passes and the bag becomes nearly entirely full of white marbles, the child becomes increasingly convinced that sunrise will come each day (Anonymous, 2000).

Mathematically, this can be represented as in equation 9 (Bayes and Price, 1763; Laplace, 1986) where equation 8 denotes the probability that a random variable ‘R’ has the value ‘r’ given that the evidence is equal to ‘e’.

$$P(R = r|e) \tag{8}$$

$$P(R = r|e) = \frac{P(e|R = r)\,P(R = r)}{P(e)} \tag{9}$$

A Bayesian belief network is a model that reflects the states of something, and how those states are described by probabilities. A Bayesian belief network can be utilised to model almost anything – with all possible states of a model representing all possible ways the states of that model can be configured. It works on the principle that some states are more likely to be true, when other states are also true. For example, if a person had a model of their body, they are more likely to have a sore throat if they also have a blocked nose. They are more likely to have sore eyes, if their eyes are watering.

Although Bayesian belief networks have many applications and are extremely effective when applied to the correct problems, for the purpose of this thesis a neural network is more easily utilisable and therefore is the approach that has been selected.

2.4.4 Machine Learning Techniques Summary

In summary, there are a large number of machine learning techniques available. One of the most flexible, with a huge amount of successful applications, is the artificial neural network (Behzadian et al., 2009; Lippman, 1987; Rowley et al., 1998) which has been proven to be a universal approximator (Cybenko, 1989; Hartman et al., 1990; Hornik, 1991; Hornik et al., 1989; Park and Sandberg, 1991) when structured as a feed-forward network. ANN's have also been successfully applied to water distribution problems (Behzadian et al., 2009) specifically, which are similar in many ways to flood risk problems.

2.5 Multi-Objective Optimisation with Machine Learning

2.5.1 Introduction

Multi-objective optimisation, by the nature of the way in which the algorithms function has an exponential increase in objective function calls as objectives are added. As multi-objective functions are generally utilised in an attempt to solve real world problems, which are typically extremely complex or even completely intractable (see section 2.3.5), the resulting implementations can be very computationally demanding.

There are a number of approaches to reducing the impact of this in some way – often involving some form of Meta modelling (Behzadian et al., 2009; Broad et al., 2005; Magnier and Haghighat, 2010; Morimoto et al., 1993; Zhou and Haghighat, 2009). An alternative approach is to use machine-learning techniques to guide population generation based on the characteristics of well-performing solutions from previous generations (di Pierro et al., 2009; Jourdan et al., 2005; Michalski et al., 2000).

Meta modelling is the practice of substituting a simplified objective function that is still accurate enough to be of use, for the real function. This may require adjustments in the way the algorithms run in order to avoid losing too much accuracy in final results, and usually the real function is only partially replaced, but is in general a very useful approach.

One of the most common meta-models used is the artificial neural network (see section 2.4.2). The concept has been around since the early 1990's (Morimoto et al., 1993) and has started to see increased uptake recently, as computer hardware and technology potential has caught up with algorithm development.

Meta-modelling with a neural network has been applied with some success to problems of building design (Magnier and Haghighat, 2010), ventilation system design and operation (Zhou and Haghighat, 2009) and water distribution system design (Behzadian et al., 2009; Broad et al., 2005).

A slightly different approach exists in LEM (Michalski et al., 2000; Wojtusiak and Michalski, 2006) which is a single-objective approach, using a machine learning technique to guide generation of new populations. LEMMO (di Pierro et al., 2009; Jourdan et al., 2005, 2004) is a multi-objective version of this algorithm using decision trees to identify rules that characterise promising solutions. LEMMO has been successfully applied to water distribution system optimisation (di Pierro et al., 2009; Jourdan et al., 2005).

Of the numerous methods available, and out of the approaches mentioned above, two have been successfully applied to water distribution systems optimisation. This kind of problem is similar to the problem detailed in this thesis, both in terms of the kinds of variables being considered (discrete pipe sizes within a complex network) and in terms of the kinds of objectives (costs, some measure of performance of the system). Therefore, these two approaches warrant further in-depth examination, which can be found below in sections 2.5.2 and 2.5.3.

2.5.2 Multi-Objective Genetic Algorithm with Adaptive Neural Networks (MOGA-ANN)

This algorithm, presented by Behzadian et al. (2009) involves the use of an adaptive neural network. Initially the algorithm is run for a number of iterations with no neural network in place. During these initial iterations the data from each objective function evaluation is collected. This data is then used after a fixed number of iterations to construct a training set for an initial training cycle for the neural network. The function is then altered so that the neural network first evaluates all solutions produced by the genetic algorithm. The solutions that rank above a specified cut-off after this first evaluation are then re-evaluated for accuracy by the full objective function. All of the data from these re-evaluations is stored and every 'n' iterations the neural network undergoes a further training cycle using this new data.

The goal of this process is to initially train a neural network, which can then be used as a kind of “filter”, to remove the poor solutions from the set before using the computationally intensive full objective function on the remaining solutions. The re-training should make this network progressively more accurate, particularly for solutions that more closely match the Pareto front (as the solutions the network is being trained on should be moving closer to the Pareto front with each training cycle) (Behzadian et al., 2009).

This approach has been successfully applied to a sampling design (Kapelan et al., 2005) approach (Behzadian et al., 2009). In the test cases described the MOGA-ANN (multi-objective genetic algorithm with adaptive neural networks) approach performs almost as well as a standard MOGA approach. It does so with far fewer evaluations of solutions with the computationally costly objective functions, thus increasing performance by around twenty-five times with only a very minor effect on Pareto front accuracy.

2.5.3 LEMMO

The LEMMO algorithm (di Pierro et al., 2009; Jourdan et al., 2005, 2004) is based on the LEM (Learning Evolution Model) single-objective algorithm (Michalski et al., 2000). The LEMMO algorithm involves a decision tree classifier within the NSGA-II algorithm which is used as a feature identifier for characteristics of solutions which perform well.

At a high level the algorithm works by performing a number of iterations of a standard NSGA-II (Deb et al., 2002) algorithm and storing data on which of the generated solutions are “good” solutions and which are “poor” solutions. It then uses this data to train a machine-learning algorithm to distinguish between good and bad functions, and uses the outcome of this training in some fashion to generate new solutions, which, according to the trained machine-learning model, are “good” solutions. These are then integrated with the main population and the algorithm continues with further iterations based on pure NSGA-II and further iterations of machine learning.

There are five variants of this algorithm that have been tested (Jourdan et al., 2005), and these are described briefly below.

- Variant 'LEMMO-1'
    - *Learning is run when there has been no change to the population for two successive iterations.*
- Variant 'LEMMO-Fix1'
    - *Every ten full iterations a learning iteration is entered, utilising the initial population of the previous evolution phase as the “bad” set and the final population of that phase as the “good” set.*
- Variant 'LEMMO-Fix2'
    - *A learning iteration is run every ten iterations, using the twenty individuals most recently inserted into the Pareto set as the “good” set and the remaining individuals as the “bad” set.*
- Variant 'LEMMO-Fix3'
    - *Every ten generations, a learning iteration is run, with random individuals from the current approximation of the Pareto set as the*

“good” set and the remaining individuals of the current population as the “bad” set.

- Variant 'LEMMO-Fix4'
    - *Run a learning iteration every ten generations using the best 30% of the individual solutions found so far on one of the objectives (randomly chosen at each learning phase) as the “good” set, and the worst 30% of those solutions as the “bad” set.*

2.6 Chapter Summary

Throughout this chapter the past of urban flood risk analysis has been investigated and various techniques and algorithms for artificial intelligence applications examined.

Urban flood risk analysis is a growing research sector for which new models are appearing with some regularity. This is in part caused by the capabilities of the technology having caught up with the ambitions of the modellers. It would be good to use these models in conjunction with optimisation techniques. The technology is not currently, and probably will not be within any reasonable time-frame, capable of performing these kind of optimisations at a useful speed. This is because urban flood risk models are such hugely intractable and complex NP-hard problems.

New approaches are being explored in many other fields in an attempt to apply optimisation algorithms combined with machine-learning heuristics to similarly complex real-world engineering problems.

These algorithms can be combined and used together in various ways to achieve a given end. One area that shows significant promise is the combination of classifier algorithms with genetic algorithms, which can be applied to either single or multiple objective genetic algorithms. Two such combinations have been examined above (sections 2.5.2 and 2.5.3).

MOGA-ANN effectively uses a neural network classifier as a filter, to distinguish between promising and non-promising solutions. LEMMO on the other hand, uses a decision tree classifier to learn the features that determine whether a solution is “good” or “bad”, and then generate new solutions that match the features of the “good” solutions. Both of the examined combinations show promise.

In terms of standard algorithms, NSGA-II is one of the most widely used algorithms, and would be a good benchmark for comparison against the two performance improvement methods detailed (2.5.2 and 2.5.3) because both those approaches build from NSGA-II. Additionally, there is a large body of previous research utilising NSGA-II (Bekele and Nicklow, 2007; Deb et al., 2002, 2000; Kannan et al., 2009), showing it can be applied widely with good success, and meaning there is good availability of test data and problems.

The literature surveyed shows that there is a lack of research involving optimisation techniques being applied successfully to flood risk problems. Additionally, applying optimisation techniques to these problems will require the application of techniques to reduce objective function calls, whether this is by replacing them with calls to meta-models, or reducing the number necessary by improving solution generation. There is a knowledge gap where this is concerned also. There may be considerable scope in terms of optimising these techniques and algorithms for a flood risk problem, and also in terms of altering the existing tools for flood risk analysis to better fit the requirements of optimisation algorithms.

Based on this literature review, this thesis aims to close the knowledge gap existing with regard to the application of optimisation algorithms to flood risk problems. It does so by taking two identified promising approaches and testing them for suitability to application to the area of flood risk intervention optimisation. The more promising of the two is then compared to a base algorithm (an algorithm without the heuristic improvements) and it’s performance on the problem is improved by further modification.

3. Urban Flood Risk Assessment

3.1 Introduction

Analysis methods for understanding the behaviour of drainage networks have evolved over the last 50 years, from relying on local knowledge of the drainage system, through to the ability now present of being able to predict frequency, depth and location of flooding events. The development of computer-based flood modelling has led to the ability to model fairly accurately the impact of extreme rainfall events on drainage systems, given the necessary data to build a sufficiently accurate model of the system. Building on these capabilities, risk-based approaches for flood risk evaluation have started to be promoted by key organisations within the field (Boelee and Kellagher, 2015; Kellagher et al., 2009).

One of the approaches to this issue has been the development of the "DTI-SAM" risk-based tool set and methodology as part of a previous research project at the sponsoring company for this thesis (Kellagher et al., 2009). The work that has been performed as part of the research project being described by this thesis is in a large extent built upon work completed as part of this previous research project, particularly in terms of the tools development. Because of this, a description of the work completed during this previous project comprises this chapter.

3.2 Risk-based Approach

When considering urban flooding in a risk-based fashion there is a complex system to analyse, as the physical flooding event must be included, along with the inhabitants of the areas to be flooded, human infrastructure present on any affected flood plains, the ecosystem present, and private personnel who will either influence or be influenced by the flooding and flood impacts.

The benefit of a risk-based approach over a more traditional approach is that a risk-based approach allows the evaluation of a system in terms of the consequences of the failure of that system, rather than in terms of the performance of that system. This allows for a decision to be made based on the consequences of system failure. In this thesis we focus on the economic consequences, however almost any consequence could be used as a measure.

Generally, risk based analysis involves modifying variables that describe the flood system in terms of pipe diameters and storage node volumes, then analysing the results of those changes on the flood model and the effect they have had on the risks of flooding occurring.

3.3 Flood Risk Analysis Toolset

The work in this thesis has been based upon the project that was undertaken by a research consortium to develop a risk-based methodology and tool set for evaluating drainage systems in terms of their flood-risk, and to improve knowledge in this area of research (Kellagher et al., 2009).

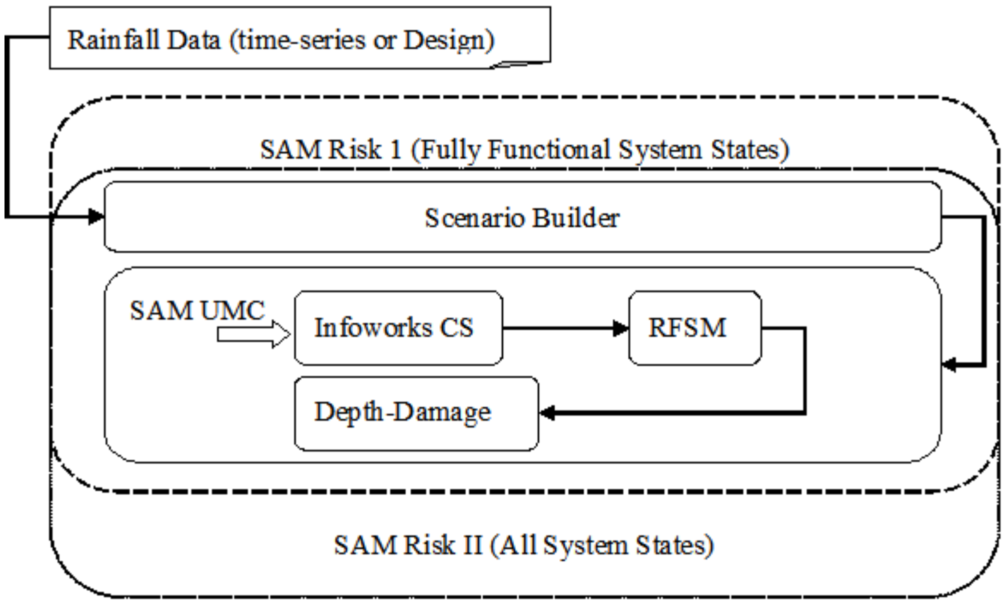


Figure 3 - Simplified DTI SAM overall diagram, separate components described in later sections of this chapter. Both risk tools contain the same subcomponents but are distinct (Kellagher et al., 2009).

Although in general terms the modelling tools existed at the project onset to apply a risk analysis, it was necessary for the project to involve a large amount of tools development. This was in order to automate the process further, allowing multiple runs with subsequent run scenarios based off the results of prior runs. Additionally, tools had to be developed for the calculation of 'risk' defined in terms of "Expected Annual Damage" or "EAD", which is a monetary measure of the amount of damage caused by flooding. In this work all EAD figures are presented in monetary terms (i.e., pounds sterling (£)) but altering the data the software utilises for pricing would allow these costs to be presented in whatever currency was most suitable.

3.4 Risk Assessment Framework

Although the flood risk assessment framework this thesis is based on (SAM-Risk) has two versions, one dealing with fully functional drainage systems and the other attempting to incorporate the chance of the systems failure, in this thesis only the former is considered. The reason for this is that the difference in generated EAD between the two versions is generally small, and when taking systems failure into account, the software takes longer to run and is more complex.

A flood risk assessment run consists of the following steps:

- Initialise flood model (Infoworks CS)
- Build scenario
    - Selected rainfall and network
    - Prepare input file for scenario
- Launch flood risk assessment with pre-generated input file
- Extract the results from previous step
- Determines next scenario (rainfall and network) according to the results
- Manages outputs – tracks the convergence of the results, and uses this convergence parameter as well as other parameters, as “stop criteria”.

The flood risk assessment tool can receive inputs from both design storms and time-series rainfall with the choice between the inputs being up to the user.

Design storms are widely available and easy to find – if time-series data is available it tends to produce more accurate EAD estimation than design storms.

3.4.1 Design Storm Risk Assessment

The classic risk calculation is based on magnitude of the probability and the damage that would be caused if that probability occurred. This can be expressed as risk (R) being equal to probability of an event (P) multiplied by damage caused (D) by that event (see equation 10) (Kellagher et al., 2009).

$$R = P \times D \tag{10}$$

In order to find the probability of the event in question not occurring, 1-P, can then be used, as the sum of all possible outcomes must have a value of one. Since a design-storm has a specific probability, the probability of non-exceedance of the threshold can be found via the equation below (equation 11) (Kellagher et al., 2009).

$$x = \left(1 - \frac{1}{n \times R_i}\right) \tag{11}$$

Where *'n'* represents the number of events in one year. Although the main purpose of applying a risk-based methodology is to evaluate the damages for a given flood event, these do vary for any given location during the same return period, depending on the duration of the storm event used. It is important that damage associated with each probability of non-exceedance is based on the critical duration for each manhole. The critical duration is the duration with which the maximum flood volume at a given node is associated. At the top of a drainage system, therefore, the critical duration will be fairly short, whereas at the bottom end of a large network it will be considerably longer (Kellagher et al., 2009).

There is also a desire to associate the flood damage with the assets that initially flooded during some analyses. In order to achieve this, the total damages occurring at all impact events must be proportionally distributed to the manholes based on the damage of the critical duration event (see equation 12 where ‘D’ is equal to damage at a manhole, ‘d’ is equal to the damage at an impact zone, ‘m’ is the total number of manholes, ‘i’ is the current manhole and ‘I’ is the total number of impact zones) (Kellagher et al., 2009). Impact zones are identified within pre-processing of the flood plain in RFSM and represent topographical depressions where water will collect in case of flooding (Lhomme et al., 2008).

$$D_i = \frac{D_i}{\sum_m D_i} \times \sum_I d_i \qquad (12)$$

This ensures consistency when associating damages at IZ’s with the manholes that initially flooded (see equation 13) (Kellagher et al., 2009).

$$\sum_m D_i = \sum_I d_j \qquad (13)$$

The EAD (expected annual damage) is the integration of the risk due to every probability of non-exceedance. The simple trapezoidal integration method is used, and as a consequence the expected annual damage for every impact zone and manhole can be written as a function of the probabilities $\left(P_{RP_i}\right)$ and damages associated $\left(D_{RP_i}\right)$ with every return period (RP_i) as shown in equation 14 (Kellagher et al., 2009).

$$EAD = D_i \times P_i + \sum_{i}^{TRP} \frac{D_{RP_i} + D_{RP_{i-1}}}{2} \times \left(P_{RP_i} - P_{RP_{i-1}}\right) \qquad (14)$$

When the modelling system is being run, expected annual damage evaluates shortest return periods first, and then continues through all return periods in ascending order until convergence is achieved.

In order to avoid an overestimation of damages, the return period threshold of every manhole/impact zone must be found. This is the return period at which flooding first occurs at this manhole/impact zone (Kellagher et al., 2009).

This means that D_1 (the damage of the shortest return period event) is a very small value (zero or close to zero) and P_1 is an event for which flooding occurs. When this process (see Figure 5) is plotted, EAD can be seen to increase asymptotically (see Figure 4) (Kellagher et al., 2009).

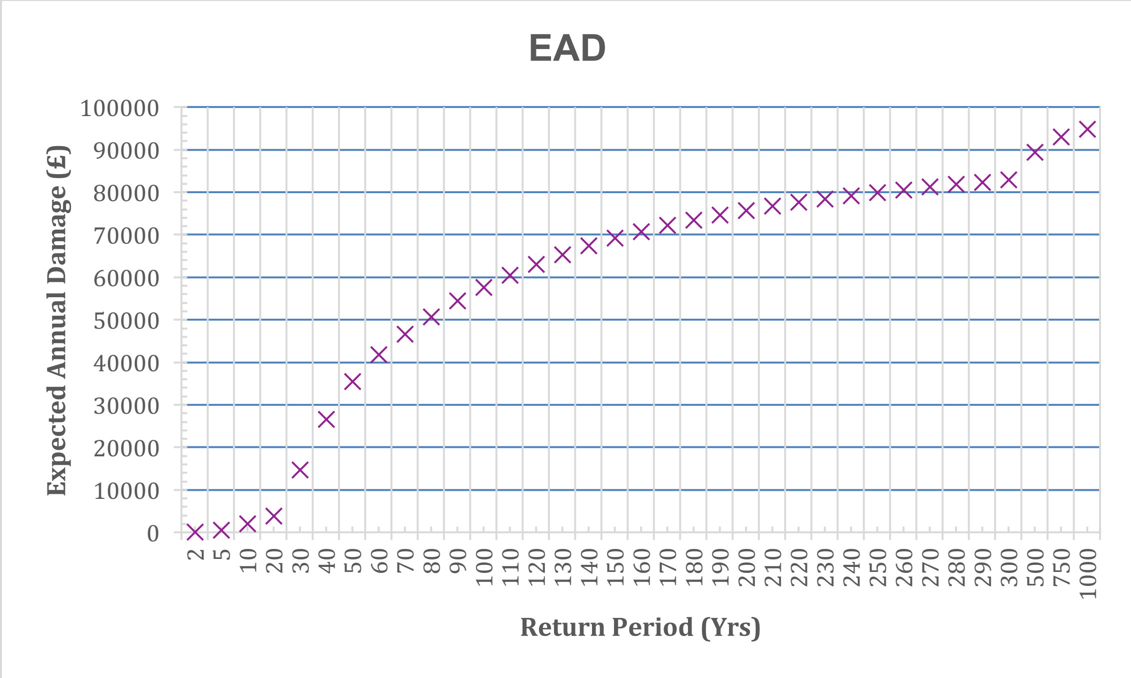


Figure 4 – Asymptotic increase of expected annual damage using a design-storm event approach.

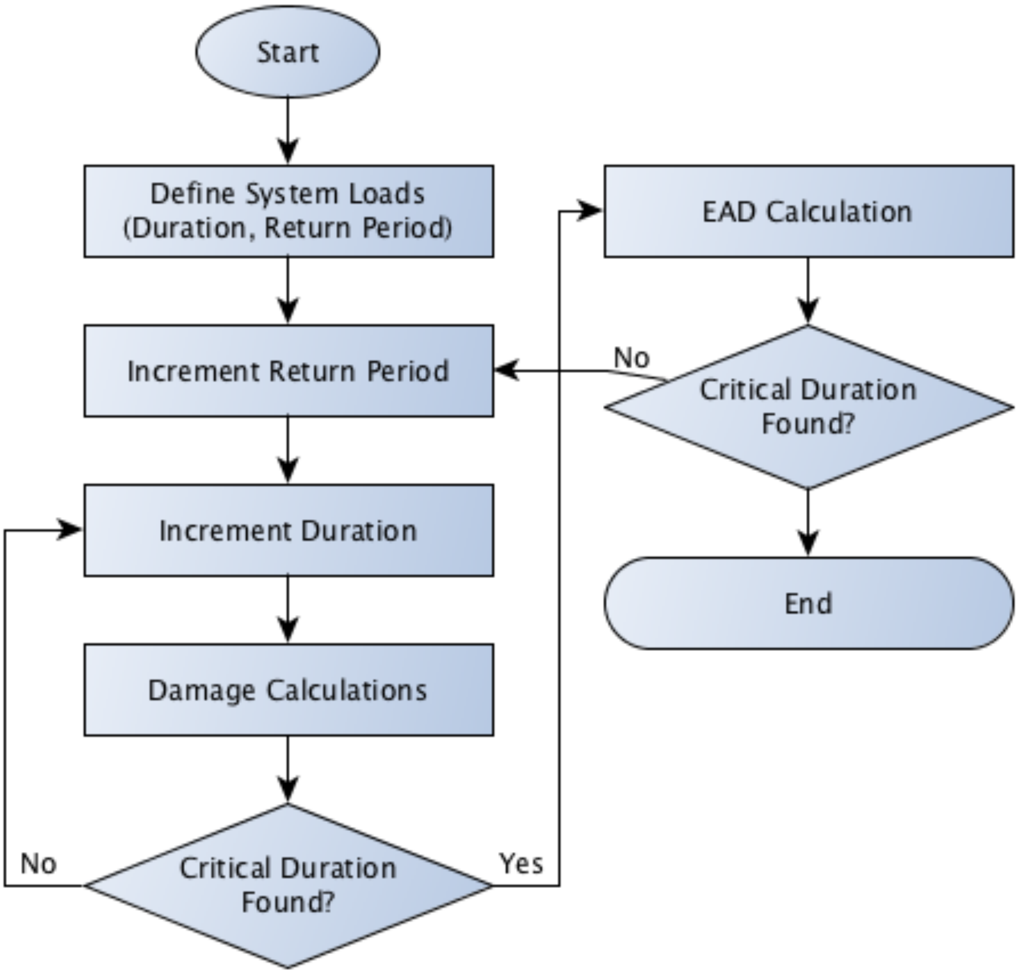


Figure 5 - Methodology for analysis of hydraulic failure using design rainfall events

3.5 Risk Assessment Framework Software Components

3.5.1 Component Object Model Interface Module

The component object model interface (Innovyze, 2007) module within this framework is known as SAM-UMC. This module facilitates interaction between Infoworks CS, RFSM and the drainage system calculations. Whilst the module allows for multiple simulation runs, they need to be set up very specifically, with a set number of runs to perform and a set number of scenarios to perform these runs on. It was in order to overcome these limitations that further tools (SAM-Risk) were developed. These tools do, however, rely upon the original software module and build upon its functionality rather than replace it.

3.5.2 Infoworks Collection Systems Drainage Model

The Infoworks Collection Systems (Innovyze, 2007, 2011) run is initiated by the flood risk assessment (Wills, 2013) module, and consists of the following steps:

- Make any necessary changes to the attributes of nodes/conduits.
- Specify the set of rainfall data and waste water data
- Specify any other parameters as necessary
- Run the simulation
- Extract results from the simulation (in terms of volume lost from manhole nodes).

3.5.3 Rapid Flood Spreading Model Module

The RFSM (Rapid Flood-Spreading Model) (Lhomme et al., 2008) performs the function of spreading volumes of water over a specified flood area. RFSM acts as a simplified hydraulic model that produces answers to an acceptable level of accuracy, but produces them considerably faster than more accurate and less simplified methods (Lhomme et al., 2008).

3.5.4 Depth-Damage Model Module

The water level in each impact zone is then used to assess the depth of flooding against a national property database provided by the environment agency and information gleaned from the Middlesex multi-coloured manual on flood damage (Penning-Rowsell et al., 2005). The damage assessment tool includes average property valuations, based on information from the land registry office, floor space and rateable values, flood damage curves for residential and non-residential areas, and varied codes associated with the multi-coloured manual (Wallingford, 2009).

3.6 Software Implementation Issues

The major performance problem with this framework involves the runtime of the Infoworks CS (Innovyze, 2007, 2011) model within this risk assessment framework. The process of evaluating expected annual damage is the most computationally intensive task involved in the optimisation algorithm. A significant proportion of that computational effort is involved in setting up and initialising Infoworks simulation runs. This is because for each expected annual damage assessment a selection of rainfall events with different return periods (two, five, and ten, then steps of ten to three hundred, five hundred, seven hundred and fifty and one thousand years) and different durations (thirty, sixty, ninety, then steps of thirty minutes all the way to six hundred) have to be run as Infoworks simulations. This totals seven hundred Infoworks simulation runs. With each of those runs of seven hundred rainfall events taking approximately five hours (depending on computational performance), this is not a tractable problem. In a genetic algorithm, which might have a population of one hundred, to be run for one thousand iterations (conservatively), the total runtime would be in excess of fifty-five years. To a certain extent that could be mitigated by running on a more modern machine than the testing machine but even with a desktop machine at the forefront of modern consumer technology, the run-time could not be expected to be reduced to a usable time-frame (i.e. weeks or months, rather than years).

3.7 Chapter Summary

There has been a substantial body of work carried out on the direct predecessors to this current project. All of this work combines to culminate in a methodology and algorithm for producing an expected annual damage estimate based on a set of rainfall data and an urban drainage system model developed in Infoworks CS. This methodology and algorithm has been utilised as an objective function in the multi-objective algorithm that is one outcome of this research project, but performance improvements were required. This is because the associated algorithm is completely intractable with regard to being utilised as part of an optimisation algorithm. Therefore, methods of decreasing the impact that this algorithm has on these computational resources had to be explored.

4. Optimisation for Urban Flood Risk Management

4.1 Introduction

In order to utilise an optimisation approach on the problem of flood risk, the main barrier is the computational efficiency of the flood risk algorithm. It is, therefore, sensible where such an algorithm has been developed and supplied outside the bounds of the project and therefore with separate priorities and goals, to examine whether that computational efficiency can be improved upon. Additionally, a costing model had to be developed, in order to provide the capital cost of making changes to the drainage network. This is in line with objective 3 (see section 1.2) and is necessary in order to give a good baseline for objective 4, and aid towards achieving good results in objective 5.

Additionally, one of the goals of the development associated with this thesis is to produce a tool, which can be utilised by flood risk engineers. In order to facilitate that, a user interface is required which allows control of the optimisation parameters, without demanding an excessive amount of specialist knowledge. This is most of objective 1 and part of objective 2 (see section 1.2).

Finally, it is necessary to develop and test optimisation approaches that can provide the "back-end" of this user interface, and utilise the above mentioned flood risk algorithm as an objective. This is a broad goal which fits in with almost all of the objectives in some way. The results of the testing of the algorithm are described in chapters 5 and 6.

This section covers these three points and how they have been approached, and achieved.

4.2 Development of the Costing Model

The second intended objective of the multi-objective algorithm is “cost”. In this case it means the cost of the proposed changes to the network (I.e. a network that matches the original network, is zero cost). These changes will take the form of alterations to pipe diameters, and storage node volumes. The constants in the cost calculation are all customizable (they are constant in the sense that they should not change during the algorithm run), and sensible defaults are provided to give a “rough idea”. These defaults are based upon work undertaken by HR Wallingford and Mouchel consulting as part of the project on which this thesis is based (Kellagher et al., 2009). This work also formed the basis for the cost model developed here. It would be expected that for a given flood risk scenario, constants for these cost calculations would be modified to be in line with the real figures for that particular scenario. These defaults have been used for testing purposes for all experiments described in this paper, they are: Mobilization Cost (M) of £50,000 for making any change to a network; Pipe Intervention Cost (I) of £1,000 per metre of pipe replaced; Storage Intervention Cost (S) of £500 per metres cubed; and Storage Base Cost (b) of £10,000 for making any change to a storage node.

$$Cost = M + \left(\sum_{i=0}^{n} I \times L_i \times c_i\right) + \left(\sum_{j=0}^{n} S \times a_j + b\right) + \left(\sum_{k=0}^{n} o_k\right) \qquad (15)$$

All cost calculations initially determine whether a network has been modified at all, or whether it is identical to the original network. All modified networks cost calculations include an initial sum to reflect the “mobilization” costs – i.e. the costs of hiring contractors, getting them and their equipment on site, and other associated costs with initially beginning a task of this nature. The pipe alteration costs are then estimated by multiplying the product of a constant “Intervention Cost” value (I) to represent costs of piping purchase, excavation, etc. and the length of pipe (L) in question by the cross-section area (c) of the pipe. Storage alteration costs are estimated by adding the product of an “Intervention Cost” constant (S) signifying cost of materials, etc., and the area of the storage node in meters squared (a), to a second “Base Cost” constant (b) that represents the costs associated with excavation, removal of existing storage node if necessary, etc. All non-modified pipes or storage nodes result in zero cost. Orifices are included in the cost model and have a flat cost (o) associated with any change to their original setting.

Therefore, the total cost of a network is the mobilization cost (M), plus the cost of each modified pipe in the network, plus the cost of each modified storage node in the network, plus the cost of each modified orifice in the network.

4.3 Improvements to EAD Calculation Tool Set

Before taking the route of applying cutting edge artificial intelligence techniques and heuristics in order to reduce runtime of the full optimisation process, the possibilities of improving the efficiency of EAD generation should be investigated. Multiple approaches should be investigated and considered towards achieving this indispensable objective.

4.3.1 Identifying Reduced Rainfall Dataset

Due to the large number of EAD calculations necessary during a multi-objective optimisation scenario, it is important that every EAD calculation takes place using the minimal computing resources possible. The standard set of design-storms previously utilised (during the DTI-SAM project) provides a very good approximation of EAD. This set of design storms was not, however, designed to be part of an optimisation algorithm and so was not subject to the sort of time considerations that are common in that scenario.

The standard set of rainfall files used during the DTI-SAM project consists of 700 complete rainfall events. These rainfall events comprise 20 durations for each return period, starting from 30 minutes, progressing to 600 minutes at steps of 30 minutes. There are a total of 35 return periods, which are 2, 5, 10 then steps of 10 up to 300, then 500, 750, and 1,000.

A return period is an estimate of the time interval between events of a similar nature occurring, and can be used as a measure of the likelihood of a given event occurring. For example, a given rainfall intensity may have a return period of 20 years. It is important to note that this does not mean that an event of that kind will occur within 20 years, or will definitely occur only once within twenty years. A rainfall duration is simply the amount of time that a rainfall event continues, in minutes.

Within the internals of an optimisation algorithm it is allowable to a certain extent to lose accuracy, provided that the change in EAD remains proportionate or close to proportionate across intervention strategies, thus allowing sufficient differentiation between higher and lower quality solutions. A possible approach, therefore, to improving the computational efficiency of producing EAD estimations is to reduce the number of return periods or durations of rainfall that are evaluated during EAD estimations.

Previous studies (Ward et al., 2011; Woodward, 2012) have shown that reducing resolution of the problem space being evaluated can result in significantly lowered model run time. It does, however, affect accuracy of results. In one study in particular (Woodward, 2012) it was found that significant performance improvements can be obtained whilst retaining sufficient accuracy for the test results to be useful as part of a multi-objective optimisation algorithm.

Initially, it is important that testing is performed on more than one network, therefore a minor, but important, first step is the generation of twenty separate testing networks from the initial network. The layout of these separate networks can be seen in Figure 7, showing the included elements, although the pipe diameters and storage node volumes will vary per network. Once this was accomplished, a full EAD estimate could be generated for each of these networks. Tests were performed with a range of different durations initially (in an attempt to identify a subset of durations that gave results as close as possible to the original EAD estimation), followed by tests utilising these durations with a subset of return periods. This gives us a combination of return periods and durations which when run, give a reasonable approximation of the EAD for a given network that maintains or nearly maintains the relative variance in EAD between different alternative solutions.

As mentioned, it is critical that any solution proposed is effective across more than one drainage system – as the NSGA-II algorithm will be generating many variants of the original drainage system in order to identify the optimal flood risk intervention.

In order to ensure this is the case, a number of drainage systems are needed to perform testing on. Twenty different networks have been generated for use as test networks. Three rainfall events that fit the following criteria were selected to generate these networks:

- One reasonably short and fairly unlikely storm (return period thirty years & duration thirty minutes, run A).
- One fairly lengthy and very unlikely storm (return period one hundred and seventy years & duration three hundred minutes, run B).
- One very long & very unlikely storm (return period one thousand years & duration six hundred minutes, run C).

The NSGA-II algorithm was then run three times; with each run evaluating EAD via one of these rainfall files alone. This is not expected to give a good approximate EAD value, but gives the algorithm a value to work towards improving. Essentially one optimisation run is extremely biased towards reducing damage from regularly occurring storm intensities, one towards rare storm intensities, and one towards very extreme storm intensities. Each run was performed for fifty iterations with a population size of twenty. Each algorithm run produces twenty networks in it’s final iteration. This gives a total of sixty (three sets of twenty) alternative network solutions to choose from once the algorithm had run its course. From each set a spread of networks (seven from run 'A', six from run 'B', and seven from run 'C', for a total of twenty) was taken which attempted to maximise the separation between EAD scores by eye. A total of twenty was aimed for in order to give a reasonably sized test set, without it being overly large. The three types of network should have produced networks with different characteristics, a larger number of results (seven) was picked from the two extremes (A & C), and a slightly smaller number (six) from the centre set (B). These twenty networks then formed the testing set.

For each of the twenty selected networks in the testing set, a full EAD evaluation was then run so that a base figure to compare the runs with fewer return periods and durations could be identified.

Each run produces a curve as the return periods are iterated through, as shown below in

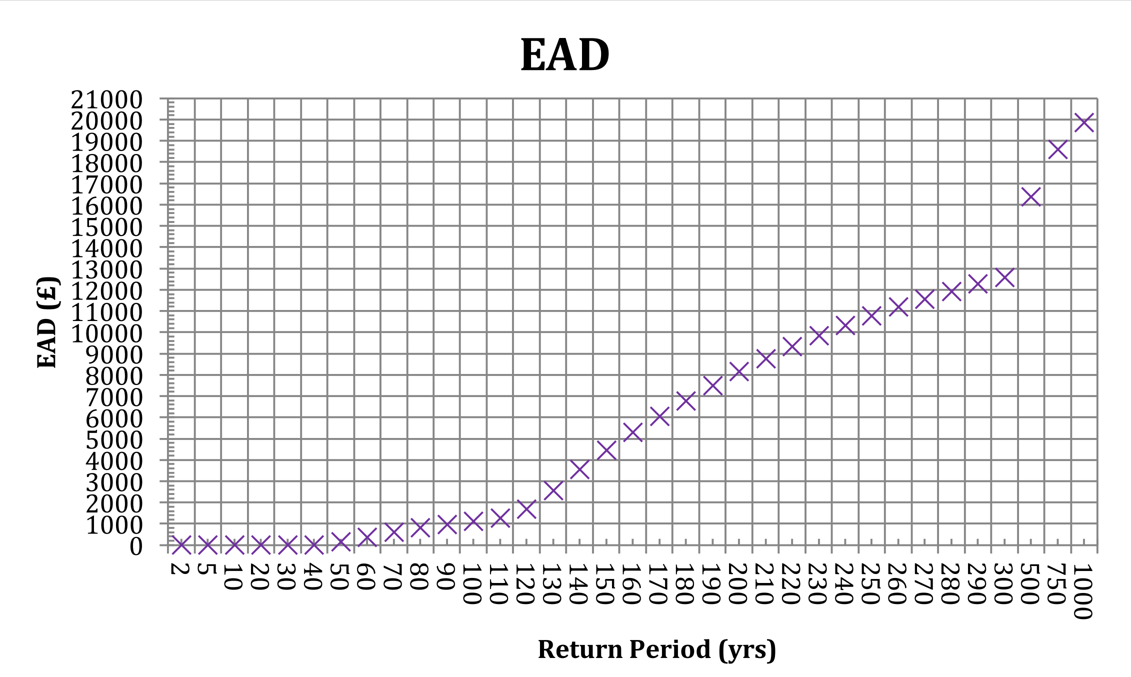


Figure 6, which is the curve for “Net 5”. The final results of these runs are shown in Table 1, Table 2 and Table 3.

Once these initial tests were completed and base figures for EAD gained, tests were initially run with fewer rainfall durations. Tests were performed with various combinations of duration, however, on all twenty networks a single duration of six hundred minutes (i.e. the most extreme duration) produced the same or very close to the same EAD score as the base figure on all networks (see Table 4, Table 5 and Table 6). The mean error of runs using only six-hundred-minute duration rainfall versus using all rainfall was £15 which given the estimated nature of EAD and the size of the sums in question, is considered small and acceptable. The decision was taken to proceed with utilising six-hundred-minute duration rainfall files, and to continue with attempting to reduce the number of return periods being analysed.

One single duration being a good match is somewhat surprising, however a possible explanation is that the section of the drainage network being optimised is at the extreme lower end of the actual catchment. The shorter durations may, therefore, have less effect as the volume of water is passing fairly quickly to the drainage network outflow, with little chance to build up within the system. In a larger drainage system there exists many more opportunities for flatter regions of pipe to be encountered and there is more chance for friction within pipes and storage to have an effect.

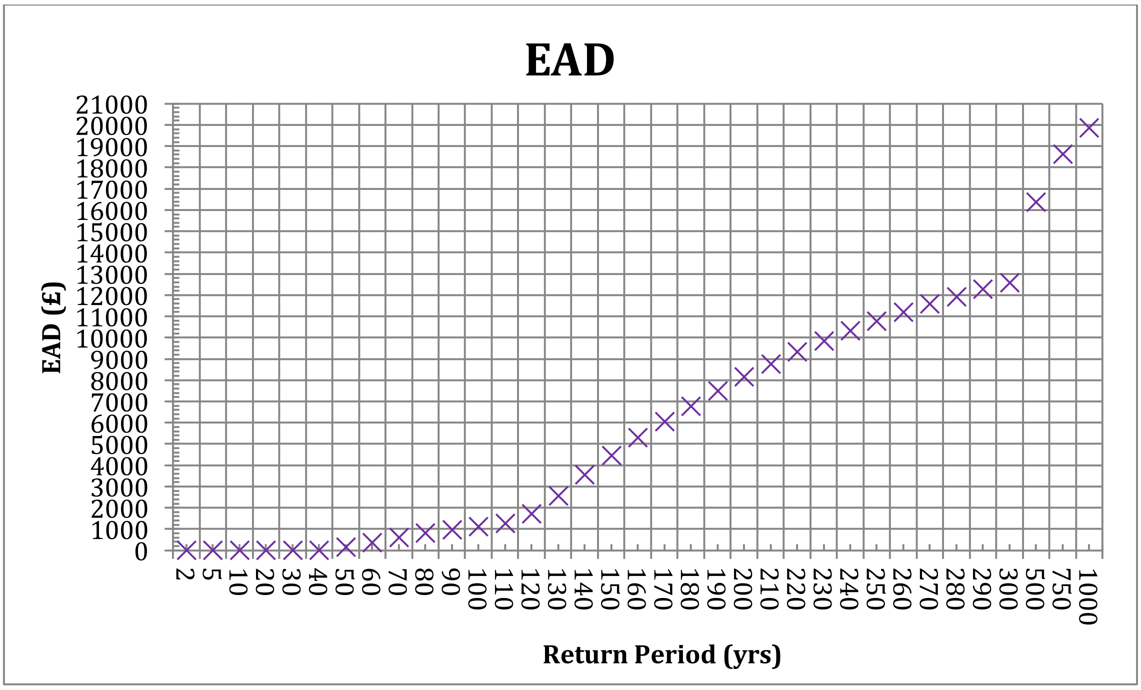


Figure 6 - Example of EAD curve (Net 5).

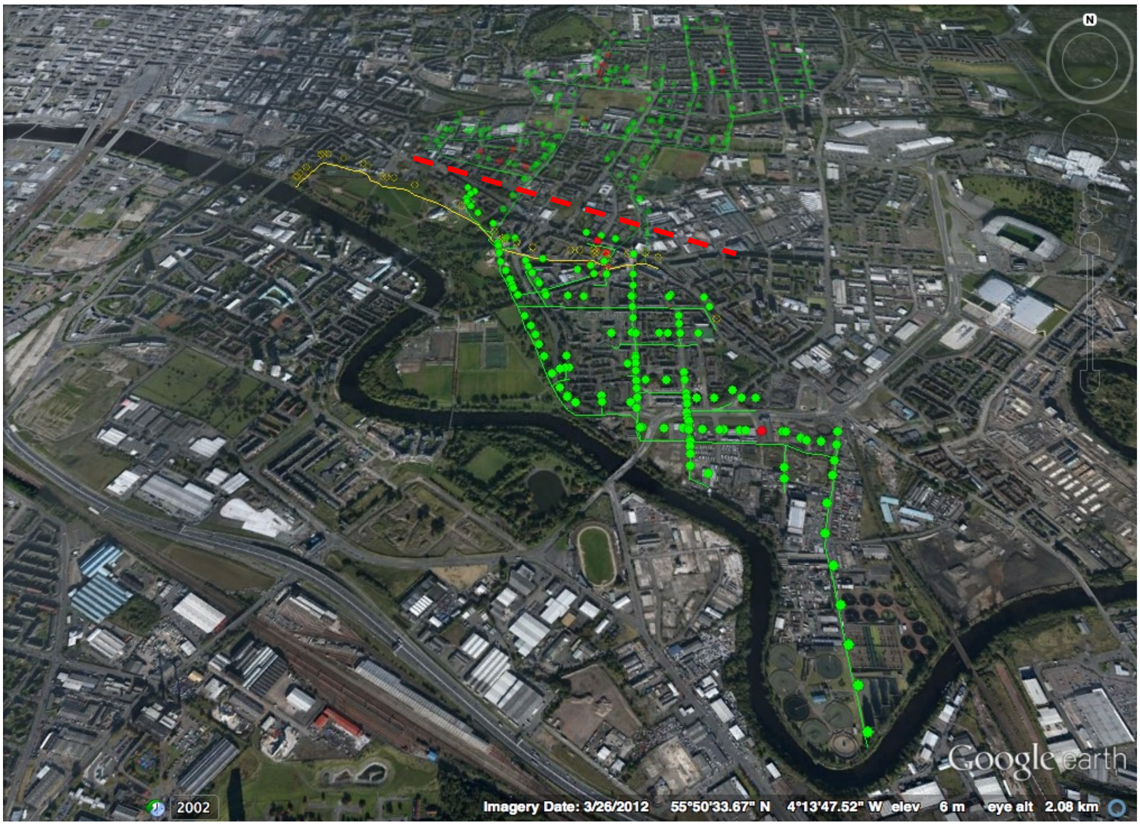


Figure 7 - Network diagram of reduced rainfall set networks (pipe diameters and storage node volumes will vary per network).

Net 1 (£)	Net 2 (£)	Net 3 (£)	Net 4 (£)	Net 5 (£)	Net 6 (£)	Net 7 (£)
2,969	4,949	11,765	12,181	19,861	1,540	1,939

Table 1: Full EAD figures from networks 1-7

Net 8 (£)	Net 9 (£)	Net 10 (£)	Net 11 (£)	Net 12 (£)	Net 13 (£)
11,045	12,407	12,429	12,440	15,363	15,706

Table 2: Full EAD figures from networks 8-13

Net 14 (£)	Net 15 (£)	Net 16 (£)	Net 17 (£)	Net 18 (£)	Net 19 (£)	Net 20 (£)
15,304	16,384	17,619	17,773	17,977	17,985	18,356

Table 3: Full EAD figures from networks 14-20

Net 1 (£)	Net 2 (£)	Net 3 (£)	Net 4 (£)	Net 5 (£)	Net 6 (£)	Net 7 (£)
2,692	4,949	11,765	12,181	19,858	1,540	1,939

Table 4: EAD values from runs using 600 durations only, networks 1-7

Net 8 (£)	Net 9 (£)	Net 10 (£)	Net 11 (£)	Net 12 (£)	Net 13 (£)
11,045	12,407	12,429	12,440	15,363	15,706

Table 5: EAD values from runs using 600 durations only, networks 8-13

Net 14 (£)	Net 15 (£)	Net 16 (£)	Net 17 (£)	Net 18 (£)	Net 19 (£)	Net 20 (£)
15,303	16,383	17,617	17,771	17,976	17,983	18,355

Table 6: EAD values from runs using 600 durations only, networks 14-20

Initially, a good spread of return periods was selected, with the goal of narrowing down which area of the spectrum requires the most focus. Previous work, where this kind of reduction in return periods has been accomplished (Woodward, 2012) with minimal impact on accuracy, suggests the use of seven return periods. Therefore, the initial number of return periods chosen was similar, with eight return periods (2, 10, 100, 200, 300, 500, 750 and 1,000 years) selected as being a fairly evenly spaced set (over our full set of return periods) and, therefore, probably a good starting point to narrow down from. It incorporates:

- Two of the very lowest return periods in our set (in order to capture a 0 damage rainfall event)
- The four most extreme return periods (as the distribution of RP's at more extreme values is fairly sparse in our set), which will definitely be damage-causing events
- Two evenly-spaced rainfall events to link these two sets together.

The selection of these rainfall events isn't crucially important, as they merely serve as a starting point for the continuing process.

Analysis of this initial test produced a mean absolute error of £4,846. Comparisons of the EAD curves generated versus the correct EAD curves suggested that the resolution of the data was too low at the bottom of the range. This is where inaccuracy in identifying the return periods at which flooding begins has a large effect on the rest of the EAD curve (Kellagher et al., 2009).

With the previous results in mind, a second test was undertaken with only seven return periods but with more of those at the lower end of the range (return periods used were 2, 20, 40, 80, 160, 750, and 1000). The results from this second test (mean absolute error of £2,220) matched the full EAD curve better than the first test at the lower end of the range, but generally diverged (either high or low) towards the middle of the EAD curve.

A third test (utilising return periods 2, 20, 40, 80, 160, 300, 500, 750 and 1000 years) was then run. This was in an attempt to compensate for the high/low divergence by providing better resolution of data at critical points along the EAD curve. This third test performed better (mean absolute error of £1,570), but there was still a lack of resolution at the low end for some networks, although it was generally better across the middle range.

A fourth test was performed with fewer of the low-end return periods, but focusing more on the low-mid range (return periods 2, 80, 160, 300, 500, 750 and 1000 years), in an attempt to cut down on the total number of return periods whilst maintaining accuracy. However, this test produced further inaccuracies, resulting in a mean absolute error of over £13,000, the highest of all tests. The decision was therefore made, for the purposes of testing, to proceed using the rainfall/duration measures from the third set as it was the best match for the full EAD curve.

It is important to note that this process of return period/duration selection would need to be run as a “pre-processing” step for each new network that an optimisation was to be performed upon. The duration/return period selection identified will be biased towards the behaviour of the network in question and will not be generalizable to other networks (Sayers et al., 2014).

It is also worth remembering that as this step is not time-consuming, a second (or third, or fourth etc.) set of rainfall files could be identified using a network or networks, produced by the first optimisation run. Also, a second (or any subsequent) optimisation run could then be undertaken, using the identified set of rainfall that is more closely matched to the optimised network.

An alternate method of reducing the number of durations required is to use mathematical techniques, such as the alternating block hyetograph (TxDOT, 2014) to compress durations by building an artificial rainfall file. This involves determining an interval for the design rainfall file end product, then taking the average rainfall intensity from each rainfall duration file. The cumulative depth and the incremental depth are then calculated (TxDOT, 2014), where the incremental depth is the increase in depth over the previous cumulative depth, with a starting point of zero.

Once these figures are identified, the rainfall is re-ordered within the file. It is ordered so that the most intense period of rainfall is within the centre of the rainfall event, with subsequent intensities positioned on alternating sides in order of intensity. This process is demonstrated in Table 7 & Figure 8 (TxDOT, 2014)

The issue with utilising this approach for this specific test-problem is that the error when using only the highest duration present in the rainfall files available is negligible. Therefore, the extra time that would be taken in converting each of the return periods into an alternating block hyetograph is not worth spending. However, if with a different catchment it was more difficult to identify a single duration with a negligible change in results, this technique could be extremely valuable to combine durations, and indeed has been put into practice by third party engineers at HR Wallingford when utilising the software developed in this thesis (Boelee and Kellagher, 2015).

Duration (Min)	Intensity (mm/Hr)	Cumulative Depth (mm)	Incremental Depth (mm)
10	105.61	17.60	17.60
20	76.25	25.65	7.82
30	59.87	29.92	4.52
40	49.35	32.92	2.97
50	42.03	35.03	2.13
60	36.65	36.65	1.60
70	32.49	37.90	1.27
80	29.18	38.86	1.02
90	26.52	39.78	0.84
100	24.28	40.49	0.71
110	22.43	41.10	0.61
120	20.83	41.63	0.53

Table 7: Calculation of alternating block hyetograph values

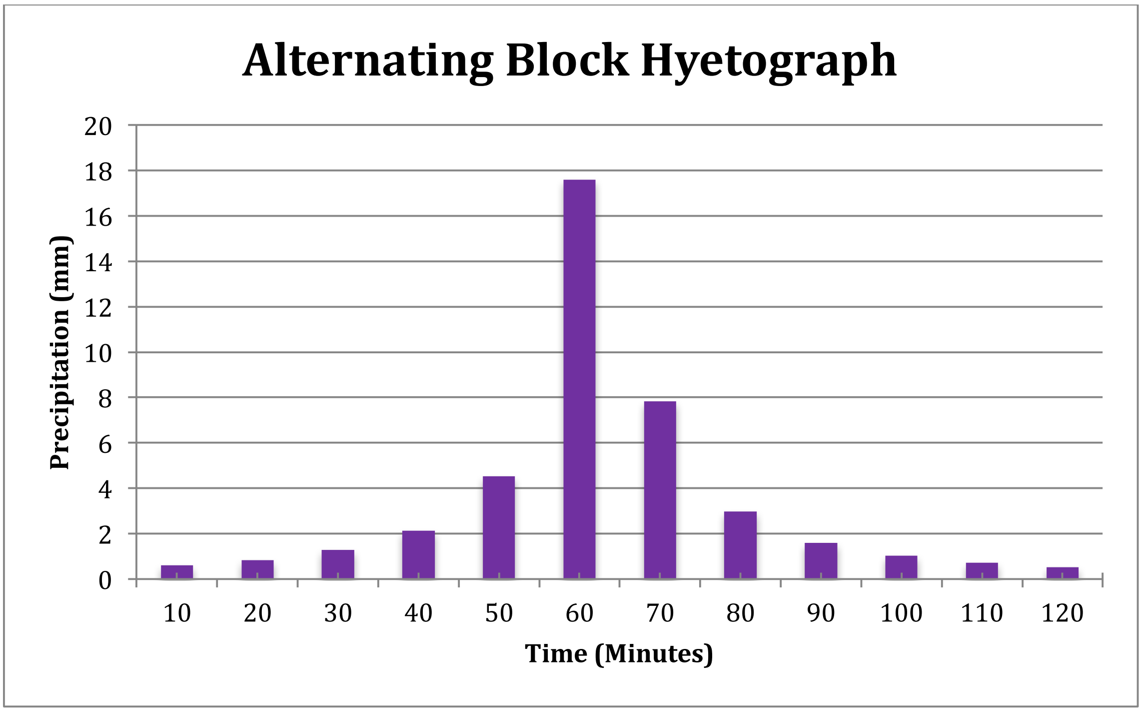


Figure 8 - Alternating block hyetograph

4.3.2 SAM-UMC Modifications

4.3.2.1 Caching of Existing Exported Network

Within the original (DTI-SAM) tool set, whenever the SAM-UMC module is called to initialise an Infoworks CS simulation run as part of the damage calculation, the Infoworks CS network is exported to the Infoworks CS working directory as CSV files, and then imported back into Infoworks.

The reason for this is that the tool set being used as a basis for ADAPT was developed alongside a second tool set which functions differently. This second toolset attempts to take into account drainage system blockages. In order to do that it has to make changes to the drainage network model in Infoworks between simulations.

In this project, however, there is no need for changes to be made between every single Infoworks CS simulation, because of the way in which the software is being used. The software was, therefore, re-written to track when changes have been made to the network, and only re-export the Infoworks network when those occur. This gives a minor performance saving with a small testing network, but dramatically improves its worth as network size increases, and the number of simulation runs without a network change increases.

It can be seen in Figure 9, that when each and every rainfall test involves the network being completely exported, the runtime increases exponentially, as the number of rainfall files goes up. When rainfall files are only exported if the network is altered, whilst the network is unaltered the run time increase for extra rainfall files is linear. Therefore, the improvement in performance is greater, the larger the amount of rainfall files being used on each network.

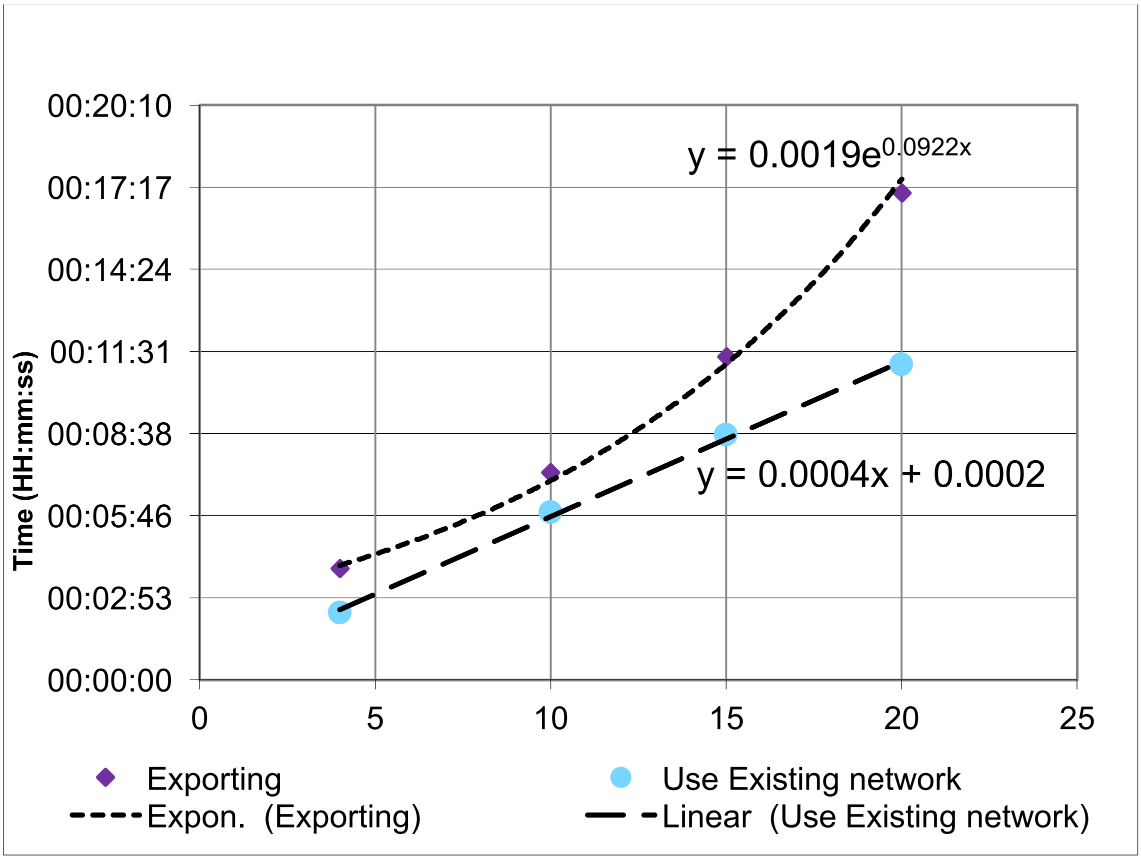


Figure 9 - Exporting drainage network vs. utilising the existing export

4.3.2.2 Software Performance Optimisation

As part of this project, optimisations and improvements to the source code developed originally for SAM-Risk were undertaken. These consisted of refactoring loops, utilising internal caching of numerical figures rather than re-calculation, and re-writing code to take advantage of C# language features introduced after the release of SAM-Risk. This work was completed by the author.

Key areas of the software for this optimisation effort to focus on were identified by means of analysis of which functions the software was spending most of its time performing. A full re-write would probably achieve even more in terms of performance benefits, but would be impractical and too large a time investment when it is impossible to guarantee the performance improvements achievable.

On the testing network, an improvement of 15.11% of time taken was shown through using this new code, i.e. a run which previously took around 6 hours, had been reduced to approximately 5 hours.

This is a considerable improvement in runtime, especially when considered that it was achieved without any loss in accuracy.

4.4 Optimisation Solution

The optimisation methodology and the tool set utilised during this project have been developed entirely during the project, in the C# programming language. The requirements of the sponsoring company, HR Wallingford, also had an influence on the software solution, as they had specific requirements they wished satisfied.

The broad goal of the software was to have a decision support application, with a graphical user interface, that would run under Microsoft Windows and use the industry standard drainage system modelling software (Infoworks CS, developed by Innovyze®) as its drainage model, as did DTI-SAM before it (see section 3.3) provided Infoworks was suitable for inclusion. Developed as part of this solution was a multi-objective optimisation algorithm, to optimise cost versus EAD. This multi-objective optimisation algorithm has been built in a loosely linked modular fashion, to allow utilisation of modules by other future software, and to improve flexibility. Some innovation has also been necessary to allow the implemented multi-objective optimisation algorithm to converge in a realistic time frame. This is not a trivial undertaking as the search space of even a modest drainage network would be too large to expect an unmodified multi-objective algorithm to converge to a useful solution in a reasonable time-period. The following sections describe how this solution was developed.

4.4.1 Development Language and Framework

The software for this thesis is developed entirely in C#, which is typically run on the .NET platform under Microsoft™ Windows ®.

C# is an object-oriented language, which means that the software can be structured in terms of a series of interacting objects. In C# these objects consist of: classes, which are groupings of methods and data for those methods to act upon, and interfaces, which are contractual obligations for a class to implement a certain set of functionality.

Classes can inherit from each other (i.e. a class of type “Motorcycle” may inherit from a class of type “Vehicle”) but only single inheritance is allowed in C#, i.e. a class can only have a single parent. A class can also implement interfaces, and can implement as many interfaces as the developer desires.

An assembly refers to a single file to be executed or linked with. I.e. a single *.exe or *.dll, when referring to .NET assemblies.

4.4.2 Optimisation Specific Performance Enhancements

In terms of specific enhancements with optimisation algorithms in mind, the software has been developed to cache EAD and results from specific solutions.

Each solution currently present in the population of solutions within the optimisation algorithm, tracks whether it has been altered since it was last evaluated for EAD / Cost. If it has, then when EAD / Cost is requested, it triggers a re-calculation of these values, including the Infoworks CS model being launched and a flood risk analysis undertaken to identify the EAD figure. If it has not, then cached values for EAD / Cost are returned.

As the optimisation progresses, a large amount of the population at any given stage is likely to be already evaluated with the results cached. Saving time by not re-evaluating these solutions should give a significant performance benefit.

As an example, if each EAD calculation takes around 40 seconds, and in a population of 100 solutions, an average of half of the solutions EAD values was cached at each iteration of the algorithm, over 5,000 iterations 115 days' worth of computation time would be saved.

4.4.3 Pipe Modelling for Optimisation

4.4.3.1 Non-Circular Pipes

Not all of the pipes within Infoworks CS are circular pipes – a reasonable number are egg-shaped pipes or other types, such as open channels. In combined sewer systems, like the ones prevalent in the United Kingdom, pipe capacities must be relatively larger than in a separated system designed for the same catchment area, as combined systems require the capacity to deal with storm water, as well as wastewater (Butler and Davies, 2010).

As Egg-shaped pipes have a smaller diameter in the lower part of the cross-section, this will be used during low-flow drier periods, whilst maintaining a large cross-section above that, which will come into play when storm flows take place. Egg shaped sewers are common in older drainage systems (Butler and Davies, 2010).

When dealing with circular pipes in Infoworks CS, it is enough to simply change the width of the pipe – the software is intelligently designed enough that it will automatically adjust the pipe as it is aware that the circular profile means that the height should be the same as the width.

When dealing with pipes that are not circular, simply changing the width will result in an altered pipe profile, as the height will remain the same, resulting in an altered pipe profile and potentially affecting the flow rates within the pipe.

One solution considered was to simply convert all non-circular pipes within the Infoworks model to equivalent circular pipes, and test to see whether the EAD figures generated were substantially altered. It was decided to retain this as a potential fall back and instead, add the capability to the software to adjust the height of pipes.

Once this capability was added, the solution followed was to alter the height in proportion with the width, so that the profile would remain the same, regardless of how the width is altered.

4.4.3.2 Pipe Groups

The unit relating to pipes that the implemented optimisation algorithm deals with is a “pipe group”. These pipe groups contain one or more pipes, and when the size of any pipe within the group is altered, all other pipes are modified to the same size.

This feature is designed so that pipes within the model which flow from one to another in sequence, could potentially be grouped together where, logically, there would be little point in changing one pipe by itself.

This would require an in depth analysis of the drainage model being used, and would limit the optimisation algorithm in terms of the potential solutions. However, it could be worthwhile, particularly if there was a circumstance where a pipe is represented in the model as several separate pipes, but is actually effectively one long section of pipe where the sub-sections could not be replaced individually.

4.4.3.3 Original Pipe widths and Restricting Size Alterations

Two additional improvements were made to the modelling of pipes within the optimisation algorithm, the first is the tracking of the original pipe widths within the pipe group, so that when a costing is requested from a pipe group object, it knows when to return £0.

A second improvement, based upon the first, is that it is possible to restrict a pipe group object from setting the pipes within it to anything smaller than their original size. This could be an important engineering consideration (as it may be known that the flow rate away from a certain part of the catchment should not be decreased) and will have the added benefit of reducing the size of the search space.

4.4.4 Storage Modelling for Optimisation

Modelling the storage nodes for use within the optimisation process proved to be less complex than modelling pipes. In a similar manner to pipes, storage nodes track their original size (and return £0 cost if they have not been altered from their original, or if they are altered back to their original), and can be set to only grow and never shrink.

There is no need for grouping of storage nodes in the same manner as is required for pipes, and as storage node area is set directly in terms of metres cubed, there is no need for any calculation of width/height etc.

It is also possible to set minimum/maximum values for the storage node area, below/above which the storage area will not be considered. This is necessary, as there may only be space available for up to a given amount of storage space.

Storage nodes consist of both a chamber area, and a shaft area, which are summed to give the total storage node area for the cost calculation.

4.4.5 Orifice Modelling for Optimisation

An orifice is a construct within a pipe network that forces the flow within a pipe to pass through an area less than that of the pipe. An orifice usually either creates a smaller circular area, or sections off a portion of the top of the pipe (Butler and Davies, 2010). The purpose of an orifice is to use Bernoulli's principle, which states that as fluid flows faster, the pressure is decreased (Batchelor, 1967). An orifice plate within a pipe causes fluid to flow faster, decreasing pressure. Orifices are also modelled within the optimisation process, although they are again simpler than pipes in terms of how they are represented to the optimisation algorithm.

Orifices within the optimisation process have a discharge minimum, a discharge maximum, and a definition of how much each "step" is between the minimum and the maximum. Together these three pieces of information identify the exact values that it's possible for an orifice to be set to. Cost for modifying an orifice is a flat figure (as any modification will cost the same as any other) and is returned if the orifice has been altered from its original state, which it tracks in the same manner as pipes and storage nodes. If it is still in its original state, cost associated is £0.

4.4.6 Cost Groups

All types of decision variables (Pipes, Orifices and Storage) can be assigned to cost groups in their various setup files. Each cost group has its own predefined set of constants for cost algorithms. Therefore, separate variables which have varying costs can be grouped together and the appropriate constants applied. This can be seen in Appendix III – SAM-Risk Settings.

4.4.7 NSGA-II

The NSGA-II implementation follows the algorithm detailed in Deb's paper (2002) (see Figure 10). It is implemented in a modular fashion, to allow further development and use of the algorithm in other work, as well as to allow the integration of meta-modelling techniques as part of this project.

A simplified class diagram for the NSGA-II implementation used in this work can be seen in Appendix I – Software Diagrams. In words, the NSGA-II main class and the chromosome class between them handle most of the functionality of the algorithm, with customisable elements (such as population creation) specified via a supplied class, which must implement the NSGA2Adaptor interface. This can be supplied from an external assembly and need not be present within the NSGA2 assembly, provided it implements the correct interfaces.

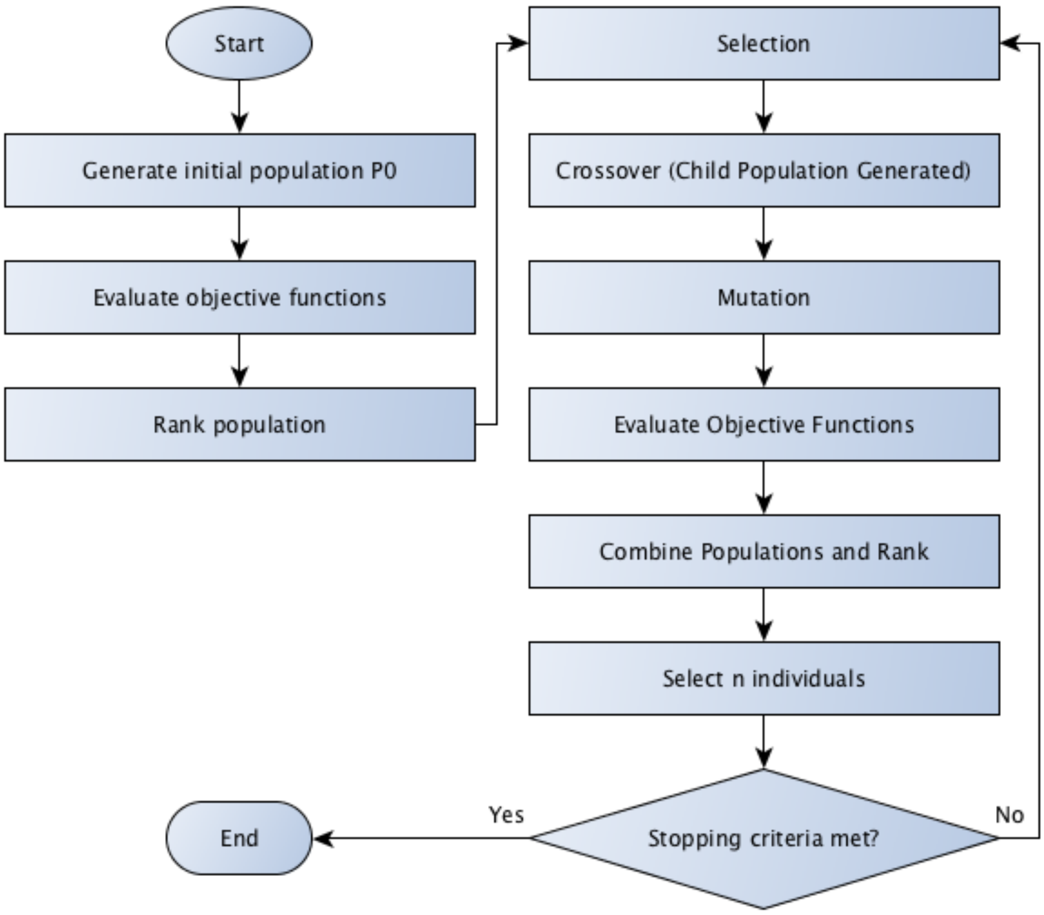


Figure 10 - NSGA-II flowchart

A population consists of a collection of chromosome objects, which are defined in the aforementioned chromosome class. Individual genes within a chromosome class are defined by the algorithm user, but must implement the “IGene” interface, and again can come from an external assembly.

The calculation of objective functions should be handled entirely within the NSGA-II adaptor class. The type of this class should be given to the NSGA-II class during instantiation and it will then be instantiated internally to that class.

4.4.8 NSGA-II and Machine Learning

There are various ways that NSGA-II could theoretically be combined with meta-modelling techniques in order to decrease the computational impact of objective function evaluation and therefore increase the speed and possible accuracy with which a reasonable Pareto front can be achieved.

These various techniques are covered in (2.5). The main technique implemented in this research project is the LEMMO technique (Jourdan et al., 2005) although other techniques were investigated, notably MOGA-ANN (Behzadian et al., 2009).

Issues were encountered using the MOGA-ANN approach detailed in Behzadian et al. ((2009). The approach used was to attempt to train an ANN to estimate EAD, with enough accuracy to allow the optimisation algorithm to discern between poor and promising solutions.

The MOGA-ANN approach involves recording the data in and out of the objective function to be supplanted, or partially supplanted (in this case EAD), by the artificial neural network, for a number of generations. This results in a collection of training data for the neural network meta-model. Once this data collection phase is complete, the neural network is initially trained on this data, to attempt to approximate the output of the objective function. After this initial training phase, all potential solutions are evaluated by the neural network and ranked according to non-dominance (Deb et al., 2002). After they have been ranked, a specified number of ranks from the first rank upwards is re-evaluated utilising the full model (the original objective function) in order to maintain full accuracy.

The data (input and output) from each run is stored and added to the training data pool. Periodically (every 'n' generations) the neural network is re-trained using all data that has been collected since the last training event. In this way, the neural network is effectively being utilised as a filter, to remove poor solutions from consideration before expending computing resources doing a full analysis. Additionally, the extra training should result in the neural network becoming increasingly more accurate with regards to higher quality solutions as the algorithm progresses.

The ANN appeared, however, to struggle to estimate the EAD figure with enough accuracy. The learning algorithm appeared to be running to as many iterations as was allowed on every training iteration, and many of the networks returned identical expected annual damage figures that proved inaccurate when the network was re-tested without the ANN.

As, at this point, LEMMO was returning promising results from its testing, the decision was made to continue and focus on LEMMO, although it is possible with enough improvement to the ANN structure that accuracy could have been improved enough to allow the algorithm to work as effectively as it has in previous studies.

The normal approach when trying to utilise meta-models as part of an optimisation process is to substitute them for part of the algorithm. Typically, this involves replacing the costly objective function or functions, reducing the number of objective function evaluations that would be required per iteration, or a combination of the two (Behzadian et al., 2009; Broad et al., 2005).

4.4.8.1 Artificial Neural Networks

A popular form of meta-model is the ANN or artificial neural network (see section 2.4.2). An ANN can be thought of in this context as a function approximator, which is capable of approximating any given function to any required degree of accuracy, provided enough data exists to train it. An additional requirement is that the ANN is structured appropriately.

An ANN needs to have enough nodes that it is capable of forming a reasonable approximation to the problem. However, it is important not to have too many nodes, or there is a risk of the network being prone to over-training, at which point it will solve the specific problems it has been trained on well, but will not generalise out to other problems. Most applications of neural networks rely on the network generalising well to problems different from the specific examples it has been trained on, so this is something to be wary of.

In order to save development time a third-party machine-learning library was utilised, called Accord. The Accord library is very well documented (Souza, 2015) and incorporates the functionality of generating many different kinds of machine learning algorithm that can be utilised with impressive flexibility.

The artificial neural network used in this project by way of the Accord library is a feed-forward neural network trained by means of a resilient propagation algorithm

(RPROP) (see section 2.4.2) (Igel and Hüsken, 2000; Riedmiller and Braun, 1993). A feed-forward neural network was selected as this type of network is robustly proven to be a universal approximator (Cybenko, 1989; Hartman et al., 1990; Hornik, 1991; Hornik et al., 1989; Park and Sandberg, 1991). The function of the network, where a given number of inputs are associated with a specified output and the weights are altered to give an input/output mapping for the problem fits well into the context of being used within another algorithm. Multilayer feed forward artificial neural networks degrade in performance gracefully, as the amount of noise in the input increases (Svozil et al., 1997). ANN's also cope well with being trained online, which is important for the applications detailed in this thesis.

The Accord library was used after trialling and comparing two other libraries – FANN (Nissen, 2012, 2011) and Encog ((Heaton, 2014). FANN caused issues because of a hard to debug memory leak – which was present either within the FANN code itself or possibly within the managed to unmanaged interface. The effect of this memory leak was that when included within a large optimisation algorithm such as being described in this thesis, the computer system running the algorithm would crash after somewhere between 100 and 300 iterations. Encog had no major issues of that sort, but did not support the same variety of algorithms as Accord. In particular, Accord supports decision tree classifiers, which was something that potentially could have been used as part of the LEMMO algorithm, although that did not end up being the case.

RPROP was chosen as it is a fast and effective alternative to standard back propagation – due to the way in which the various approaches integrate the artificial neural network, having a large or complex training algorithm is likely to have a significant impact on performance, so something relatively simple, but proven and effective (Igel and Hüsken, 2000; Riedmiller and Braun, 1993) was required.

4.4.8.2 Development of LEMMO with ANN

Two main approaches were followed when testing the combination of neural network meta-models and NSGA-II, the MOGA-ANN approach (Behzadian et al., 2009) and the LEMMO approach (Jourdan et al., 2005).

Learnable evolution models (LEM) have been used successfully on single objective optimisations (Michalski et al., 2000). A LEM approach to optimisation consists of normal evolutionary algorithm execution, interspersed with “Learning phases” (Michalski et al., 2000). During a learning phase, data collected within the evolution phase is used to train a machine-learning algorithm to differentiate between more and less optimal solutions.

The machine learning algorithms used in previous implementations of LEMMO are decision tree classifiers. In this case the machine-learning algorithm is an artificial neural network, which necessitated some development of the algorithm. This machine-learning algorithm is then utilised to generate a new population for the next evolution phase to use as a starting point. This functionality had to be adapted somewhat to make it applicable to multiple-objective optimisation.

Five variants were examined in Jourdan's (2005) paper, referred to as LEMMO-1, LEMMO-fix1, LEMMO-fix2, LEMMO-fix3 and LEMMO-fix4. These are described in section 2.5.3.

An approach based upon LEMMO-Fix4 has been used as this was recommended in Jourdan et al. (2005) and testing in di Pierro et al. (2009) led to that study also following this recommendation. Unfortunately, due to memory constraints it is not possible to maintain a full list of all individual solutions found so far.

This problem might be surmountable by modifying the code to maintain a database of found solutions, or by altering the architecture of the program to allow access to a larger amount of memory. As neither of those approaches was available to this project, all found solutions between the last LEMMO iteration and the current iteration (i.e. ten iterations worth of solutions) were stored. The approach taken can be seen in Figure 11.

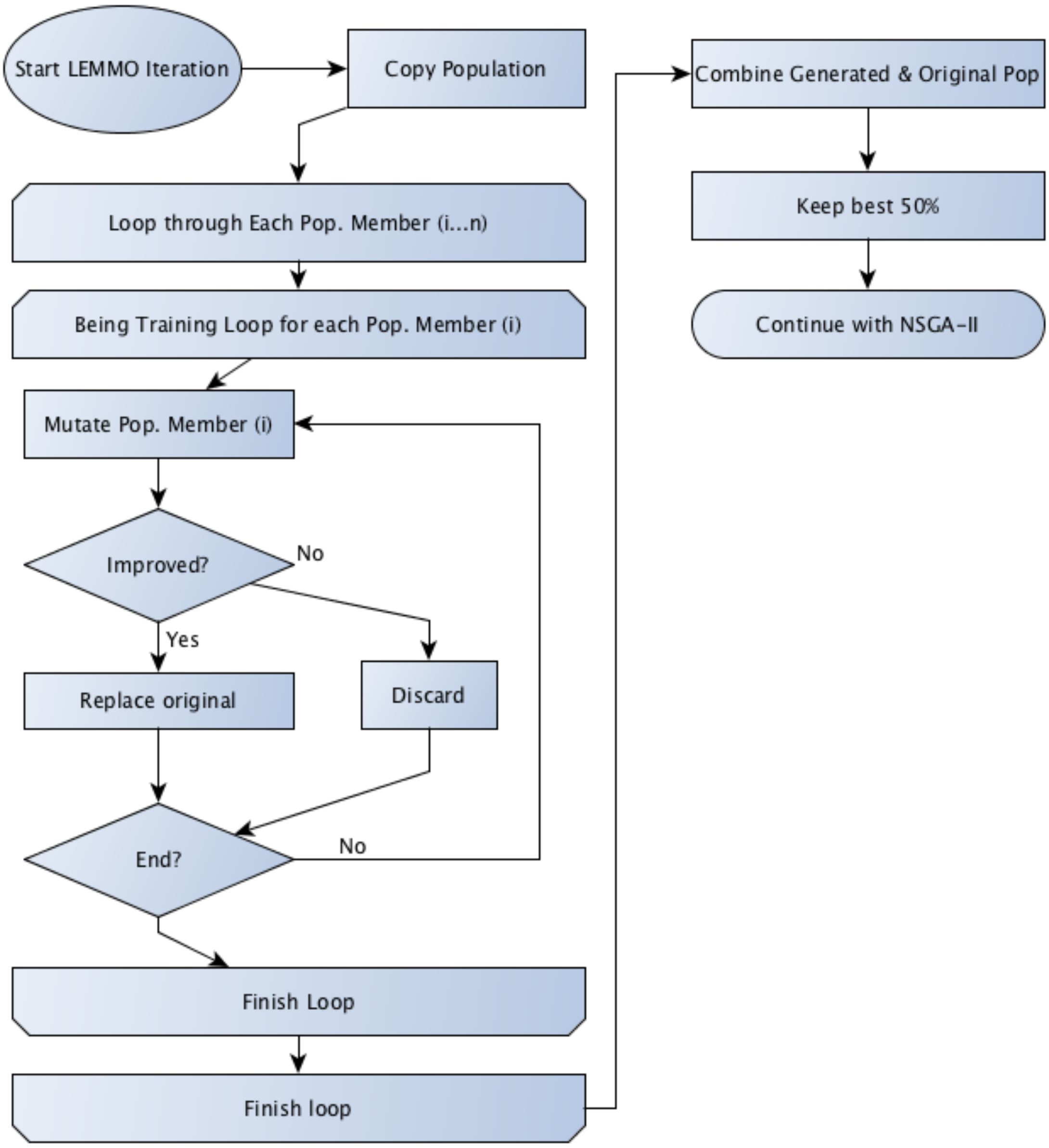


Figure 11 - Process to create a new LEMMO population using ANN

The feed-forward artificial neural network was trained using RPROP with the best thirty percent of these solutions from the last ten generations as the “good” set and the worst thirty percent as the “bad” set. In order to generate solutions that match the “good” set and do not match the “bad”, solutions are generated for each of the population members in turn. These solutions are then mutated and evaluated (by the ANN), discarding poor mutations and retaining the mutations that improve the solution. At this stage, no solution can enter the population if a solution already exists with the same characteristics. Once either an iteration hard-limit has been reached, or the solution matches the “good” and doesn’t match the “bad” solutions with a certain degree of accuracy, the solution is retained. Once this procedure has finished for all population members, the population is combined with the original population, and the best 50% retained.

At this stage, no solution can be entered into the population that already exists there. Finally, this newly generated population is treated as a new child population within the NSGA2 algorithm. This means that a conglomerated population of the current solutions, plus these newly generated solutions is created, evaluated, ranked, analysed for crowding distance, sorted by rank then crowding distance, and the best 50% retained for the next iteration of NSGA2. In this way it is ensured that only improved solutions generated by LEMMO will persist into the NSGA-II population, the rest will be discarded.

This is a novel development, as LEMMO has previously been used only with classifiers where it is easier to extract rules upon which the classification is being based. Research into this area, such as has been performed here, is therefore of high value.

How effective this algorithm is when applied to flood risk optimisation problems will be shown in chapters 5 and 6

4.5 Optimisation Assessment

4.5.1 Introduction

In this section the approach that has been used to assess the previously described algorithms will be covered. These algorithms needed to be analysed in two ways, firstly, the effectiveness of each algorithm in relation to the others at approximating the true Pareto front. Secondly, their relative efficiencies in doing so, i.e., can some approaches produce a better or equal approximation with fewer objective function calls. Finally, how well the identified most effective algorithm optimises using a real test case.

4.5.2 Optimisation Set Up

A selection of test problems and associated Pareto fronts specifically designed to be utilised as a benchmark for the testing of optimisation algorithms have recently been published by the Centre for Water Systems at Exeter University (Wang et al., 2014).

These test problems are an extremely good candidate for testing LEMMO with ANN's and comparing it to NSGA-II. There are marked similarities between these water distribution system problems and flood risk problems, in terms of the variables that control them (pipe sizes, and arrangement) and the non-linear complexity of the problem. Additionally, they have been very thoroughly researched (Wang et al., 2014) and a reasonable estimated Pareto front found for each of them.

Testing the developed algorithm on these problems gives a very good indication as to whether it converges to a reasonable Pareto front estimation (see chapter 5) and how well it works on complex combinatorial non-linear problems. No surface runoff needs to be analysed for these problems, plus these problem setups are developed with optimisation algorithms in mind, using software models it is easy to interact with and that runs quickly, EPANET2 (Rossman, 2000).

A subset of these tests has been utilised and repeated with the algorithms developed for this thesis, in order to test their capability to converge to a reasonable approximation of the true Pareto front. The tests have also been analysed to determine how efficiently and effectively they converge before any attempt to utilise them on a real test case. This testing has been performed by the author.

Once the algorithms are converging to a reasonable approximation of a Pareto front, and it is known which algorithm is achieving this most efficiently and effectively, this algorithm can be utilised with a real test-case and a reduced rainfall set (as described in section 4.3.2). The results generated by this algorithm and the real test case were then analysed for EAD using the original SAM-RISK approach and a full set of rainfall files.

For the purposes of this thesis, several collections of settings are required for the various tests that have been utilised to analyse performance of the various algorithms and how well they are matching a Pareto front. These can be separated into two distinct groups, one comprising several collections of settings, and one containing only one. These groups are, artificial test case settings, and ADAPT optimisation settings.

The settings of the artificial test cases described in chapter 5 have been arrived at by initially using similar settings to those used in the paper Wang et al. (2014). Where settings are unique to the specific implementation being described in this thesis, sensible seeming defaults have been selected. This combination of settings based on previous experiments and sensible defaults will then be iteratively improved by trial and error, until the final combinations are arrived at.

4.5.3 Estimating Pareto Front Accuracy

Estimating the accuracy of the Pareto front is challenging because it is impossible to know with certainty what a completely perfect Pareto front looks like for any problem which is too large to exhaustively evaluate in a reasonable time-period. An exhaustive evaluation of any of the reasonably sized problems available would take a prohibitively long time to complete under any circumstances.

It is necessary, because of this limitation, that the final output of the algorithm is evaluated without any prior knowledge of the optimal output. In the case of the chosen test problems, as previously mentioned, several Pareto fronts have been generated by means of running a number of algorithms a number of times, then performing a non-dominated sort on the combined output to achieve a super Pareto front as reported by Wang et al. (2014). Where the optimal Pareto front is known for these test problems (i.e. where the problems are trivial enough that they can be exhaustively computed) the Pareto fronts generated by this technique correlate very well and all fronts appear to be high quality results. This work was intended to produce a “reference set” of Pareto fronts for several problems.

The research generated Pareto fronts using these test problems and newly developed algorithms. The results were compared to the Pareto fronts that form the “reference set”. The Pareto fronts were then compared visually, in order to check for reasonable correlation, then in terms of diversity of the Pareto front, in terms of convergence to the “reference” fronts and in terms of the Hypervolume metric (Fonseca et al., 2006). These tests were performed using both the standard NSGA-II algorithm, and with the adapted NSGA-II with the LEMMO algorithm included. This allowed the comparison of the performance of these two algorithms, as well as confirmed that both algorithms were performing as expected.

The performance of the two approaches was, therefore, firstly compared in order to confirm which of the two was the better performing. Once this had been determined, the better performing algorithm was applied to the Dalmarnock model (see chapter 6). This is in order to confirm that the algorithm continues to perform well and converges when applied to a drainage network model via the expected annual damage optimisation objective.

4.5.4 Diversity Metric

The first of the three selected performance metrics is the diversity measure described by Deb et al. (2002). This measure involves calculating the Euclidean distance between each member of the generated Pareto front and its neighbour.

The extreme solutions are then calculated in Deb's implementation by fitting a curve parallel to that of the true Pareto-optimal front. The extreme solutions are found by calculating the values of both objectives for the problem in question for two cases. The first case being where all pipes and storage nodes are the maximum allowed size, and the second case being where all pipes and storage nodes are their initial size (i.e. cost will be 0, EAD will be at its starting value).

In equation 16 the process for calculating the diversity metric is described, where "d_f" and "d_l" are the Euclidean distances between the extreme solutions and the boundary solutions of the non-dominated set. Meanwhile "d" represents the average of all distances for the non-dominated set.

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} \left| d_i - \bar{d} \right|}{d_f + d_l + (N-1)\bar{d}} \qquad (16)$$

With this measure, lower numbers are better, as they indicate a more uniform spread of solutions along the estimated Pareto front, covering larger areas of the estimated Pareto front. This measure has a benefit in that it can be applied to problems where the true Pareto front is unknown provided one can calculate the extreme end-points of the true Pareto front (Deb et al., 2002).

4.5.5 Convergence Metric

This metric involves measuring how close the various points in a non-dominated set are to another set of coordinates (representing either a true Pareto front, or another estimated Pareto front which is believed to be a superior approximation). It is based upon the measure described in Deb's (2002) paper on NSGA-II. In

Deb’s original metric a set of 500 uniformly spaced solutions is selected from the superior front. For each calculated solution to be compared, the minimum Euclidean distance of that point from the chosen solutions in the superior front is then computed. The average of all these distances is used as the metric. Therefore, the lower the average of these distances, the better the score.

The issue encountered with Deb’s metric is that in a situation where there are fewer solutions on the estimated Pareto front than the true Pareto front a very low value can be obtained. This could give a false impression as to how close to matching the Pareto front an estimated front may be.

A modification has, therefore, been made by the author to overcome this problem. The solutions on the best-known front are taken and for each of those solutions the minimum Euclidean distance to a member of the set of algorithmically generated solutions is identified. The average of those distances is then taken.

The difference can be seen in Table 8, Table 9, Table 10 and Figure 12. These tables and figure contain unitless example data, simply to demonstrate the mathematics (units in a real world application would depend upon the parameters being measures). Table 8 contains the coordinates for data section ‘A’, as well as the minimum Euclidean distance from each of these points to the points in the Pareto front. The average of these points is 1.21.

Table 9 contains the coordinates for data section ‘B’ from Figure 12 and, again, the minimum Euclidean distances for these points to the points in the Pareto front. These distances average to 0.35.

The data in these two tables, combined with a visual check on Figure 12, indicates that dataset ‘B’ is a poorer fit than dataset ‘A”. However, because of the different numbers of data points in each dataset, dataset ‘B’ achieves a better convergence value than dataset ‘A’.

On the other hand, looking at Table 10 (in the final two columns) the figures for the minimum distances from each data point in the Pareto front, to the data points in ‘A’ and ‘B’, can be seen. These figures average to 1.04 and 2.47 respectively, giving a better estimation of how far from matching the true Pareto front these two datasets are. Much like the original measure, if there is a perfect match (including identical data-points being found) this measure will produce zero. Therefore, the lower the number, the closer the estimated front is to the true Pareto front.

The mathematical expression for this metric can be seen in equation 17 where ‘x’ and ‘y’ are the coordinates for the Pareto front and accented ‘x’ and ‘y’ are the coordinates for the estimated Pareto front.

A		
X	**Y**	**Distances**
7	0	2.00
6	1	1.41
5	2	1.41
3	3	1.00
3	4	1.41
1	4	0.00
Average distance:		1.21

Table 8 - Convergence metric example data 'A'

B		
X	**Y**	**Distances**
4	1	0.00
5.5	0.5	0.71
Average distance:		0.35

Table 9 - Convergence metric example data 'B'

Pareto Front			
X	**Y**	**A Distances**	**B Distances**
5	0	1.41	0.71
4	1	1.41	0.00
3	2	1.00	1.41
2	3	1.00	2.83
1	4	0.00	4.24
0	5	1.41	5.66
Average distances:		1.04	2.47

Table 10 - Convergence metric example data "Pareto Front"

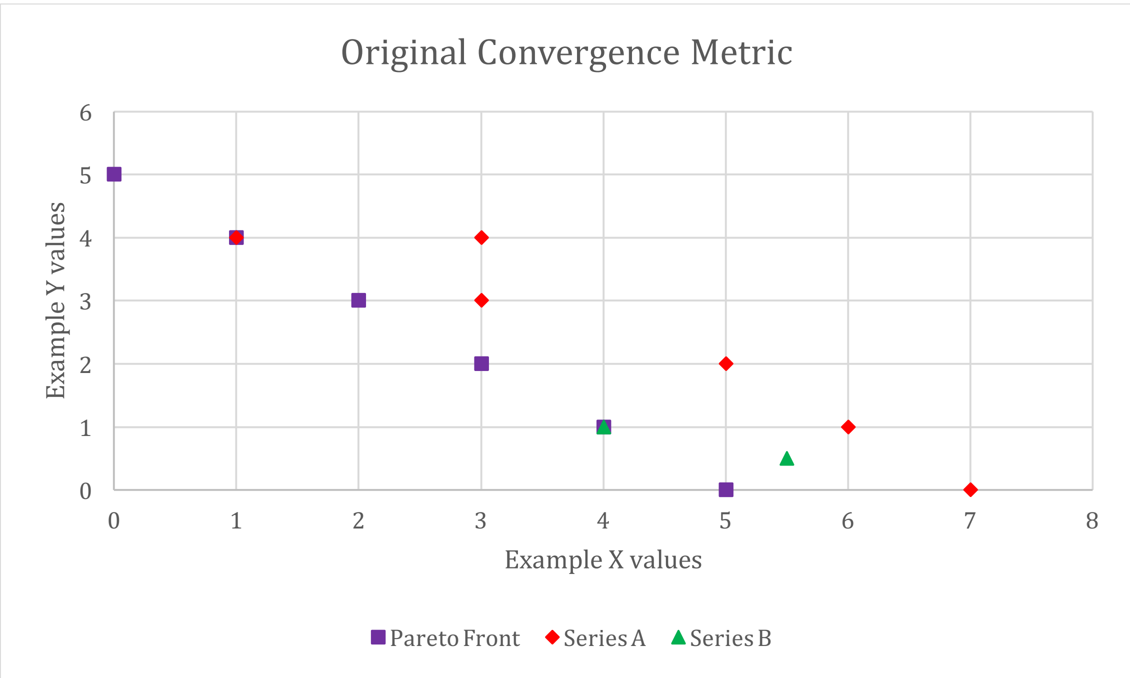


Figure 12 - Convergence metric example data

$$\frac{\sum_{i=1}^{n}\left(\sum_{j=1}^{m} min\sqrt{\left(x_i - \acute{x}_j\right)^2 + \left(y_i - \acute{y}_j\right)^2}\right)}{n} \qquad (17)$$

4.5.6 Dominated Hypervolume Metric

The dominated hypervolume or “S-Metric” is a measure of the Hypervolume dominated by the estimated front (Zitzler and Thiele, 1998). Given a reference point, the volume of space dominated by the estimated Pareto front can be calculated, resulting in a measure by which to compare different estimated fronts. The larger the volume of dominated space (i.e. the higher the numerical value of the metric) the better the estimation of the Pareto front. This can be seen in Figure 13, where the dominated region is indicated by shading.

The reference has to be in such a position that it will encompass the entire Pareto front to be measured. Additionally, the reference point must be the same between separate tests, if they are intended to be compared. Differing reference points could result in wildly different results. Finally, in the given example (see Figure 13) both objectives are being minimized – whereas in our test networks, one objective (resiliency) is being maximised rather than minimised.

We are using the DEAP library implementation of the Hypervolume metric, written in Python (Fortin et al., 2012; Wessing, 2010), executed in the .NET environment alongside C# code using the IronPython python implementation (Foord and Muirhead, 2009). This implementation assumes minimisation on both objectives.

This problem was solved by inverting the scores from the particular objective that is being maximised, before applying the hypervolume metric. This results in a "flip" of the curve for that particular objective, meaning a reference point based upon this new, but equivalent, curve can be provided and the algorithm works without issue or alteration.

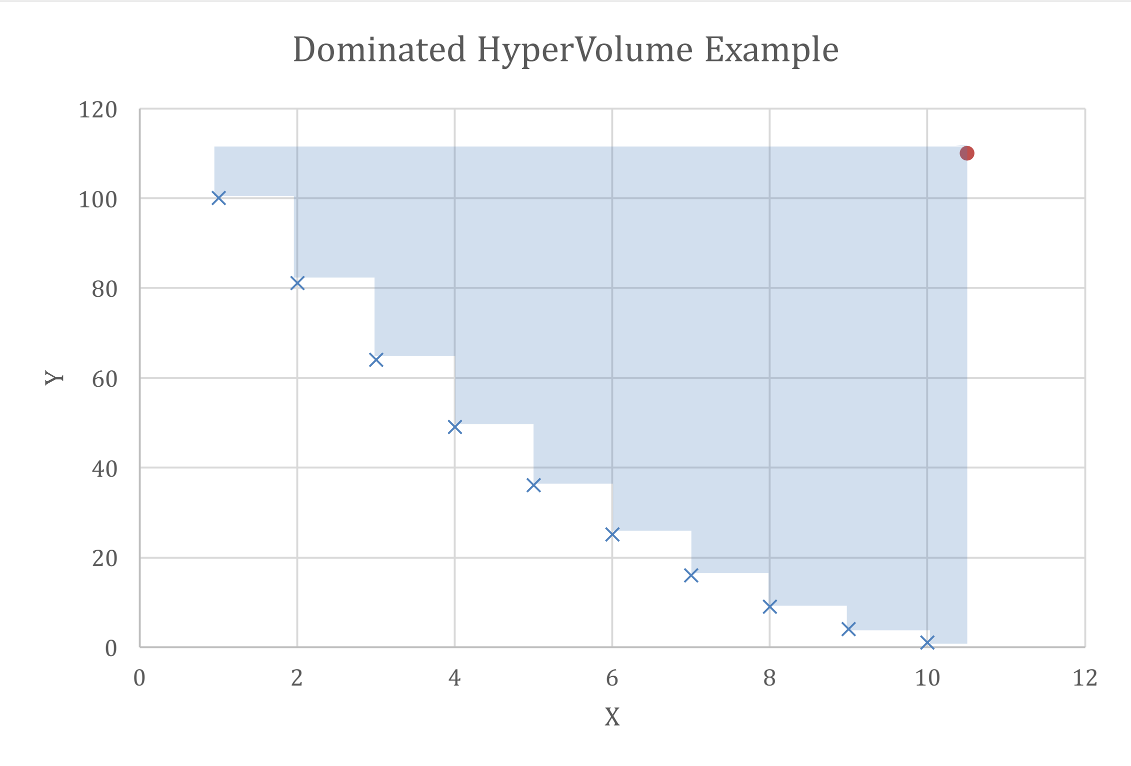


Figure 13 - Dominated Hypervolume example, shaded area represents dominated volume from the single red reference point, to the blue line.

4.6 Chapter Summary

In this chapter, the improvements made to the existing software have been identified. The process of identifying a reduced rainfall set with an expected annual damage that correlates closely to the produced expected annual damage from the full rainfall set has been described. Optimisation methodologies, how they can be improved and how two previously studied methodologies (Behzadian et al., 2009; Jourdan et al., 2005) compare, has been discussed. Finally, how the optimisation processes were set up and the metrics by which they were evaluated has been described in detail. This has laid the foundation for the following two chapters, which examine the results produced from this testing.

5. Water Distribution System Test-Cases

5.1 Introduction

It is important when selecting test cases to ensure that the test problems are as close as possible to the real problem one is attempting to solve. In the case of this thesis, that requires the problem is combinatorial, is non-linear in nature, has a highly complex problem-space and (in order to be useful for the purposes of testing) can be analysed in less time than the full problem could be. The test problems analysed by Wang et al. (2014) fit all of these criteria. Additionally, there are marked similarities in the decision variables (pipe sizes) and the objective functions (cost of the network, plus a measure of network performance). Therefore, it was decided to use a subset of these test problems as the testing set for the LEMMO algorithm developed within this thesis.

5.2 Selection of Tests

In the original paper describing the benchmark problems (Wang et al., 2014) twelve WDS (Water Distribution System) design problems are examined, which fit into four categories: small, medium, large and very large (see Table 11). This is done on the basis of the size of the search space defined by the problem.

Small	Medium	Large	Very Large
Two-Reservoir network (TRN) (3.28×10^{7})	New York Tunnel network (NYT) (1.93×10^{25})	Fossolo network (FOS) (7.25×10^{77})	Modena network (MOD) (1.32×10^{353})
Two-loop network (TLN) (1.48×10^{9})	Blacksburg network (BLA) (2.30×10^{26})	Pescara network (PES) (1.91×10^{110})	Balerma irrigation network (BIN) (1.00×10^{455})
BakRyan network (BAK) (2.36×10^{9})	Hanoi network (HAN) (2.87×10^{26})		Exeter network (EXN) (2.95×10^{590})
	GoYang network (GOY) $(1.24\times10^{27}$		

Table 11: Test problem categories

It was considered that at least one small problem should be included to allow for easy bug testing and modification of software with a small and fast to run problem. Additionally, a smaller problem has the advantage that the best estimated Pareto front has been found by exhaustive search and is therefore known correct. A medium problem was then included, in order to ensure that problems were tested upon across a reasonable range of the complexities available (see Table 11). Finally, two very large problems were included, as these most accurately represent the scale and type of problem that the new approach is designed to solve.

Taking these considerations into account the selected problems are the two-loop network (TLN), the GoYang network (GOY), the Modena network (MOD) and the Balerma irrigation network (BIN). The details of these problems can be seen in Table 12.

Problem	Water Sources	Decision Variables (Pipes)	Pipe Diameter Options	Search Space Size
TLN	1	8	14	1.48×10^{7}
GOY	1	30	8	1.24×10^{27}
MOD	4	317	13	1.32×10^{353}
BIN	4	454	10	1.00×10^{455}

Table 12: Test problem details

In the benchmark test paper (Wang et al., 2014) a computational budget is fixed, in order that the results were repeatable easily by maintaining a similar computational budget. The computational budgets used in this paper for the chosen tests can be seen in Table 13.

Problem	Number of Evaluations	Group 1 Population	Group 2 Population	Group 3 Population
TLN	100,000	40	80	160
GOY	600,000	60	120	240
MOD	2,000,000	200	400	800
BIN	2,000,000	200	400	800

Table 13: Test problem computational budget in original benchmarking

The best-known Pareto set was identified in Wang et al. (2014) by running a large number of different optimisation algorithms, conglomerating the results, and identifying the best non-dominated set from those conglomerated results. Because of this all the results within the best-known Pareto front were not generated using NSGA-II, and it was not expected that during our testing the algorithm would identify every single result that the Wang et al. (2014) identified. The amount and percentage (against the overall total) of solutions identified by

NSGA-II in the best known-pareto fronts for each problem selected can be seen in Table 14 and Table 15.

Problem	Group 1 Contribution	Group 2 Contribution	Group 3 Contribution
TLN	54	74	77
GOY	4	23	31
MOD	71	61	26
BIN	8	67	179

Table 14: Contribution to best-known Pareto front from NSGA-II (Wang et al., 2014)

Problem	Total Solutions in Best-Known Pareto front	Percent Discovered by NSGA-II (%)
TLN	77	100
GOY	67	43.3
MOD	196	57.7
BIN	265	72.5

Table 15: Percentage contribution to the best-known Pareto front from NSGA-II in percentages (Wang et al., 2014)

In Table 14 the number of contributions to the best-known Pareto front can be seen from each NSGA-II group run within Wang's tests. It can be seen in these results that certain problems seem to lend themselves to higher populations, which means a broader exploration of the available search space. Meanwhile, other problems lend themselves to smaller populations but necessarily higher numbers of iterations (to keep to the same computational budget), which means a deeper exploration of the available search space. With regard to the very large problems (particularly interesting for this thesis) one of each of these variants is included (the MOD and BIN problems).

Additionally in the second table (Table 15) it can be seen that during Wang et al. (2014)'s tests, NSGA-II performs very well on TLN, more poorly on MOD and GOY, and then better again on BIN. This pattern is mirrored by the NSGA-II implementation developed for this thesis, as would be expected.

5.3 Objective Function Formulations

The two objectives used for these test problems are network resilience and capital expenditure, network resilience is a measure of how reliable the water distribution network is. The formulation of the objectives can be seen in equations 18, 19 and 20 (Wang et al., 2014).

$$\min C = \sum_{i=1}^{np} a \times D_i^b \times L_i \quad (18)$$

$$\max I_n = \frac{\sum_{j=1}^{nn} C_j Q_j \left(H_j - H_j^{req}\right)}{\left(\sum_{k=1}^{nr} Q_k H_k + \sum_{i=1}^{npu} \frac{P_i}{\gamma}\right) - \sum_{j=1}^{nn} Q_j H_j^{req}} \quad (19)$$

$$C_j = \frac{\sum_{i=1}^{npj} D_i}{npj \times \max\{D_i\}} \quad (20)$$

In equation 18 C represents total cost (monetary units are problem dependant); np represents the total number of pipes; a and b represent constants depending on specific problems; D_i is the diameter of pipe i and L_i is the length of pipe i (Wang et al., 2014).

In equations 19 and 20, I_n is network resilience; nn represents number of demand nodes; C_j, Q_j, H_j and H_j^{req} represent uniformity, demand, actual head, and minimum required head of node j; nr is number of reservoirs; Q_k and H_k are discharge and actual head of reservoir k; npu is the number of pumps; P_i is the power of pump i; γ represents the specific weight of water; npj is the number of pipes that are connected to node j; D_i is the diameter of pipe i connected to demand node j (Wang et al., 2014).

Mathematically, the cost function for these test problems is similar, but less complex than the cost function developed in this thesis. The resilience function is fairly dissimilar mathematically, but conceptually is similar, as it is a measure of network performance, albeit a considerably less complex measure.

5.4 Testing Conditions

The main purpose of these tests is to ensure that the implementations of the basic NSGA-II process, and the heuristic additions converge successfully to a reasonable approximation of the best estimated Pareto front. The best estimated Pareto front is as generated in the paper of Wang et al. (2014). The performance differences between the basic NSGA-II process and the heuristic additions on these test problems were then examined.

In the paper of Wang et al. (2014) the parameters for the various algorithms being tested are specified in detail. The parameter values selected within this study approximate these parameter values (see Table 16). The algorithms used are the standard NSGA-II and an NSGA-II variant with LEMMO (Jourdan et al., 2005).

Both these algorithms use simulated binary crossover, mapped to an allowed set of real values, combined with a mutation operator that selects a random value from within the allowed set of values. This mutation parameter is applied at the gene level – every chromosome is up for mutation, then each gene within that chromosome has the probability of mutation applied. The level of mutation probability (see Table 16) is set so that the probability is equal to the number of genes present in each chromosome divided by one and thus the probability is that roughly one gene will be altered per chromosome. The NSGA-II algorithm functions in such a way that this does not affect the elitism present in the algorithm.

It can be seen (in Table 16) that TLN has a million evaluations with a fairly small population. This is because it is very quick to execute and thus it can be run for a high number of iterations without a large impact on time. GOY runs for six hundred thousand evaluations, as this problem takes longer to run than TLN, and this value matches the number of evaluations allotted in Wang et al. (2014). MOD runs for five thousand iterations with a population of four hundred to give two million evaluations. It was felt important that one of the two very large problems have as long a runtime as was practical, and MOD is the problem which NSGA-II performed most poorly on in Wang et al. (2014). Finally, BIN runs for one million evaluations, as time constraints were strong, BIN was the longest running of the test functions, and NSGA-II had performed reasonably well on BIN in Wang et al. (2014).

	Crossover	Mutation	Maximum Iter.	Pop. Size	Parent Pop. Size	No. Evals.
TLN	1.0	0.125	12,500	80	40	1,000,000
GOY	1.0	0.030	5,000	120	60	600,000
MOD	1.0	0.003	5,000	400	200	2,000,000
BIN	1.0	0.002	2,500	400	200	1,000,000

Table 16: Settings for NSGA-II and NSGA-II LEMMO

5.5 Testing of NSGA-II Base

The NSGA-II base algorithm needs to be tested to ensure that it is converging to a reasonable Pareto front. In this case, a "reasonable" Pareto front is identified partially by comparing to the best-known Pareto fronts available from Wang et al. (2014) and partially by way of the optimisation metrics.

The NSGA-II settings are identical to those for LEMMO (see Table 16), and the standard NSGA-II algorithm is implemented as described in the original literature (Deb et al., 2000, 2002). The crossover technique used is simulated binary crossover (often abbreviated to SBX) (Deb and Agrawal, 1994).

The mutation technique used in the test problems is implemented at the gene level and involves replacing the "value" of the gene to be mutated with another, random, valid value.

5.6 Meta-model Evaluation

It has been necessary to identify a suitable structure for the neural network being utilised. It has been shown that a feed-forward neural network with one hidden layer (see section 2.4.2) is a universal approximator (Cybenko, 1989; Hornik, 1991; Hornik et al., 1989) given the correct parameters (i.e. weight values). However, the question still remains, which is how to find these parameters and how many nodes should be present in a hidden layer to solve a given problem.

Additionally – these papers (Cybenko, 1989; Hornik, 1991; Hornik et al., 1989) do not necessarily prove that the three-layer approach is the most efficient to train and effective for a given problem, only that it theoretically should be able to achieve an approximation. There is no universal rule or system for selection of the correct number of hidden neurons for a given problem, but most suggested guidelines are between zero and 'n' with 'n' being the number of decision variables. Initially, therefore a three-layer network was used, where the number of hidden nodes was equal to the number of decision variables divided by two (see Figure 14).

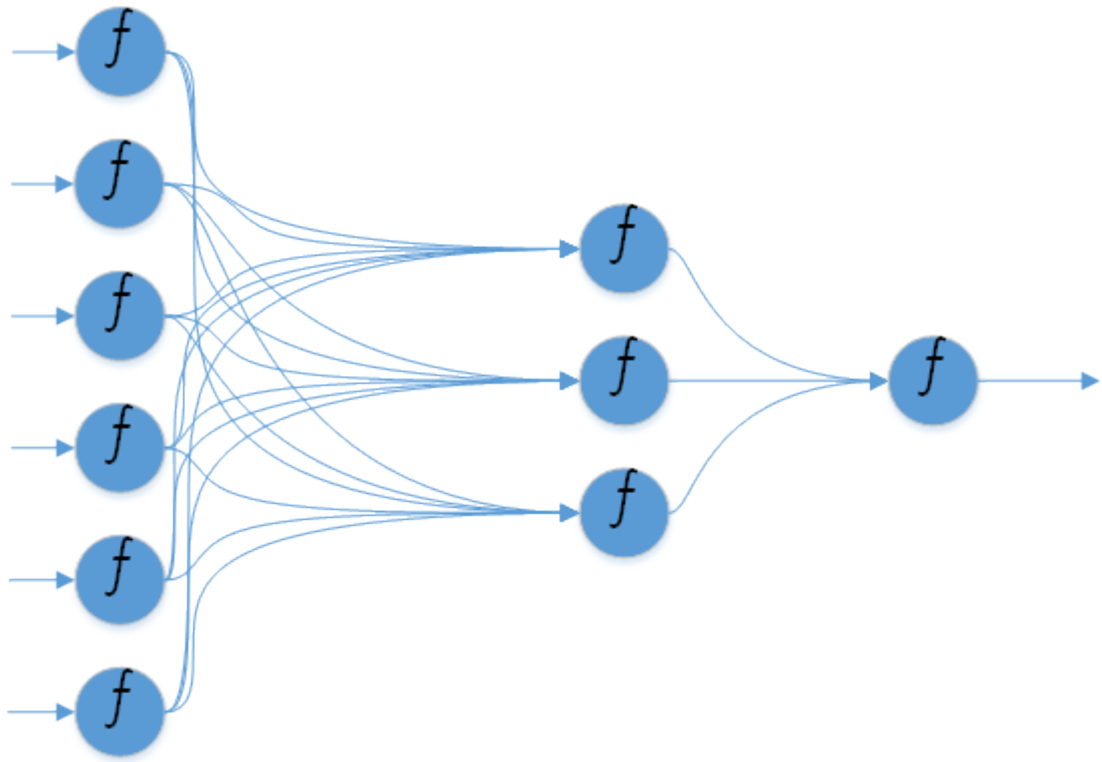


Figure 14 - Initial neural network structure with six input nodes

The training algorithm used was resilient back-propagation (RPROP+) (Riedmiller and Braun, 1993) this seemed to produce an improvement in approximating the Pareto front on smaller problems (such as TLN) but failed to the same for the larger problems. Additionally, by analysing the execution of the code, it could be seen that the neural network-training algorithm was running until it hit a hard limit imposed to prevent endless loops. This was interpreted as meaning that the neural network was struggling to classify the inputs, and was not producing a meaningful enough answer for the LEMMO algorithm to function correctly (di Pierro et al., 2009; Jourdan et al., 2005, 2004).

It has been suggested (Masters, 1993) that a network with two hidden layers could perform better than one with a single hidden layer when approximating complex discontinuous functions. Based on this, it was decided to experiment with adding an extra layer, comprised of half the number of hidden nodes in the previously existing hidden layer (number of decision variables divided by two). This layer is between the previously existing hidden layer and the input layer (see Figure 15). This network arrangement produced considerably improved results over the previous arrangement (see section 5.7) and with time being a factor it was decided to progress using this network architecture.

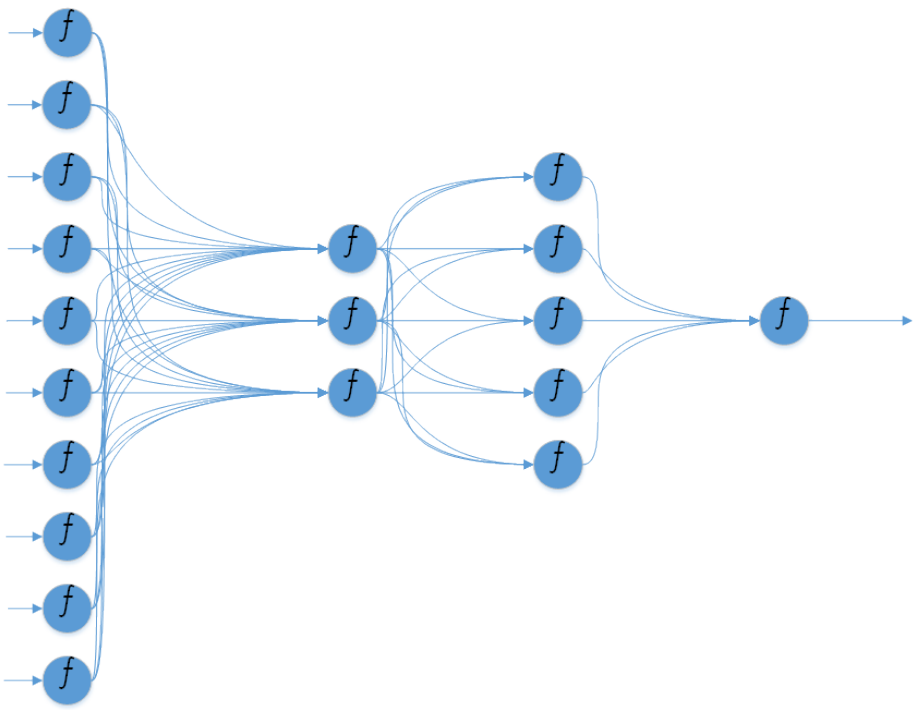


Figure 15 - Graphical representation of the neural network structure with ten input nodes (for illustration only - test-problems and real problems should have considerably more inputs).

5.7 Accuracy of Generated Pareto Fronts

In order for the two different algorithm types to be analysed for each of the four problems, twenty tests were run for each, first without and then with LEMMO. Additionally, a sample of the initial results with LEMMO (before the neural network structure was changed from three to four layers) are included. These tests were halted when it became clear that the LEMMO algorithm with a three-layered ANN was not offering the improvement hoped for, and the algorithm checked to attempt to determine why the improvement seen in the original LEMMO paper (Jourdan et al., 2005) was not being realised. Therefore, there are fewer of these results (18 for BIN, 17 for MOD, and 19 for TLN) and the GOY test problem was not run with this configuration, as the tests were brought to a halt earlier than originally planned. Finally, the full twenty tests were performed for each test case with the LEMMO algorithm with a four-layered ANN.

For each of these twenty tests (or in the case of the three layer ANN, 18, 17 and 19 respectively for BIN, MOD and TLN) two tests that are a reasonable representation of the overall results were selected to be shown in more detail in this chapter. Each of these selected tests were separated into evenly spaced iterations, in order to show clear progression of the optimisation in the way which would be expected. A visual comparison of these tests also holds some value. It may be worth noting that for every single iteration, of every single result, a graph was generated and inspected. However, for brevity's sake, not all were included.

The numerical analysis of the results has been undertaken via the use of three metrics, convergence, diversity and dominated hypervolume, which are based on the mean results from all iterations of each test. The full results tables for all these graphs can be seen in Appendix II – BIN Data Tables.

5.7.1 NSGA-II Base Algorithm

5.7.1.1 TLN

TLN is the simplest of the four test problems being utilised in this thesis, and this is reflected in the graphs seen below (Figure 16 & Figure 17). It can be seen that by iteration ten (our first measure) the algorithm is already reasonably progressed towards the Pareto front, and by the last iterations the Pareto front is effectively found – with many of the points being identical to the Pareto front. This is because TLN is a trivial problem to solve and can in fact be exhaustively computed. Because of this, it is very useful as a measure of whether a multi-objective algorithm is functioning as expected.

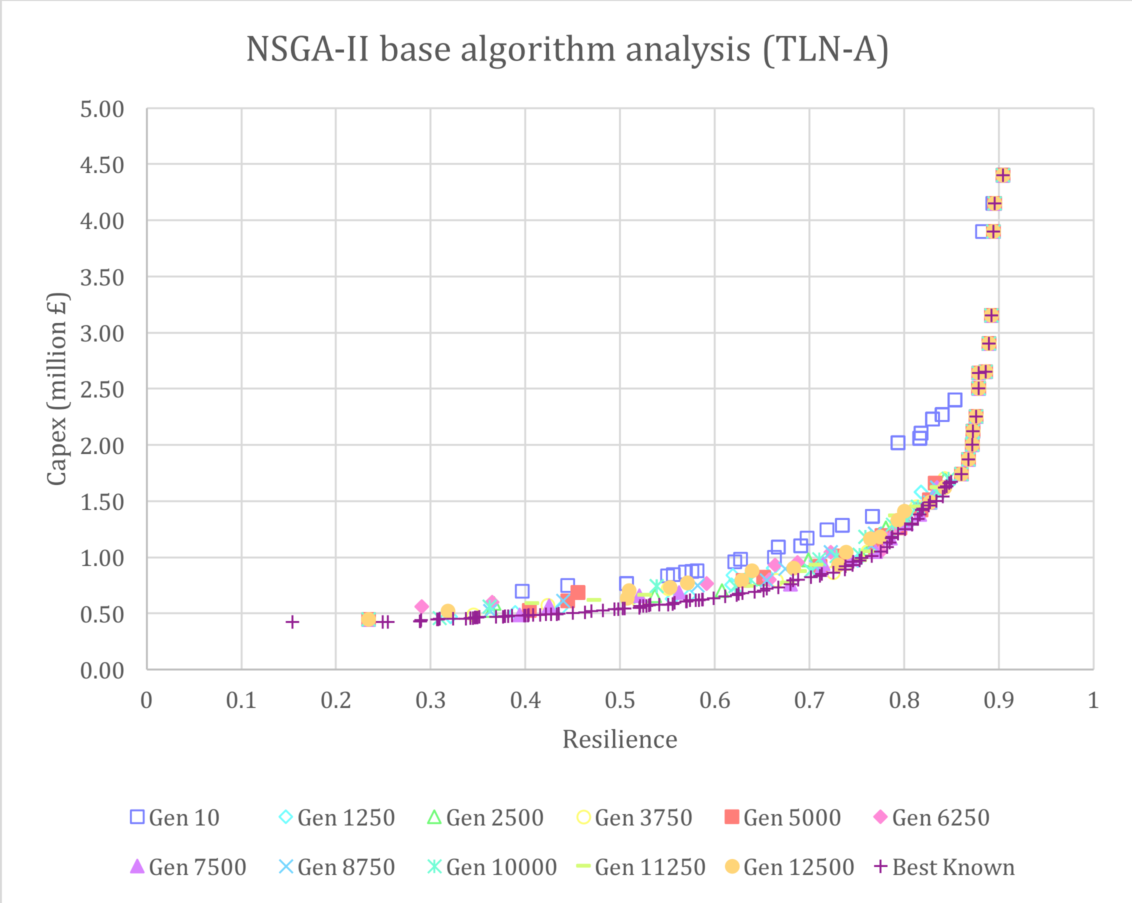


Figure 16 - NSGA-II, base algorithm analysis, TLN-A

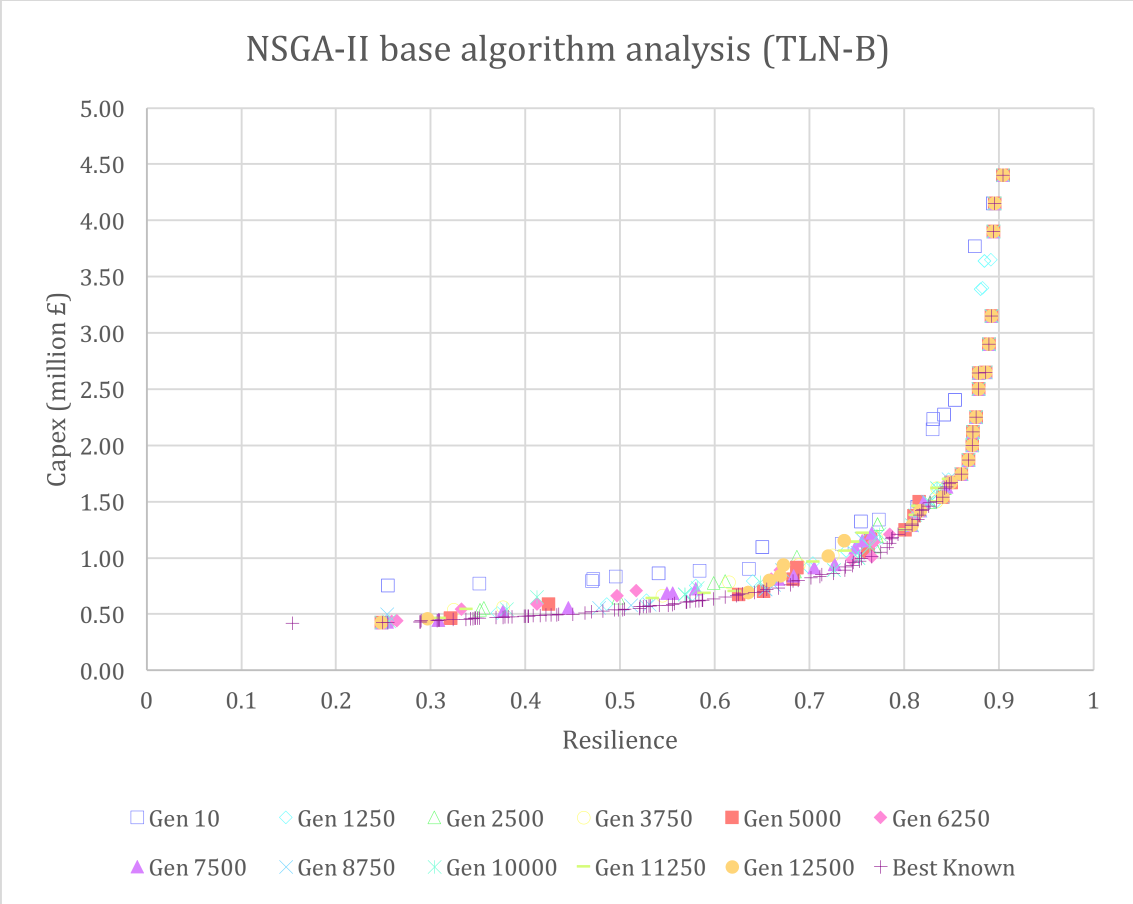


Figure 17 - NSGA-II, base algorithm analysis, TLN-B

5.7.1.2 GOY

The GOY testing problem is significantly more complex than TLN, but still on a fairly small scale compared to the two other test problems being utilised in this thesis. This can be seen (Figure 18 & Figure 19) in that the solutions are not as close to the overall best-known Pareto front by iteration ten as the TLN test problem solutions were, but by iteration one thousand the algorithm has converged reasonably well.

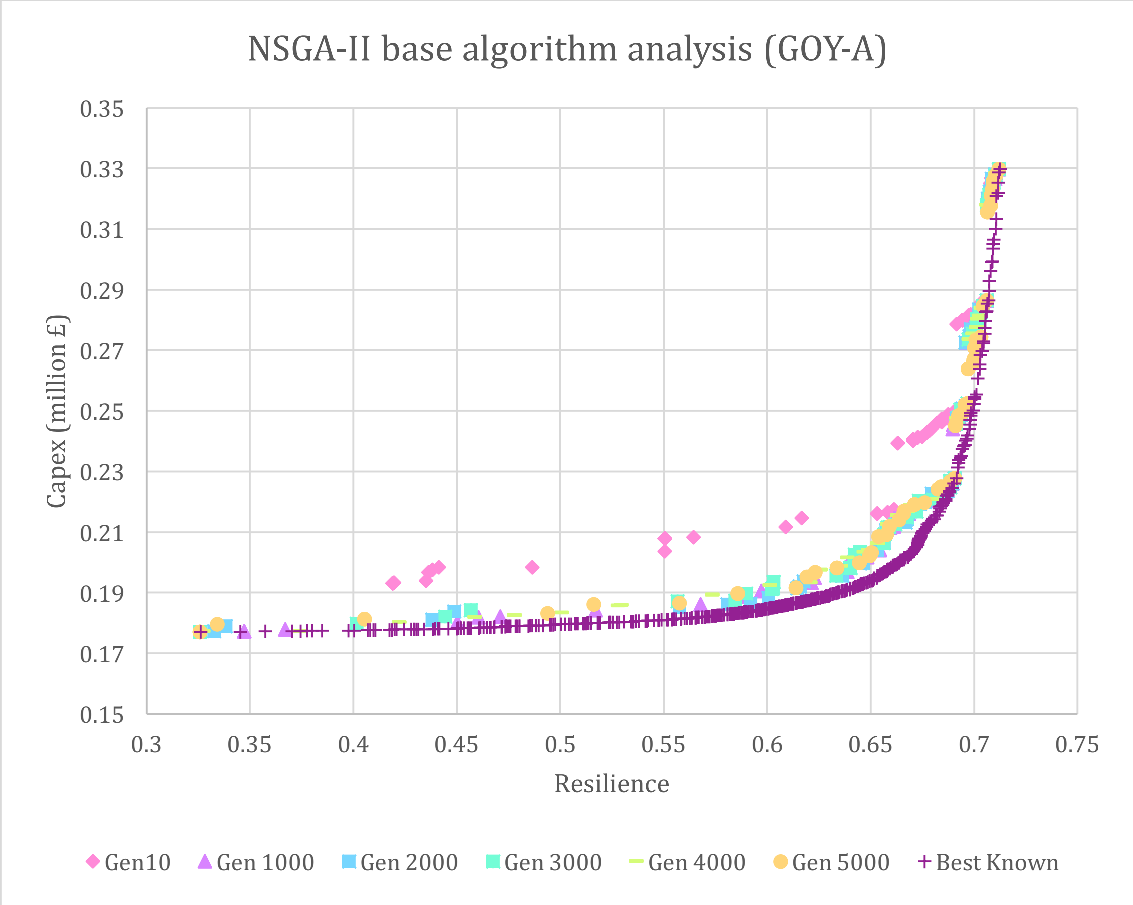


Figure 18 - NSGA-II, base algorithm analysis, GOY-A

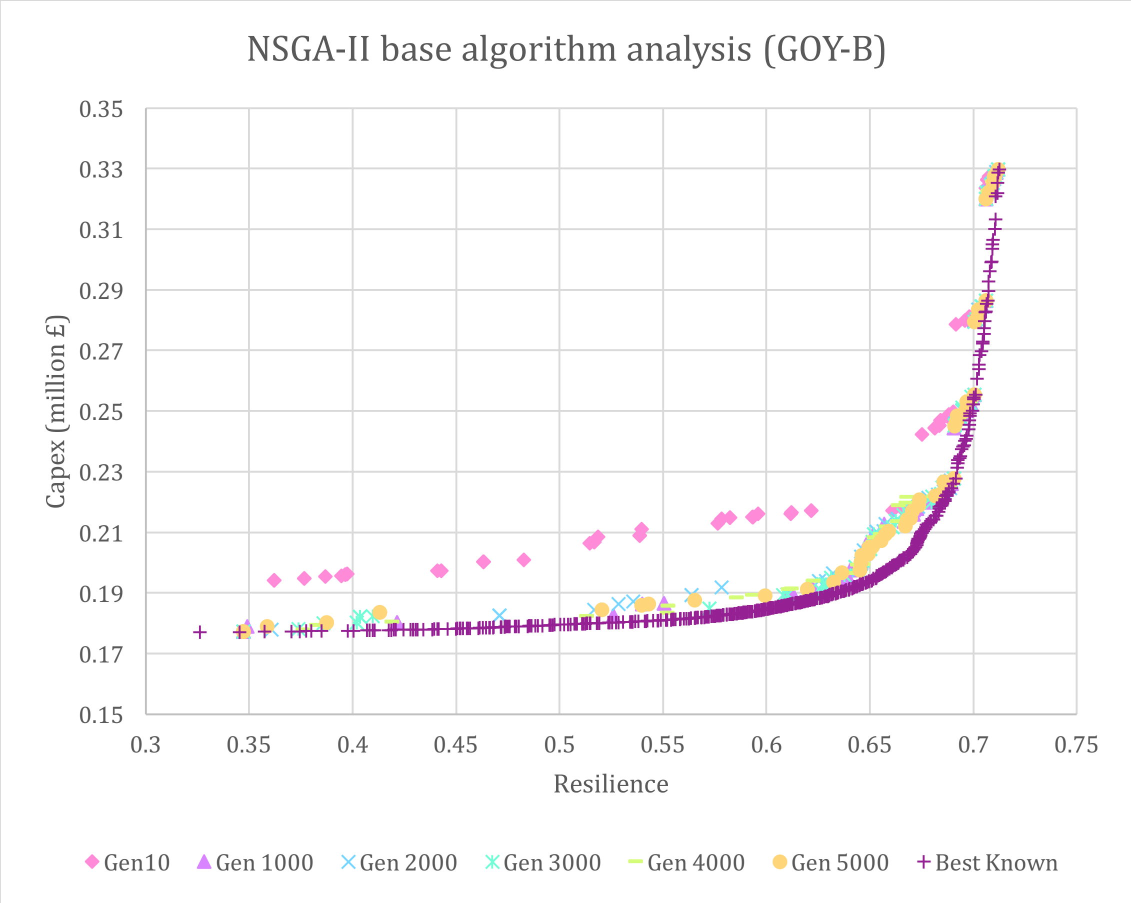


Figure 19 - NSGA-II, base algorithm analysis, GOY-B

5.7.1.3 MOD

In the following two graphs (Figure 20 & Figure 21) it can be seen that the NSGA-II algorithm converges reasonably well to the bottom half (low cost – low resilience) of the best-known front. The other (more vertical, i.e., high cost – high resilience) part of the best-known front is presumably more difficult to find. In order to make these graphs easier to read, versions with the axes altered to show the portion with the estimated fronts more clearly were also produced (Figure 22 & Figure 23). It seems that NSGA-II is steadily approaching the lower portions of the best known Pareto front, but would require more evaluations to reach it. The high cost – high resilience part of the best known Pareto front contains a smaller number of solutions with large gaps between them. This indicates that even the multi-algorithm strategy employed by Wang et al. (2014) struggled to identify solutions in that region of the Pareto front. Since only one algorithm is being used here, instead of the multi-algorithm strategy used in Wang et al. (2014), it is not expected that this algorithm necessarily finds the full Pareto front.

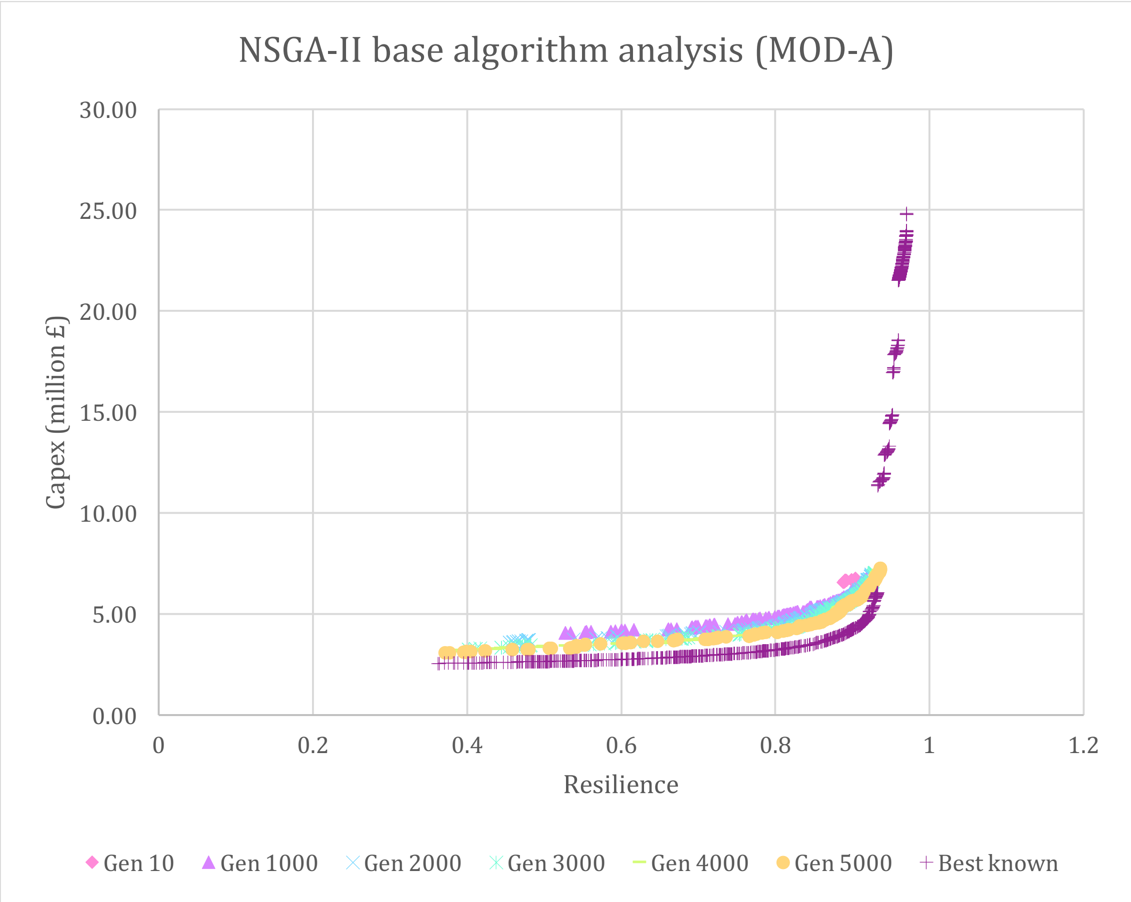


Figure 20 - NSGA-II, base algorithm analysis, MOD-A

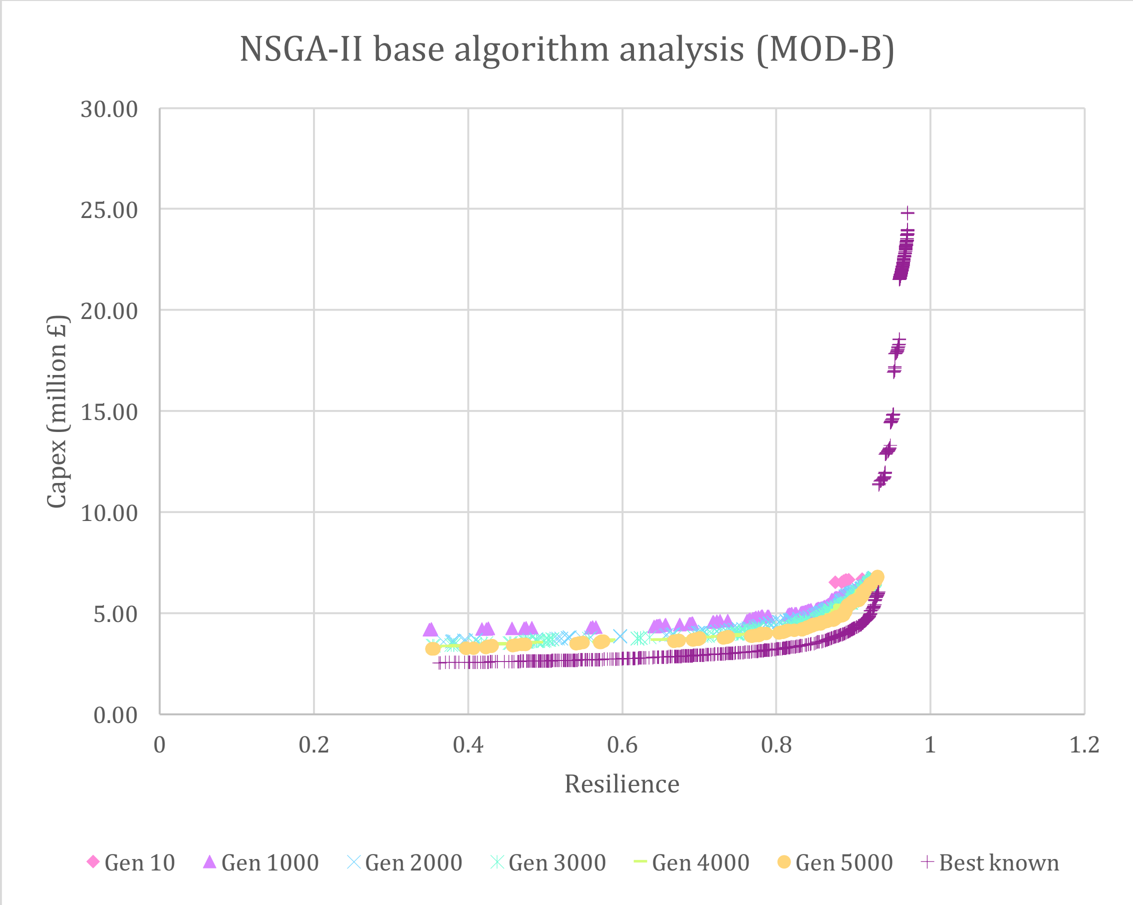


Figure 21 - NSGA-II, base algorithm analysis, MOD-B

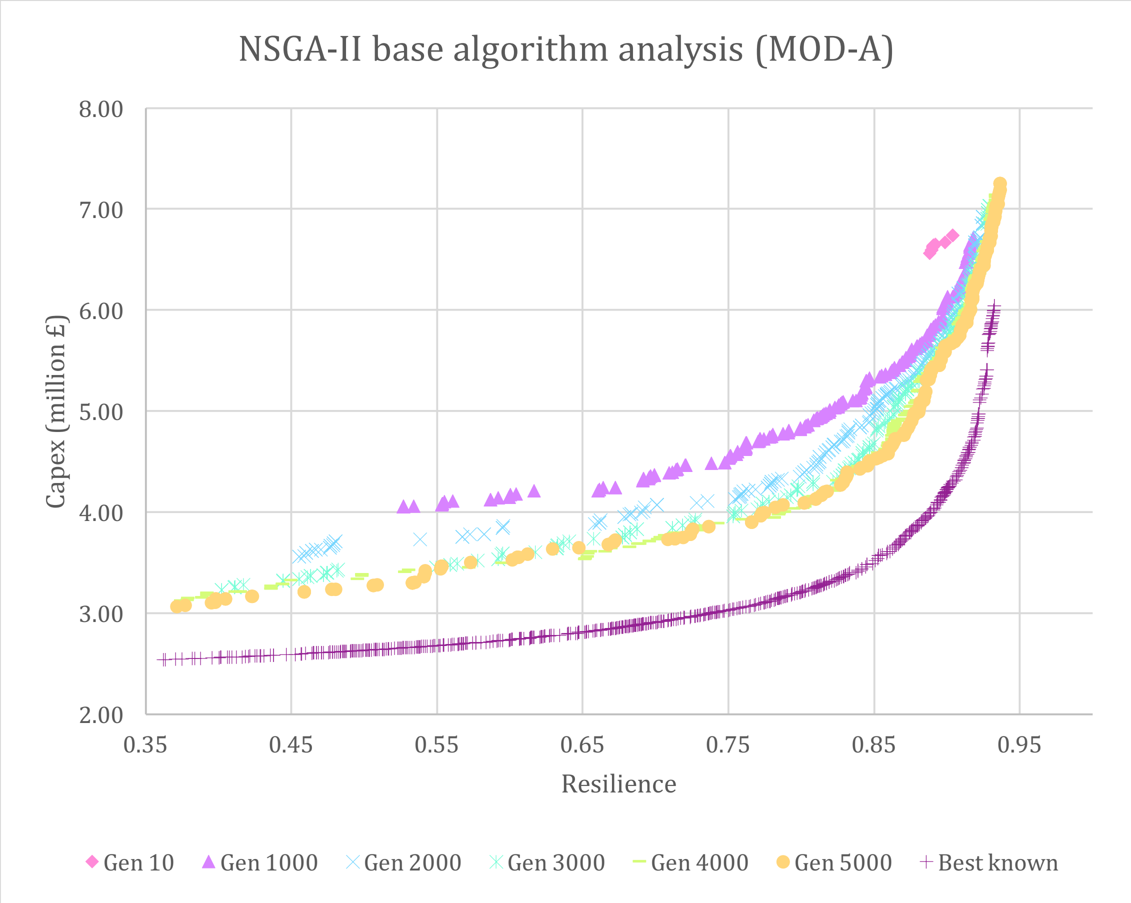


Figure 22 - NSGA-II, base algorithm analysis, MOD-A (altered axes)

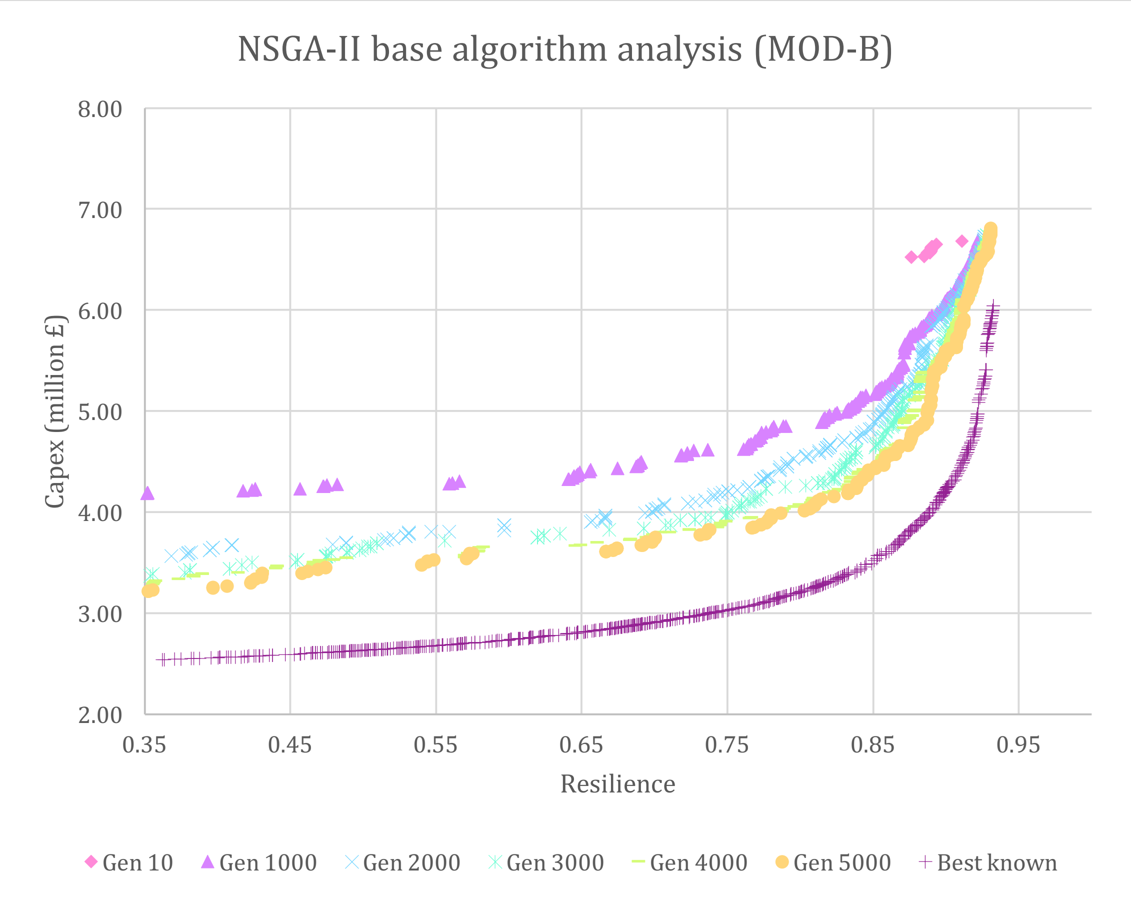


Figure 23 - NSGA-II, base algorithm analysis, MOD-B (altered axes)

5.7.1.4 BIN

It can be seen visually in Figure 24 and Figure 25 that the estimated Pareto fronts come closer to the best Pareto front as the algorithm progresses through iterations in the manner that might be expected. It is also interesting to note that a full spread of solutions (i.e., coverage) of the entire Pareto front is achieved by the algorithm in its approach to the best known. This suggests that given enough iterations the algorithm would have achieved full coverage for MOD also. In these two graphs, selected iterations are shown, iteration ten, followed by iteration five hundred, and then with an interval of five hundred iterations.

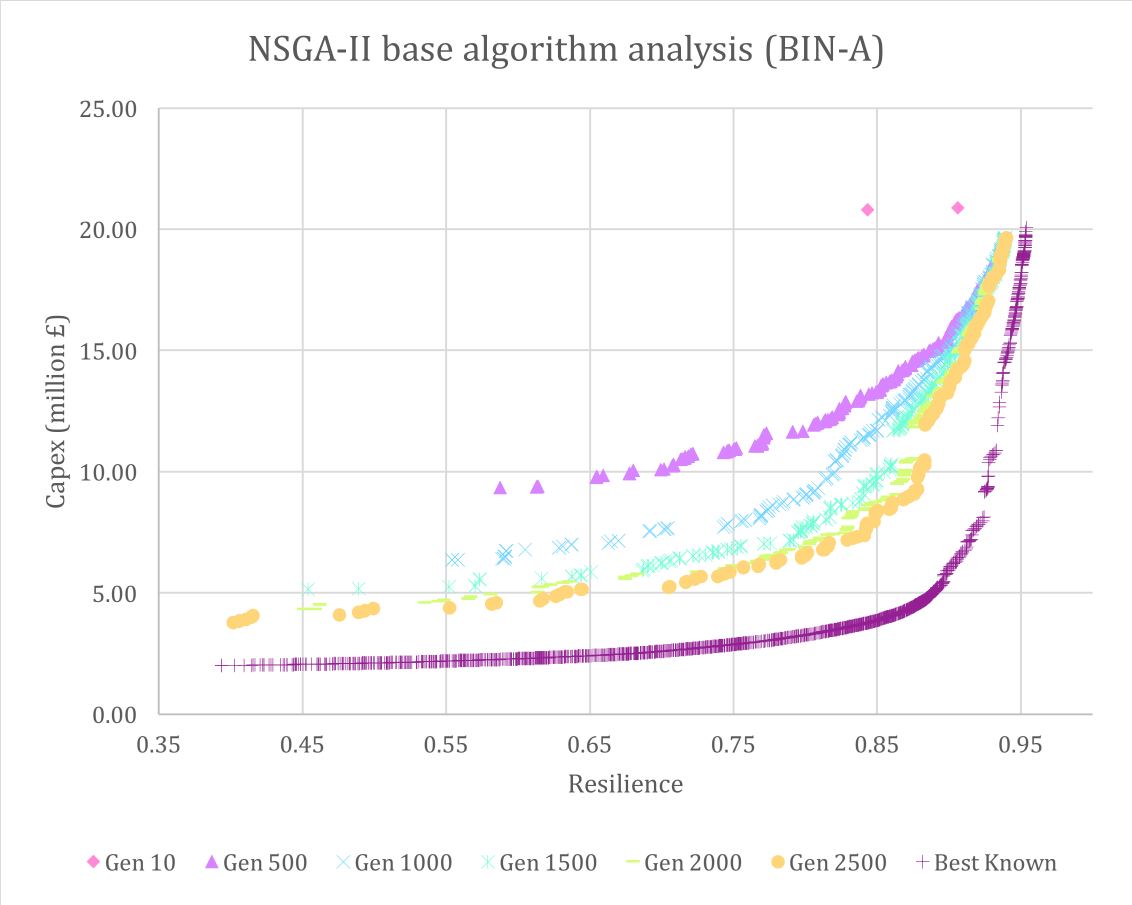


Figure 24 - NSGA-II, base algorithm analysis, BIN-A

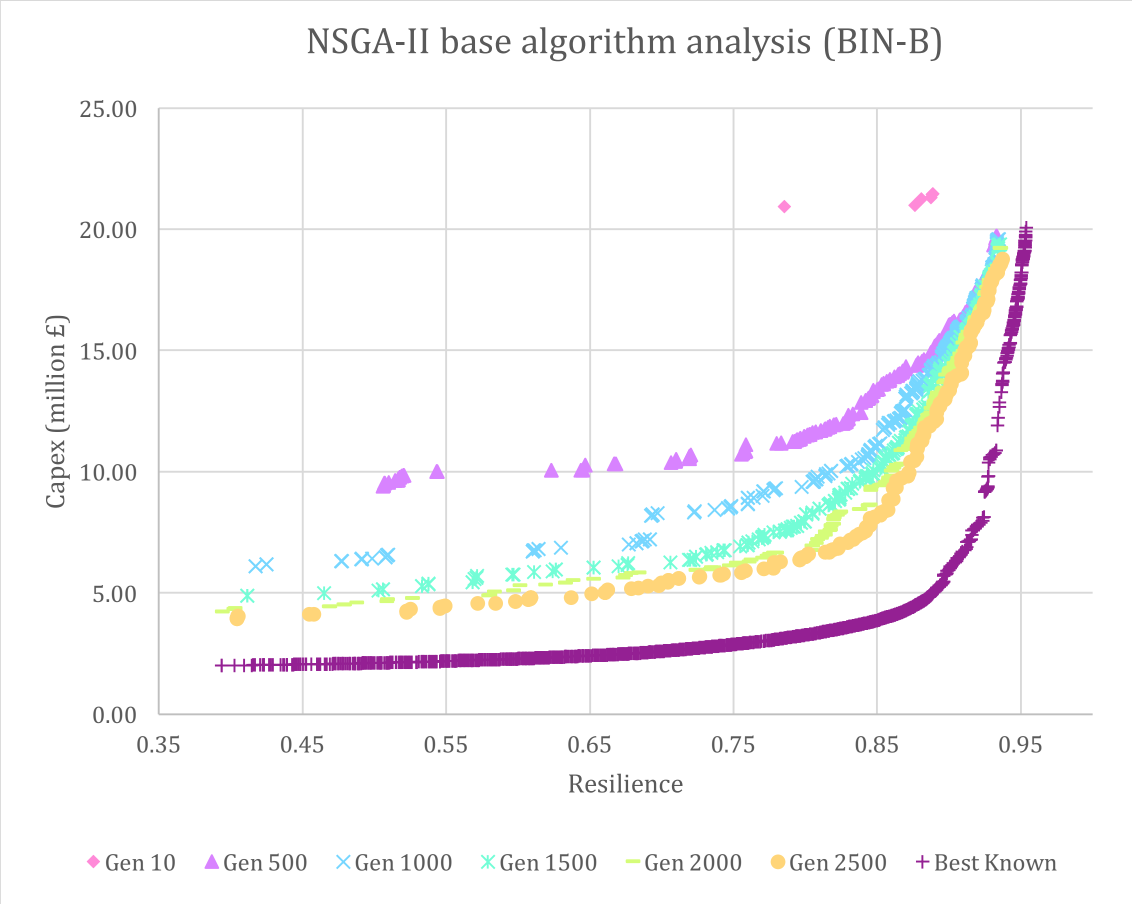


Figure 25 - NSGA-II, base algorithm analysis, BIN-B

5.7.2 NSGA-II with LEMMO and Initial ANN Structure

The test results within this section are from runs of NSGA-II plus LEMMO, with a three layer artificial neural network architecture (see section 5.6, Figure 14). Whilst there was some slight improvement over standard NSGA-II with this meta-model in place within the algorithm, the improvement was not felt to be significant. It was, for the most part, only present in any noticeable way on the simpler problems. An analysis of the algorithms execution showed that the ANN training was not completing but was instead hitting the hard limit on ANN training iterations. These two facts combined suggested that the ANN was having trouble accurately representing the more complex problems with its current structure. These tests were therefore halted, and tests started with a four-layer network. These tests had more promising results, which are presented in section 5.7.3. The results for the three-layer ANN tests are included, despite their being superseded by the four-layer ANN tests, to demonstrate the process undertaken when designing the ANN meta-model.

5.7.2.1 TLN

This initial trial with the LEMMO algorithm and the three-layer neural network shows reasonable results visually, with the estimated Pareto front closely matching the reference Pareto front. This problem is fairly trivial, however, and any properly functioning algorithm should solve it with ease.

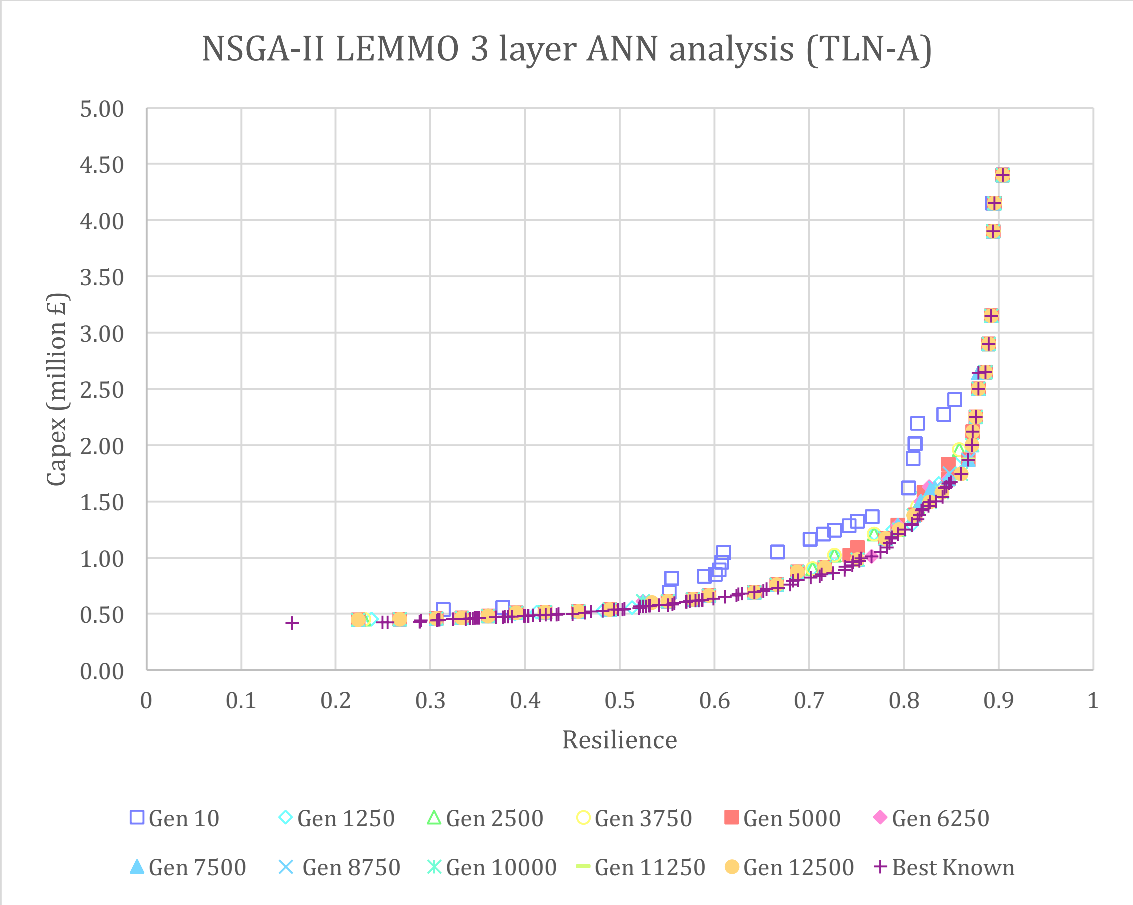


Figure 26 - NSGA-II, LEMMO with three layer ANN analysis, TLN-A

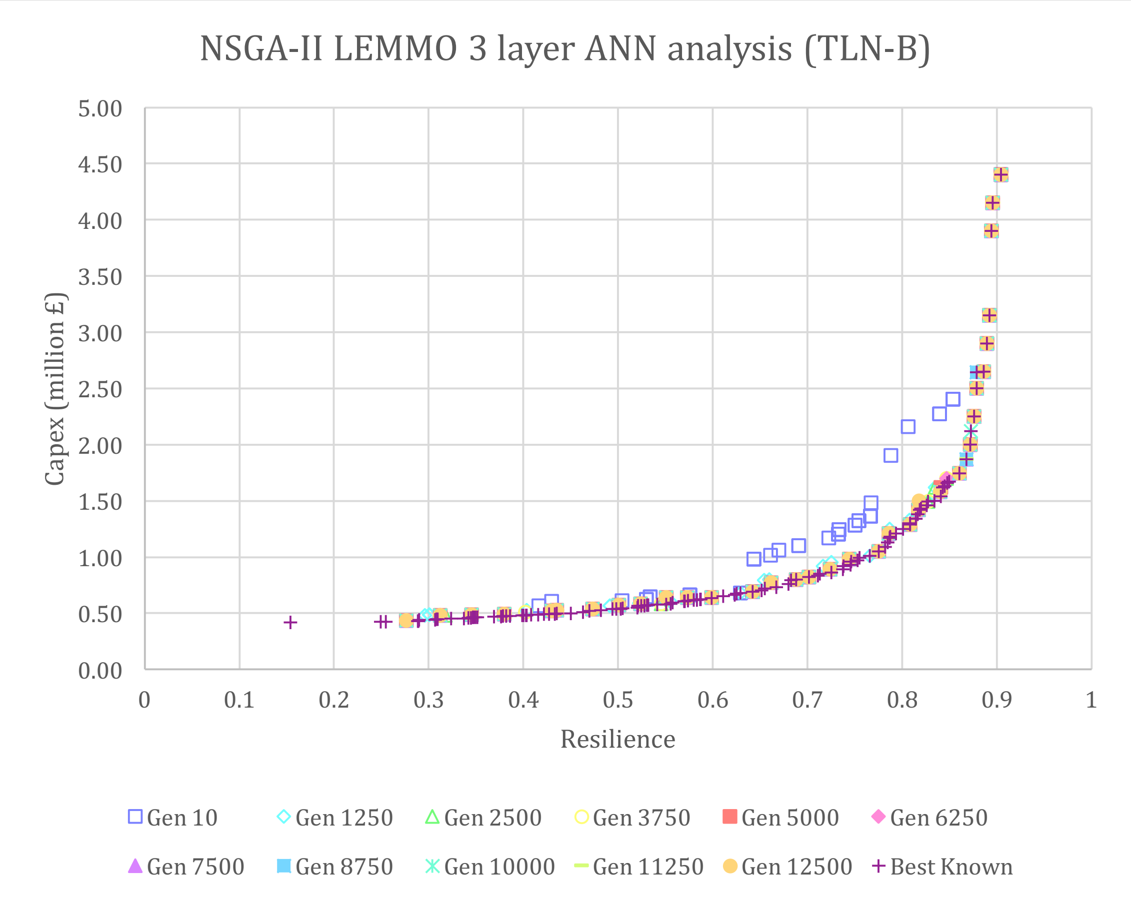


Figure 27 - NSGA-II, LEMMO with three layer ANN analysis, TLN-B

5.7.2.2 MOD

Similarly to with the standard algorithm (see section 5.7.1), the test results match well against the lower portion of the best known results curve, however, they do not match at all with the more vertical portion. As it is difficult to differentiate the lines when the whole curve is shown, versions of the graph with altered axes are also available where the individual lines are easier to discern, and the part of the best known curve against which the test results do not match, is not shown.

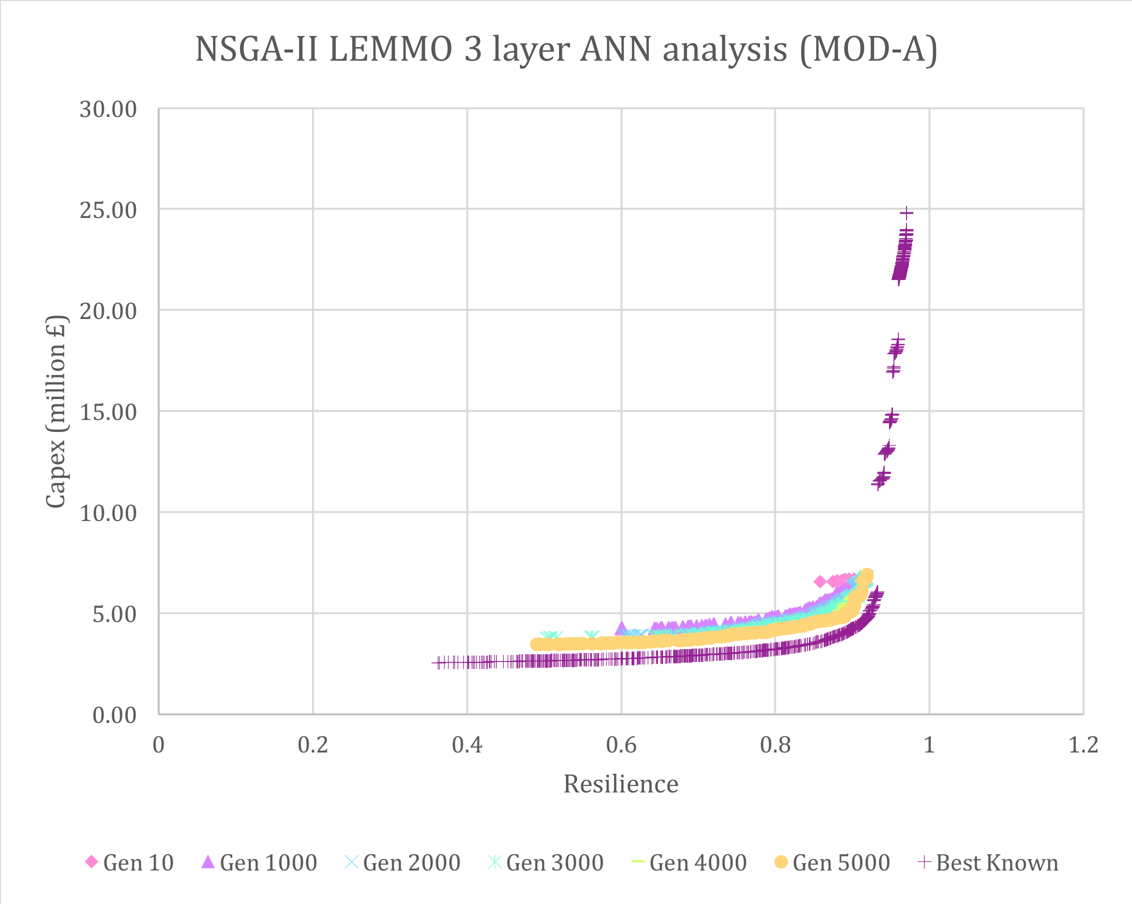


Figure 28 - NSGA-II, LEMMO with three layer ANN analysis, MOD-A

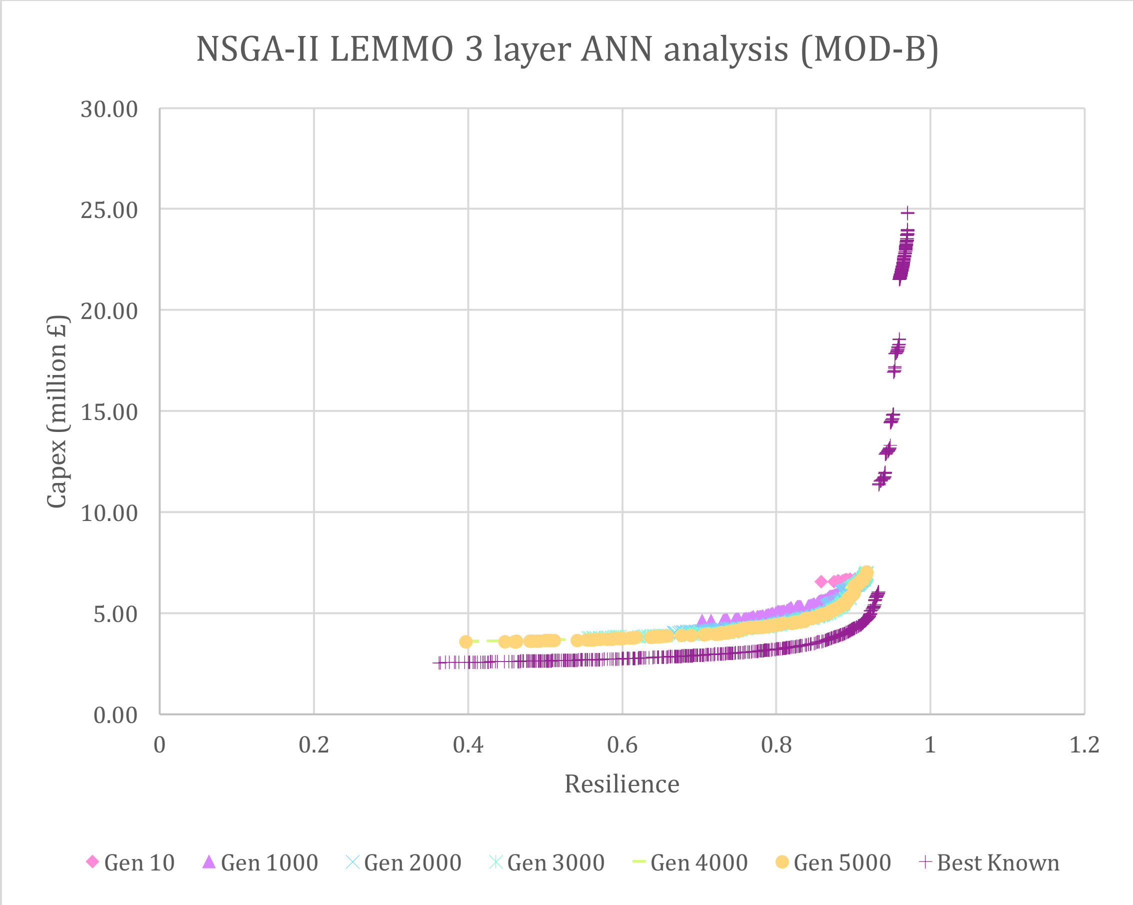


Figure 29 - NSGA-II, LEMMO with three layer ANN analysis, MOD-B

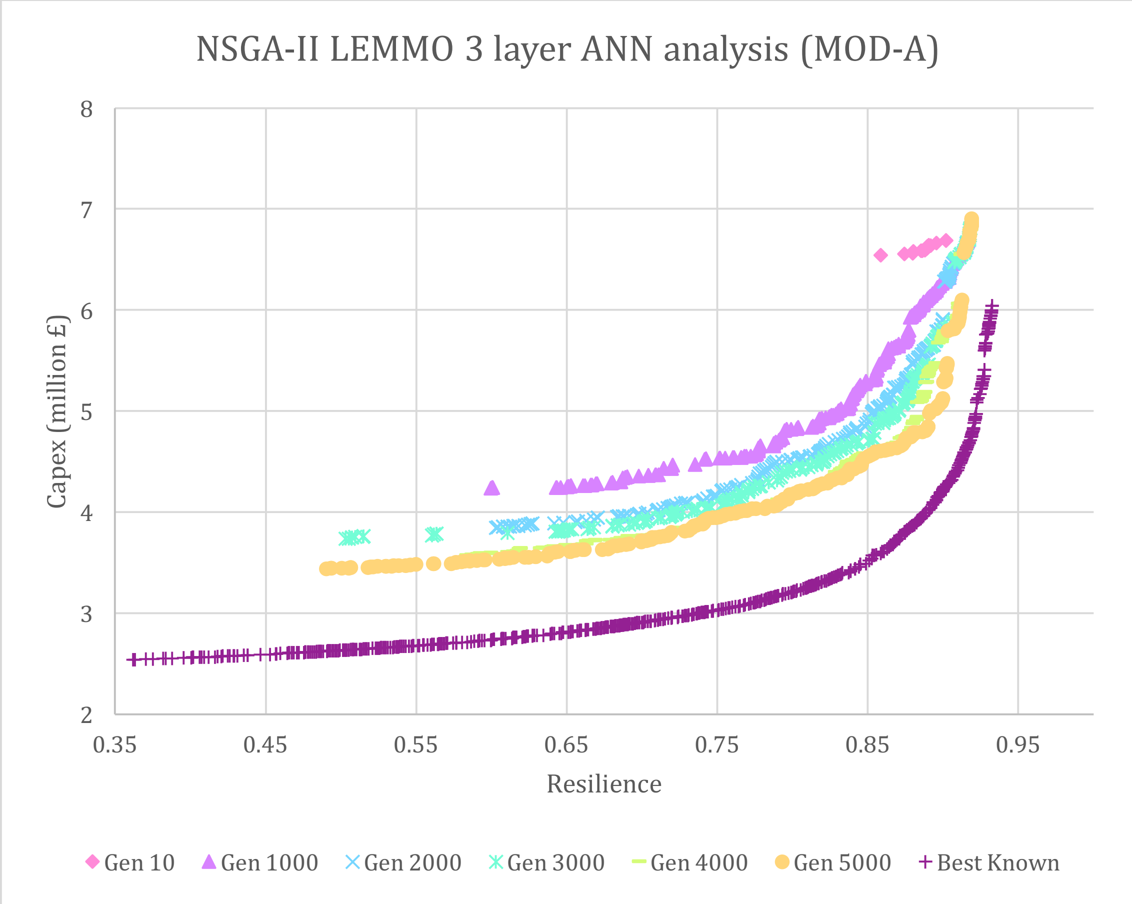


Figure 30 - NSGA-II, LEMMO with three layer ANN analysis, MOD-A (altered axes)

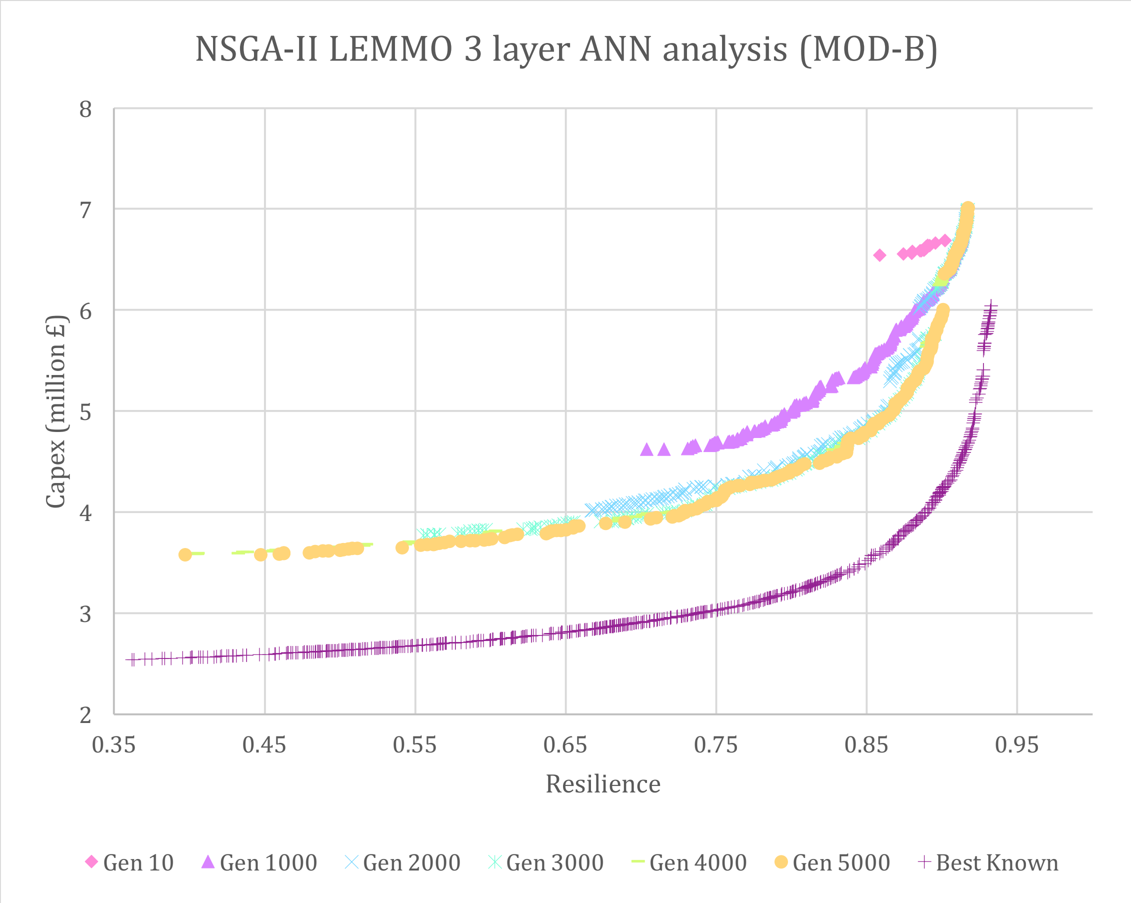


Figure 31 - NSGA-II, LEMMO with three layer ANN analysis, MOD-B (altered axes)

5.7.2.3 BIN

Visually it is hard to tell if these results are any better or worse than those presented with the base algorithm (see section 5.7.1.4). However, it is clear that no significant improvement has taken place from a visual inspection. When the logs for the algorithm run were inspected, the artificial neural network did not appear to be easily identifying which solutions are good or bad to a reasonable error rate.

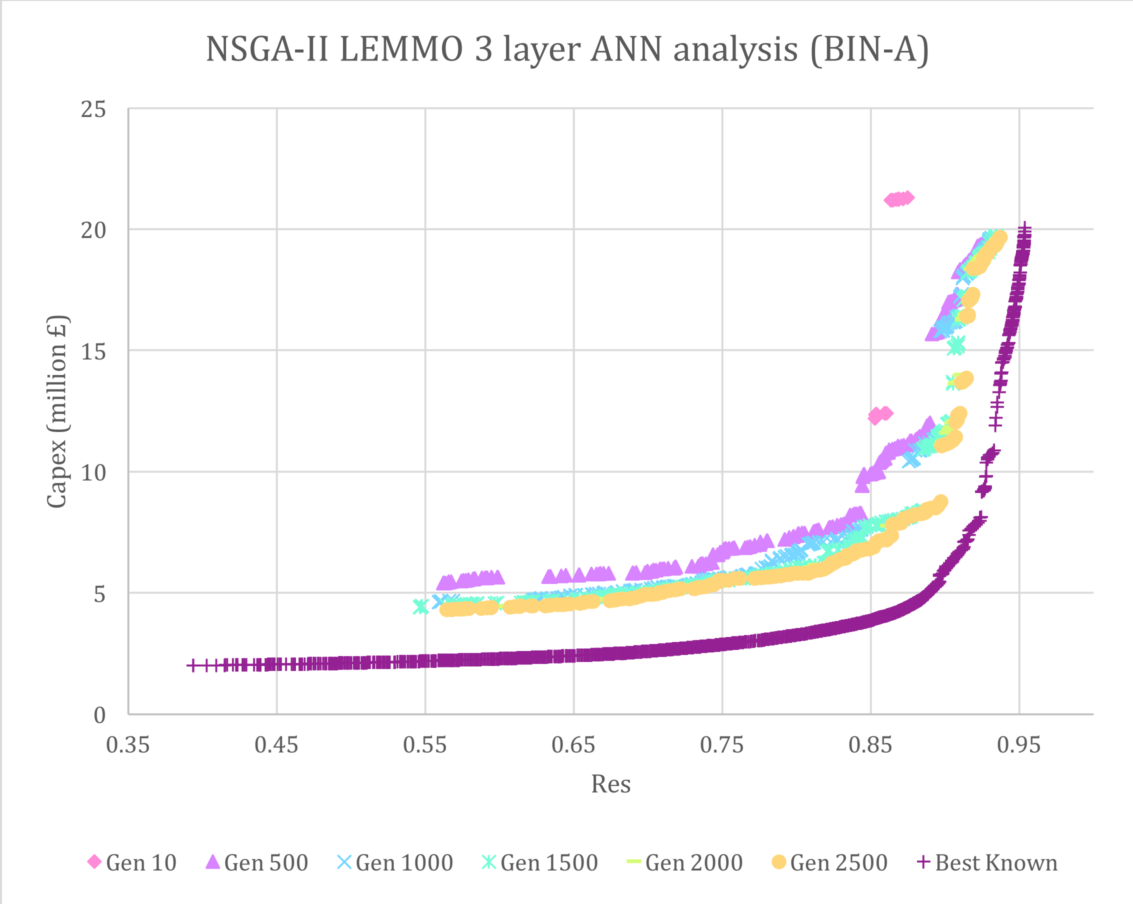


Figure 32 - NSGA-II, LEMMO with three layer ANN analysis, BIN-A

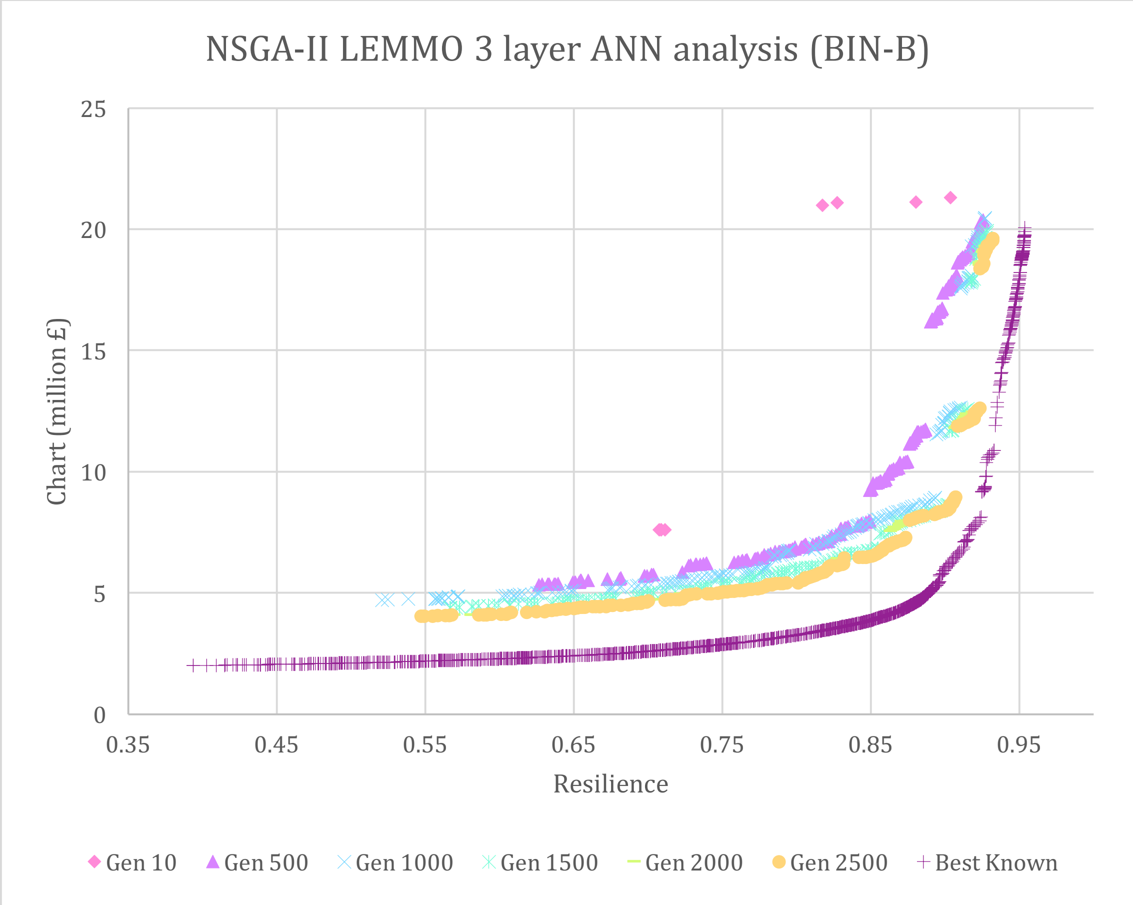


Figure 33 - NSGA-II, LEMMO with three layer ANN analysis, BIN-B

5.7.3 NSGA-II with LEMMO and Final ANN Structure

This set of test results were attained with the NSGA plus LEMMO algorithm, with a four layered artificial neural network meta-model.

5.7.3.1 TLN

As is expected with a fairly trivial test, the algorithm quickly converges to a good approximation of the optimal Pareto front.

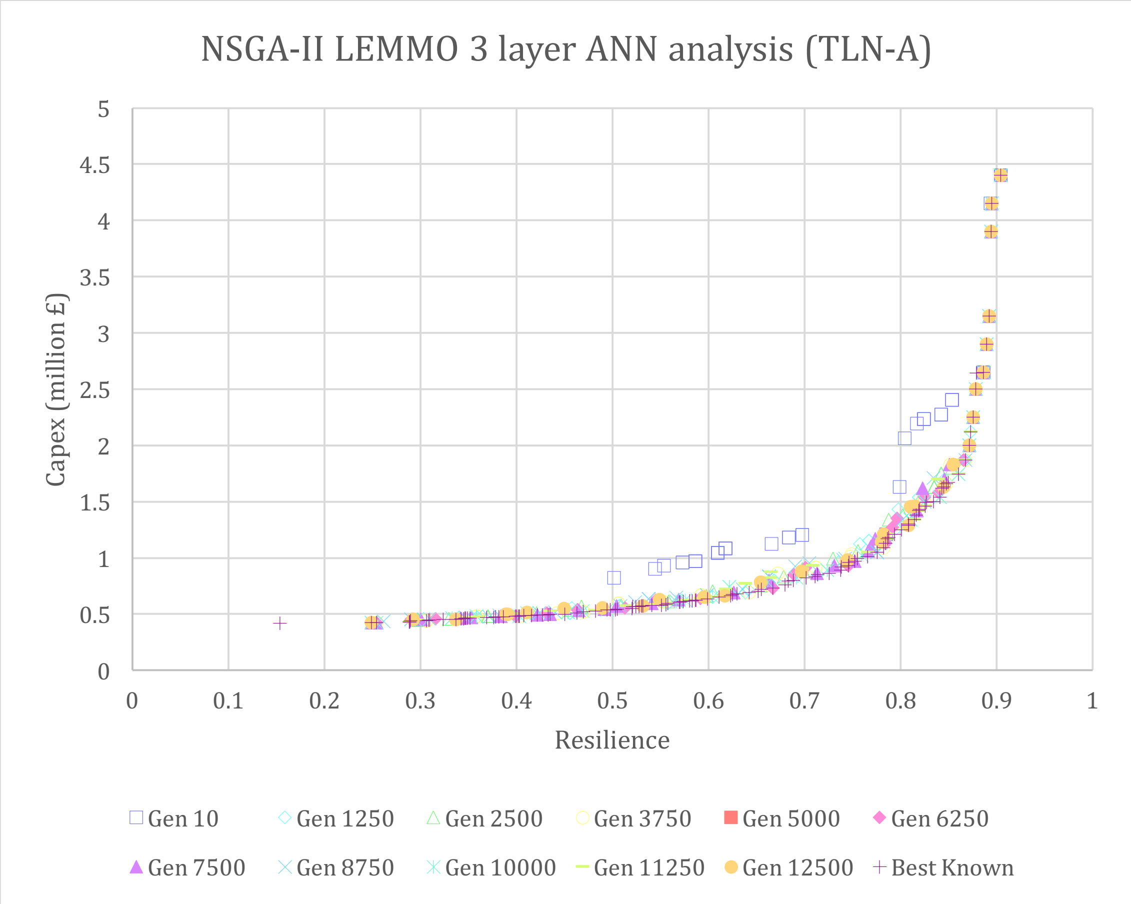


Figure 34 - NSGA-II, LEMMO with four layer ANN analysis, TLN-A

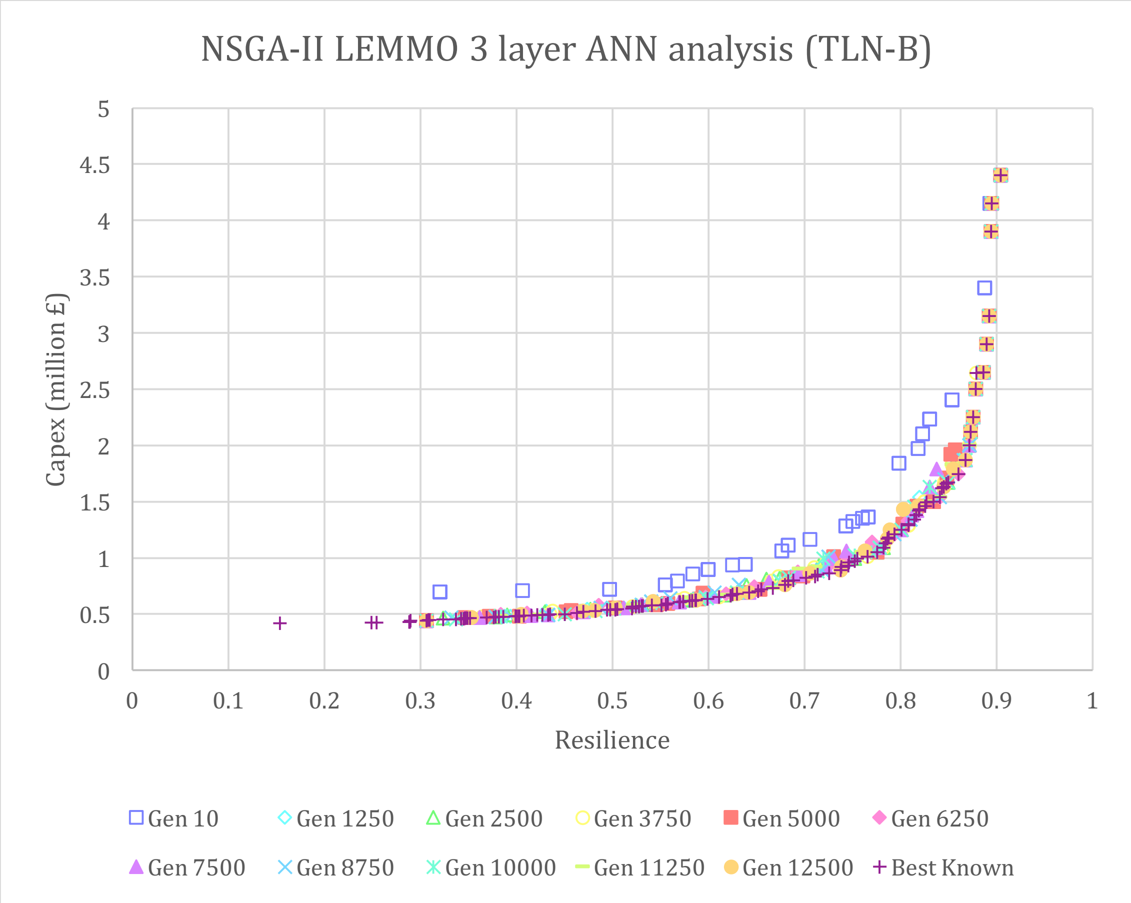


Figure 35 - NSGA-II, LEMMO with four layer ANN analysis, TLN-B

5.7.3.2 GOY

The GOY test is significantly more complex, but still manageable. The algorithm converges well by iteration one thousand.

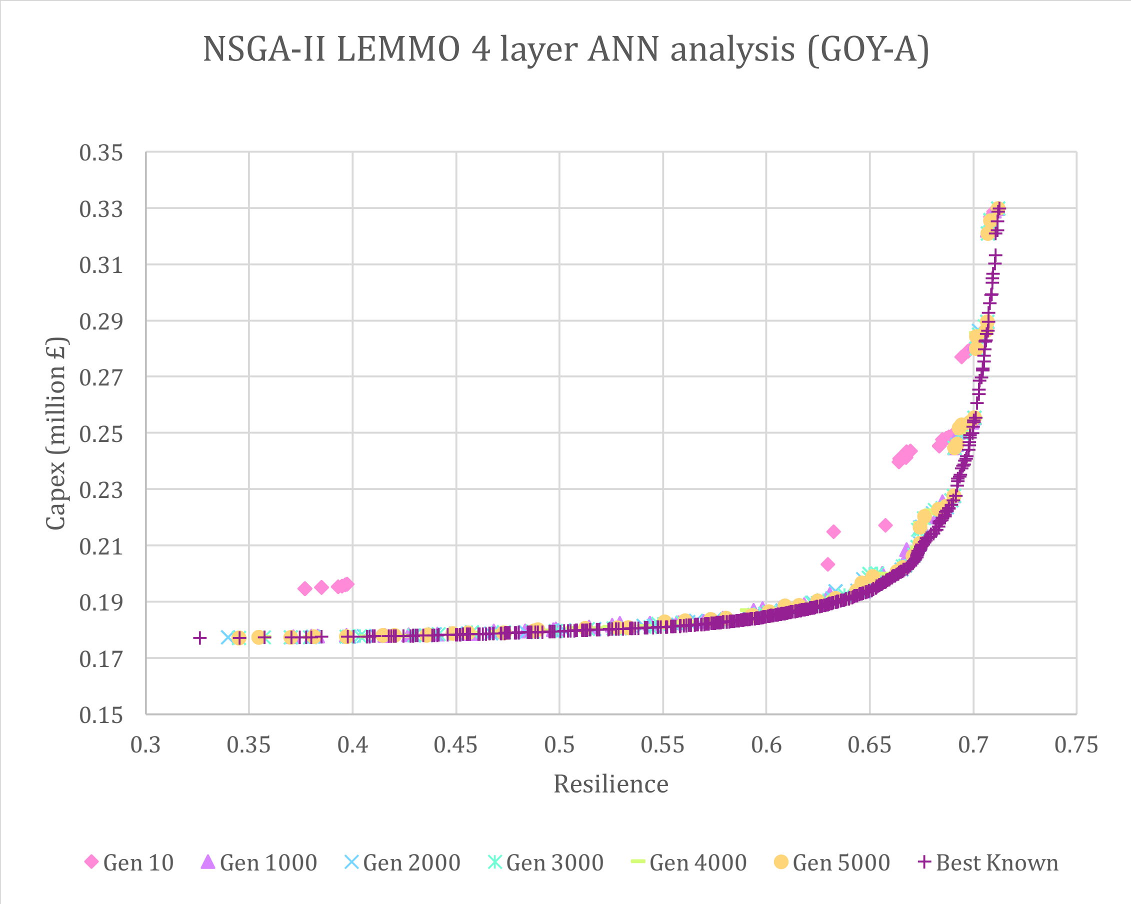


Figure 36 - NSGA-II, LEMMO with four layer ANN analysis, GOY-A

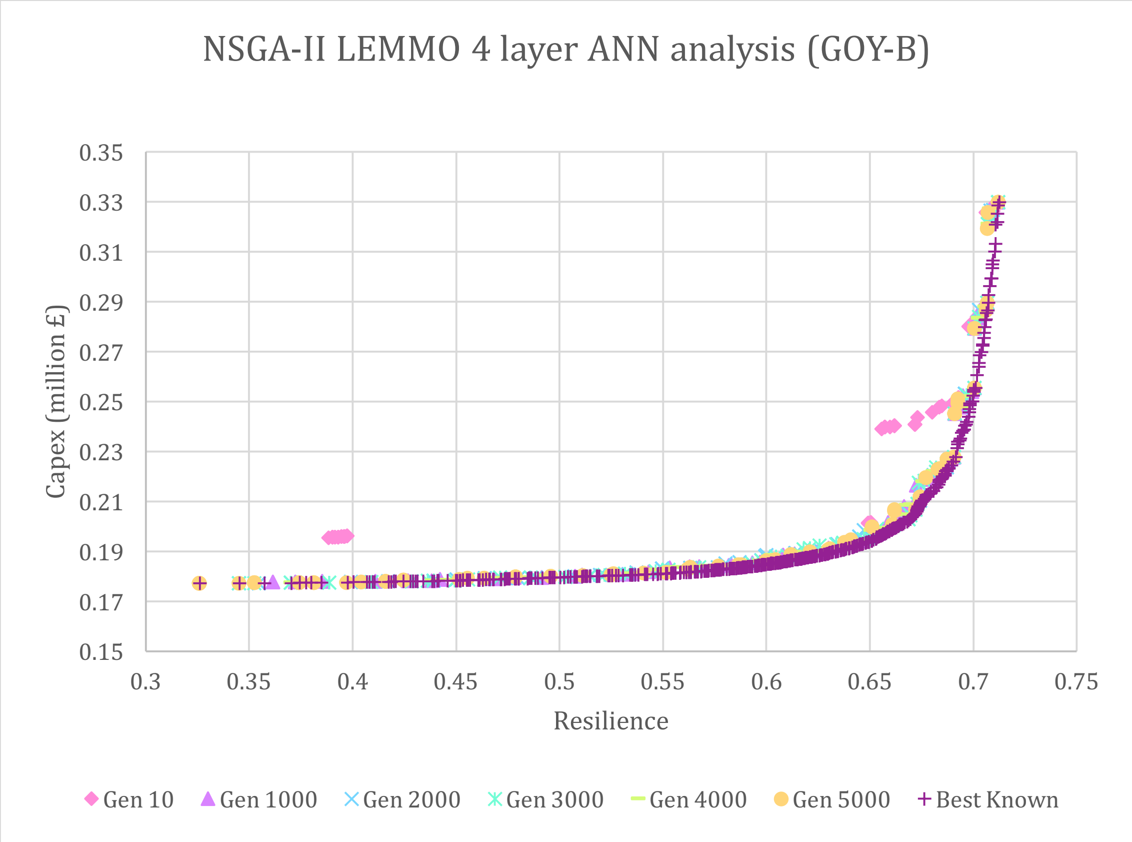


Figure 37 - NSGA-II, LEMMO with four layer ANN analysis, GOY-B

5.7.3.3 MOD

Although difficult to discern visually, the convergence appears to be better for MOD with the four layer ANN analysis than in previous tests on the same problem. Again, as with previous MOD tests the vertical solutions on the best Pareto front seem to be difficult for the algorithm to converge to, so two graphs with altered axes allow comparison of the horizontal portion of the best known front.

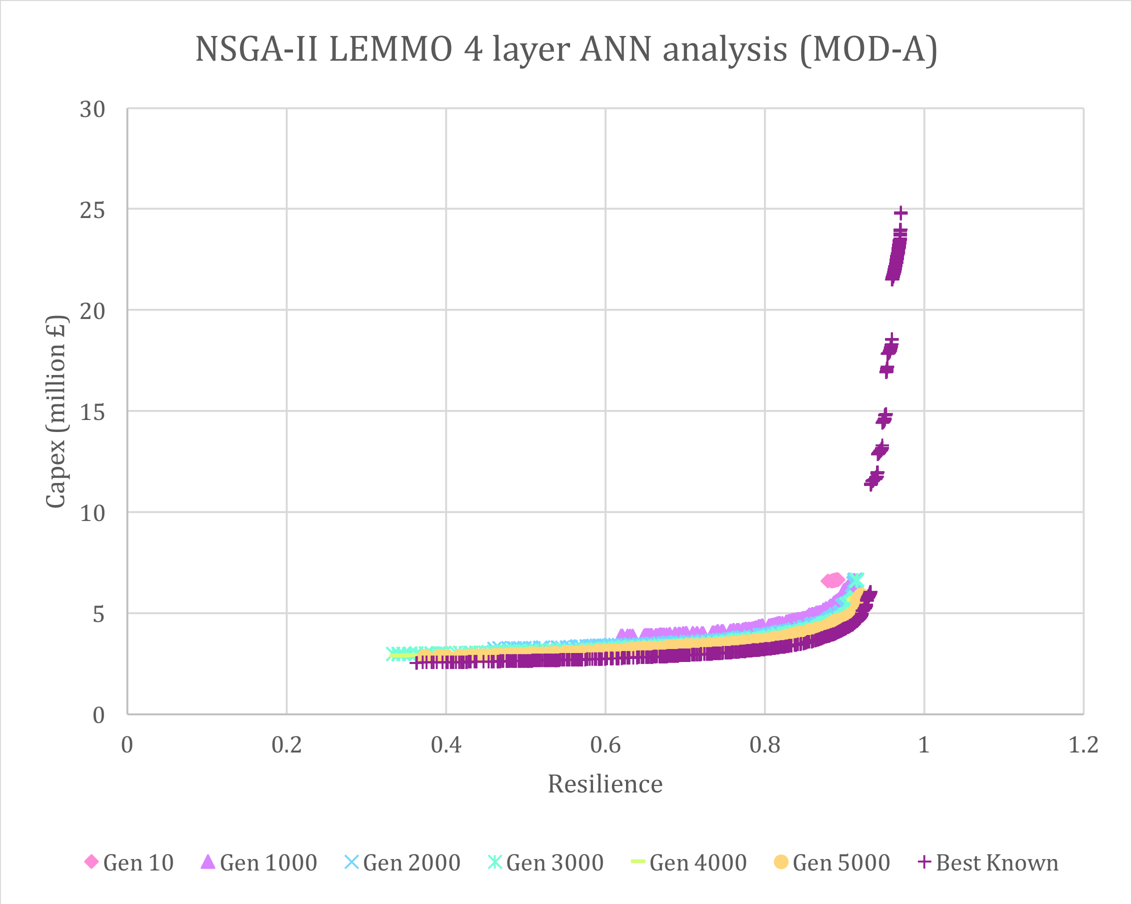


Figure 38 - NSGA-II, LEMMO with four layer ANN analysis, MOD- A

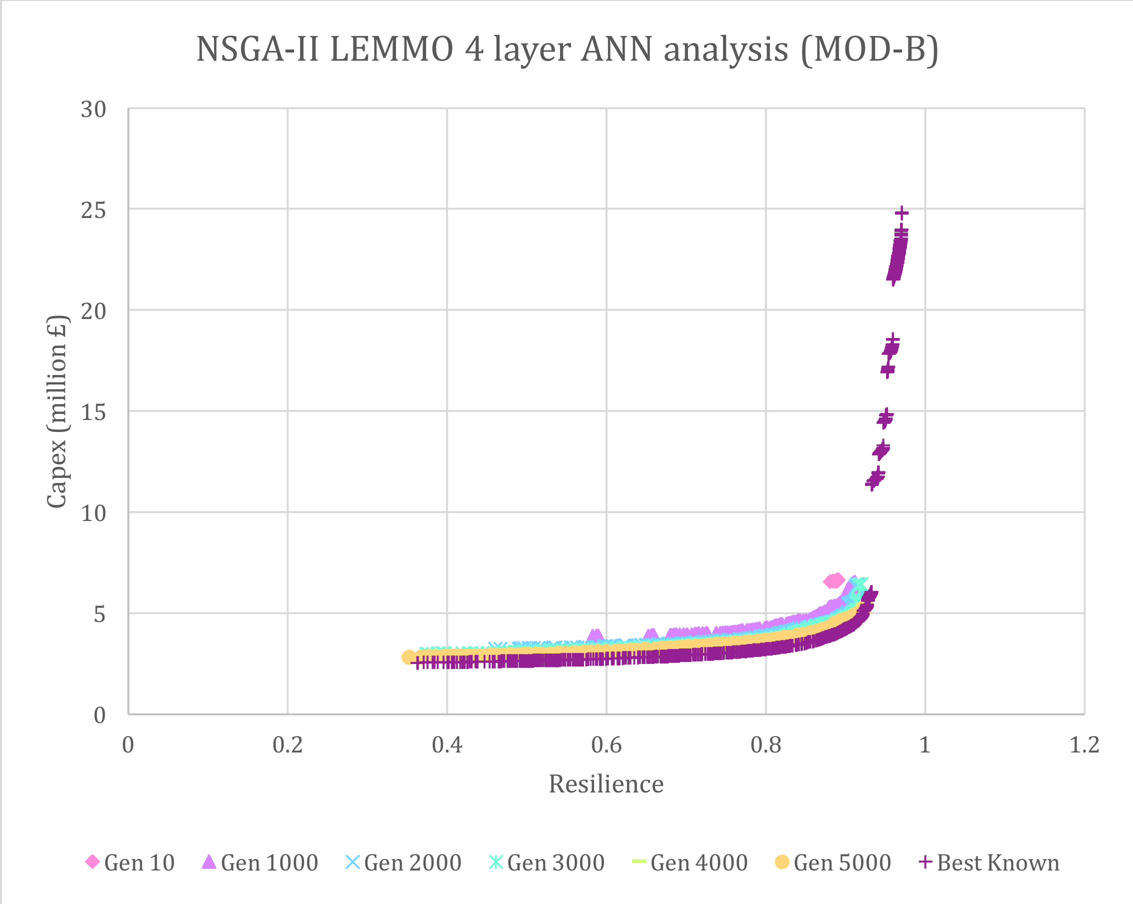


Figure 39 - NSGA-II, LEMMO with four layer ANN analysis, MOD-B

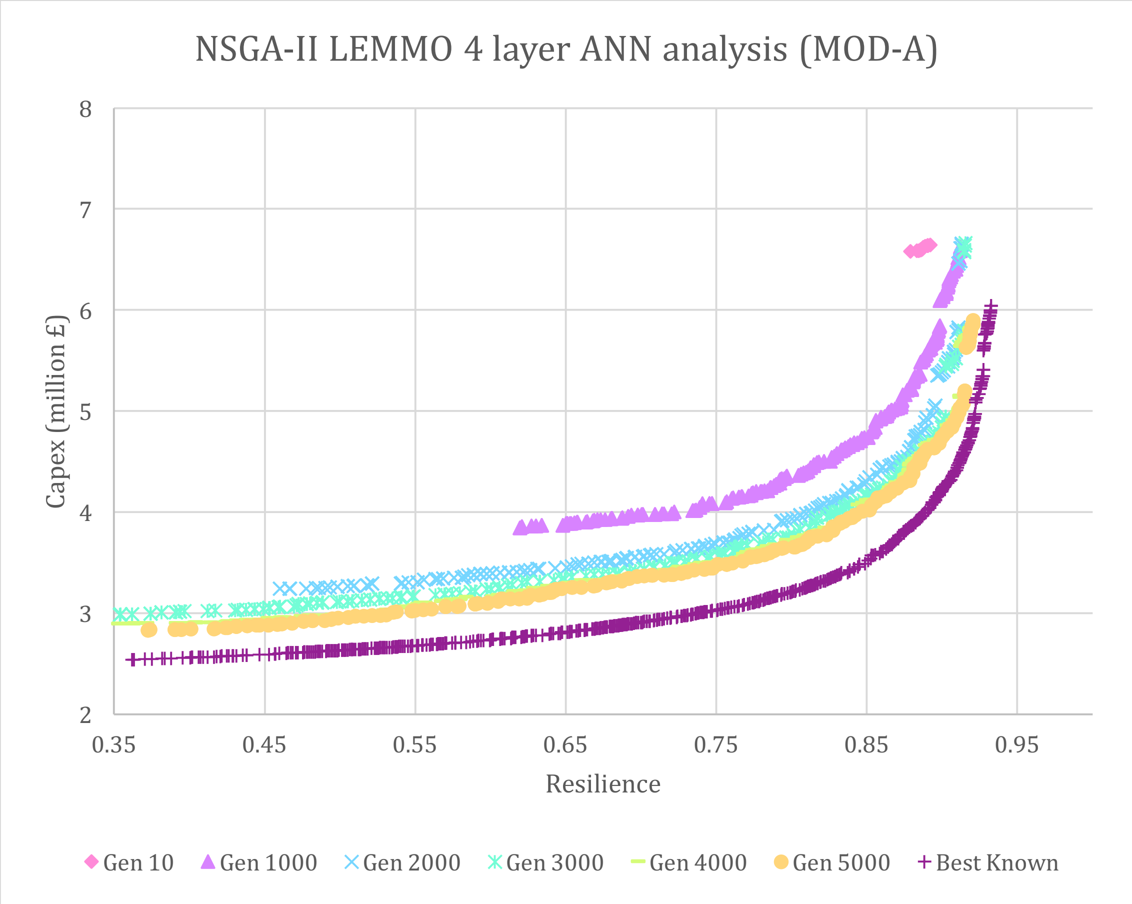


Figure 40 - NSGA-II, LEMMO with four layer ANN analysis, MOD- A (altered axes)

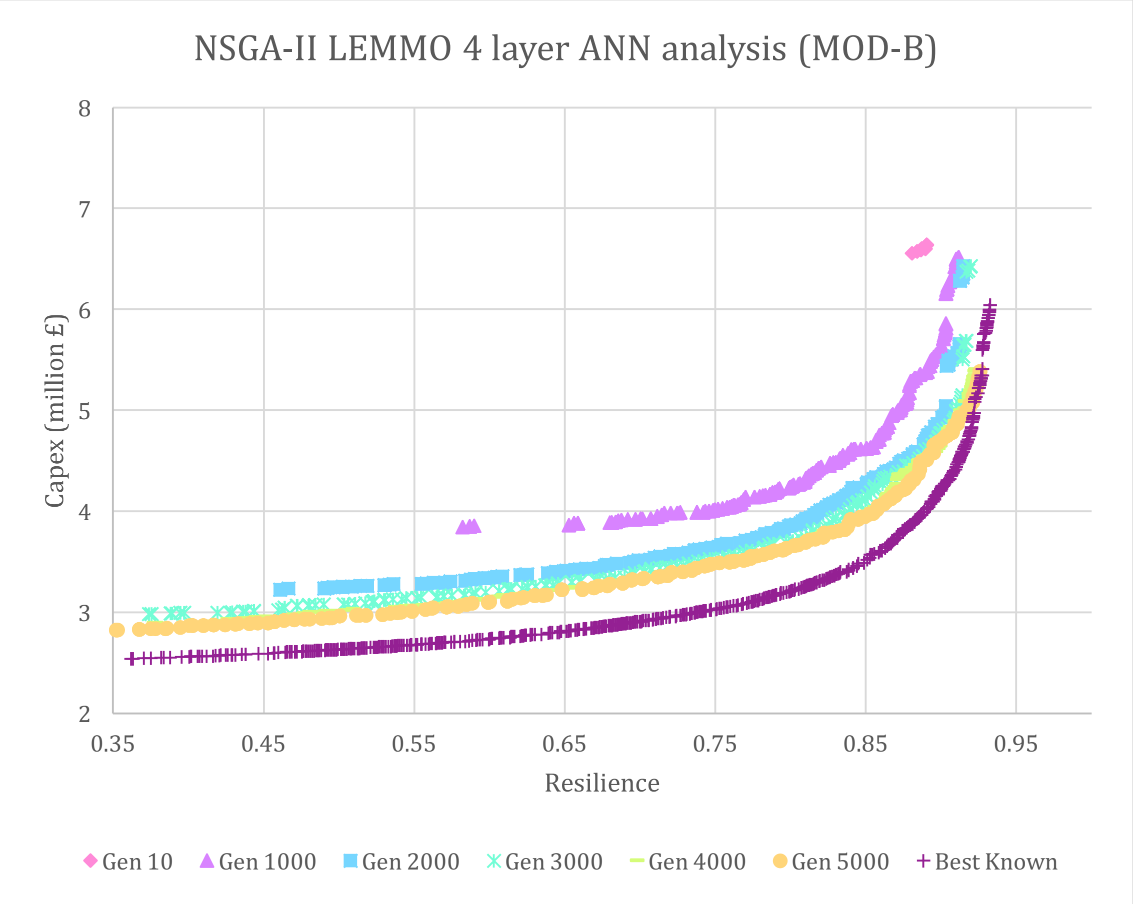


Figure 41 - NSGA-II, LEMMO with four layer ANN analysis, MOD-B (altered axes)

5.7.3.4 BIN

It is clear from a visual inspection of two selected problems that with a four-layer artificial neural network the BIN WDS problem is converging to a far better estimation of the best known Pareto front.

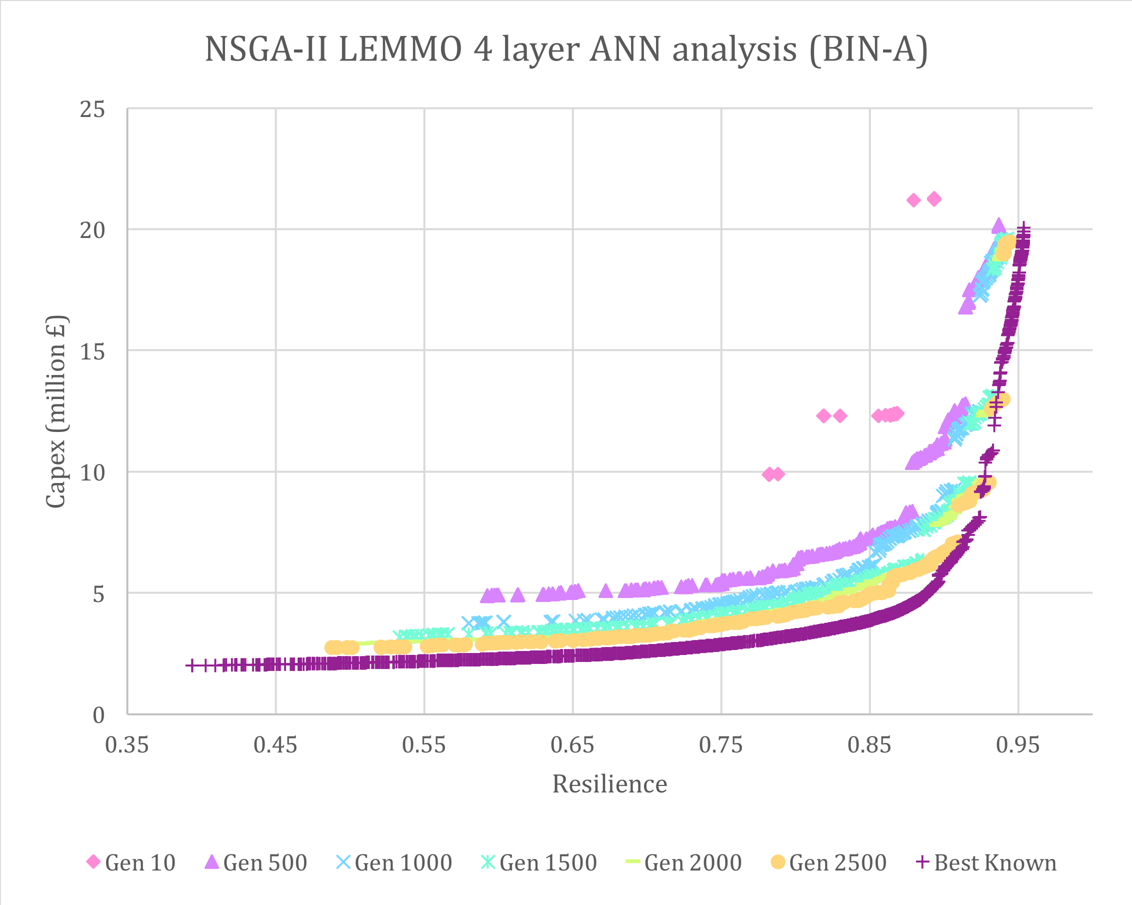


Figure 42 - NSGA-II, LEMMO with four layer ANN analysis, BIN-A

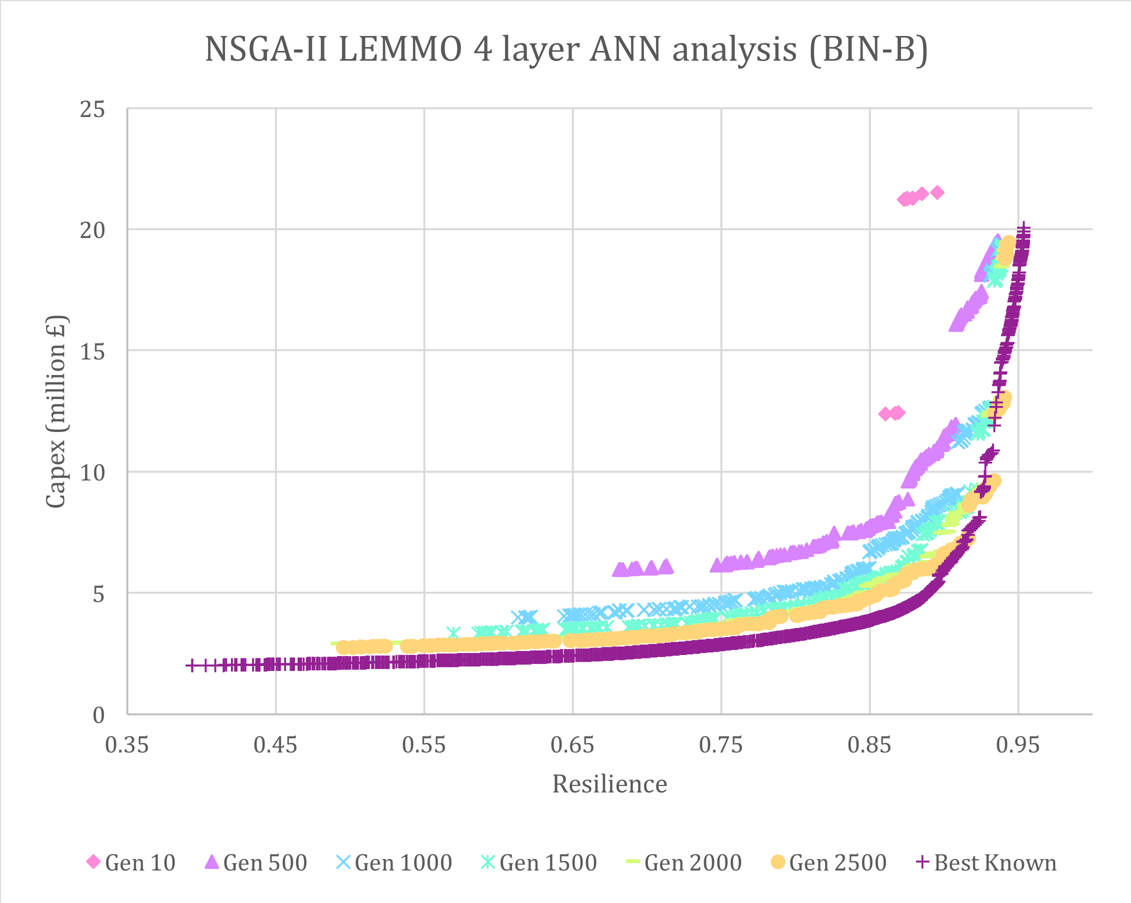


Figure 43 - NSGA-II, LEMMO with four layer ANN analysis, BIN-B

5.7.4 Analysis of Results

Three metrics have been utilised in the analysis of these results, these are convergence, diversity, and dominated hypervolume (see section 4.5). For each test WDS problem (bar the three-layer ANN tests which were cut short due to time constraints and their poor results) there are twenty runs of the tested algorithm, for which the results for every tenth iteration of that run have been recorded.

As well as the results that have thus far been presented (in the sample results above), all other results from all twenty runs are also included within this analysis.

This is achieved by calculating all three metrics for each tenth iteration of every single test. These metrics for each iteration are then averaged across common tests. For example, the metric results for TLN with NSGA-II and no LEMMO consist of a set of averages of the metrics produced for each iteration. Further details of the results presented and the calculated metrics results can be seen in Appendix II – BIN Data Tables.

5.7.4.1 TLN Analysis

The analysis for TLN (see Figure 44, Figure 45 and Figure 46) shows that in terms of convergence towards the best known Pareto front, both LEMMO tests show improved results over the NSGA-II base algorithm. The three-layer ANN version of LEMMO very slightly out-performs the four-layer ANN version in terms of this metric – but the difference is minor, and it is worth noting that the four-layer version converges faster.

In terms of diversity, the TLN analysis shows some interesting results. Both the four-layer ANN version of LEMMO and the NSGA-II base algorithm start off at a diversity of approximately point five. This diversity then decreases slightly before remaining fairly static. In contrast, the three-layer version of LEMMO starts at around a diversity of point five, before increasing, and becoming static at approximately point six five. Further investigation of the results files for TLN shows that TLN with three-layer ANN LEMMO consistently has a lower number of solutions in rank one at its final iteration than either of the other two variants. This would explain the higher diversity (as fewer points to cover the same curve will be further apart).

A hypothesis to explain this occurrence, could be that the meta-model is helping the genetic algorithm initially, resulting in the generally better results as the algorithm progresses faster. Three-layer ANN is then struggling to accurately represent the TLN problem. The NSGA-II algorithm continues to optimise from its (improved relative to running without the meta-models) position. This could result in a better overall convergence to the best-known Pareto front, but only elitism preserved solutions generated by NSGA-II being part of that convergence. This would happen to an increasing extent as the iterations progress.

In terms of dominated hypervolume, both of the LEMMO algorithm variants out-perform the NSGA-II base algorithm. The four-layer version of LEMMO appears to converge to its solution slightly faster – this is supported by the data given by the convergence metric.

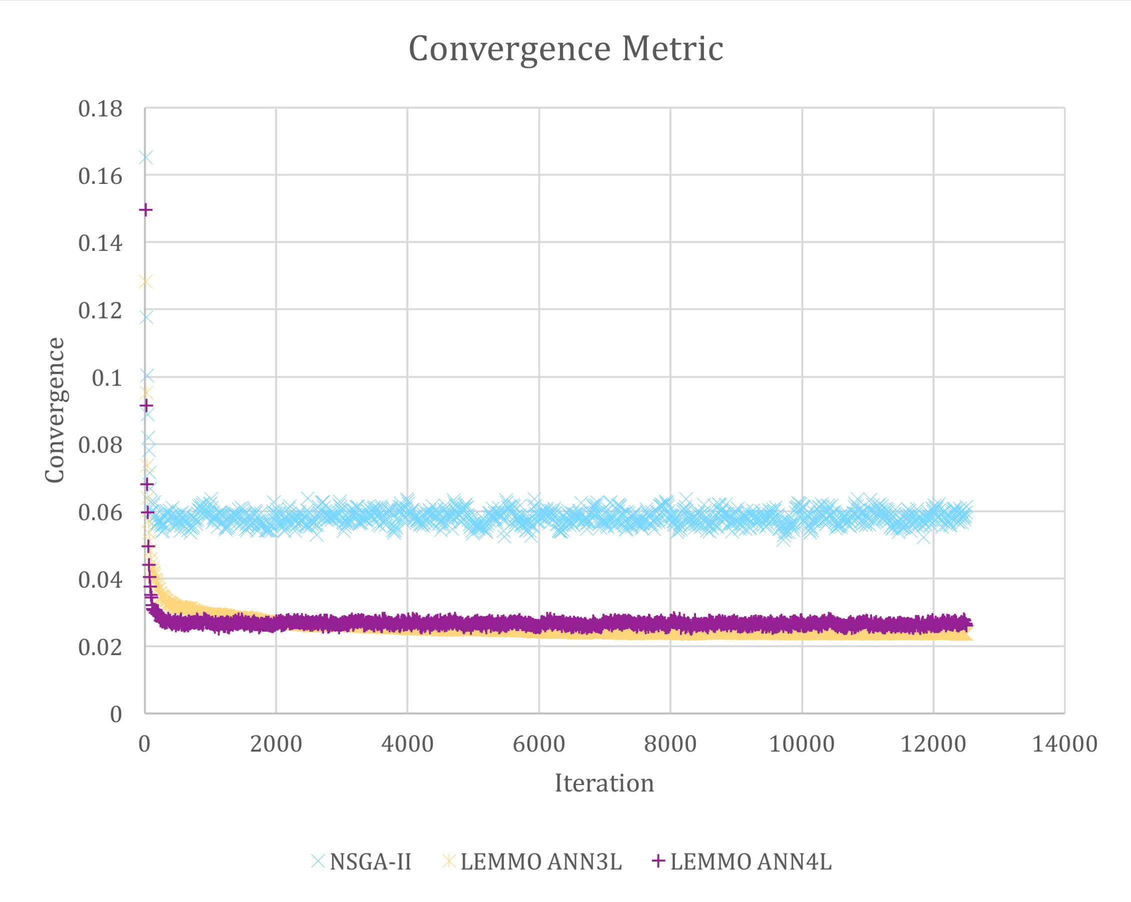


Figure 44 - Averaged convergence metric for TLN

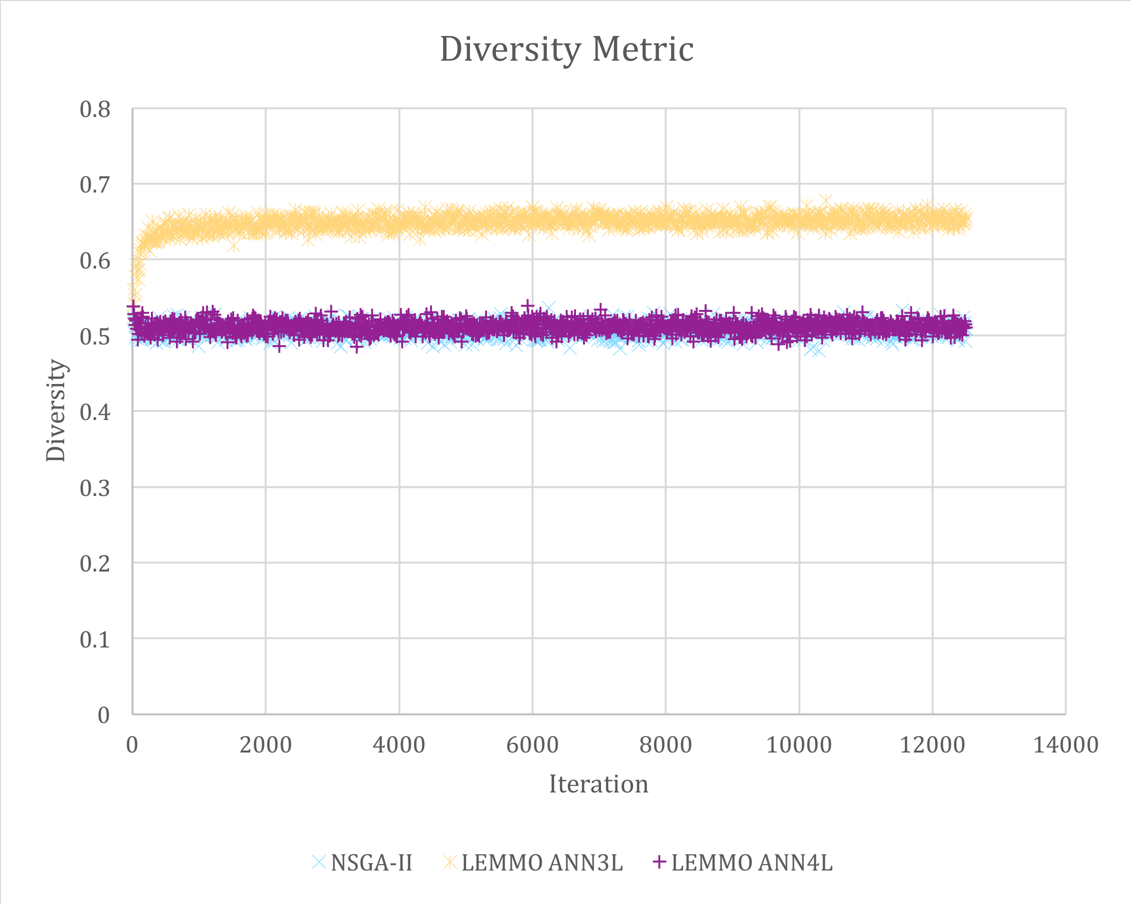


Figure 45 - Averaged diversity metric for TLN

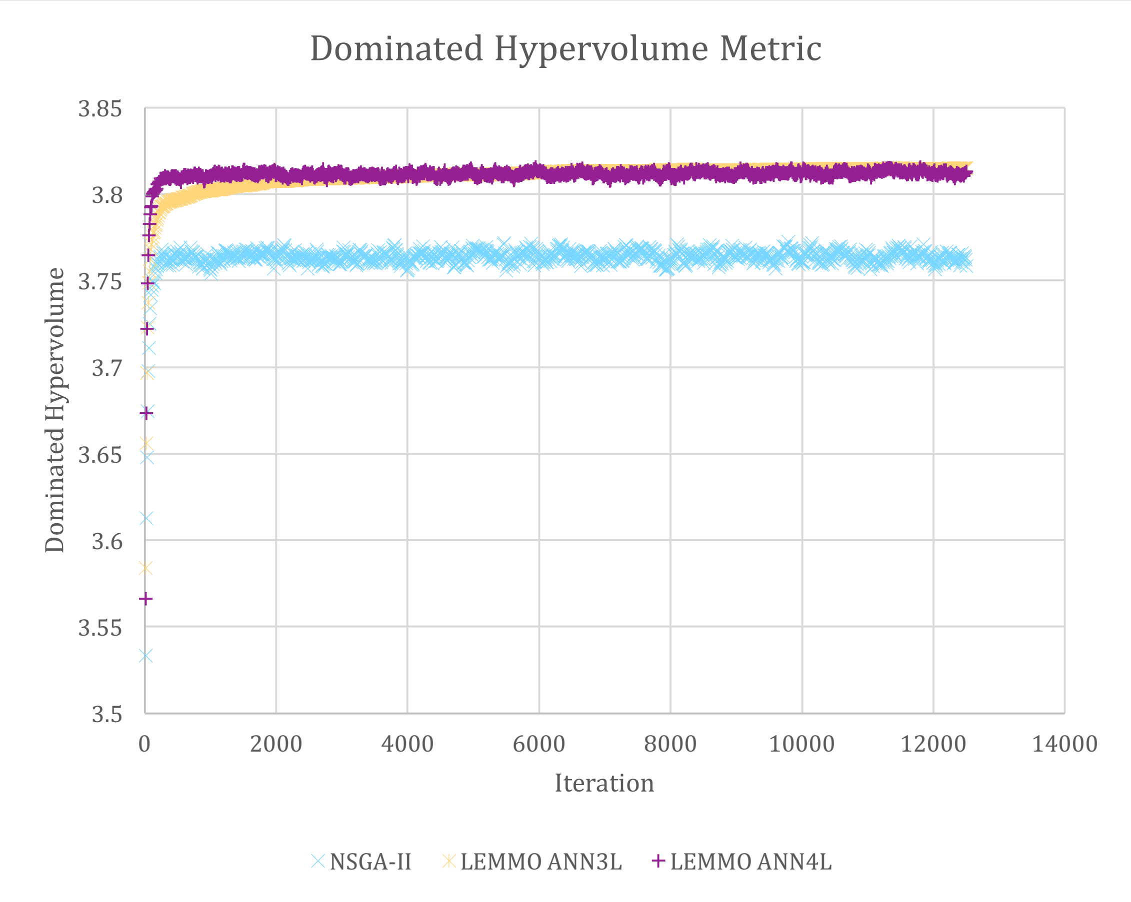


Figure 46 - Averaged dominated hypervolume metric for TLN

5.7.4.2 GOY Analysis

The analysis of the GOY test (see Figure 47, Figure 48 and Figure 49) shows that in terms of convergence and dominated hypervolume, the LEMMO implementation with a four-layer ANN meta-model consistently out-performs the NSGA-II base algorithm. In terms of diversity, the metric appears to be extremely variable from iteration to iteration. However, the overall mean diversity for the NGSA-II base algorithm is 0.427, whereas the overall mean diversity for the LEMMO implementation with four-layer ANN is 0.422. So on average the NSGA-

II base algorithm is slightly out-performing the LEMMO implementation in terms of diversity.

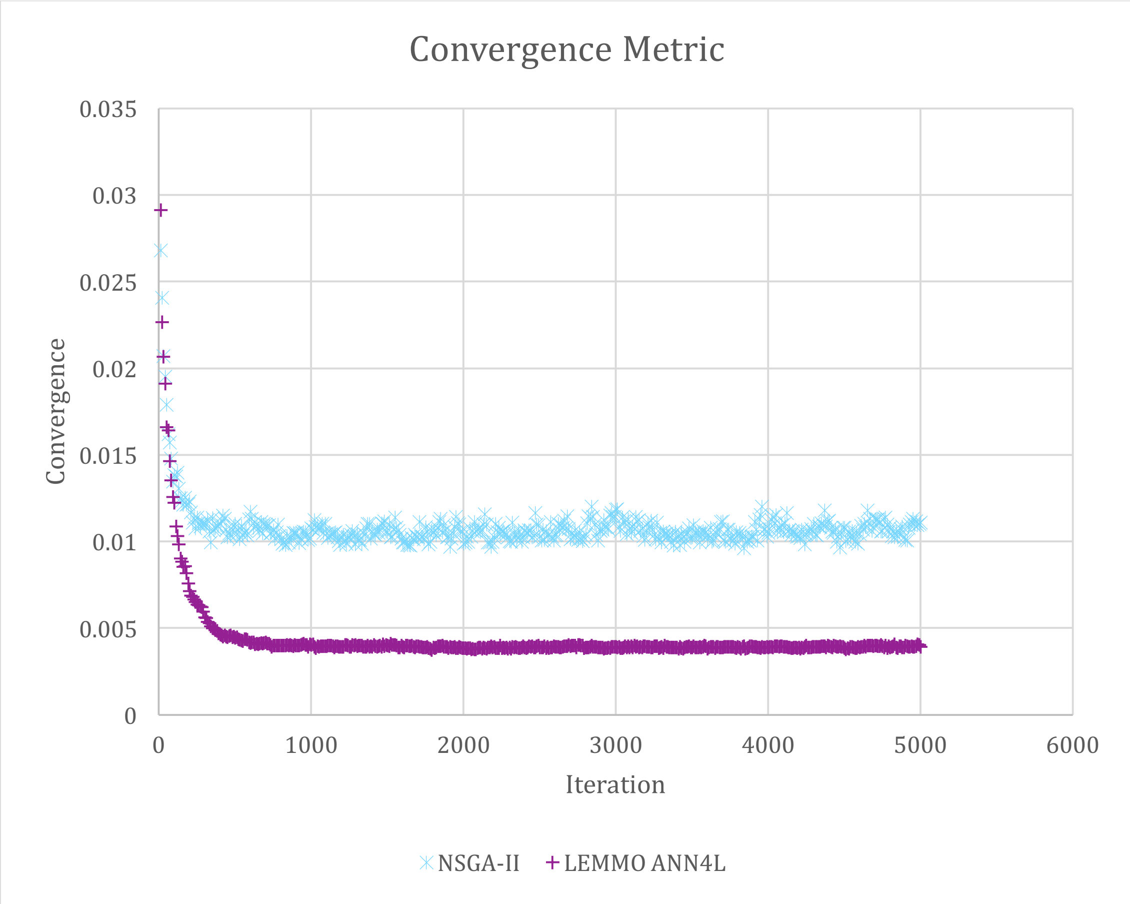


Figure 47 - Averaged convergence metric for GOY

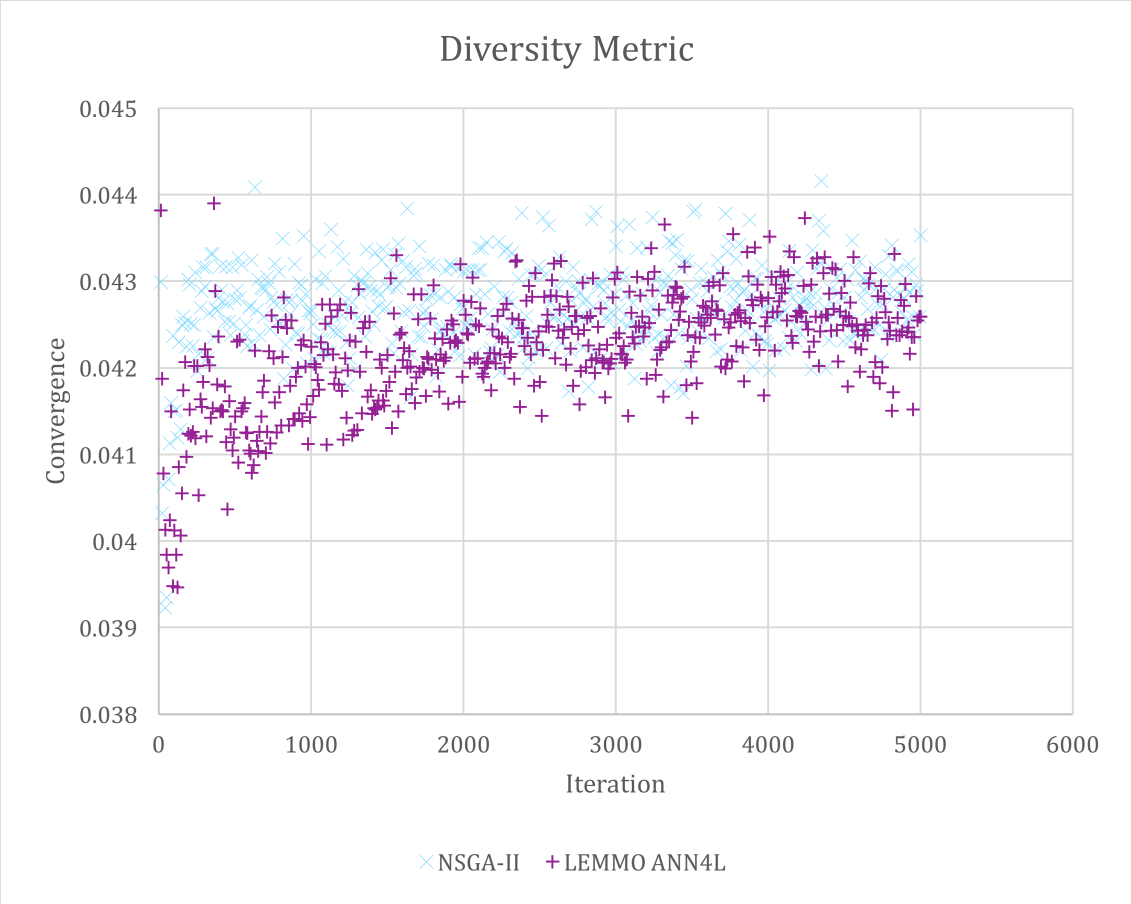


Figure 48 - Averaged diversity metric for GOY

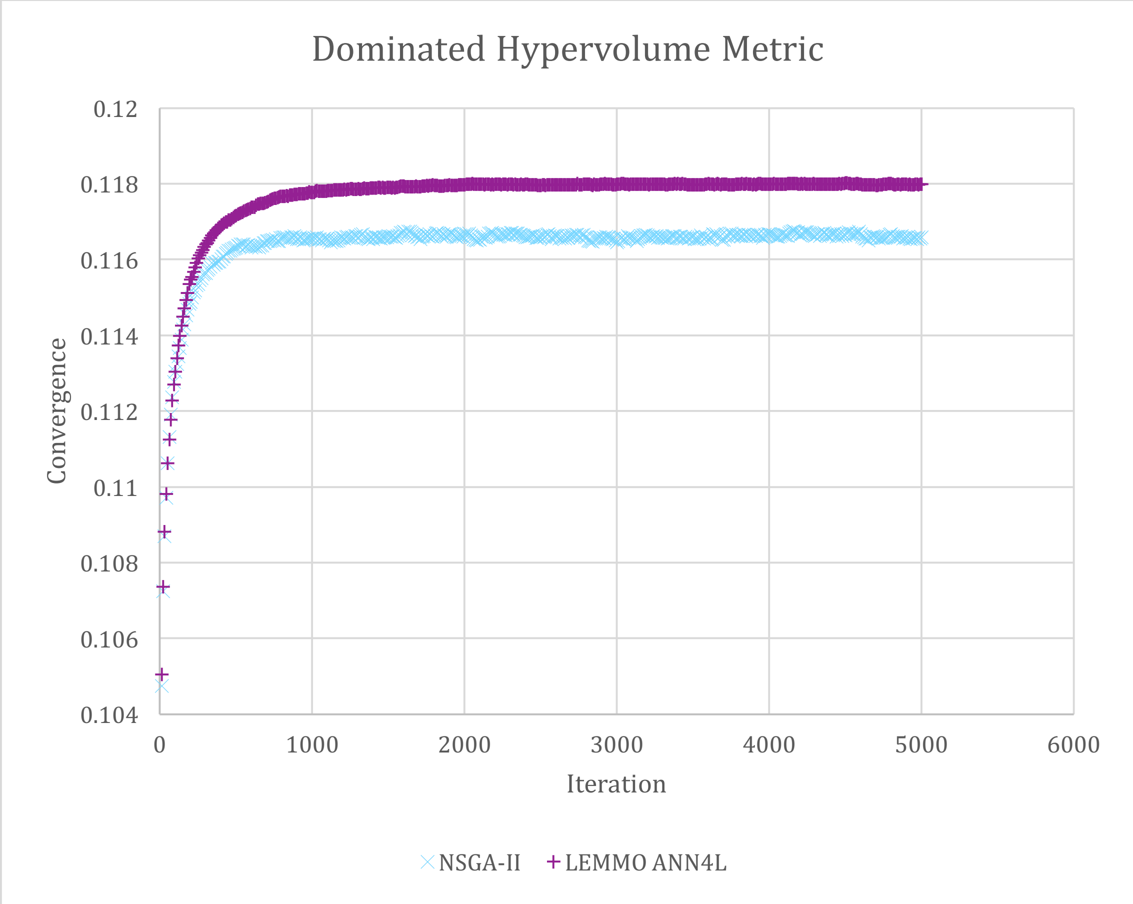


Figure 49 - Averaged dominated hypervolume metric for GOY

5.7.4.3 MOD Analysis

MOD's analysis shows that for convergence, the NSGA-II base algorithm out-performs both LEMMO variants. However, in terms of dominated hypervolume the three-layer LEMMO is least effective, behind the NSGA-II base algorithm, while the four-layer ANN LEMMO performs best. This suggests that the four-layer ANN is achieving a better estimation of the Pareto front, but with solutions that differ from the other two variants results. It also suggests that the LEMMO variant with a three-layer ANN is not modelling the problem well, biasing the algorithm towards a local optimum as a result. In terms of diversity it appears that LEMMO with a three-layer ANN has the most diversity at the stopping point. The NSGA-II base algorithm has slightly less diversity, followed by the four-layer ANN LEMMO variant with the least diversity.

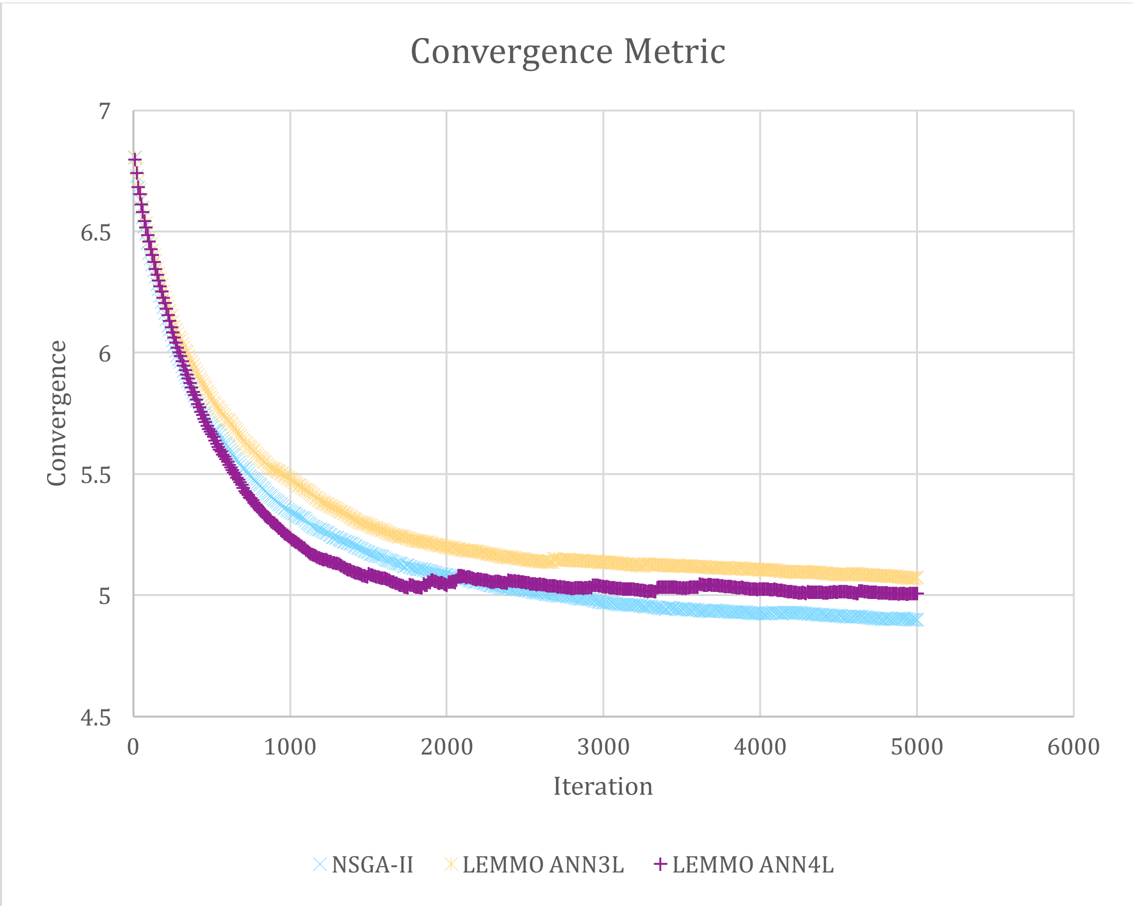


Figure 50 - Averaged convergence metric for MOD

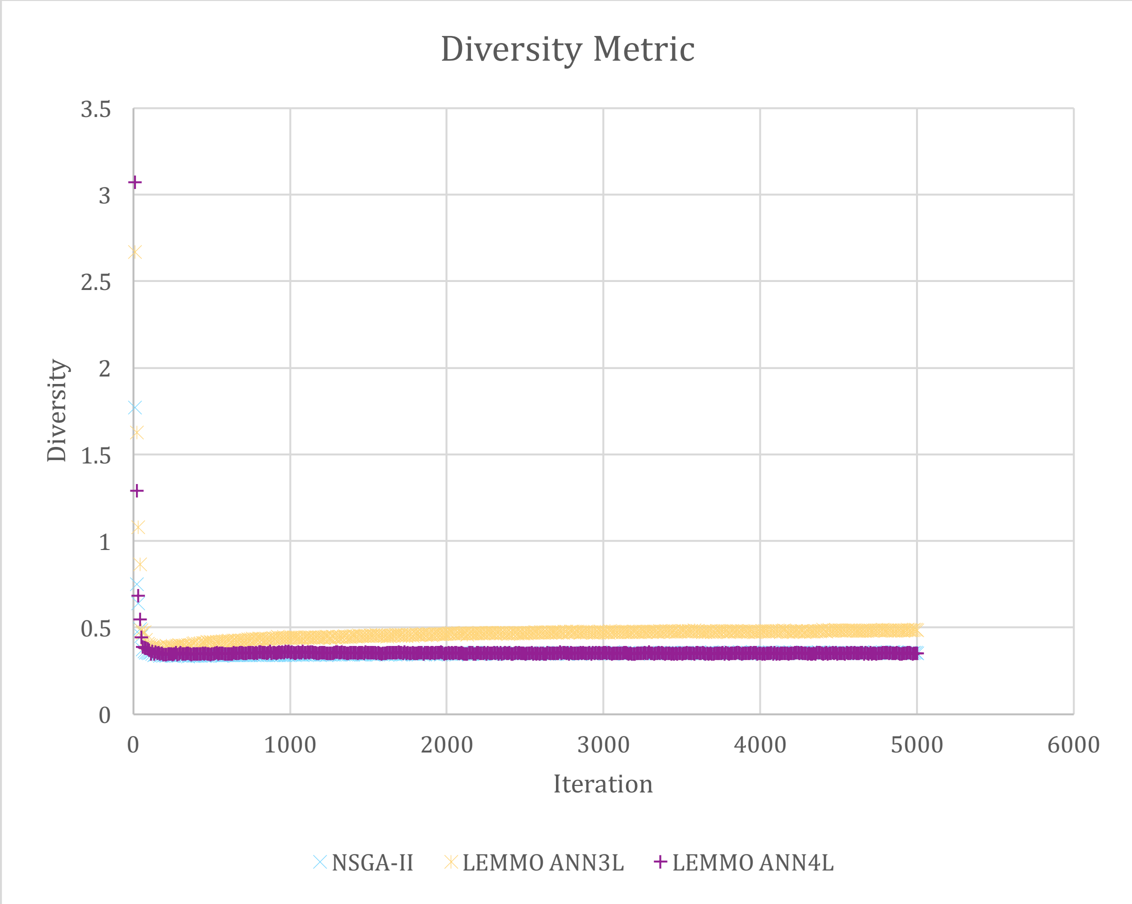


Figure 51 - Averaged diversity metric for MOD

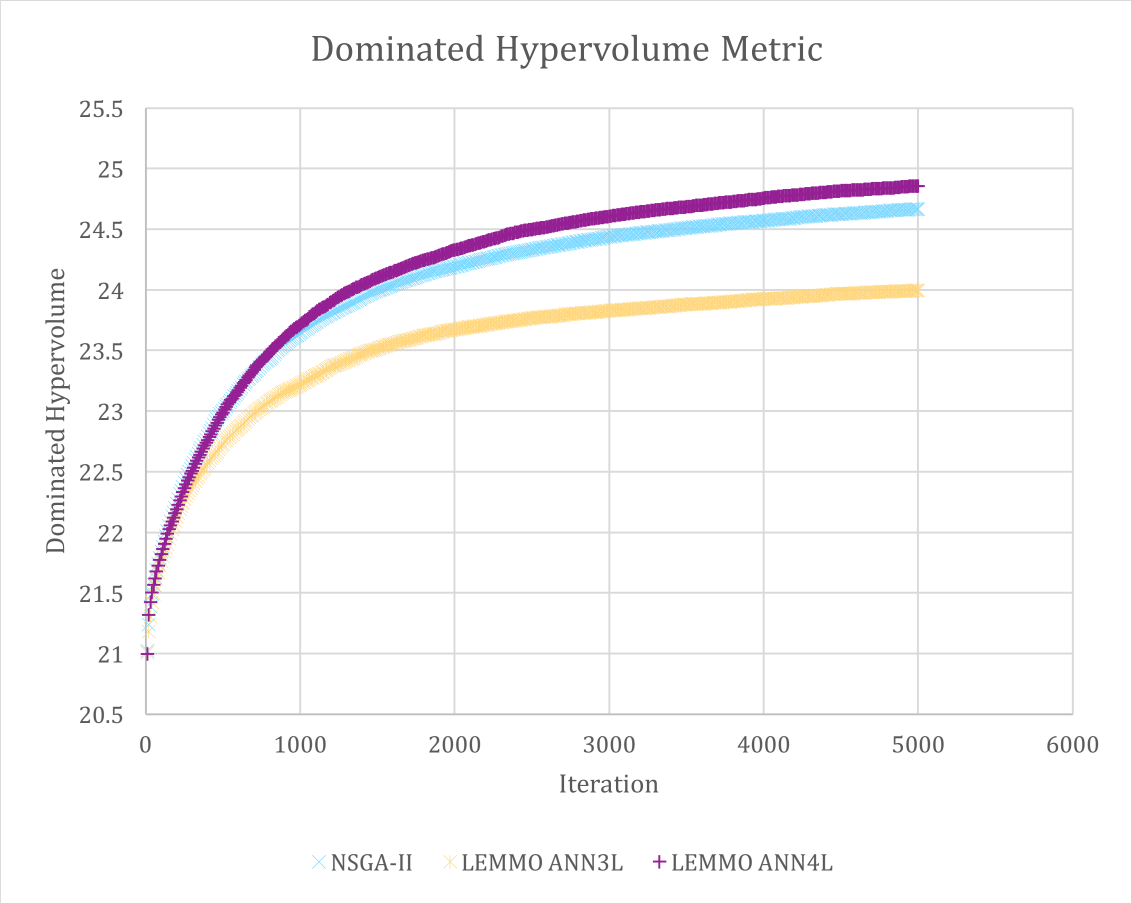


Figure 52 - Averaged dominated hypervolume metric for MOD

5.7.4.4 BIN Analysis

In the analysis of the metrics for the BIN WDS test problem (see Figure 53, Figure 54 and Figure 55) it is clear that LEMMO with a four-layer ANN meta model consistently out-performs both the other algorithm variants for which results are shown. The three-layer ANN is the worst performer of the three variants in terms of convergence and dominated hypervolume. However, it does perform slightly better in terms of diversity and it progresses more quickly during the early iterations.

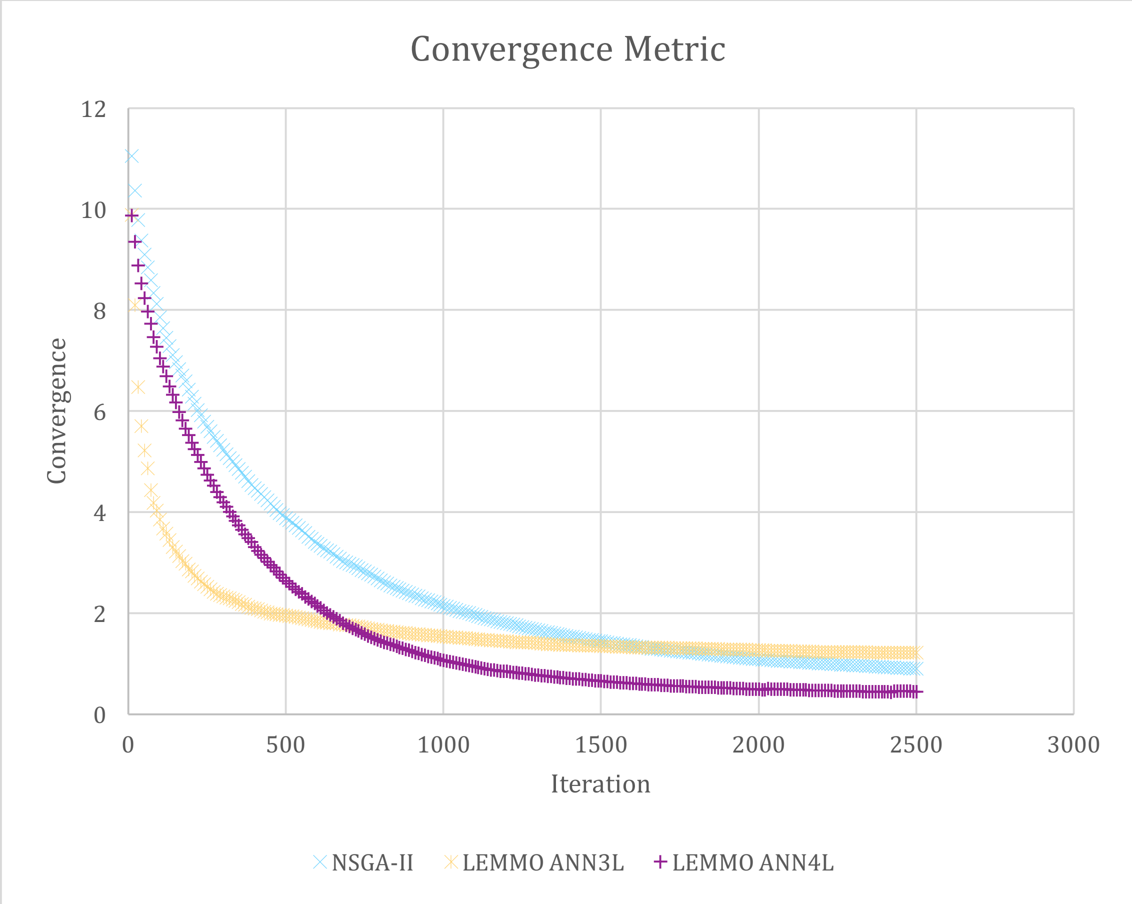


Figure 53 - Averaged convergence metric for BIN

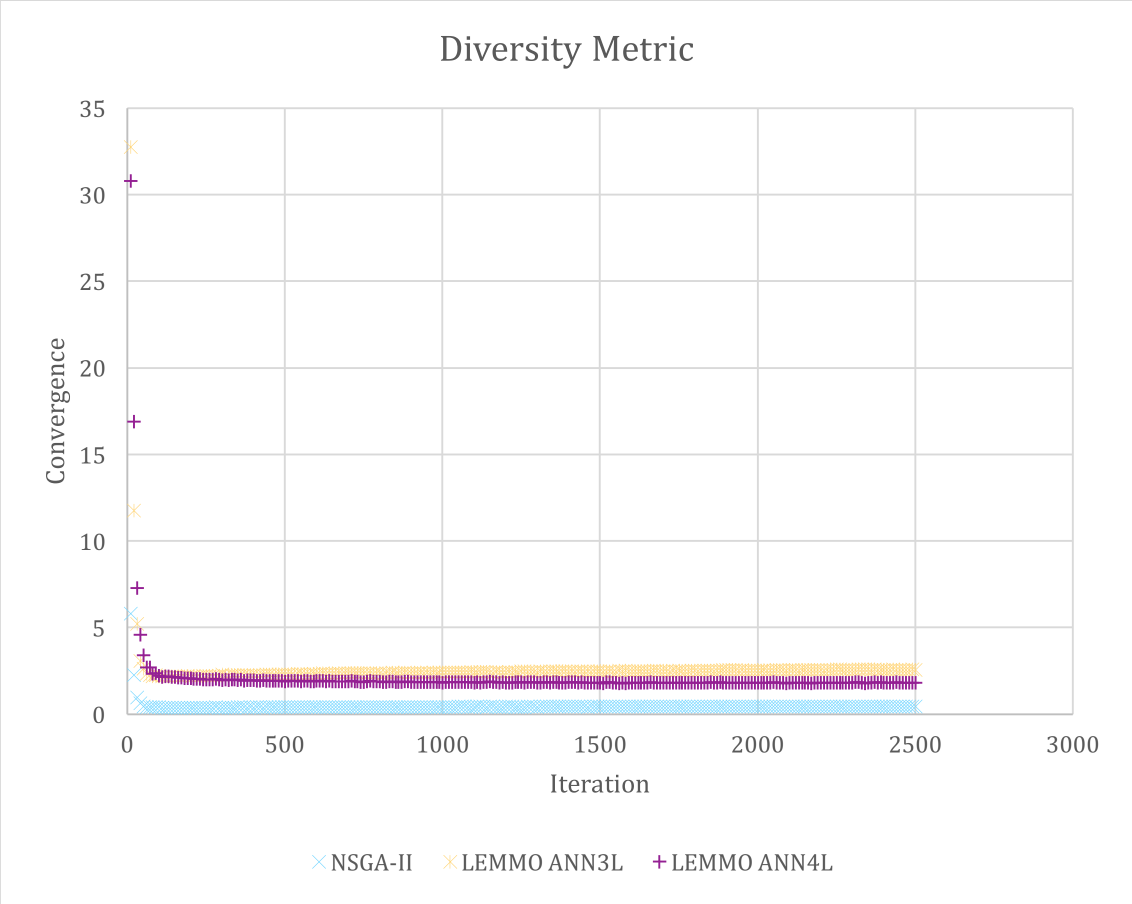


Figure 54 - Averaged diversity metric for BIN

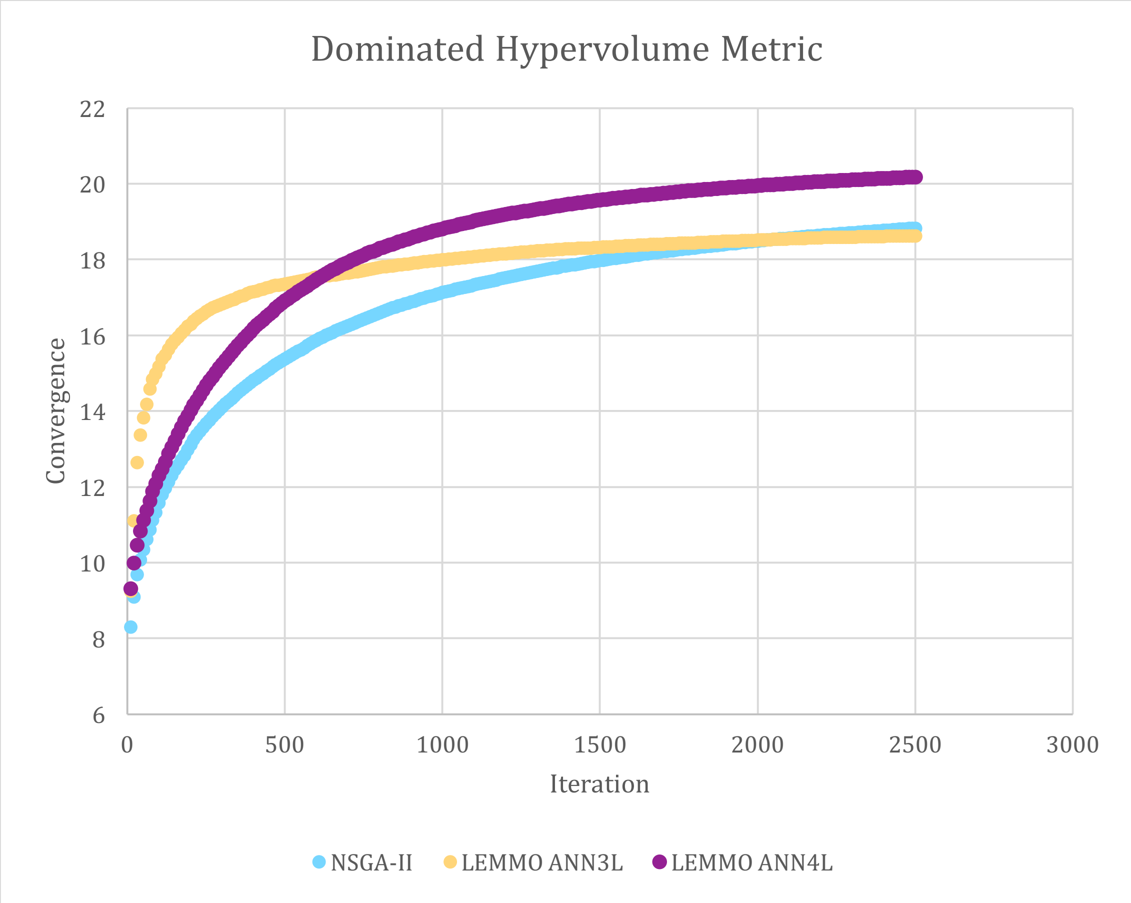


Figure 55 - Averaged dominated hypervolume metric for BIN

5.8 Chapter Summary

It can be seen from the presented results here that the NSGA-II (Deb et al., 2002) algorithm converges well on test problems, approaching the best known Pareto fronts that are being used for comparison (Wang et al., 2014). The LEMMO (Jourdan et al., 2005) approach with a four-layer neural network meta-model can be seen to achieve better convergence towards the same best-known Pareto fronts when using the same number of objective function evaluations. Additionally, it can be seen that the LEMMO approach, with the four-layer neural network, achieves equivalent (to the NSGA-II base algorithm) or better convergence to the best-known Pareto fronts in fewer iterations (and thus fewer objective function evaluations). It is also entirely possible that further tuning the ANN structure and training strategy could further improve the algorithms results in terms of how quickly they approach the optimal pareto front, and how closely they match it.

Logically, no accuracy is lost through this process. The LEMMO algorithm integrates into the NSGA-II algorithm in such a way that if a LEMMO iteration produces only very poor solutions to the problem, they will not enter the population. The only possible negative effect could be that if the meta-model cannot model the complexities of the problem well, it could bias the algorithm towards convergence in a local optimum. This is likely to be the explanation behind results such as in Figure 52 where the LEMMO approach with a three-layer neural network underperforms compared to the two other test variants. Additionally, the time taken to run a LEMMO iteration versus running a full iteration is negligible, meaning that it is very cheap in terms of computational demand to use this technique to improve the results of the NSGA-II algorithm.

The conclusion, therefore, is that the LEMMO approach used with NSGA-II performs well with ANN meta-models. It generally improves the results compared to a standard NSGA-II base algorithm, and achieves comparable results in fewer iterations. The caveat is that the ANN used must be structured and trained well enough that it will approximate the testing function well, otherwise it could bias the algorithm towards local optima.

6. Case Study: Dalmarnock Catchment

6.1 Introduction

The Dalmarnock drainage system is a flood risk catchment used by HR Wallingford for testing the SAM-Risk approach and software (Kellagher et al., 2009). It, therefore, seemed reasonable to utilise this same catchment model as a case-study test for ADAPT and the optimisation algorithms that have been developed during this thesis.

6.2 Dalmarnock Catchment Description

6.2.1 Original Dalmarnock Model

The original Dalmarnock model is a reasonably large, verified, Infoworks CS model of a drainage system covering 96 km^2 and including 5501 nodes, 5468 links and 2172 sub-catchments (see Table 17) (Kellagher et al., 2009).

The Dalmarnock drainage system is mostly combined, but does have a limited amount of separate wastewater pipe work and storm water pipework.

With this model, assuming that all decision variables (pipe diameters, storage node volumes and orifice settings) were included individually rather than grouped together, and using the allowed values defined in the algorithm, the search space would be 2.01×10^{370}, which is in the region of the very large WDS test problems covered in chapter 5.

Area (km^2)	96
No. Nodes	5501
No. Links	5468
No. Sub-catchments	2172
No. weirs	145
No. sluices	46
No. pumps	10
No. flumes	1
No. orifices	41
No. screens	1
No. flap valves	11
Slope (m/m)	0.01

Table 17 - Original Dalmarnock catchment details (Kellagher et al., 2009)

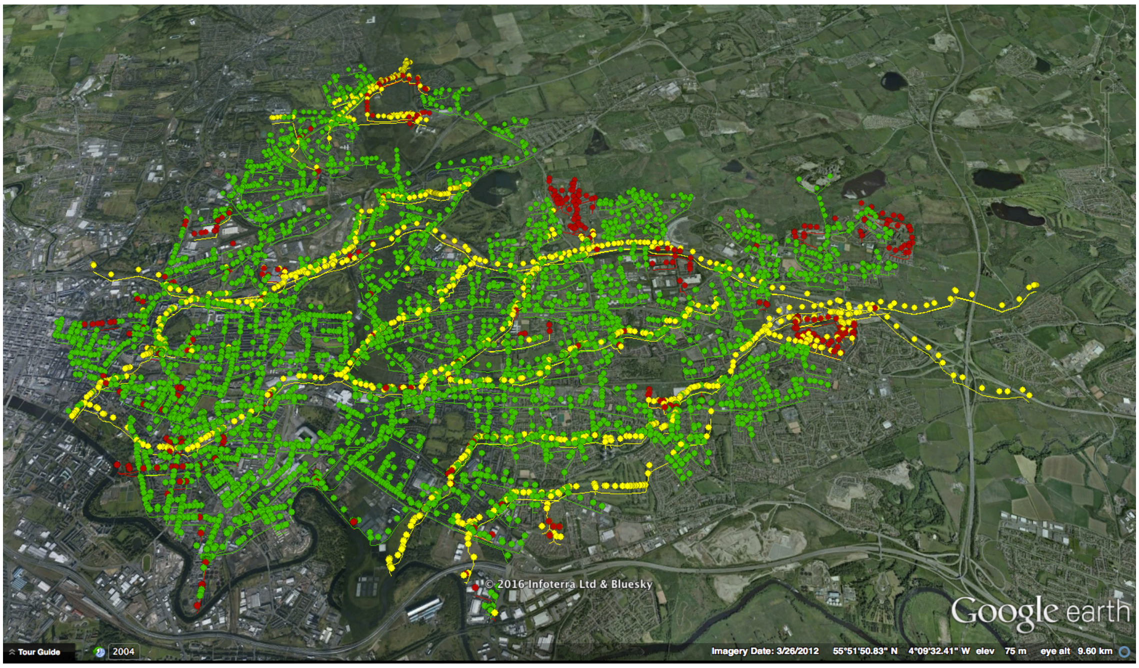


Figure 56 - Full Dalmarnock network model (Kellagher et al., 2009)

6.2.2 Testing Dalmarnock Model

As the algorithms the Dalmarnock model has been tested with are intensive and require repeated calls to the Infoworks CS model during the DTI-SAM project a sub-set of the original model was used. This is the network being used. This subset of the Dalmarnock model covers only 15.34% of the area in the original model, and has only 6.89% of the pipes and 6.83% of the nodes (see Table 18).

This sub-set consists almost entirely of combined pipework, but does include a watercourse routed through a culvert (see Figure 57). With the reduced numbers of pipes and storage nodes in the testing network the new search space is roughly $8.78{\times}10^{254}$ which is significantly reduced, although still substantial.

This test set gives us a large combined system, with a very few waste water only nodes, and a culvert which is storm-flow only. It is, therefore, ideal as a test network, as it encompasses a range of different types of pipes.

Area (km^2)	14.73
No. Nodes	376
No. Links	377
No. Sub-Catchments	153
No. Weirs	0
No. Sluices	1
No. Pumps	2
No. Flumes	0
No. Orifices	3
No. Screens	0
No. Flap Valves	0
Slope (m/m)	0.01

Table 18 - Sub-set of Original Dalmarnock Catchment Model (Kellagher et al., 2009)

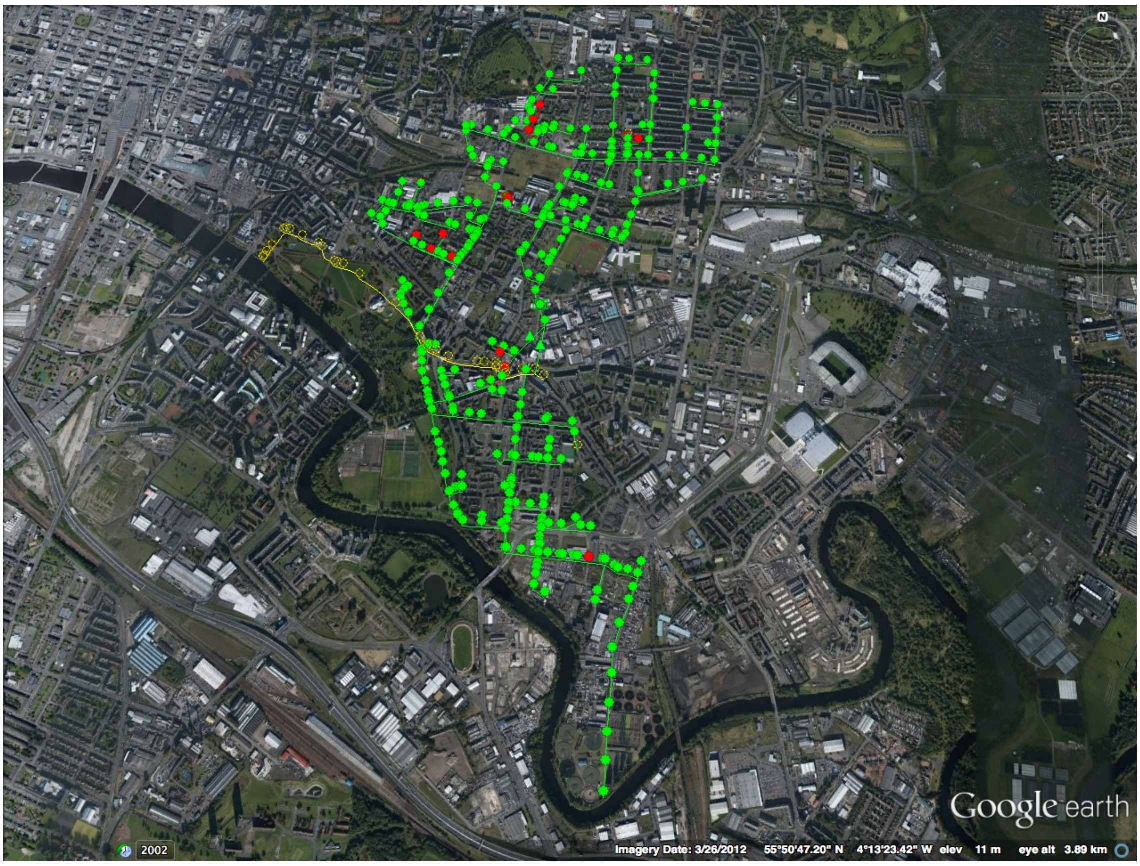


Figure 57 - Testing Dalmarnock Model, Green features are combined, Red are wastewater, and Yellow are storm flow.

6.3 Allowed Decision Variable Values

Of the three kinds of decision variable possible within the developed algorithm, there are limitations on the possible values.

The possible pipe sizes are taken from HR Wallingford and D.I.H. Barr's "Tables for the Hydraulic Design of Pipes, Sewers and Channels" (2006) (see Table 19). Additionally, every pipe can also be its original size, even if that size doesn't fit into the allowed sizes giving them a maximum number of 54 possible values.

Allowed Pipe Sizes (mm)										
150	200	225	250	275	300	350	375	400	450	500
525	600	630	675	700	750	800	825	900	975	1000
1050	1100	1125	1200	1250	1300	1350	1400	1500	1600	1650
1800	1950	2000	2100	2200	2250	2400	2500	2550	2600	2700
2800	2850	3000	3200	3400	3500	3600	4000	4500		

Table 19 - Allowed Pipe Sizes

Orifices are allowed to be any setting between 0 m^3/s and 10 m^3/s in steps of 0.1 m^3/s, giving them a range of 100 possible settings. Meanwhile storage nodes can have a storage area of anywhere from 1 m^3 to 100 m^3 in steps of 1 m^3, giving storage nodes 99 possible values. Additionally, similarly to pipes, nodes can also be their original value, regardless of whether that is in the allowed set.

6.4 Mutation Operator for Dalmarnock

The mutation operator for the Dalmarnock problem had to be developed slightly differently from the mutation operator used in the WDS problems. This was in order to fit with the requirements of restricted sizes and the requirements for maintaining consistency within the pipe groups. For the storage nodes, the mutation operator involved selecting a new size from a normal distribution, then validating that this new size complies with the various rules that may or may not be turned on. These rules include the disallowance of reduction of pipe sizes and storage node sizes (meaning they cannot be reduced below their original value, in the base network), a min and a max size for storage nodes and min/max limits on orifice discharges. Pipe groups mutate by selecting a random increase or decrease in pipe sizes by 0-3 steps on a discrete triangular distribution (within the ordered list of allowed pipe sizes). The selected change is then checked to ensure it meets all currently applied rules before applying that change to the pipes in the group. Finally, orifices select a new value from a normal distribution similarly to storage nodes, before checking that the new value complies with all rules before applying.

6.5 Optimisation Testing Introduction

Time is a factor, and even with the improvements and optimisations that have been made, running an optimisation with a large flood risk problem such as this is an extremely time-consuming undertaking.

Without the various optimisations and reductions in inputs, a full flood-risk analysis to calculate EAD took in the region of 6 hours to test (see section 4.3.2.2). The search space for the used testing network (i.e. the subset of Dalmarnock) with all pipes included is 8.78×10^{254}, and therefore an exhaustive evaluation of this network would take approximately 6.0×10^{251} years. To put this figure into perspective – the current age of the universe is roughly 13.8×10^{9} years.

An optimisation algorithm like NSGA-II improves upon this considerably, by searching heuristically so that it can ignore large portions of the search space and narrow in on the useful portions. An NSGA-II algorithm, running an objective function that takes this long to compute, assuming 5,000 iterations and a population of 100, would take roughly 342 years to complete.

Using the methodology outlined in this thesis, that 6 hour runtime has been improved upon, reducing it to 40 seconds in our case once the reduced rainfall set is found (see section 4.3.1), which is a reduction of more than 99% and a considerable achievement.

With this new runtime, the exhaustive search and the NSGA-II run mentioned previously would take respectively 1.0×10^{249} and 231 days (maximum, depending on how many runs along the way were cached). With caching, it is reasonable to assume that as the algorithm progresses, roughly 50% of the population at each iteration will be carried over from the previous iteration and therefore already evaluated. Reducing that runtime by 50% gives us a figure of 115 days, or roughly 3.9 months. So it can be said with some confidence that the run time of that particular setup would be somewhere between 115 days and 231 days.

In order to reduce the search-space further, and hopefully give a better likelihood of reasonable results with fewer iterations, the optimisation algorithm is only altering a sub-set of these pipes, from the lower end of the network (see Figure 58) although the full network continues to be simulated in the flood risk analysis to generate EAD. This means that the hope would be for our resulting network options to be a small, but significant, improvement in terms of EAD from the original Dalmarnock model. No large improvement is likely without modifying the full selection of pipes and storage nodes. The pipes and nodes included were selected by identifying a point in the network where only two pipes carrying fluid between the two halves (see Figure 58).

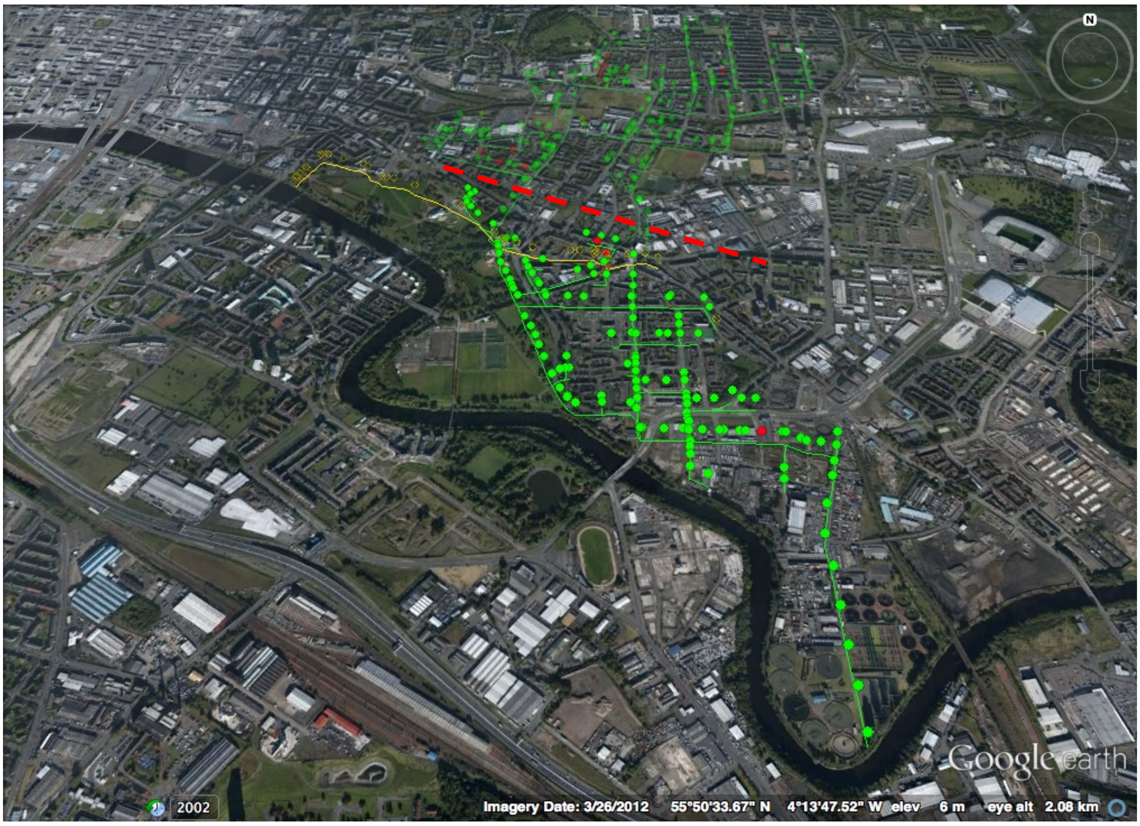


Figure 58 - Decision variables from test Dalmarnock model (decision variable elements are highlighted). Red dashed line indicates separation from un-modified section of Dalmarnock.

No. Nodes	174
No. Links	176
No. Sub-Catchments	56

Figure 59 - Decision variables for Dalmarnock catchment testing

There are a large number of parameters needed for the optimisation algorithm, as well as settings files that are used by the EAD calculation. The settings that have been used can be seen in Appendix III – SAM-Risk Settings.

More details on the initial values of the decision variables used in the optimisation can be found in Appendix IV – Decision Variable Details.

LEMMO is capable of accelerating the progress of the optimisation, such that for a given number of iterations, the progress made is improved over base NSGA-II. Therefore, it may be suitable to run LEMMO with fewer iterations than NSGA-II would require, to present a reasonable answer. However, it can be seen that running this algorithm, even with the huge improvements in performance that have been made, is an extremely time-consuming process and ideally should be performed on a powerful machine with an uninterruptible power supply.

Additionally, as a limitation on the ways in which one might manage to run multiple time-consuming problems in parallel, only one test can be run at a time. This is due to the licensing of the Infoworks software being contained on a USB dongle which can only connect to one computer at a time.

Four runs were undertaken, three runs to ascertain that the algorithm was running correctly when applied to our test model (of 45, 100 and 100 generations respectively), followed by a long run (1049 iterations) to obtain a result which can be investigated and compared to the base system (the Dalmarnock test model with no alterations). This 1049 generation run took from 20:42 on 09/05/2015 until 00:30 on 04/07/2015, a total of over 55 days of runtime. The computer used to perform these tests was as described in section 1.1.

6.6 Reduced Data-set Identification

The original rainfall set comprises 700 different rainfall files, encompassing return periods 2,5,10 then steps of 10 to 300, 500,750 and 1000 years. For each of those return periods, there are durations of 30 minutes, 60, 90, then in steps of 30 minutes all the way to 600.

As a first stage of optimisation on the Dalmarnock flood risk model, the process detailed in section 4.3.1 was followed to generate a reduced data-set to be used for the optimisation of input data used to compute the objective function.

This results in the rainfall setup detailed in Table 20 being identified.

Return period (Yrs)	Duration (S)
2	36,000
20	36,000
40	36,000
80	36,000
160	36,000
750	36,000
1000	36,000

Table 20 - Rainfall setup for reduced data-set identification

6.7 NSGA-II and LEMMO Optimisation

6.7.1 Basic Run Parameters

This section gives the basic run parameters for all four optimisations run on Dalmarnock. The full details of the run parameters can be seen in Appendix III – SAM-Risk Settings. The parameters were selected to give a reasonable chance of running the algorithm to completion. These were a population size of 80, generation’s limit of 2,000, crossover rate of 1.0, and a mutation rate of 0.002 (1/n where ‘n’ is the number of decision variables). The objective functions are network cost and expected annual damage.

6.7.2 Optimisation Results

6.7.2.1 45 Iteration Run Results

It can be seen (Figure 60) in this very short run that the algorithm is improving and quickly forming an estimated Pareto front. By the 10th iteration, a Pareto front is already forming. This 45 iteration run took around two days total to complete.

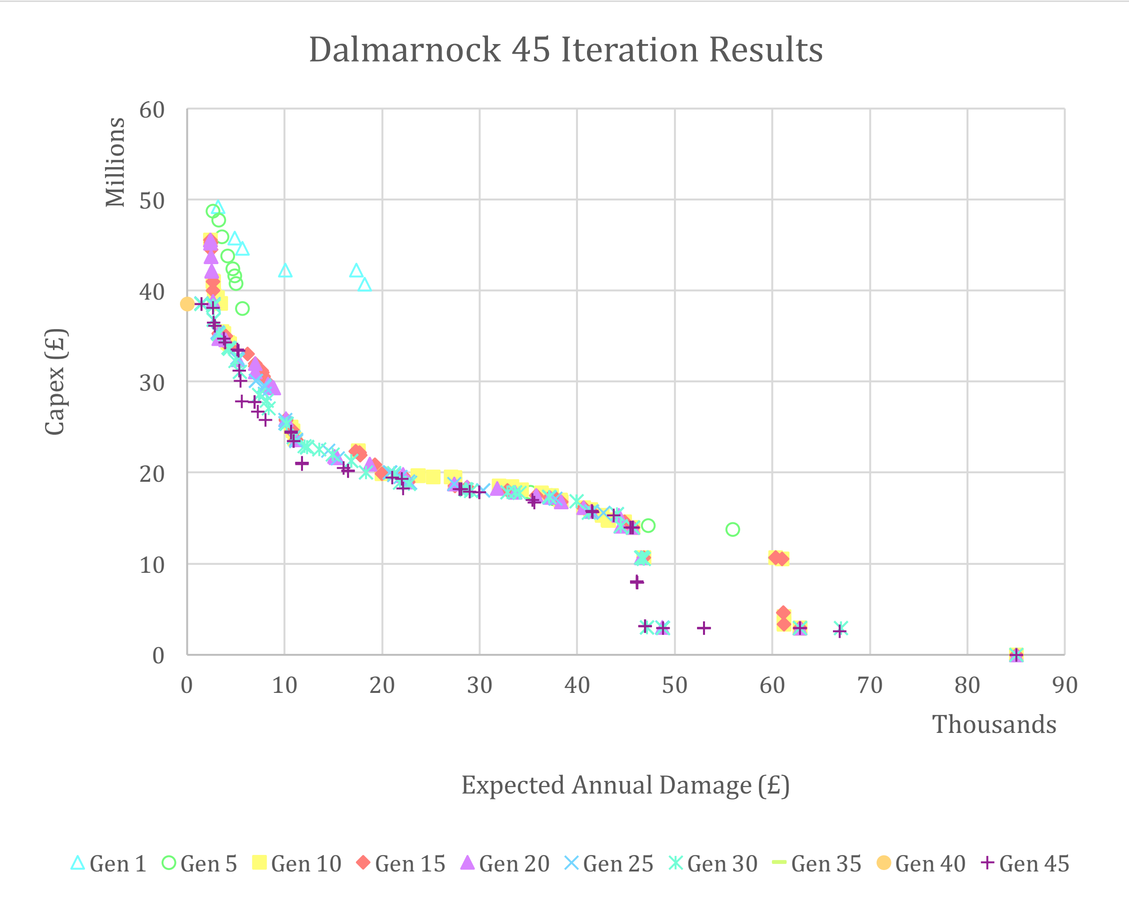


Figure 60 - Dalmarnock 45 Iteration Run Results

In the dominated hypervolume analysis for the 45 iteration run (Figure 61), it can be seen that the algorithm initially progresses very rapidly (within the first ten iterations) then slows in progress, although progress is still being made. The reference point for this hypervolume calculation is at x = 90,000 and y = 60,000,000.

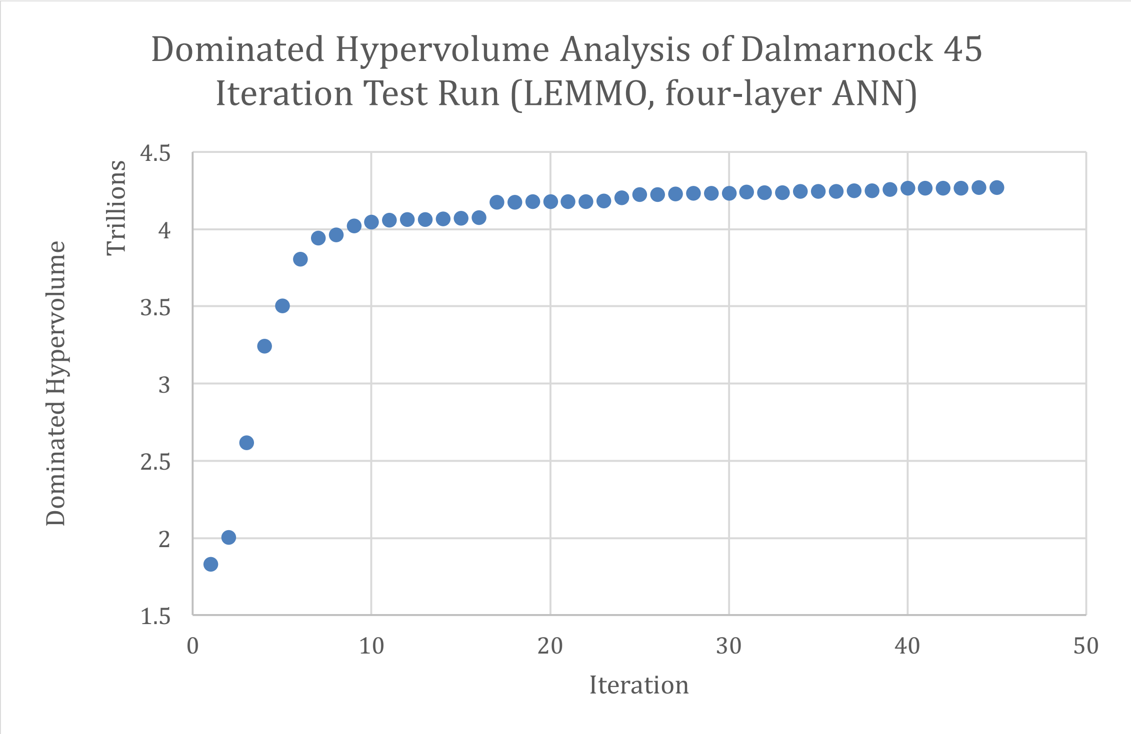


Figure 61 - Dominated Hypervolume Analysis of Dalmarnock 45 Iteration Test Run (LEMMO, four-layer ANN)

6.7.2.2 100 Iteration Run ‘A’ Results

In the results of the 100 iteration ‘A’ run (see Figure 62) the same quick start of convergence can be seen. Additionally, it can be seen more clearly that the algorithm starts converging towards the 0,0 point as the “waves” of estimated Pareto front can be discerned to be moving in that direction.

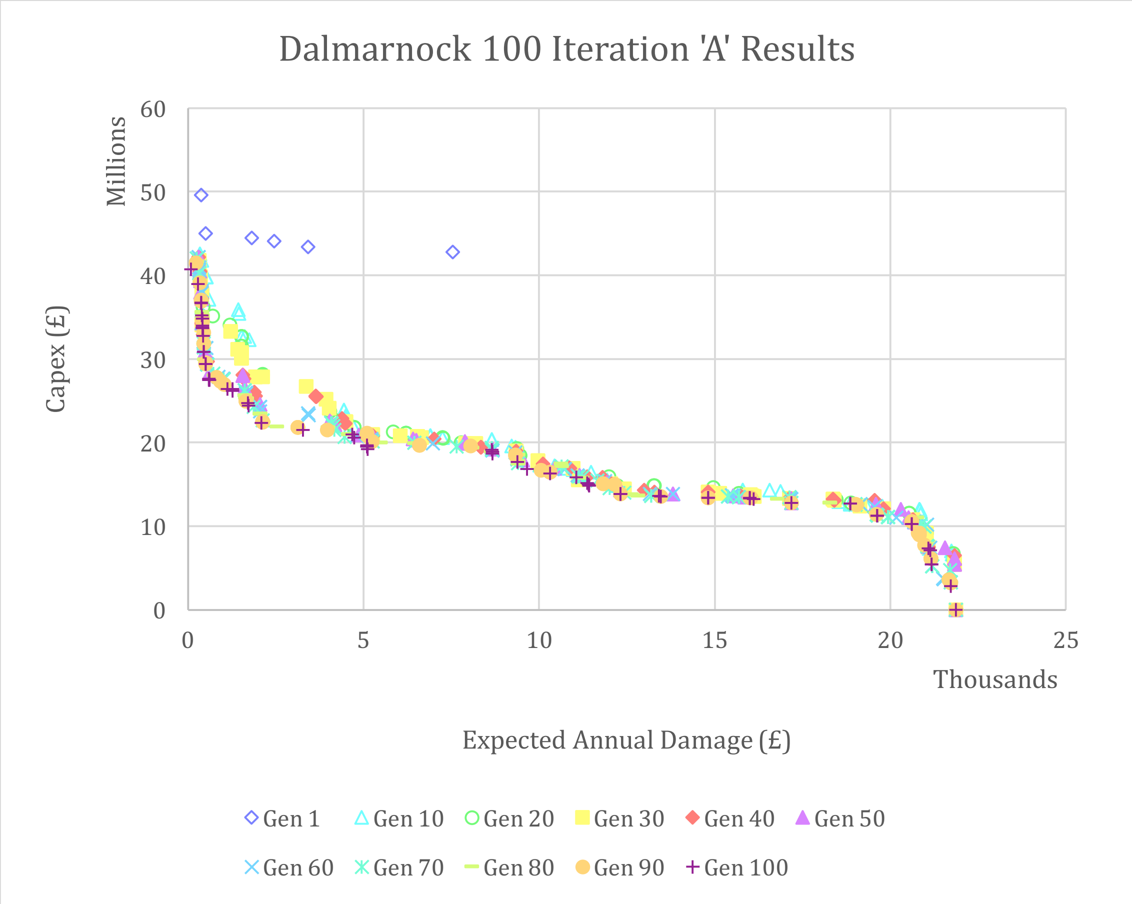


Figure 62 - Dalmarnock 100 Iteration 'A' Results

It may be noted that the maximum EAD is considerably less than in either the 45 iteration or the 1049 iteration run, this is due to the optimisation running with a cut down model that only incorporated the portion of Dalmarnock that encompasses the decision variables being used. This was in order to speed up execution whilst performing test runs. This test run took approximately 3 days to complete – whereas, it could have been considerably longer had it been exporting the entire test-model of Dalmarnock at each simulation.

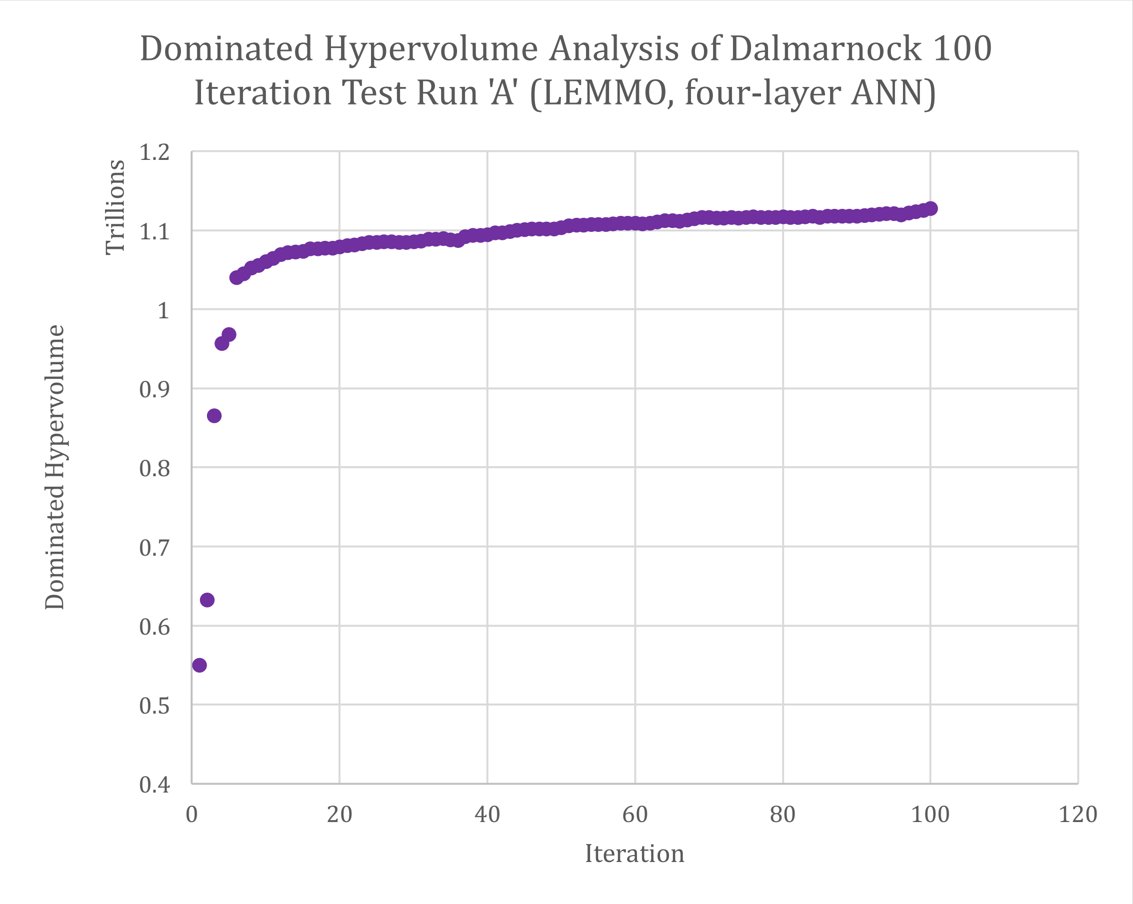


Figure 63 - Dominated Hypervolume Analysis of Dalmarnock 100 Iteration Test Run (LEMMO, four-layer ANN)

The dominated hypervolume for the 100 iteration run 'A' can be seen (Figure 63) to be progressing extremely rapidly in the initial generations, followed by a tapering off into steady progress. This is on par with the results that have been seen on other runs. Small jumps in progress can be seen, which could be an indication that the LEMMO algorithm is performing well and making "leaps" of intuition.

6.7.2.3 100 Iteration Run ‘B’ Results

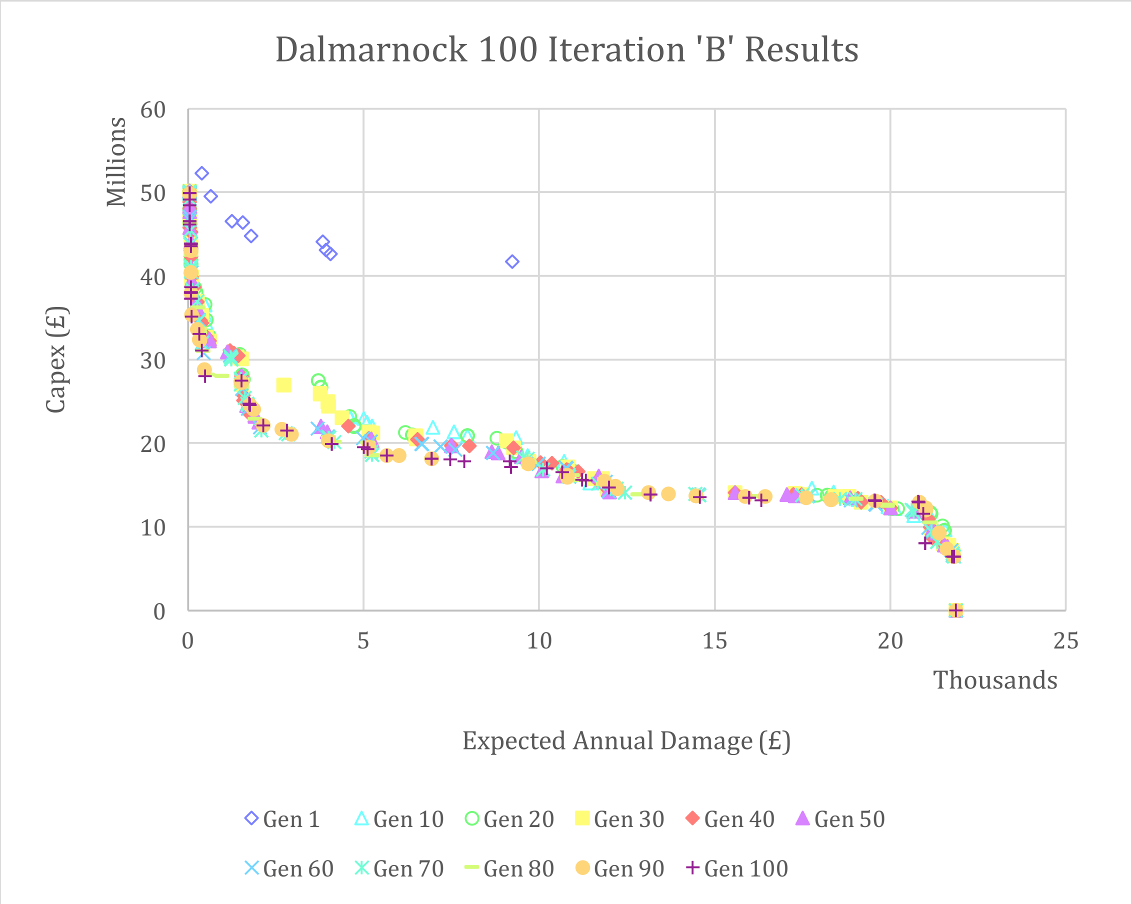


Figure 64 - 100 Iteration Run 'B' Results

It can be seen (see Figure 64) that the algorithm progresses well. As with run ‘A’ this test was run with a cut-down network to speed up optimisation progress. This run also took around 3 days to complete.

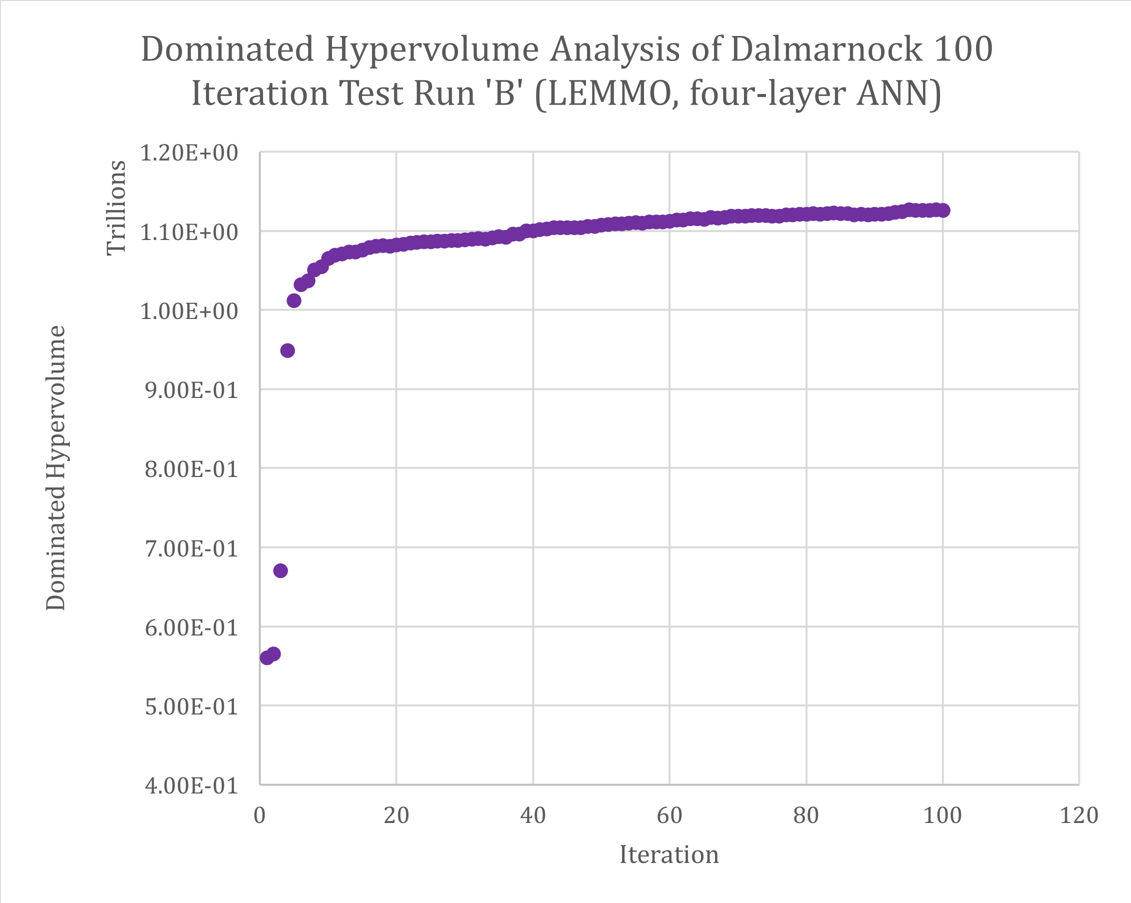


Figure 65 - Dominated Hypervolume Analysis of Dalmarnock 100 Iteration Test Run 'B' (LEMMO, four-layer ANN)

A similar pattern can be seen in the dominated hypervolume metric as well (when compared to Dalmarnock 100 iteration run 'A'). Where very fast initial progress is shortly followed by a more gradual, but steady increase in the dominated hypervolume.

6.7.2.4 Full Case Study Results

The optimisation results for the Dalmarnock case study show a steady convergence taking place, towards the minimisation of both objectives (Capex and EAD). It is more difficult to discern the individual lines as they converge, so two extra graphs with altered axes are also presented, to make it easier to see the differentiation.

It can be seen that convergence towards the ideal (0, 0) point commences throughout the algorithm, and as the algorithm continues significant progress is made, particularly in the mid-range. It can be discerned that the mid-range progresses from an average of around £20m capex and between 5 – 60k EAD, to an average across the range of EAD of roughly £5m capex. This an extremely significant saving which would be well worth the time investment in running an algorithm.

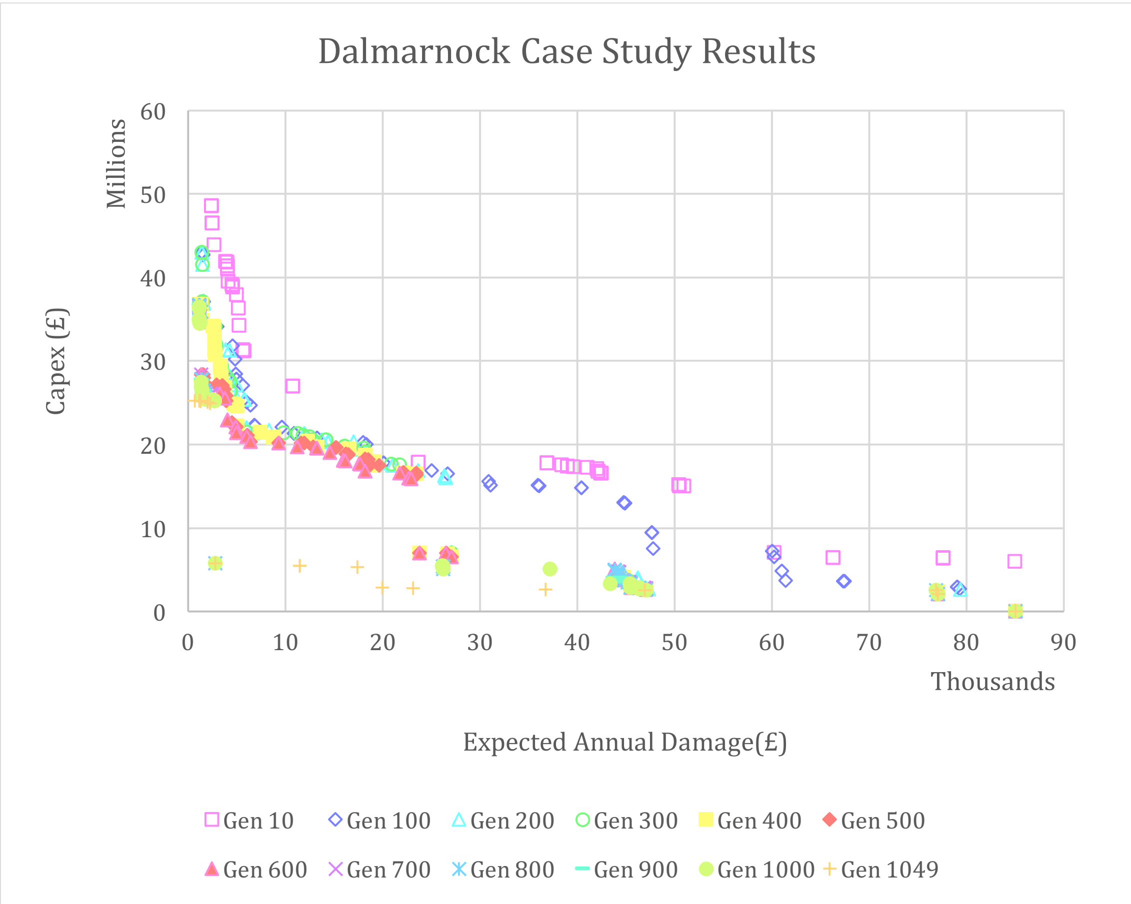


Figure 66 - Overall Dalmarnock case study results

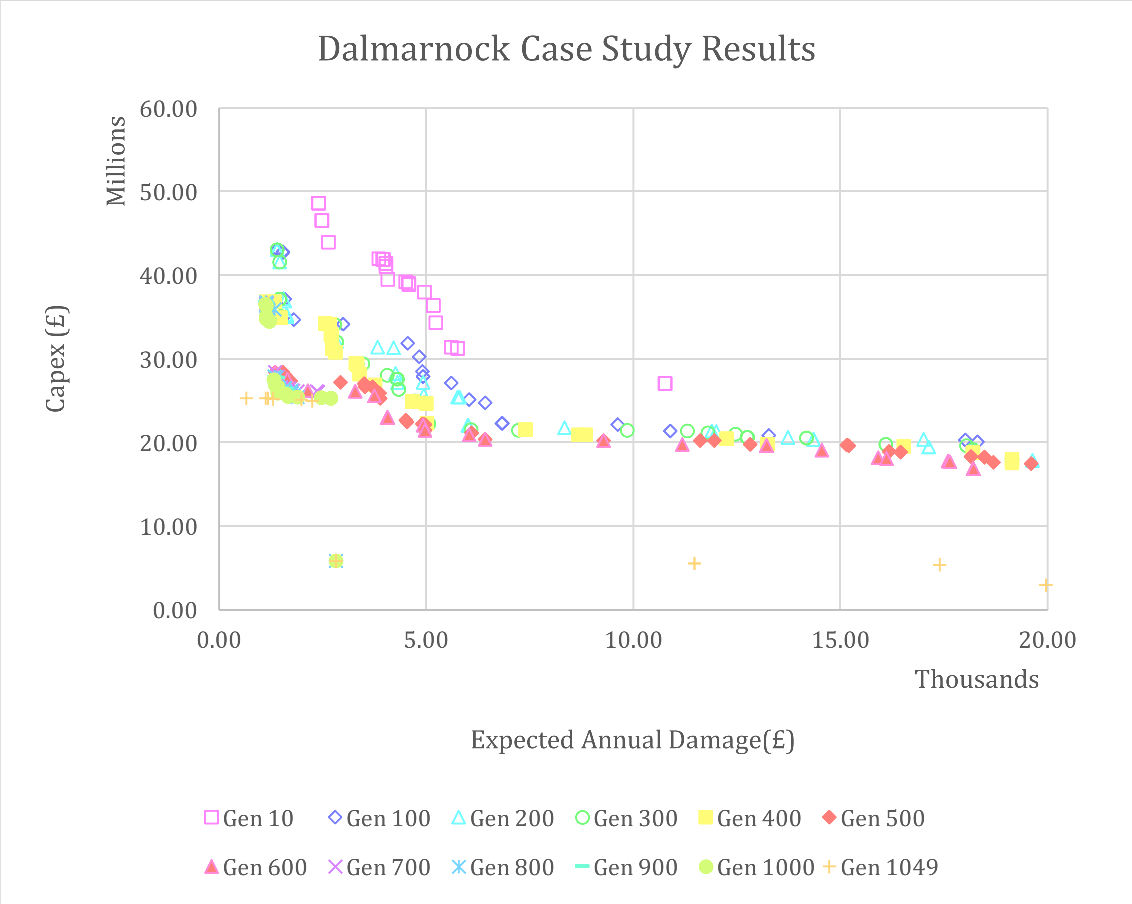


Figure 67 – Dalmarnock case study results, EAD 0 – 20k

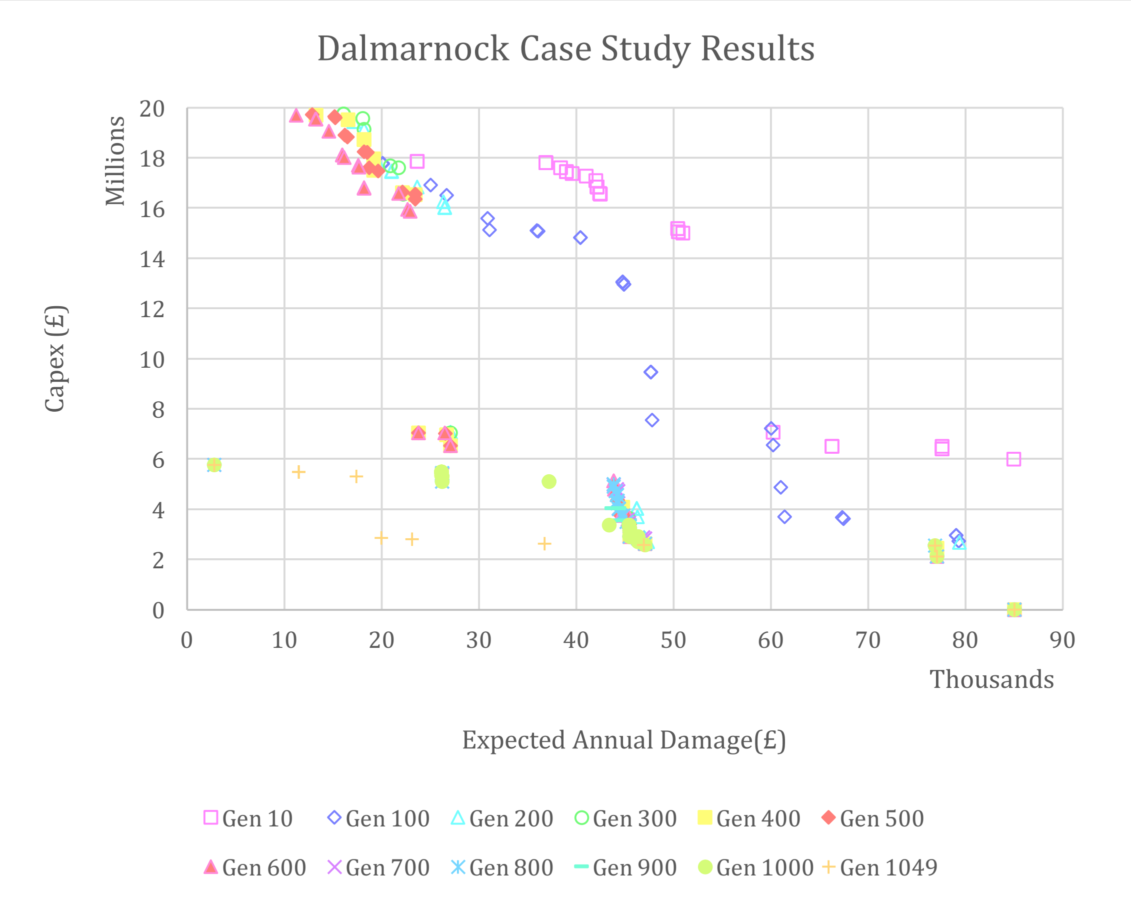


Figure 68 - Dalmarnock case study results, Capex 0-20m

6.8 Dalmarnock Optimisation Solution Analysis

The dominated hypervolume metric (see Figure 69) applied to the results from the optimisation run on the Dalmarnock problem, shows a steady increase in dominated hypervolume, as it would be expected from a converging algorithm. For this generation of dominated hypervolume, the reference point was set at x=90,000 and y=60,000,000. Initial progress is very fast, followed by a smoother more gradual progression. This smoother part contains periodic “jumps” (circled in Figure 69) which could either be due to mutation, or the effects of the LEMMO algorithm resulting in intuition-like “leaps”.

The initial progress tallies well in general shape with both Figure 61 and Figure 63 thus giving an indication that those two runs, had they continued, would likely have followed the same, or a similar curve.

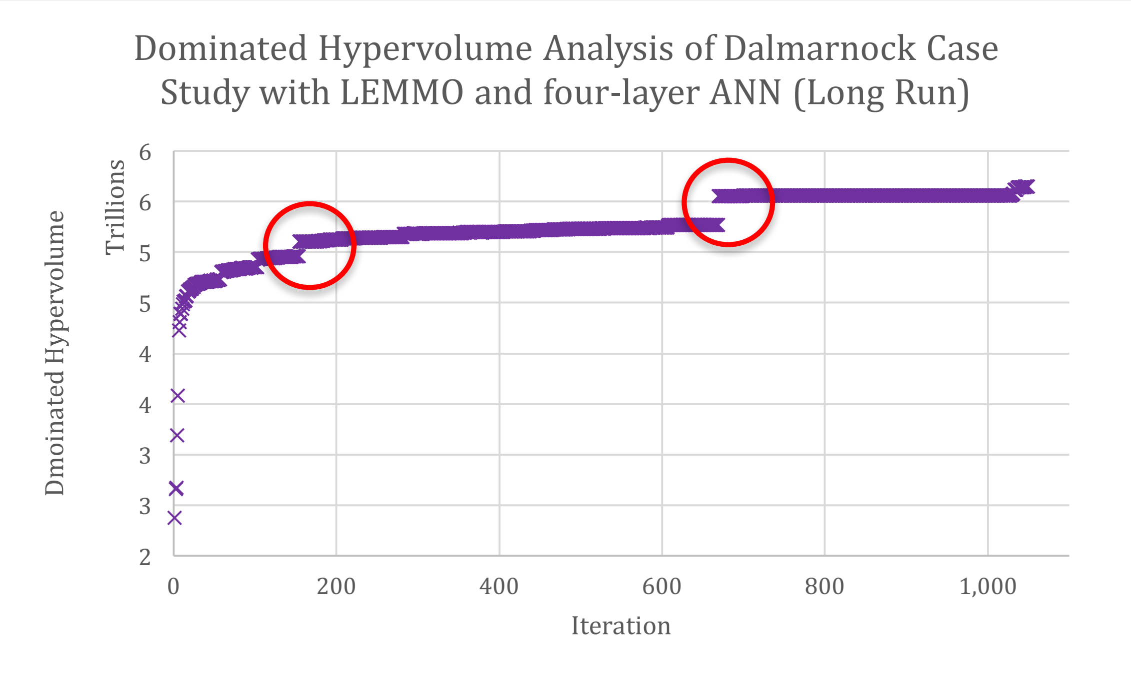


Figure 69 - Dominated hypervolume metric for Dalmarnock case study with LEMMO and four-layer ANN (Long Run)

As the Dalmarnock catchment has been optimised by means of a reduced-rainfall set, it would be prudent to analyse a selection of points along its estimated Pareto set with the full rainfall set. To this end, three points have been selected from the optimisation results and have been circled in red in Figure 70. These points were selected to give a range. The first point (starting from the leftmost) is part of a cluster of very high cost, low EAD solutions, this is referred to as point A. The second (point B) is mid-way along our estimated Pareto front, and given the large jump in Capex that occurs when one attempts to move any further to the left (i.e.

to even lower EAD results) is a point which would likely be of interest to engineers investigating this catchment. The final point (point C) is part of a relatively flat portion of the estimated front, with a high EAD in comparison to the other two, but a low cost.

A full SAM-Risk assessment was then run on these selected points (A, B, and C) and the original drainage network, the results of which can be seen in

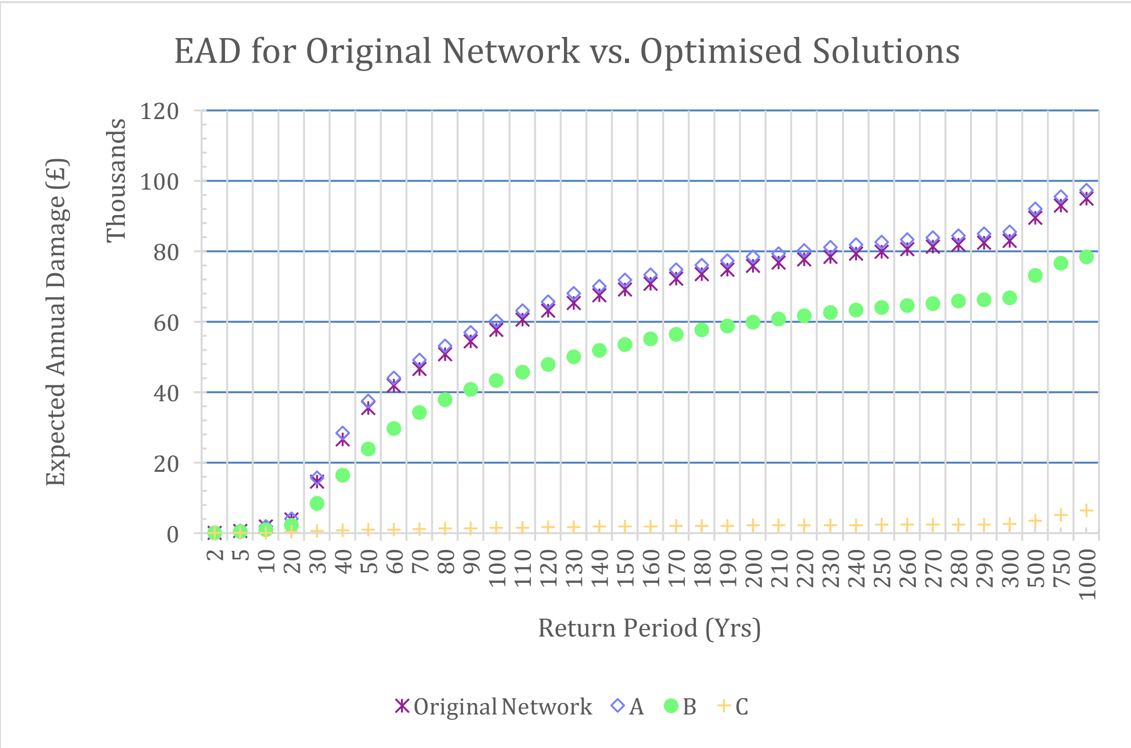


Figure 71. The optimised result can be seen to be significantly lower in EAD across the entire range of return periods analysed.

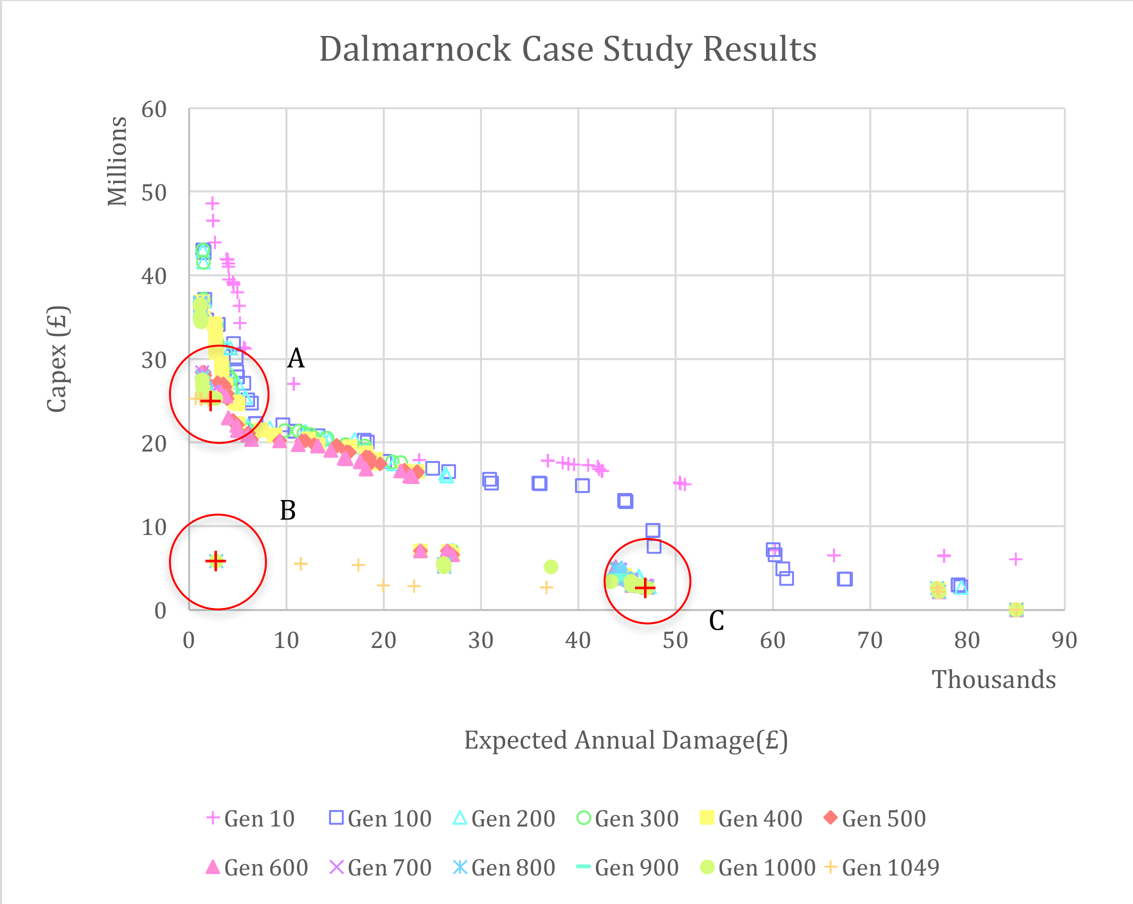


Figure 70 - Selected points A, B, and C from Dalmarnock Case Study Results (points circled and coloured red).

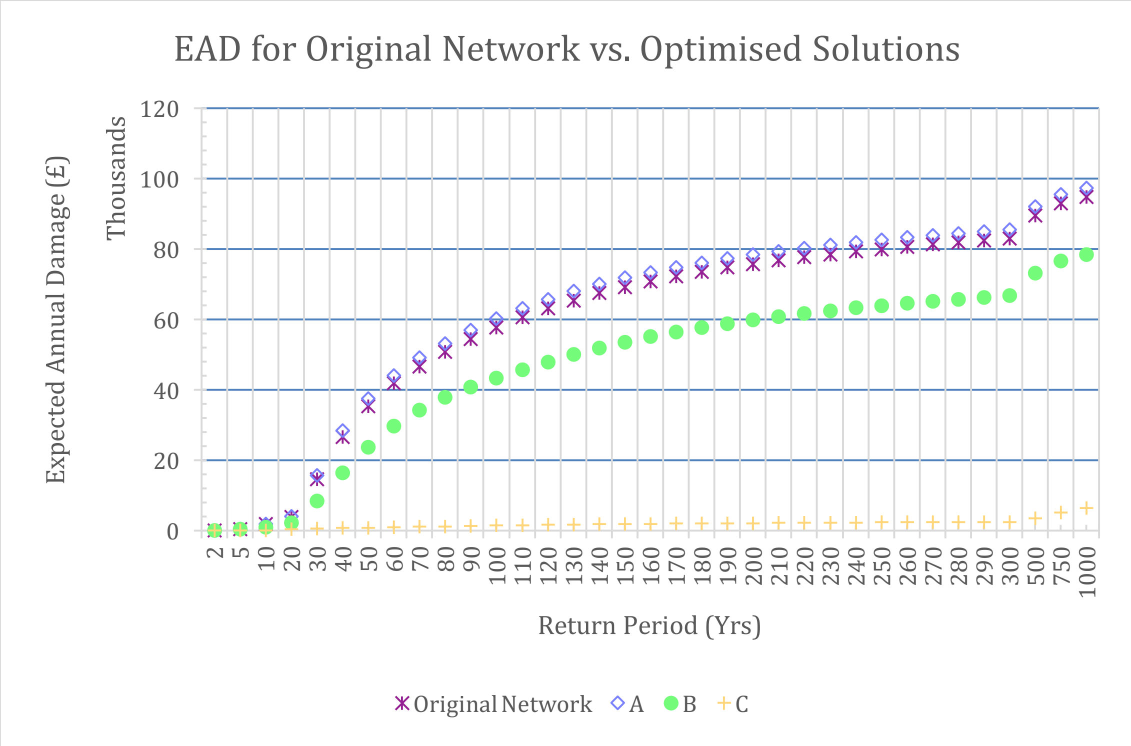


Figure 71 - EAD for Original Network vs Selected Optimised Result

Analysis of the decision variable values for these selected points (see Appendix IV – Decision Variable Details) shows that point 'C' has been extensively altered. Almost all pipe sizes have been increased in width, and almost all storage nodes increased in chamber area. This can be seen reflected in the averages in Table 21. Point 'B' has experienced hardly any storage chamber increase (an average of 0.15) however, a reasonable amount of pipe width increase. This suggests that the algorithm is finding that altering specific pipes can show an improvement in EAD for a very reasonable cost. In order to gain large improvements, however, it is necessary to start increasing both pipe sizes and storage node sizes, which leads to rapid cost increases.

It can be seen in

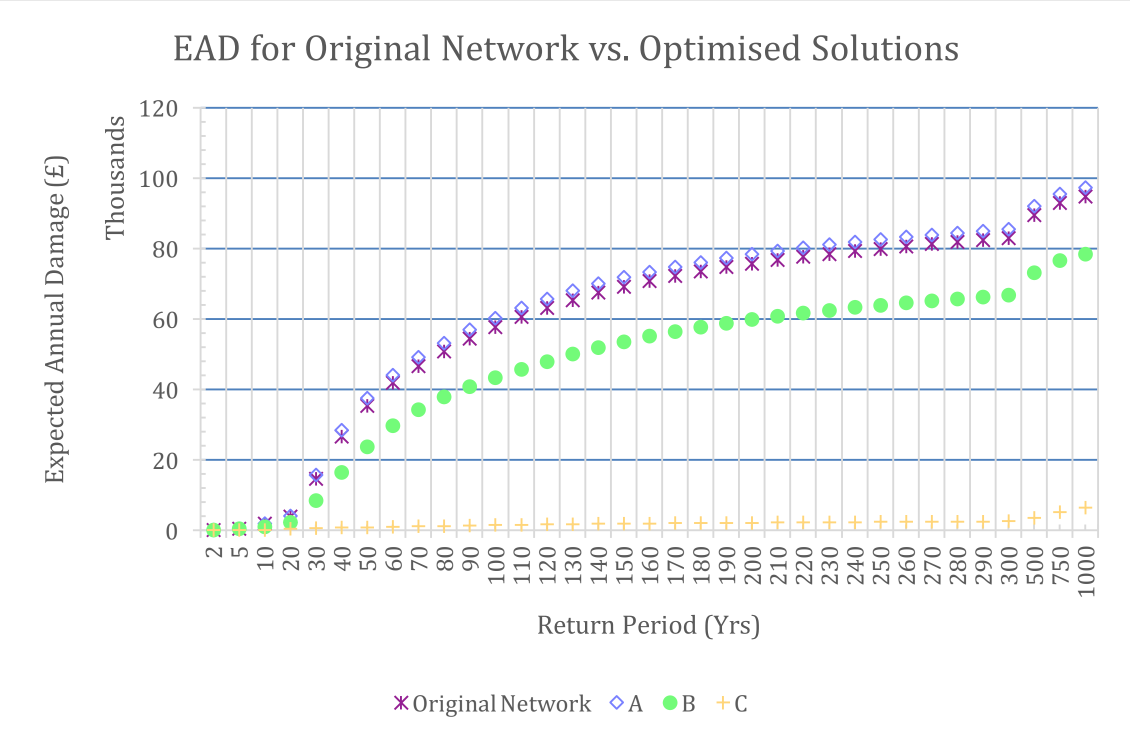


Figure 71 that in fact point 'A', when analysed with the full rainfall set, performs more poorly than the original network. This can probably be attributed to error introduced by the reduced rainfall set. The mean error of the set used in testing was £1,570 and the difference between the two Pareto fronts ('C' and Original) is £2515.46. This optimised network is not identical to the network on which the original error was measured, and so it could be expected that results may become more error prone as that network is moved away from.

The important point to note here is that where significant improvements are identified, they follow through to the full rainfall set. Where minor improvements are identified, they may not follow through to the final set. A better set of rainfall files could potentially reduce the chances of this issue occurring, however, any time that the full rainfall set is not being used, by necessity the optimisation is trading accuracy for speed. These rainfall files do seem to be enough to guide the algorithm, as witnessed by the results with ‘A’ and ‘B’,

	A	**B**	**C**
Average Chamber Area (m^2)	8.25	2.71	2.56
Average Pipe Width (m)	1.13	0.91	0.90

Table 21 - Average Chamber Area and Pipe Width for Optimised Points

6.9 Chapter Summary

This chapter shows that the same algorithm which performs well on WDS test problems, also applies well to our specific flood risk study. It appears to be optimising well, the dominated hypervolume metric is increasing as would be expected. The visual inspection of the Pareto fronts from iteration to iteration shows the progression of the estimated Pareto front well as both objectives are minimised.

Additionally, it has been shown that a point on the produced estimated Pareto front, generated using a reduced rainfall-set, is still better than the baseline case (in terms of EAD) when re-analysed using the full rainfall set.

7. Summary and Conclusions

7.1 Summary

The specific objectives of this thesis (see section 1.2), have all been achieved during the course of this EngD research project, summarised here below.

Specific Objective 1: Identification of a multi-objective optimisation algorithm to utilise as a starting and comparison point for the optimisation process.

This objective was achieved through a thorough investigation of the state of the art multi-objective optimisation algorithms. The investigation was performed and a benchmark algorithm (NSGA-II) was selected on the basis of it showing excellent performance across a number of computationally complex optimisation problems and also being commonly used in research and practice. Additionally, the two main machine-learning approaches that were being considered as a basis for improving performance had built upon NSGA-II.

This choice of NSGA-II was verified by the performance of the algorithm on the set of benchmark problems for the water distribution system design, which are similar in nature to the problem studied in this thesis. The testing on these problems and on the Dalmarnock catchment systems demonstrated the validity of the algorithm choice.

Specific Objective 2: To develop a multi-objective optimisation methodology and implement it through an object-oriented structured software engineering approach with a suitable user-interface, as one of the requirements for this EngD is software that can be further utilised in practice

Once the multi-objective optimisation algorithm had been selected, the software was developed identifying and following the requirements of the industrial partner co-sponsoring this project. This software allowed for the running of our selected multi-objective optimisation algorithm in a modular fashion (which allows for the addition of elements to the code-base with ease, as the modules are loosely linked) and further development and testing of various methodologies investigated in this thesis.

A suitable user interface was then developed for this software (see Appendix III – SAM-Risk Settings, which contains screenshots of this user interface in order to show settings). This user interface met HR Wallingford's requirements and was developed with practitioner's needs in mind. In addition to the utility of the software and the new interface being demonstrated on the Dalmarnock case study in this thesis, the successful achievement of this objective was verified by subsequent application by third-party modellers at HR Wallingford on the EU TRUST project (Boelee and Kellagher, 2015). In this application of the technology, a different approach to reducing rainfall periods needed was used, and the base NSGA-II algorithm was used for optimisation. However, the software used is the software developed for this thesis (with minor modifications).

Specific Objective 3: To formulate the overall optimisation problem for the multi-objective optimisation algorithm that will best describe the drainage system flood risk management problem based on:

a. **Expected annual damage, and**
b. **Capital cost of intervention strategy.**

The previously developed methodology and software (SAM-Risk) was modified in order to improve its user-interface code and make it usable in a number of practical situations. This was replaced by the user-interface software developed in this thesis. In the process the SAM-Risk implementation was developed as a module, which could be run easily by any other software to calculate expected annual damage for an arbitrary drainage system. This was to be utilised for the computation of the first objective function in this thesis. The formulation allows a reduced of set of rainfall duration/return-period events to be used and compared to the full set so that a suitable improvement in computational speed can be achieved without a noticeable loss in accuracy.

The second objective function for the drainage system risk optimisation multi-objective algorithm, was developed to give a cost estimate of the changes being undertaken to the network. It is based upon the cost of the pipes required, plus the cost of excavation for storage, and fixed material costs. The cost calculation is customisable so that scaling can be applied to update this measure and make it appropriate for a particular situation.

Specific Objective 4: To test the objective functions for performance, and investigate methodologies for reducing the computational burden of these functions to allow efficient and effective drainage system flood risk optimisation

Evaluation of the EAD objective function requires a large number of simulation model runs rendering optimisation almost impossible due to excessive times needed to complete it.

To alleviate this, the EAD objective function was modified to cache drainage networks where the network is unchanged between rainfall runs, the solutions were modified to cache objective function scores where they have not been altered between evaluations, and some optimisation of the EAD software such as using LINQ (Pialorsi and Russo, 2007) where appropriate and restructuring iterations to reduce unnecessary complexity. This yielded a performance gain of around 15% for each EAD value calculated, and the caching reduced the time taken for multiple EAD value calculations to a linearly increasing value rather than an exponentially increasing value.

To further improve this performance, the development of a methodology for identifying a reduced rainfall set as undertaken. This methodology allows for EAD to be estimated using a reduced number of rainfall events for a given network, vastly reducing the complexity of calculating EAD. This reduced the runtime to around 45 seconds per EAD calculation, from a value of 5 hours per calculation.

Finally, even a 45 second objective function is too large to allow for a large number of iterations to be completed, so methods were investigated to reduce the number of necessary iterations, or improve the performance of the optimisation.

The LEMMO algorithm (Jourdan et al., 2005), which was originally used for optimisation of WDS design decisions, uses classifier based meta-models to improve computational efficiency of an optimisation algorithm. The original algorithm was based on the use of decision-tree methods as the main classifier. An approach based on employing Artificial Neural Networks (ANN) has been investigated in this thesis and shown to be effective in combination with a multiobjective genetic algorithm. This approach was developed by the author.

The new methodology was applied and tested in depth on a set of WDS test problems that are similar in nature to the problem of drainage system risk optimisation, but require less simulation times, hence are faster to converge to a good Pareto front. Furthermore, a best-known Pareto front is available for each of the test cases (Wang et al., 2014), thus allowing easy evaluation of any optimisation algorithm. The new LEMMO-ANN algorithm was shown to work best with a four-layered ANN achieving a good approximation of the best-known Pareto fronts on a range of WDS problem sizes, ranging from 1.48×10^{9} to 1.00×10^{455}.

Specific Objective 5: To test and verify computational efficiency and effectiveness of the new methodology on a real case study involving drainage system flood risk optimisation.

After the successful test of the new methodology on the WDS design test cases, it was applied to the Dalmarnock test problem, with the EAD objective and the capital expenditure (cost of network alteration) objective. The size and complexity of the problem were such that this particular case study could not be tackled using existing tools. Even with the new LEMMO-ANN methodology, only one long (1000 iterations plus) run on this case study was performed due to excessive run times and computational resources required. Based on this run it was seen that selected solutions from the Pareto front represent a considerable improvement over the base unmodified network which was the optimisation starting point.

7.2 Novel Contributions

The research work undertaken in this thesis resulted in the following key contributions:

1. Developed a novel methodology to optimise urban flood risk management by linking a newly developed LEMMO-ANN optimisation method to a modified version of the flood risk assessment tools and methodology developed by HR Wallingford.

   Modifications to the flood risk assessment tools resulted in improving performance by an average of 15% in terms of time taken. This was achieved by applying more modern features of the C# language, such as LINQ (Pialorsi and Russo, 2007) and refactoring iterative sections of the algorithm.

When this modified toolset is utilised within the NSGA-II LEMMO-ANN algorithm developed within this thesis and combined with a reduced rainfall set (see point 2), the algorithm can complete in a reasonable (i.e. months, not hundreds of years) time frame.

This formulation of the drainage system flood risk problem is unique as no previous study has been able to combine the statistical analysis of flood damage consequences and the cost of drainage system network improvements in a single multi-objective algorithm to identify the trade-off between the two conflicting objectives.

2. Developed a novel methodology to identify a reduced set of rainfall events that can be used by an optimisation to approximate flood risk with enough accuracy to allow for optimisation to take place effectively. This methodology has shown to be effective during testing performed within this thesis, and is flexible enough to be combined with other state of the art efforts in this direction, if it were to be desirable, which may vary depending on the models being used.

   When the output of this methodology is combined with the modified flood risk assessment toolset mentioned in point 1 the overall run time for an evaluation of EAD for Dalmarnock is brought from around 5 hours, to 45 seconds. Whilst this is still a very lengthy time-span for an objective function evaluation within a multi-objective algorithm, the improvements gained are extremely large and represent a significant contribution of this thesis.

3. Developed a novel adaptation of the LEMMO algorithm (di Pierro et al., 2009; Jourdan et al., 2005, 2004) that functions using ANN's as the machine-learning component. This has shown extremely promising results in the tests undertaken during this thesis. As part of the development of this algorithm, a neural network structure has been empirically identified that is effective for the size and complexity of problems that have been tested during this thesis, as part of the newly developed LEMMO-ANN algorithm.

   This LEMMO-ANN algorithm has the potential to be applied both to future problems of the type described within this thesis, and other highly complex optimisation problems with exceptionally large search-spaces. It has been tested and validated for effectiveness against current state of the art baseline algorithms, in order to ensure that the developed algorithm is an overall improvement. Using a large number of runs on WDS test problems where near optimal Pareto fronts are known, the effectiveness of the algorithm was verified before it was applied to the flood risk problem formulated and developed within this thesis.

7.3 Conclusions

From this completed work several conclusions can be drawn. The main is that it is possible to run a full flood risk versus capital expenditure multi-objective optimisation on mainstream desktop computer hardware. This optimisation progresses towards a Pareto front that should aid in identifying networks with an improved performance in terms of EAD vs capital expenditure. The caveat to that is, that it is only possible with appropriate improvements to the state-of-the-art optimisation methodology, including reduction of rainfall set, and use of a cutting edge machine learning and multi-objective optimisation algorithm.

As a product of the above conclusion, it is possible to reduce the rainfall set used to evaluate flood-risk, provided that a specific set of rainfall suitable for the catchment is identified. This can be achieved with only minor loss of accuracy, which allows for this less accurate EAD to be used as a multi-objective optimisation algorithm objective. The end-product of that optimisation can then be checked using the full rainfall set to ensure that full accuracy checks of the lower-accuracy result have occurred.

Additionally, one can conclude that the use of machine-learning based meta-models within optimisation algorithms is highly promising. Both for application to future problems of this nature, and to other highly complex combinatorial type problems. For this particular application, it has been shown to produce good results on test-problems. Improving on a base NSGA-II algorithm both in terms of final output and throughout the algorithms execution, and optimising well and producing good results on a true flood-risk vs. network modification cost case-study.

7.4 Recommendations for Future Work

The first steps to continuing the work outlined within this thesis would be to continue the testing performed. In particular, with more real case-studies to examine how well the LEMMO-ANN combination and reduced rainfall set identification perform on different kinds of urban environment. There is also potential for combining an alternating block hyetograph type method with the methodology outlined here for identification of a reduced rainfall set. The durations could, for example, be combined for each return period and then the methodology applied to identify return periods.

There is some potential in improving the way in which LEMMO is integrated into the NSGA-II base algorithm. It could be possible to develop an approach which would avoid the outcomes seen in Chapter 5, where a poorly performing meta-model within the LEMMO algorithm results in less optimal performance than the base NSGA-II network

This work would be a clear candidate for the application of distributed or high-powered computing due to the highly parallel nature of genetic algorithm based optimisation algorithms. In theory, applying enough computers to this problem could reduce the time taken for each iteration of the algorithm to the time required for one simulation. The downside to experimenting with this kind of approach, is that it would render the software unable to run on a single desktop computer, and would require extensive licensing costs, were Infoworks CS still utilised as part of the software.

There is also the prospect of developing a methodology for identifying a suitable neural network architecture for a given flood-drainage problem. In this research, this was achieved by experimentation on suitable architectures, until one was discovered which performed well. It is, however, possible that other architectures exist which would perform better for this work. There are several artificial neural network training algorithms which build the structure of the neural network in the process of training. Some of these can have a tendency to over-train but this tendency is may be less of an issue with a problem of this complexity, where the problem the ANN is training on is so complex that they are likely to struggle to learn it well.

Another prospect for further research, would be developing the EAD generating methodology and toolset to function with a different flood drainage model. Whilst Infoworks CS is the industry standard (and is therefore trusted and well-recognised in practice), it is not developed with automation in mind. Several problems were experienced that related directly to this software, and performance could potentially be improved by using software that offers an improved application programming interface (API). However, it would be important to rigorously evaluate any potential replacement flood drainage models. There would also be significant amounts of work involved in altering the rest of the flood-risk framework to work well with an alternative drainage model.

Appendices

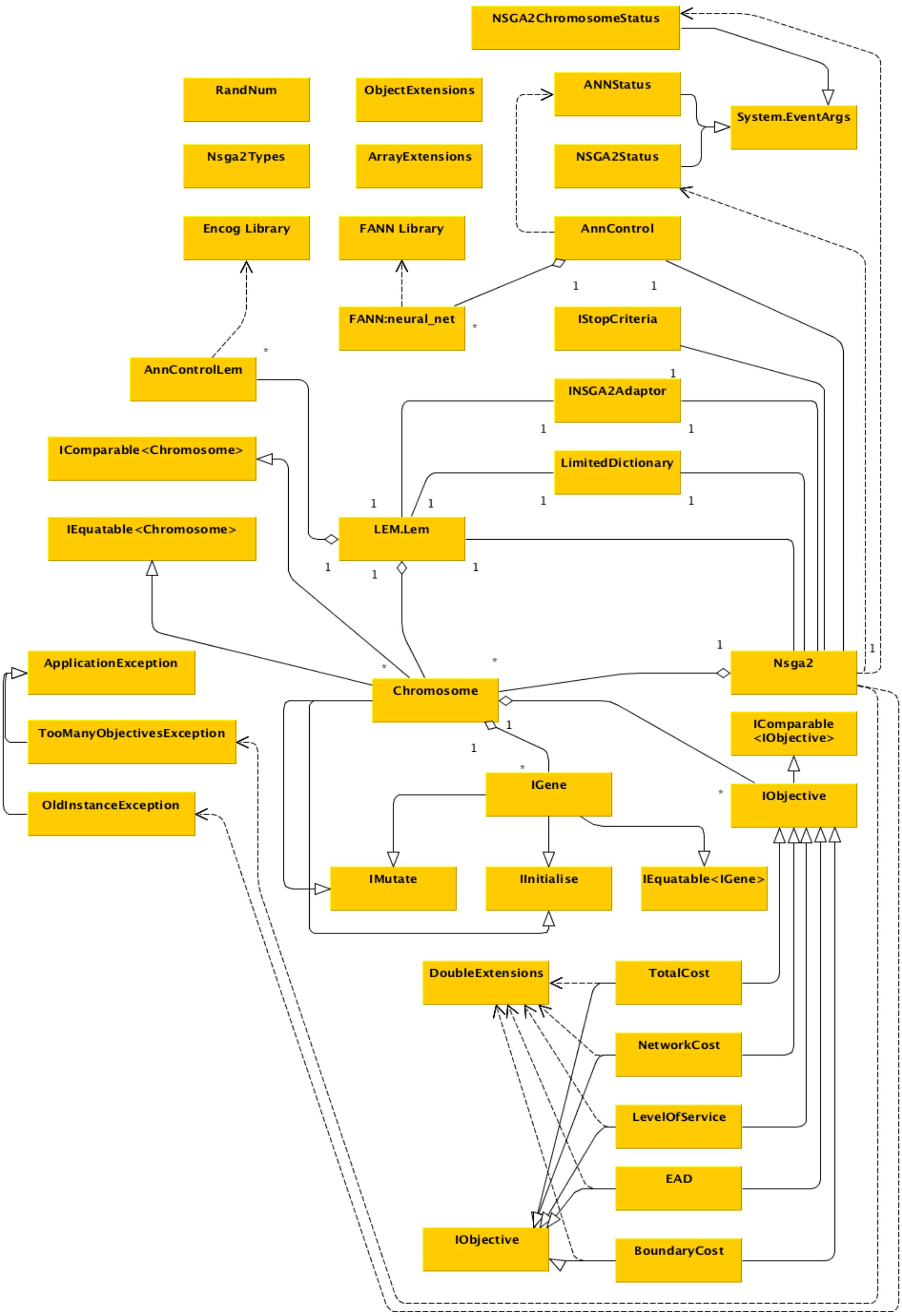
Appendix I – Software Diagrams

This appendix contains class diagrams (without the extreme detail that a fill software engineering design would entail) to illustrate the relationships between the classes developed as part of the ADAPT software.

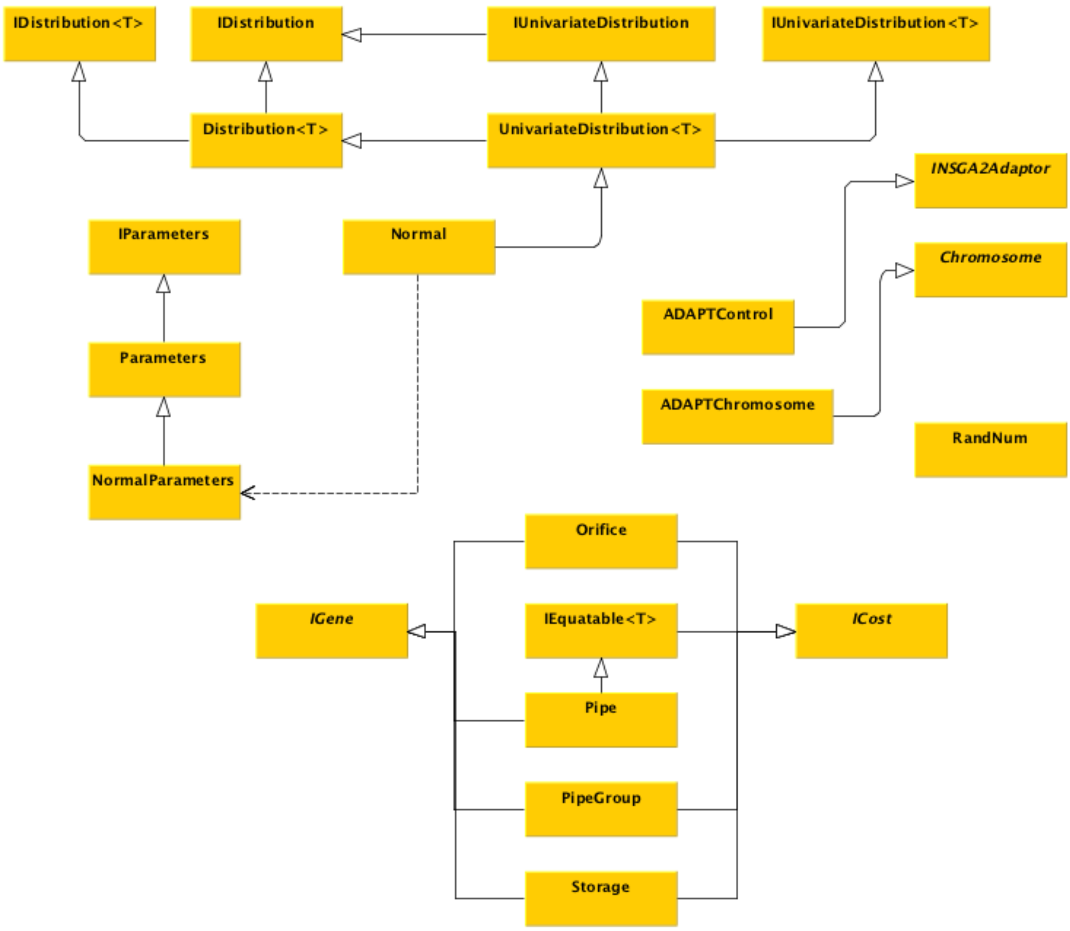
Whilst this software could have been written in a simpler fashion, one of the goals for the software development process was to have a modular and loosely linked arrangement, in order that as much of the code as possible would be available for code re-use by HR Wallingford.

Although there is a large amount of software components that form the ADAPT solution, only three of those are covered here. This is because these are the main three governing functionality of the NSGA-II and LEMMO algorithm during the optimisation performed for this thesis.

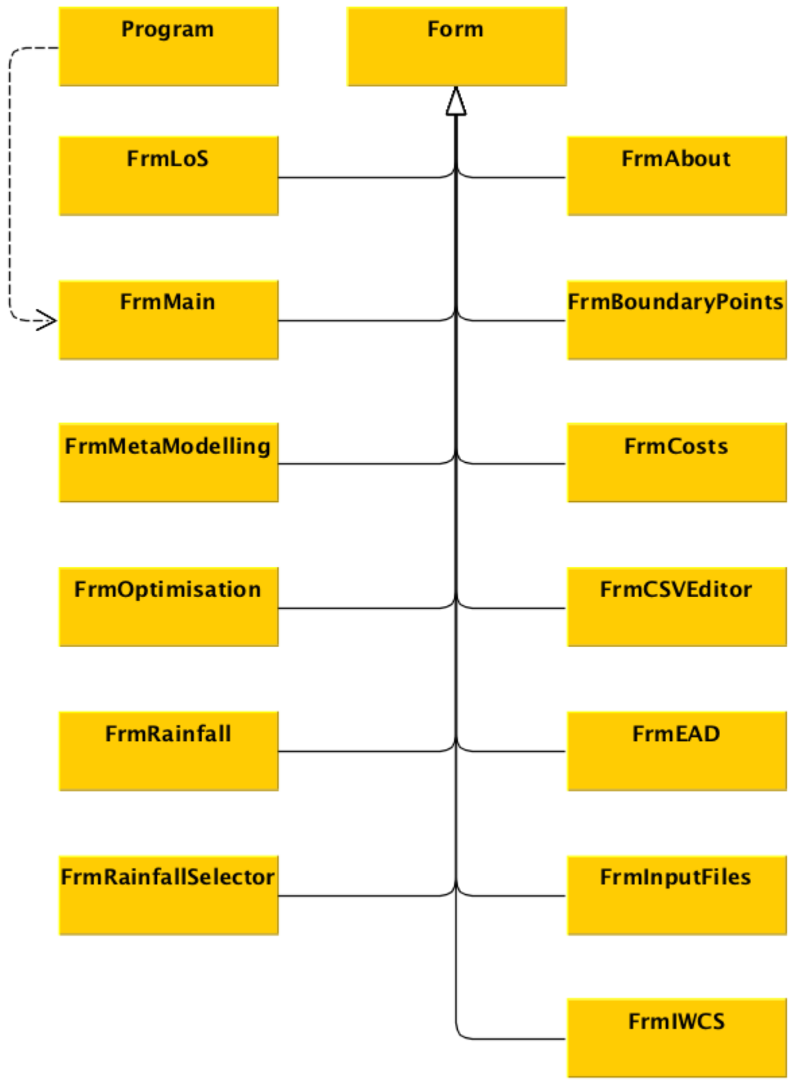
NSGA2CS Class Diagram



ADAPTController Class Diagram



ADAPT User Interface Class Diagram



Appendix II – BIN Data Tables

This appendix contains a selection of the raw data tables associated with the testing performed in this thesis. It is not practical to include all the data, due to the sheer volume.

Presented within this section are, therefore, results from the final iteration from one of the two BIN problems for each test variant plus the averaged results from the metric calculations for BIN.

Additionally, the same sets of results from the Dalmarnock run are also included.

NSGA-II Base Algorithm

These are the results from iteration 2500 of BIN "20141215080454444".

Resilience	Capex
0.402	3.772
0.402	3.772
0.405	3.811
0.406	3.858
0.410	3.901
0.414	3.982
0.414	4.011
0.415	4.036
0.415	4.068
0.476	4.084
0.489	4.180
0.489	4.180
0.489	4.180
0.493	4.252
0.493	4.266
0.499	4.338
0.499	4.338
0.552	4.387
0.552	4.387
0.582	4.529
0.582	4.529
0.584	4.592
0.584	4.592
0.615	4.666
0.615	4.666
0.617	4.743
0.617	4.748
0.617	4.748
0.626	4.852
0.629	4.926
0.629	4.926
0.633	5.027
0.633	5.027
0.634	5.040
0.643	5.137
0.645	5.141
0.705	5.216
0.705	5.239
0.705	5.239
0.705	5.239
0.716	5.420
0.717	5.442
0.717	5.451
0.723	5.559
0.723	5.563
0.727	5.657
0.739	5.671
0.739	5.671
0.743	5.780
0.743	5.780
0.747	5.843
0.747	5.843
0.757	6.048
0.757	6.048
0.757	6.048
0.757	6.048
0.757	6.066
0.767	6.096
0.767	6.096
0.768	6.140
0.768	6.140
0.768	6.178
0.779	6.213
0.779	6.300
0.782	6.356
0.797	6.443
0.797	6.500
0.799	6.512
0.799	6.512
0.799	6.512
0.800	6.547
0.801	6.616
0.801	6.646
0.801	6.646
0.812	6.786
0.812	6.786
0.813	6.790
0.813	6.790
0.814	6.909
0.816	6.918
0.816	6.918
0.816	7.011
0.816	7.079
0.816	7.079
0.829	7.176
0.833	7.243
0.833	7.243
0.836	7.315
0.836	7.315
0.841	7.348
0.842	7.627
0.842	7.630
0.842	7.630
0.842	7.630
0.842	7.819
0.843	7.850
0.847	7.945
0.847	7.945
0.847	7.945
0.847	7.945
0.849	8.222
0.849	8.222
0.849	8.300
0.849	8.316

0.850	8.387
0.850	8.393
0.858	8.430
0.859	8.461
0.859	8.473
0.860	8.494
0.860	8.682
0.860	8.682
0.860	8.705
0.861	8.712
0.868	8.846
0.868	8.865
0.871	8.935
0.871	8.937
0.872	8.955
0.872	8.955
0.873	9.029
0.875	9.060
0.875	9.060
0.876	9.213
0.876	9.213
0.878	9.265
0.878	9.265
0.878	9.750
0.878	9.767
0.878	9.849
0.879	9.904
0.879	9.904
0.879	10.044
0.879	10.044
0.879	10.099
0.880	10.189
0.880	10.189
0.882	10.257
0.882	10.262
0.883	10.269
0.883	10.269
0.883	10.269
0.883	10.414
0.883	10.414

0.883	10.414
0.883	10.467
0.883	10.467
0.883	11.937
0.883	11.937
0.883	11.937
0.883	11.940
0.883	11.994
0.885	12.051
0.885	12.065
0.885	12.085
0.885	12.085
0.886	12.180
0.886	12.183
0.886	12.326
0.889	12.363
0.889	12.490
0.889	12.490
0.891	12.578
0.892	12.657
0.892	12.657
0.892	12.657
0.892	12.657
0.892	12.749
0.892	12.749
0.892	12.786
0.892	12.786
0.892	12.786
0.893	12.808
0.893	13.054
0.894	13.088
0.894	13.088
0.895	13.166
0.899	13.234
0.899	13.234
0.900	13.472
0.900	13.484
0.900	13.557
0.900	13.573
0.900	13.573

0.900	13.658
0.900	13.658
0.901	13.738
0.902	13.745
0.903	13.844
0.903	13.862
0.904	13.871
0.904	13.875
0.904	13.875
0.905	14.115
0.906	14.190
0.906	14.190
0.906	14.263
0.908	14.313
0.909	14.336
0.910	14.418
0.910	14.425
0.910	14.505
0.911	14.552
0.911	14.552
0.911	15.018
0.911	15.018
0.911	15.018
0.911	15.026
0.911	15.084
0.912	15.171
0.912	15.171
0.913	15.255
0.914	15.263
0.914	15.263
0.914	15.338
0.914	15.403
0.914	15.408
0.914	15.428
0.915	15.455
0.916	15.548
0.916	15.548
0.916	15.548
0.916	15.548
0.916	15.619

0.917	15.682
0.917	15.696
0.917	15.696
0.917	15.696
0.918	15.865
0.918	15.865
0.918	15.865
0.918	15.865
0.918	15.873
0.918	15.988
0.919	16.019
0.920	16.111
0.920	16.111
0.920	16.111
0.920	16.161
0.921	16.183
0.921	16.183
0.921	16.183
0.921	16.246
0.921	16.246
0.922	16.266
0.922	16.338
0.922	16.338
0.923	16.356
0.923	16.440
0.924	16.489
0.924	16.517
0.925	16.548
0.925	16.641

0.925	16.709
0.925	16.712
0.925	16.782
0.925	16.794
0.925	16.863
0.926	16.873
0.926	16.944
0.927	16.969
0.927	16.990
0.928	17.038
0.928	17.038
0.928	17.624
0.928	17.624
0.928	17.645
0.928	17.691
0.928	17.700
0.928	17.773
0.928	17.773
0.929	17.888
0.931	17.954
0.931	18.035
0.932	18.075
0.933	18.154
0.933	18.235
0.935	18.304
0.935	18.304
0.935	18.630
0.935	18.630
0.935	18.674

0.936	18.711
0.936	18.837
0.936	18.837
0.936	18.837
0.936	18.910
0.936	18.910
0.936	18.982
0.937	19.000
0.937	19.024
0.937	19.152
0.937	19.163
0.938	19.256
0.938	19.256
0.938	19.314
0.938	19.314
0.938	19.388
0.938	19.457
0.939	19.466
0.939	19.548
0.939	19.548
0.939	19.548
0.939	19.548
0.940	19.607
0.940	19.618
0.940	19.639
0.940	19.639
0.940	19.639

NSGA-II with LEMMO and Initial ANN Structure

These are the results from iteration 2500 of BIN “2015030420295674”.

Resilience	Capex
0.564	4.304
0.568	4.307
0.569	4.318
0.572	4.328
0.573	4.330
0.576	4.335
0.577	4.336
0.579	4.344
0.588	4.359
0.591	4.375
0.594	4.384
0.594	4.395
0.607	4.405
0.610	4.428
0.613	4.438
0.613	4.446
0.614	4.456
0.621	4.459
0.622	4.467
0.631	4.468
0.633	4.482
0.634	4.494
0.638	4.496
0.640	4.502
0.643	4.507
0.645	4.524
0.647	4.539
0.654	4.569
0.656	4.578
0.657	4.611
0.661	4.650
0.663	4.651
0.674	4.665
0.677	4.688
0.677	4.690
0.679	4.728
0.682	4.748
0.687	4.755
0.689	4.771
0.691	4.792
0.694	4.853
0.696	4.890
0.697	4.922
0.698	4.925
0.701	4.931
0.703	4.934
0.706	4.952
0.707	5.004
0.709	5.009
0.712	5.061
0.714	5.083
0.718	5.113
0.720	5.135
0.722	5.161
0.731	5.167
0.731	5.178
0.734	5.211
0.735	5.224
0.739	5.249
0.742	5.323
0.744	5.333
0.744	5.345
0.747	5.436
0.747	5.499
0.750	5.510
0.750	5.517
0.754	5.521
0.758	5.579
0.762	5.592
0.772	5.601
0.774	5.610
0.776	5.624
0.779	5.642
0.782	5.650
0.784	5.675
0.786	5.678
0.787	5.689
0.789	5.704
0.791	5.729
0.792	5.755
0.795	5.761
0.796	5.770
0.798	5.792
0.800	5.792
0.801	5.805
0.804	5.805
0.808	5.813
0.808	5.813
0.810	5.842
0.810	5.877
0.812	5.913
0.814	5.922
0.816	5.926
0.817	5.952
0.819	5.987
0.821	6.071
0.822	6.112
0.823	6.163
0.823	6.211
0.825	6.233
0.826	6.287
0.827	6.340
0.829	6.387
0.832	6.402

0.833	6.439
0.834	6.547
0.836	6.578
0.837	6.595
0.839	6.644
0.840	6.716
0.841	6.756
0.844	6.764
0.845	6.785
0.846	6.795
0.849	6.817
0.850	6.848
0.852	6.889
0.852	6.889
0.852	6.889
0.852	6.997
0.854	7.080
0.855	7.137
0.858	7.178
0.861	7.185
0.862	7.289
0.863	7.348
0.864	7.362
0.864	7.362
0.864	7.362
0.864	7.801
0.865	7.837
0.865	7.837
0.869	7.856
0.870	7.887
0.870	7.894
0.870	7.894
0.870	7.894
0.870	7.894
0.872	8.043
0.874	8.085
0.875	8.136

0.877	8.161
0.879	8.191
0.880	8.241
0.883	8.263
0.885	8.291
0.887	8.326
0.887	8.332
0.887	8.432
0.889	8.463
0.891	8.481
0.893	8.500
0.895	8.611
0.896	8.658
0.897	8.746
0.897	8.767
0.897	11.060
0.899	11.101
0.899	11.106
0.901	11.134
0.902	11.189
0.902	11.194
0.904	11.245
0.905	11.295
0.906	11.356
0.906	11.399
0.907	11.413
0.907	11.413
0.907	12.033
0.908	12.101
0.908	12.126
0.908	12.328
0.909	12.363
0.910	12.400
0.911	13.674
0.912	13.703
0.913	13.766
0.914	13.838

0.914	13.838
0.914	16.374
0.914	16.417
0.915	16.423
0.915	16.423
0.915	16.468
0.915	16.468
0.916	17.042
0.917	17.062
0.917	17.152
0.918	17.157
0.918	17.233
0.918	17.311
0.918	17.311
0.919	18.358
0.922	18.401
0.923	18.454
0.924	18.529
0.925	18.592
0.925	18.691
0.926	18.695
0.926	18.831
0.927	18.962
0.928	18.971
0.929	19.023
0.930	19.136
0.930	19.158
0.931	19.259
0.932	19.272
0.933	19.309
0.934	19.381
0.935	19.498
0.935	19.502
0.936	19.564
0.936	19.620
0.937	19.645

NSGA-II with LEMMO and Final ANN Structure

These are the results from iteration 2500 of BIN “2015032511374346”.

Resilience	Capex
0.495	2.729
0.496	2.736
0.502	2.745
0.502	2.745
0.507	2.752
0.507	2.752
0.507	2.752
0.508	2.753
0.513	2.756
0.516	2.780
0.520	2.783
0.520	2.785
0.522	2.793
0.524	2.794
0.538	2.798
0.542	2.804
0.542	2.804
0.542	2.804
0.542	2.804
0.549	2.812
0.549	2.817
0.550	2.827
0.554	2.827
0.554	2.827
0.554	2.828
0.559	2.835
0.559	2.835
0.563	2.835
0.563	2.840
0.567	2.845
0.567	2.845
0.571	2.854
0.576	2.867
0.579	2.874
0.579	2.874
0.584	2.875
0.584	2.883
0.588	2.886
0.589	2.889
0.589	2.889
0.593	2.902
0.595	2.905
0.597	2.914
0.599	2.921
0.603	2.923
0.603	2.923
0.603	2.923
0.609	2.927
0.611	2.930
0.614	2.961
0.614	2.961
0.614	2.961
0.614	2.961
0.618	2.961
0.620	2.966
0.625	2.968
0.626	2.986
0.631	2.990
0.631	2.991
0.636	3.006
0.638	3.015
0.648	3.022
0.649	3.023
0.654	3.027
0.655	3.029
0.657	3.037
0.660	3.063
0.660	3.063
0.665	3.063
0.666	3.070
0.666	3.070
0.666	3.070
0.670	3.071
0.670	3.072
0.675	3.094
0.675	3.095
0.679	3.114
0.679	3.121
0.683	3.135
0.683	3.135
0.686	3.141
0.689	3.159
0.689	3.162
0.689	3.162
0.693	3.162
0.693	3.162
0.693	3.162
0.697	3.204
0.698	3.205
0.700	3.206
0.703	3.214
0.704	3.215
0.704	3.215
0.708	3.216
0.709	3.225
0.712	3.259
0.715	3.277
0.715	3.277
0.719	3.314
0.719	3.314
0.725	3.350
0.725	3.350
0.726	3.352
0.727	3.360

0.727	3.360
0.730	3.384
0.734	3.385
0.735	3.389
0.735	3.389
0.738	3.457
0.743	3.478
0.743	3.478
0.744	3.481
0.745	3.481
0.750	3.500
0.750	3.500
0.750	3.500
0.754	3.526
0.754	3.526
0.757	3.551
0.757	3.551
0.759	3.566
0.759	3.566
0.759	3.566
0.762	3.658
0.762	3.660
0.762	3.662
0.762	3.662
0.768	3.703
0.771	3.712
0.771	3.712
0.775	3.717
0.775	3.717
0.775	3.717
0.778	3.766
0.782	3.776
0.782	3.778
0.783	3.950
0.783	3.958
0.784	3.961
0.786	3.993
0.788	4.000
0.790	4.032
0.790	4.034

0.790	4.034
0.790	4.034
0.801	4.053
0.801	4.053
0.801	4.053
0.801	4.059
0.802	4.075
0.803	4.077
0.805	4.137
0.805	4.137
0.809	4.160
0.809	4.160
0.812	4.196
0.813	4.212
0.816	4.226
0.816	4.226
0.817	4.359
0.819	4.363
0.819	4.363
0.819	4.363
0.822	4.391
0.822	4.393
0.822	4.393
0.825	4.410
0.826	4.412
0.828	4.442
0.830	4.453
0.830	4.453
0.831	4.457
0.834	4.486
0.834	4.486
0.837	4.498
0.837	4.498
0.841	4.522
0.842	4.657
0.844	4.658
0.844	4.662
0.844	4.662
0.847	4.755
0.848	4.777

0.848	4.777
0.850	4.836
0.851	4.838
0.851	4.839
0.853	4.887
0.854	4.895
0.855	5.000
0.855	5.004
0.858	5.097
0.858	5.097
0.858	5.097
0.863	5.121
0.863	5.121
0.865	5.179
0.865	5.179
0.866	5.320
0.866	5.320
0.866	5.323
0.869	5.428
0.871	5.448
0.871	5.448
0.873	5.508
0.873	5.508
0.873	5.508
0.874	5.780
0.874	5.781
0.875	5.783
0.879	5.837
0.879	5.837
0.879	5.837
0.880	5.924
0.881	5.927
0.882	5.964
0.884	5.966
0.888	5.985
0.889	5.987
0.892	6.071
0.895	6.167
0.896	6.182
0.896	6.182

0.899	6.208
0.899	6.482
0.899	6.482
0.899	6.580
0.900	6.583
0.900	6.585
0.900	6.585
0.901	6.654
0.903	6.660
0.904	6.668
0.904	6.668
0.905	6.783
0.905	6.785
0.905	6.785
0.905	6.785
0.905	6.787
0.908	6.817
0.908	6.840
0.909	6.899
0.911	7.032
0.911	7.032
0.911	7.032
0.911	7.032
0.913	7.055
0.913	7.055
0.915	7.247
0.915	7.247
0.915	7.247
0.916	7.262

0.916	7.262
0.916	7.262
0.916	8.603
0.917	8.631
0.918	8.805
0.918	8.805
0.918	8.805
0.918	8.805
0.919	8.893
0.921	8.936
0.926	8.940
0.926	8.941
0.927	9.080
0.927	9.080
0.928	9.111
0.928	9.288
0.928	9.314
0.928	9.314
0.929	9.362
0.931	9.388
0.931	9.388
0.931	9.388
0.931	9.388
0.931	9.562
0.932	9.595
0.933	9.604
0.933	9.616
0.933	12.393
0.933	12.393

0.935	12.493
0.936	12.562
0.936	12.566
0.937	12.578
0.937	12.733
0.938	12.812
0.940	12.830
0.940	12.830
0.940	12.830
0.940	13.023
0.940	13.023
0.941	13.063
0.941	13.063
0.941	18.777
0.941	18.777
0.941	18.777
0.941	18.803
0.942	18.919
0.942	18.919
0.942	19.150
0.942	19.162
0.942	19.302
0.942	19.302
0.943	19.355
0.943	19.440
0.943	19.440
0.943	19.440

NSGA-II Base Algorithm Analysis Metric Results

Convergence	Diversity	Dominated Hypervolume	Iteration
11.040	5.787	8.290	10.000
10.355	2.243	9.081	20.000
9.781	0.947	9.671	30.000
9.377	0.617	10.064	40.000
9.099	0.451	10.341	50.000
8.835	0.404	10.609	60.000
8.584	0.407	10.859	70.000
8.338	0.388	11.109	80.000
8.115	0.381	11.324	90.000
7.851	0.388	11.575	100.000
7.637	0.376	11.782	110.000
7.442	0.365	11.966	120.000
7.286	0.369	12.121	130.000
7.108	0.376	12.295	140.000
6.961	0.367	12.441	150.000
6.829	0.366	12.572	160.000
6.705	0.370	12.696	170.000
6.581	0.373	12.818	180.000
6.415	0.367	12.971	190.000
6.278	0.366	13.104	200.000
6.143	0.367	13.235	210.000
6.018	0.374	13.358	220.000
5.907	0.364	13.461	230.000
5.790	0.363	13.572	240.000
5.687	0.368	13.669	250.000
5.597	0.363	13.758	260.000
5.487	0.372	13.857	270.000
5.408	0.382	13.939	280.000
5.319	0.366	14.022	290.000
5.232	0.371	14.103	300.000
5.136	0.375	14.190	310.000
5.069	0.376	14.260	320.000
4.999	0.372	14.326	330.000
4.927	0.369	14.395	340.000
4.848	0.385	14.469	350.000
4.767	0.384	14.542	360.000

4.692	0.387	14.608	370.000
4.612	0.371	14.686	380.000
4.537	0.383	14.754	390.000
4.470	0.377	14.817	400.000
4.418	0.381	14.869	410.000
4.358	0.388	14.926	420.000
4.291	0.378	14.991	430.000
4.227	0.388	15.048	440.000
4.169	0.382	15.105	450.000
4.099	0.383	15.166	460.000
4.037	0.385	15.223	470.000
3.979	0.382	15.278	480.000
3.932	0.391	15.326	490.000
3.872	0.391	15.379	500.000
3.832	0.381	15.423	510.000
3.786	0.390	15.470	520.000
3.739	0.388	15.516	530.000
3.680	0.397	15.573	540.000
3.637	0.399	15.614	550.000
3.576	0.394	15.666	560.000
3.509	0.396	15.725	570.000
3.463	0.391	15.770	580.000
3.411	0.397	15.819	590.000
3.362	0.392	15.865	600.000
3.325	0.397	15.904	610.000
3.281	0.398	15.947	620.000
3.236	0.394	15.991	630.000
3.210	0.394	16.023	640.000
3.166	0.398	16.065	650.000
3.118	0.396	16.108	660.000
3.080	0.397	16.143	670.000
3.031	0.395	16.187	680.000
3.002	0.406	16.217	690.000
2.973	0.400	16.246	700.000
2.951	0.399	16.290	710.000
2.925	0.403	16.317	720.000
2.890	0.400	16.353	730.000
2.850	0.400	16.391	740.000
2.816	0.400	16.422	750.000
2.782	0.402	16.455	760.000

2.758	0.397	16.483	770.000
2.718	0.402	16.518	780.000
2.677	0.403	16.555	790.000
2.645	0.392	16.586	800.000
2.605	0.399	16.623	810.000
2.578	0.403	16.652	820.000
2.550	0.402	16.684	830.000
2.513	0.404	16.718	840.000
2.488	0.405	16.745	850.000
2.465	0.406	16.769	860.000
2.443	0.407	16.792	870.000
2.417	0.405	16.821	880.000
2.397	0.405	16.846	890.000
2.367	0.411	16.875	900.000
2.346	0.403	16.897	910.000
2.324	0.404	16.921	920.000
2.293	0.404	16.952	930.000
2.268	0.412	16.980	940.000
2.250	0.415	17.002	950.000
2.228	0.410	17.029	960.000
2.209	0.407	17.051	970.000
2.185	0.412	17.078	980.000
2.164	0.408	17.101	990.000
2.143	0.413	17.121	1,000.000
2.123	0.416	17.142	1,010.000
2.106	0.415	17.163	1,020.000
2.082	0.415	17.185	1,030.000
2.062	0.415	17.208	1,040.000
2.041	0.417	17.228	1,050.000
2.026	0.417	17.246	1,060.000
2.012	0.417	17.263	1,070.000
2.000	0.413	17.280	1,080.000
1.977	0.413	17.304	1,090.000
1.957	0.413	17.326	1,100.000
1.937	0.422	17.344	1,110.000
1.919	0.422	17.363	1,120.000
1.902	0.422	17.384	1,130.000
1.887	0.413	17.402	1,140.000
1.870	0.419	17.421	1,150.000
1.856	0.424	17.439	1,160.000

1.840	0.413	17.457	1,170.000
1.824	0.411	17.475	1,180.000
1.812	0.414	17.493	1,190.000
1.795	0.418	17.510	1,200.000
1.781	0.429	17.528	1,210.000
1.767	0.417	17.547	1,220.000
1.753	0.413	17.564	1,230.000
1.736	0.415	17.583	1,240.000
1.724	0.410	17.600	1,250.000
1.708	0.415	17.618	1,260.000
1.691	0.419	17.637	1,270.000
1.679	0.418	17.654	1,280.000
1.669	0.423	17.668	1,290.000
1.653	0.415	17.687	1,300.000
1.638	0.423	17.704	1,310.000
1.625	0.421	17.720	1,320.000
1.617	0.424	17.732	1,330.000
1.608	0.419	17.747	1,340.000
1.594	0.424	17.762	1,350.000
1.582	0.417	17.778	1,360.000
1.567	0.419	17.794	1,370.000
1.553	0.421	17.812	1,380.000
1.543	0.419	17.827	1,390.000
1.532	0.430	17.840	1,400.000
1.526	0.428	17.850	1,410.000
1.517	0.425	17.862	1,420.000
1.507	0.428	17.875	1,430.000
1.495	0.423	17.890	1,440.000
1.483	0.428	17.904	1,450.000
1.472	0.415	17.917	1,460.000
1.460	0.420	17.931	1,470.000
1.452	0.422	17.941	1,480.000
1.443	0.424	17.953	1,490.000
1.435	0.431	17.966	1,500.000
1.425	0.428	17.979	1,510.000
1.411	0.430	17.994	1,520.000
1.405	0.435	18.003	1,530.000
1.397	0.425	18.015	1,540.000
1.388	0.431	18.029	1,550.000
1.380	0.427	18.041	1,560.000

1.373	0.421	18.052	1,570.000
1.366	0.424	18.065	1,580.000
1.359	0.429	18.077	1,590.000
1.350	0.426	18.088	1,600.000
1.340	0.424	18.100	1,610.000
1.332	0.421	18.111	1,620.000
1.323	0.426	18.123	1,630.000
1.312	0.423	18.135	1,640.000
1.303	0.430	18.148	1,650.000
1.294	0.419	18.161	1,660.000
1.289	0.435	18.169	1,670.000
1.283	0.428	18.178	1,680.000
1.276	0.432	18.189	1,690.000
1.269	0.430	18.201	1,700.000
1.261	0.425	18.211	1,710.000
1.253	0.431	18.222	1,720.000
1.248	0.430	18.232	1,730.000
1.242	0.422	18.243	1,740.000
1.233	0.434	18.254	1,750.000
1.228	0.427	18.263	1,760.000
1.222	0.426	18.273	1,770.000
1.217	0.431	18.281	1,780.000
1.209	0.434	18.291	1,790.000
1.207	0.436	18.299	1,800.000
1.198	0.426	18.312	1,810.000
1.191	0.433	18.322	1,820.000
1.185	0.431	18.331	1,830.000
1.179	0.425	18.340	1,840.000
1.172	0.435	18.350	1,850.000
1.164	0.431	18.361	1,860.000
1.158	0.440	18.369	1,870.000
1.152	0.431	18.378	1,880.000
1.146	0.433	18.386	1,890.000
1.141	0.443	18.394	1,900.000
1.134	0.431	18.404	1,910.000
1.129	0.432	18.412	1,920.000
1.123	0.438	18.418	1,930.000
1.116	0.431	18.430	1,940.000
1.111	0.432	18.440	1,950.000
1.104	0.440	18.450	1,960.000

1.097	0.430	18.460	1,970.000
1.091	0.434	18.469	1,980.000
1.090	0.436	18.475	1,990.000
1.085	0.434	18.482	2,000.000
1.080	0.432	18.490	2,010.000
1.073	0.437	18.501	2,020.000
1.068	0.436	18.507	2,030.000
1.064	0.441	18.514	2,040.000
1.059	0.429	18.522	2,050.000
1.054	0.440	18.531	2,060.000
1.049	0.432	18.539	2,070.000
1.047	0.429	18.545	2,080.000
1.044	0.436	18.552	2,090.000
1.039	0.431	18.561	2,100.000
1.034	0.432	18.569	2,110.000
1.030	0.429	18.578	2,120.000
1.026	0.433	18.586	2,130.000
1.021	0.440	18.594	2,140.000
1.018	0.431	18.600	2,150.000
1.014	0.442	18.607	2,160.000
1.008	0.426	18.617	2,170.000
1.006	0.444	18.623	2,180.000
1.002	0.433	18.630	2,190.000
0.998	0.443	18.637	2,200.000
0.994	0.432	18.643	2,210.000
0.993	0.434	18.648	2,220.000
0.988	0.430	18.656	2,230.000
0.983	0.429	18.664	2,240.000
0.982	0.433	18.670	2,250.000
0.978	0.435	18.675	2,260.000
0.976	0.436	18.681	2,270.000
0.973	0.433	18.686	2,280.000
0.969	0.427	18.693	2,290.000
0.964	0.433	18.700	2,300.000
0.959	0.435	18.706	2,310.000
0.957	0.433	18.712	2,320.000
0.955	0.428	18.717	2,330.000
0.951	0.425	18.725	2,340.000
0.946	0.444	18.730	2,350.000
0.941	0.428	18.737	2,360.000

0.938	0.435	18.742	2,370.000
0.934	0.438	18.750	2,380.000
0.928	0.434	18.758	2,390.000
0.926	0.440	18.763	2,400.000
0.923	0.438	18.769	2,410.000
0.920	0.435	18.775	2,420.000
0.916	0.433	18.781	2,430.000
0.914	0.435	18.787	2,440.000
0.911	0.433	18.793	2,450.000
0.908	0.442	18.798	2,460.000
0.904	0.442	18.806	2,470.000
0.901	0.433	18.811	2,480.000
0.898	0.434	18.817	2,490.000
0.896	0.450	18.822	2,500.000

NSGA-II with LEMMO and Initial ANN Structure Analysis Metric Results

Convergence	Diversity	Dominated Hypervolume	Iteration
9.880	32.720	9.232	10.000
8.093	11.739	11.090	20.000
6.471	5.214	12.627	30.000
5.704	3.047	13.356	40.000
5.212	2.602	13.826	50.000
4.864	2.321	14.165	60.000
4.436	2.231	14.571	70.000
4.178	2.167	14.828	80.000
4.021	2.249	14.987	90.000
3.845	2.199	15.167	100.000
3.665	2.175	15.365	110.000
3.568	2.166	15.474	120.000
3.447	2.150	15.619	130.000
3.309	2.163	15.753	140.000
3.212	2.158	15.856	150.000
3.132	2.156	15.944	160.000
3.029	2.155	16.052	170.000
2.968	2.183	16.125	180.000
2.862	2.200	16.227	190.000
2.820	2.132	16.279	200.000
2.734	2.155	16.371	210.000
2.680	2.137	16.430	220.000
2.621	2.172	16.497	230.000
2.574	2.221	16.553	240.000
2.515	2.190	16.615	250.000
2.462	2.167	16.670	260.000
2.412	2.175	16.724	270.000
2.378	2.182	16.762	280.000
2.353	2.211	16.797	290.000
2.328	2.264	16.829	300.000
2.302	2.226	16.858	310.000
2.284	2.213	16.888	320.000
2.260	2.236	16.917	330.000
2.241	2.235	16.948	340.000

2.211	2.238	16.985	350.000
2.179	2.254	17.024	360.000
2.164	2.238	17.050	370.000
2.125	2.261	17.092	380.000
2.102	2.195	17.121	390.000
2.087	2.228	17.145	400.000
2.072	2.200	17.169	410.000
2.056	2.241	17.192	420.000
2.029	2.247	17.219	430.000
2.011	2.287	17.247	440.000
1.990	2.279	17.270	450.000
1.980	2.255	17.291	460.000
1.974	2.258	17.306	470.000
1.966	2.268	17.322	480.000
1.957	2.283	17.337	490.000
1.947	2.290	17.356	500.000
1.940	2.279	17.371	510.000
1.935	2.278	17.383	520.000
1.923	2.245	17.400	530.000
1.906	2.308	17.422	540.000
1.893	2.303	17.440	550.000
1.880	2.290	17.457	560.000
1.873	2.293	17.470	570.000
1.856	2.284	17.489	580.000
1.841	2.317	17.510	590.000
1.830	2.289	17.525	600.000
1.820	2.321	17.540	610.000
1.814	2.338	17.551	620.000
1.808	2.302	17.560	630.000
1.801	2.337	17.571	640.000
1.796	2.313	17.581	650.000
1.791	2.329	17.590	660.000
1.774	2.355	17.607	670.000
1.770	2.323	17.617	680.000
1.764	2.339	17.628	690.000
1.756	2.342	17.639	700.000
1.749	2.353	17.649	710.000
1.739	2.352	17.662	720.000
1.730	2.341	17.672	730.000
1.719	2.367	17.689	740.000

1.705	2.364	17.706	750.000
1.693	2.323	17.722	760.000
1.678	2.342	17.741	770.000
1.668	2.314	17.758	780.000
1.663	2.362	17.772	790.000
1.655	2.345	17.786	800.000
1.647	2.300	17.798	810.000
1.640	2.364	17.809	820.000
1.634	2.332	17.818	830.000
1.628	2.396	17.827	840.000
1.616	2.375	17.841	850.000
1.609	2.355	17.852	860.000
1.604	2.373	17.860	870.000
1.603	2.409	17.866	880.000
1.598	2.374	17.877	890.000
1.587	2.375	17.892	900.000
1.579	2.356	17.903	910.000
1.576	2.365	17.911	920.000
1.571	2.393	17.922	930.000
1.564	2.362	17.932	940.000
1.559	2.379	17.941	950.000
1.553	2.371	17.951	960.000
1.551	2.398	17.960	970.000
1.547	2.371	17.967	980.000
1.543	2.394	17.974	990.000
1.535	2.392	17.983	1,000.000
1.529	2.405	17.991	1,010.000
1.524	2.395	17.999	1,020.000
1.519	2.404	18.009	1,030.000
1.513	2.380	18.020	1,040.000
1.508	2.397	18.030	1,050.000
1.505	2.397	18.035	1,060.000
1.501	2.396	18.043	1,070.000
1.496	2.400	18.050	1,080.000
1.489	2.427	18.059	1,090.000
1.485	2.403	18.067	1,100.000
1.478	2.387	18.077	1,110.000
1.473	2.418	18.084	1,120.000
1.468	2.419	18.092	1,130.000
1.463	2.421	18.100	1,140.000

1.457	2.381	18.112	1,150.000
1.453	2.367	18.121	1,160.000
1.448	2.431	18.128	1,170.000
1.446	2.393	18.135	1,180.000
1.441	2.395	18.141	1,190.000
1.436	2.401	18.148	1,200.000
1.428	2.440	18.159	1,210.000
1.425	2.400	18.166	1,220.000
1.422	2.427	18.172	1,230.000
1.420	2.397	18.179	1,240.000
1.416	2.452	18.185	1,250.000
1.411	2.460	18.192	1,260.000
1.411	2.418	18.197	1,270.000
1.411	2.437	18.201	1,280.000
1.403	2.428	18.211	1,290.000
1.397	2.408	18.219	1,300.000
1.393	2.449	18.225	1,310.000
1.387	2.429	18.233	1,320.000
1.382	2.428	18.239	1,330.000
1.380	2.459	18.244	1,340.000
1.378	2.455	18.249	1,350.000
1.375	2.452	18.254	1,360.000
1.369	2.435	18.261	1,370.000
1.368	2.473	18.265	1,380.000
1.367	2.463	18.269	1,390.000
1.362	2.445	18.276	1,400.000
1.361	2.465	18.280	1,410.000
1.359	2.431	18.283	1,420.000
1.359	2.452	18.285	1,430.000
1.356	2.461	18.290	1,440.000
1.354	2.448	18.295	1,450.000
1.353	2.450	18.297	1,460.000
1.351	2.440	18.302	1,470.000
1.351	2.455	18.305	1,480.000
1.348	2.449	18.311	1,490.000
1.346	2.455	18.316	1,500.000
1.343	2.444	18.320	1,510.000
1.341	2.445	18.326	1,520.000
1.337	2.452	18.331	1,530.000
1.335	2.448	18.335	1,540.000

1.331	2.439	18.341	1,550.000
1.329	2.497	18.346	1,560.000
1.327	2.471	18.350	1,570.000
1.325	2.497	18.354	1,580.000
1.322	2.467	18.359	1,590.000
1.319	2.473	18.363	1,600.000
1.317	2.470	18.367	1,610.000
1.315	2.466	18.372	1,620.000
1.315	2.471	18.375	1,630.000
1.314	2.474	18.380	1,640.000
1.313	2.449	18.382	1,650.000
1.316	2.468	18.388	1,660.000
1.316	2.490	18.391	1,670.000
1.315	2.496	18.394	1,680.000
1.315	2.471	18.396	1,690.000
1.313	2.471	18.400	1,700.000
1.310	2.477	18.404	1,710.000
1.306	2.433	18.411	1,720.000
1.304	2.487	18.414	1,730.000
1.302	2.476	18.418	1,740.000
1.299	2.490	18.422	1,750.000
1.298	2.485	18.425	1,760.000
1.298	2.457	18.428	1,770.000
1.298	2.481	18.430	1,780.000
1.297	2.507	18.434	1,790.000
1.296	2.491	18.437	1,800.000
1.292	2.481	18.443	1,810.000
1.291	2.514	18.447	1,820.000
1.290	2.513	18.451	1,830.000
1.289	2.502	18.453	1,840.000
1.287	2.511	18.457	1,850.000
1.286	2.483	18.460	1,860.000
1.284	2.507	18.464	1,870.000
1.282	2.510	18.467	1,880.000
1.280	2.525	18.474	1,890.000
1.278	2.526	18.477	1,900.000
1.278	2.534	18.479	1,910.000
1.277	2.533	18.482	1,920.000
1.277	2.523	18.484	1,930.000
1.274	2.509	18.488	1,940.000

1.271	2.529	18.492	1,950.000
1.271	2.488	18.494	1,960.000
1.270	2.515	18.497	1,970.000
1.268	2.504	18.500	1,980.000
1.266	2.499	18.504	1,990.000
1.262	2.498	18.509	2,000.000
1.261	2.515	18.512	2,010.000
1.259	2.534	18.515	2,020.000
1.257	2.522	18.517	2,030.000
1.256	2.518	18.520	2,040.000
1.254	2.522	18.523	2,050.000
1.252	2.545	18.527	2,060.000
1.250	2.540	18.530	2,070.000
1.247	2.524	18.534	2,080.000
1.247	2.500	18.537	2,090.000
1.246	2.526	18.539	2,100.000
1.244	2.548	18.542	2,110.000
1.242	2.551	18.546	2,120.000
1.241	2.514	18.548	2,130.000
1.240	2.540	18.551	2,140.000
1.236	2.530	18.555	2,150.000
1.235	2.554	18.557	2,160.000
1.233	2.537	18.560	2,170.000
1.232	2.532	18.563	2,180.000
1.231	2.531	18.565	2,190.000
1.227	2.527	18.570	2,200.000
1.227	2.518	18.573	2,210.000
1.227	2.549	18.576	2,220.000
1.226	2.550	18.577	2,230.000
1.226	2.529	18.579	2,240.000
1.225	2.569	18.580	2,250.000
1.225	2.561	18.582	2,260.000
1.225	2.533	18.583	2,270.000
1.223	2.536	18.585	2,280.000
1.223	2.555	18.587	2,290.000
1.222	2.552	18.588	2,300.000
1.222	2.549	18.589	2,310.000
1.222	2.563	18.591	2,320.000
1.221	2.567	18.593	2,330.000
1.221	2.569	18.595	2,340.000

1.219	2.561	18.598	2,350.000
1.218	2.556	18.600	2,360.000
1.216	2.567	18.602	2,370.000
1.216	2.548	18.603	2,380.000
1.216	2.558	18.605	2,390.000
1.215	2.557	18.606	2,400.000
1.215	2.553	18.608	2,410.000
1.215	2.554	18.609	2,420.000
1.215	2.575	18.611	2,430.000
1.215	2.553	18.612	2,440.000
1.215	2.557	18.613	2,450.000
1.214	2.546	18.615	2,460.000
1.215	2.551	18.616	2,470.000
1.215	2.546	18.618	2,480.000
1.213	2.573	18.620	2,490.000
1.213	2.559	18.621	2,500.000

NSGA-II with LEMMO and Final ANN Structure Analysis Metric

Results

Convergence	Diversity	Dominated Hypervolume	Iteration
9.862	30.787	9.307	10.000
9.344	16.873	9.985	20.000
8.884	7.267	10.461	30.000
8.524	4.564	10.826	40.000
8.234	3.375	11.115	50.000
7.972	2.684	11.375	60.000
7.722	2.675	11.622	70.000
7.464	2.314	11.875	80.000
7.265	2.342	12.072	90.000
7.045	2.212	12.295	100.000
6.872	2.152	12.470	110.000
6.692	2.180	12.652	120.000
6.488	2.169	12.864	130.000
6.321	2.146	13.043	140.000
6.162	2.143	13.210	150.000
5.982	2.080	13.393	160.000
5.809	2.085	13.568	170.000
5.647	2.061	13.730	180.000
5.525	2.045	13.861	190.000
5.369	2.059	14.020	200.000
5.236	2.016	14.157	210.000
5.122	2.016	14.276	220.000
4.984	2.028	14.416	230.000
4.857	1.989	14.549	240.000
4.735	1.982	14.673	250.000
4.617	1.995	14.794	260.000
4.520	2.003	14.896	270.000
4.393	2.007	15.024	280.000
4.292	1.980	15.130	290.000
4.192	1.940	15.234	300.000
4.099	1.972	15.335	310.000
4.003	1.967	15.437	320.000
3.911	1.973	15.536	330.000
3.828	1.969	15.626	340.000

3.738	1.956	15.720	350.000
3.651	1.981	15.813	360.000
3.563	1.903	15.905	370.000
3.475	1.959	15.999	380.000
3.403	1.957	16.080	390.000
3.299	1.931	16.184	400.000
3.227	1.930	16.261	410.000
3.154	1.915	16.337	420.000
3.093	1.934	16.405	430.000
3.012	1.903	16.488	440.000
2.955	1.918	16.553	450.000
2.900	1.916	16.616	460.000
2.821	1.895	16.698	470.000
2.755	1.925	16.771	480.000
2.696	1.916	16.834	490.000
2.635	1.880	16.901	500.000
2.577	1.914	16.965	510.000
2.524	1.899	17.023	520.000
2.473	1.911	17.079	530.000
2.420	1.912	17.139	540.000
2.374	1.872	17.191	550.000
2.321	1.895	17.249	560.000
2.283	1.916	17.298	570.000
2.226	1.881	17.358	580.000
2.184	1.876	17.407	590.000
2.136	1.918	17.463	600.000
2.094	1.917	17.512	610.000
2.050	1.912	17.563	620.000
2.001	1.890	17.616	630.000
1.965	1.920	17.661	640.000
1.921	1.891	17.711	650.000
1.884	1.877	17.755	660.000
1.847	1.878	17.797	670.000
1.796	1.866	17.849	680.000
1.760	1.893	17.890	690.000
1.727	1.884	17.930	700.000
1.696	1.895	17.968	710.000
1.672	1.877	18.000	720.000
1.642	1.887	18.039	730.000
1.606	1.842	18.081	740.000

1.578	1.850	18.114	750.000
1.548	1.868	18.151	760.000
1.518	1.896	18.186	770.000
1.495	1.880	18.217	780.000
1.467	1.868	18.251	790.000
1.445	1.869	18.286	800.000
1.421	1.848	18.315	810.000
1.402	1.857	18.342	820.000
1.380	1.871	18.371	830.000
1.360	1.866	18.399	840.000
1.335	1.837	18.429	850.000
1.314	1.827	18.456	860.000
1.297	1.849	18.480	870.000
1.278	1.879	18.506	880.000
1.259	1.864	18.533	890.000
1.238	1.843	18.564	900.000
1.218	1.843	18.590	910.000
1.199	1.828	18.616	920.000
1.179	1.846	18.643	930.000
1.160	1.821	18.672	940.000
1.142	1.855	18.697	950.000
1.124	1.849	18.723	960.000
1.108	1.825	18.744	970.000
1.095	1.833	18.765	980.000
1.079	1.833	18.788	990.000
1.067	1.818	18.807	1,000.000
1.050	1.831	18.829	1,010.000
1.037	1.830	18.848	1,020.000
1.025	1.844	18.867	1,030.000
1.008	1.840	18.890	1,040.000
0.996	1.839	18.911	1,050.000
0.981	1.829	18.932	1,060.000
0.968	1.821	18.952	1,070.000
0.957	1.823	18.970	1,080.000
0.945	1.857	18.989	1,090.000
0.930	1.801	19.012	1,100.000
0.918	1.830	19.030	1,110.000
0.907	1.811	19.046	1,120.000
0.898	1.833	19.064	1,130.000
0.890	1.829	19.080	1,140.000

0.876	1.860	19.100	1,150.000
0.867	1.822	19.115	1,160.000
0.853	1.821	19.136	1,170.000
0.841	1.819	19.155	1,180.000
0.847	1.832	19.170	1,190.000
0.841	1.819	19.186	1,200.000
0.832	1.812	19.200	1,210.000
0.823	1.812	19.214	1,220.000
0.816	1.840	19.228	1,230.000
0.810	1.824	19.241	1,240.000
0.802	1.824	19.254	1,250.000
0.797	1.818	19.268	1,260.000
0.790	1.840	19.280	1,270.000
0.782	1.837	19.294	1,280.000
0.774	1.842	19.309	1,290.000
0.766	1.811	19.324	1,300.000
0.759	1.814	19.338	1,310.000
0.753	1.828	19.348	1,320.000
0.746	1.840	19.365	1,330.000
0.741	1.805	19.377	1,340.000
0.736	1.823	19.390	1,350.000
0.728	1.821	19.402	1,360.000
0.722	1.812	19.416	1,370.000
0.715	1.819	19.428	1,380.000
0.709	1.788	19.438	1,390.000
0.704	1.836	19.452	1,400.000
0.699	1.842	19.463	1,410.000
0.693	1.832	19.475	1,420.000
0.689	1.802	19.486	1,430.000
0.682	1.830	19.499	1,440.000
0.677	1.807	19.509	1,450.000
0.672	1.810	19.520	1,460.000
0.667	1.816	19.531	1,470.000
0.662	1.806	19.541	1,480.000
0.655	1.813	19.554	1,490.000
0.650	1.820	19.562	1,500.000
0.643	1.776	19.573	1,510.000
0.638	1.829	19.584	1,520.000
0.633	1.823	19.595	1,530.000
0.630	1.816	19.607	1,540.000

0.625	1.797	19.616	1,550.000
0.623	1.780	19.624	1,560.000
0.619	1.809	19.632	1,570.000
0.615	1.783	19.642	1,580.000
0.610	1.816	19.652	1,590.000
0.605	1.803	19.660	1,600.000
0.601	1.813	19.669	1,610.000
0.596	1.809	19.681	1,620.000
0.592	1.789	19.687	1,630.000
0.589	1.813	19.695	1,640.000
0.586	1.820	19.704	1,650.000
0.581	1.823	19.713	1,660.000
0.578	1.805	19.718	1,670.000
0.574	1.794	19.728	1,680.000
0.571	1.797	19.736	1,690.000
0.567	1.798	19.744	1,700.000
0.562	1.805	19.754	1,710.000
0.560	1.801	19.762	1,720.000
0.557	1.819	19.771	1,730.000
0.553	1.814	19.778	1,740.000
0.551	1.807	19.784	1,750.000
0.548	1.799	19.791	1,760.000
0.546	1.814	19.799	1,770.000
0.543	1.808	19.806	1,780.000
0.541	1.810	19.814	1,790.000
0.536	1.791	19.822	1,800.000
0.534	1.803	19.829	1,810.000
0.532	1.803	19.837	1,820.000
0.528	1.807	19.844	1,830.000
0.527	1.810	19.850	1,840.000
0.525	1.834	19.855	1,850.000
0.523	1.816	19.863	1,860.000
0.521	1.796	19.869	1,870.000
0.519	1.824	19.876	1,880.000
0.516	1.808	19.882	1,890.000
0.514	1.802	19.887	1,900.000
0.511	1.816	19.894	1,910.000
0.508	1.797	19.900	1,920.000
0.505	1.805	19.906	1,930.000
0.503	1.794	19.911	1,940.000

0.500	1.794	19.918	1,950.000
0.497	1.814	19.923	1,960.000
0.494	1.803	19.928	1,970.000
0.491	1.818	19.935	1,980.000
0.489	1.794	19.941	1,990.000
0.486	1.817	19.946	2,000.000
0.483	1.811	19.951	2,010.000
0.481	1.806	19.958	2,020.000
0.498	1.813	19.963	2,030.000
0.497	1.810	19.969	2,040.000
0.494	1.832	19.974	2,050.000
0.493	1.809	19.979	2,060.000
0.490	1.803	19.983	2,070.000
0.488	1.805	19.988	2,080.000
0.485	1.779	19.996	2,090.000
0.483	1.806	20.002	2,100.000
0.482	1.814	20.007	2,110.000
0.480	1.816	20.012	2,120.000
0.478	1.816	20.018	2,130.000
0.476	1.813	20.023	2,140.000
0.473	1.791	20.029	2,150.000
0.470	1.806	20.034	2,160.000
0.469	1.779	20.040	2,170.000
0.467	1.809	20.045	2,180.000
0.465	1.818	20.050	2,190.000
0.463	1.812	20.054	2,200.000
0.462	1.812	20.058	2,210.000
0.462	1.810	20.063	2,220.000
0.461	1.788	20.068	2,230.000
0.458	1.801	20.072	2,240.000
0.457	1.805	20.075	2,250.000
0.455	1.809	20.080	2,260.000
0.454	1.799	20.085	2,270.000
0.454	1.812	20.088	2,280.000
0.451	1.788	20.092	2,290.000
0.450	1.794	20.096	2,300.000
0.449	1.827	20.100	2,310.000
0.447	1.826	20.104	2,320.000
0.446	1.814	20.107	2,330.000
0.445	1.783	20.112	2,340.000

0.443	1.818	20.116	2,350.000
0.442	1.790	20.120	2,360.000
0.441	1.822	20.125	2,370.000
0.439	1.820	20.129	2,380.000
0.438	1.821	20.132	2,390.000
0.436	1.819	20.136	2,400.000
0.434	1.788	20.140	2,410.000
0.433	1.791	20.143	2,420.000
0.453	1.816	20.147	2,430.000
0.451	1.820	20.150	2,440.000
0.450	1.793	20.154	2,450.000
0.448	1.793	20.158	2,460.000
0.448	1.788	20.163	2,470.000
0.447	1.802	20.166	2,480.000
0.446	1.808	20.170	2,490.000
0.444	1.820	20.174	2,500.000

Dalmarnock Case Study Results

These are the results from the final complete iteration from the Dalmarnock case study.

Network cost (£)	Expected Annual Damage (£)	Rank	Crowding distance
25243757.17	651.4286466	1	Infinity
25239716.21	1119.875394	1	0.008600801
25185446.5	1182.020857	1	0.006180114
25136252.15	1295.436244	1	0.014419537
25059271.79	1976.638749	1	0.021186697
24885612.37	2244.96083	1	0.774726192
5750455.941	2805.855535	1	0.877837503
5485101.292	11464.51351	1	0.080764945
5485101.292	11464.51351	1	0.110403976
5289620.428	17389.87322	1	0.204892654
2852647.653	19950.35613	1	0.166836575
2793530.332	23121.56678	1	0.208412792
2621097.045	36754.84341	1	0.291594523
2554687.125	46917.47306	1	0.478635833
2538550.951	76848.55754	1	0.374724157
2117206.39	77061.09463	1	0.197140252
0	84994.28223	1	Infinity

Dalmarnock Analysis Metric Results

Presented here are the results from the analysis metric run on the Dalmarnock case study results.

Dominated Hypervolume	Iteration
2.38E+12	1
2.66E+12	2
2.67E+12	3
3.19E+12	4
3.58E+12	5
4.22E+12	6
4.31E+12	7
4.39E+12	8
4.39E+12	9
4.44E+12	10

4.49E+12	11
4.51E+12	12
4.51E+12	13
4.52E+12	14
4.56E+12	15
4.57E+12	16
4.61E+12	17
4.61E+12	18
4.61E+12	19
4.63E+12	20
4.63E+12	21
4.64E+12	22
4.65E+12	23
4.65E+12	24
4.67E+12	25
4.68E+12	26
4.68E+12	27
4.68E+12	28
4.68E+12	29
4.69E+12	30
4.69E+12	31
4.69E+12	32
4.70E+12	33
4.70E+12	34
4.70E+12	35
4.70E+12	36
4.71E+12	37
4.71E+12	38
4.71E+12	39
4.70E+12	40
4.70E+12	41
4.71E+12	42
4.71E+12	43
4.71E+12	44
4.72E+12	45
4.71E+12	46
4.71E+12	47
4.72E+12	48
4.72E+12	49
4.71E+12	50

4.71E+12	51
4.72E+12	52
4.73E+12	53
4.73E+12	54
4.73E+12	55
4.73E+12	56
4.73E+12	57
4.80E+12	58
4.80E+12	59
4.80E+12	60
4.80E+12	61
4.81E+12	62
4.81E+12	63
4.81E+12	64
4.81E+12	65
4.82E+12	66
4.82E+12	67
4.82E+12	68
4.82E+12	69
4.82E+12	70
4.82E+12	71
4.82E+12	72
4.82E+12	73
4.82E+12	74
4.83E+12	75
4.83E+12	76
4.83E+12	77
4.83E+12	78
4.83E+12	79
4.83E+12	80
4.83E+12	81
4.84E+12	82
4.84E+12	83
4.84E+12	84
4.84E+12	85
4.84E+12	86
4.84E+12	87
4.84E+12	88
4.84E+12	89
4.84E+12	90

4.85E+12	91
4.85E+12	92
4.85E+12	93
4.85E+12	94
4.85E+12	95
4.85E+12	96
4.85E+12	97
4.85E+12	98
4.85E+12	99
4.85E+12	100
4.85E+12	101
4.85E+12	102
4.93E+12	103
4.93E+12	104
4.93E+12	105
4.93E+12	106
4.93E+12	107
4.93E+12	108
4.93E+12	109
4.93E+12	110
4.93E+12	111
4.93E+12	112
4.94E+12	113
4.94E+12	114
4.94E+12	115
4.94E+12	116
4.94E+12	117
4.94E+12	118
4.94E+12	119
4.94E+12	120
4.94E+12	121
4.94E+12	122
4.94E+12	123
4.94E+12	124
4.94E+12	125
4.94E+12	126
4.94E+12	127
4.94E+12	128
4.95E+12	129
4.95E+12	130

4.95E+12	131
4.95E+12	132
4.95E+12	133
4.95E+12	134
4.95E+12	135
4.95E+12	136
4.95E+12	137
4.95E+12	138
4.95E+12	139
4.95E+12	140
4.95E+12	141
4.95E+12	142
4.95E+12	143
4.95E+12	144
4.95E+12	145
4.96E+12	146
4.96E+12	147
4.96E+12	148
4.96E+12	149
4.96E+12	150
4.96E+12	151
4.96E+12	152
4.96E+12	153
5.10E+12	154
5.10E+12	155
5.10E+12	156
5.11E+12	157
5.11E+12	158
5.11E+12	159
5.11E+12	160
5.11E+12	161
5.11E+12	162
5.11E+12	163
5.11E+12	164
5.11E+12	165
5.11E+12	166
5.10E+12	167
5.10E+12	168
5.10E+12	169
5.10E+12	170

5.10E+12	171
5.10E+12	172
5.10E+12	173
5.10E+12	174
5.11E+12	175
5.11E+12	176
5.11E+12	177
5.11E+12	178
5.10E+12	179
5.11E+12	180
5.11E+12	181
5.11E+12	182
5.11E+12	183
5.11E+12	184
5.12E+12	185
5.12E+12	186
5.12E+12	187
5.12E+12	188
5.12E+12	189
5.12E+12	190
5.12E+12	191
5.12E+12	192
5.12E+12	193
5.12E+12	194
5.12E+12	195
5.12E+12	196
5.12E+12	197
5.12E+12	198
5.12E+12	199
5.12E+12	200
5.12E+12	201
5.12E+12	202
5.12E+12	203
5.12E+12	204
5.12E+12	205
5.12E+12	206
5.13E+12	207
5.13E+12	208
5.13E+12	209
5.14E+12	210

5.14E+12	211
5.14E+12	212
5.14E+12	213
5.14E+12	214
5.14E+12	215
5.14E+12	216
5.14E+12	217
5.14E+12	218
5.14E+12	219
5.14E+12	220
5.14E+12	221
5.14E+12	222
5.14E+12	223
5.14E+12	224
5.14E+12	225
5.14E+12	226
5.14E+12	227
5.14E+12	228
5.14E+12	229
5.14E+12	230
5.14E+12	231
5.14E+12	232
5.14E+12	233
5.14E+12	234
5.14E+12	235
5.14E+12	236
5.14E+12	237
5.14E+12	238
5.14E+12	239
5.14E+12	240
5.14E+12	241
5.14E+12	242
5.14E+12	243
5.14E+12	244
5.14E+12	245
5.14E+12	246
5.14E+12	247
5.14E+12	248
5.14E+12	249
5.14E+12	250

5.14E+12	251	5.18E+12	291
5.14E+12	252	5.18E+12	292
5.14E+12	253	5.18E+12	293
5.14E+12	254	5.18E+12	294
5.14E+12	255	5.18E+12	295
5.14E+12	256	5.18E+12	296
5.15E+12	257	5.18E+12	297
5.15E+12	258	5.18E+12	298
5.15E+12	259	5.18E+12	299
5.15E+12	260	5.18E+12	300
5.15E+12	261	5.18E+12	301
5.15E+12	262	5.18E+12	302
5.15E+12	263	5.18E+12	303
5.15E+12	264	5.18E+12	304
5.15E+12	265	5.18E+12	305
5.15E+12	266	5.18E+12	306
5.15E+12	267	5.18E+12	307
5.15E+12	268	5.18E+12	308
5.15E+12	269	5.18E+12	309
5.15E+12	270	5.18E+12	310
5.15E+12	271	5.18E+12	311
5.15E+12	272	5.18E+12	312
5.15E+12	273	5.18E+12	313
5.15E+12	274	5.18E+12	314
5.15E+12	275	5.18E+12	315
5.15E+12	276	5.18E+12	316
5.15E+12	277	5.18E+12	317
5.15E+12	278	5.18E+12	318
5.15E+12	279	5.18E+12	319
5.15E+12	280	5.18E+12	320
5.15E+12	281	5.18E+12	321
5.18E+12	282	5.18E+12	322
5.18E+12	283	5.18E+12	323
5.18E+12	284	5.18E+12	324
5.18E+12	285	5.18E+12	325
5.18E+12	286	5.18E+12	326
5.18E+12	287	5.18E+12	327
5.18E+12	288	5.19E+12	328
5.18E+12	289	5.19E+12	329
5.18E+12	290	5.19E+12	330

5.19E+12	331
5.19E+12	332
5.19E+12	333
5.19E+12	334
5.19E+12	335
5.19E+12	336
5.19E+12	337
5.19E+12	338
5.19E+12	339
5.19E+12	340
5.19E+12	341
5.19E+12	342
5.19E+12	343
5.19E+12	344
5.19E+12	345
5.19E+12	346
5.19E+12	347
5.19E+12	348
5.19E+12	349
5.19E+12	350
5.19E+12	351
5.19E+12	352
5.19E+12	353
5.19E+12	354
5.19E+12	355
5.19E+12	356
5.19E+12	357
5.19E+12	358
5.20E+12	359
5.20E+12	360
5.20E+12	361
5.20E+12	362
5.19E+12	363
5.20E+12	364
5.20E+12	365
5.20E+12	366
5.20E+12	367
5.20E+12	368
5.19E+12	369
5.20E+12	370

5.20E+12	371
5.19E+12	372
5.20E+12	373
5.19E+12	374
5.19E+12	375
5.19E+12	376
5.19E+12	377
5.19E+12	378
5.19E+12	379
5.19E+12	380
5.20E+12	381
5.20E+12	382
5.20E+12	383
5.20E+12	384
5.20E+12	385
5.20E+12	386
5.20E+12	387
5.20E+12	388
5.20E+12	389
5.19E+12	390
5.19E+12	391
5.20E+12	392
5.20E+12	393
5.20E+12	394
5.20E+12	395
5.20E+12	396
5.20E+12	397
5.20E+12	398
5.20E+12	399
5.20E+12	400
5.20E+12	401
5.20E+12	402
5.20E+12	403
5.20E+12	404
5.20E+12	405
5.20E+12	406
5.20E+12	407
5.20E+12	408
5.20E+12	409
5.20E+12	410

5.20E+12	411
5.20E+12	412
5.20E+12	413
5.20E+12	414
5.20E+12	415
5.20E+12	416
5.20E+12	417
5.20E+12	418
5.20E+12	419
5.20E+12	420
5.20E+12	421
5.20E+12	422
5.20E+12	423
5.20E+12	424
5.20E+12	425
5.20E+12	426
5.20E+12	427
5.20E+12	428
5.20E+12	429
5.20E+12	430
5.20E+12	431
5.20E+12	432
5.20E+12	433
5.20E+12	434
5.20E+12	435
5.20E+12	436
5.20E+12	437
5.21E+12	438
5.21E+12	439
5.21E+12	440
5.21E+12	441
5.21E+12	442
5.21E+12	443
5.21E+12	444
5.21E+12	445
5.21E+12	446
5.21E+12	447
5.21E+12	448
5.22E+12	449
5.22E+12	450

5.22E+12	451
5.22E+12	452
5.22E+12	453
5.22E+12	454
5.22E+12	455
5.22E+12	456
5.22E+12	457
5.22E+12	458
5.22E+12	459
5.22E+12	460
5.22E+12	461
5.22E+12	462
5.22E+12	463
5.22E+12	464
5.22E+12	465
5.22E+12	466
5.22E+12	467
5.22E+12	468
5.22E+12	469
5.22E+12	470
5.22E+12	471
5.22E+12	472
5.22E+12	473
5.22E+12	474
5.22E+12	475
5.22E+12	476
5.22E+12	477
5.22E+12	478
5.22E+12	479
5.22E+12	480
5.23E+12	481
5.23E+12	482
5.23E+12	483
5.23E+12	484
5.23E+12	485
5.23E+12	486
5.23E+12	487
5.23E+12	488
5.23E+12	489
5.23E+12	490

5.23E+12	491
5.23E+12	492
5.23E+12	493
5.23E+12	494
5.23E+12	495
5.23E+12	496
5.23E+12	497
5.23E+12	498
5.23E+12	499
5.23E+12	500
5.23E+12	501
5.23E+12	502
5.23E+12	503
5.23E+12	504
5.23E+12	505
5.23E+12	506
5.23E+12	507
5.23E+12	508
5.23E+12	509
5.23E+12	510
5.23E+12	511
5.23E+12	512
5.23E+12	513
5.23E+12	514
5.23E+12	515
5.23E+12	516
5.23E+12	517
5.23E+12	518
5.23E+12	519
5.23E+12	520
5.23E+12	521
5.23E+12	522
5.23E+12	523
5.23E+12	524
5.23E+12	525
5.23E+12	526
5.23E+12	527
5.23E+12	528
5.23E+12	529
5.23E+12	530

5.23E+12	531
5.23E+12	532
5.23E+12	533
5.23E+12	534
5.23E+12	535
5.23E+12	536
5.24E+12	537
5.24E+12	538
5.24E+12	539
5.24E+12	540
5.24E+12	541
5.24E+12	542
5.24E+12	543
5.24E+12	544
5.24E+12	545
5.24E+12	546
5.24E+12	547
5.24E+12	548
5.24E+12	549
5.24E+12	550
5.24E+12	551
5.24E+12	552
5.24E+12	553
5.24E+12	554
5.24E+12	555
5.24E+12	556
5.24E+12	557
5.24E+12	558
5.24E+12	559
5.24E+12	560
5.24E+12	561
5.24E+12	562
5.24E+12	563
5.24E+12	564
5.24E+12	565
5.24E+12	566
5.24E+12	567
5.24E+12	568
5.24E+12	569
5.24E+12	570

5.24E+12	571
5.24E+12	572
5.24E+12	573
5.24E+12	574
5.24E+12	575
5.24E+12	576
5.24E+12	577
5.24E+12	578
5.24E+12	579
5.24E+12	580
5.24E+12	581
5.24E+12	582
5.24E+12	583
5.24E+12	584
5.24E+12	585
5.24E+12	586
5.24E+12	587
5.24E+12	588
5.24E+12	589
5.24E+12	590
5.24E+12	591
5.24E+12	592
5.24E+12	593
5.24E+12	594
5.24E+12	595
5.24E+12	596
5.24E+12	597
5.24E+12	598
5.24E+12	599
5.24E+12	600
5.24E+12	601
5.24E+12	602
5.24E+12	603
5.24E+12	604
5.24E+12	605
5.24E+12	606
5.26E+12	607
5.26E+12	608
5.26E+12	609
5.26E+12	610

5.26E+12	611
5.26E+12	612
5.26E+12	613
5.27E+12	614
5.27E+12	615
5.26E+12	616
5.26E+12	617
5.26E+12	618
5.26E+12	619
5.26E+12	620
5.27E+12	621
5.27E+12	622
5.27E+12	623
5.27E+12	624
5.27E+12	625
5.27E+12	626
5.27E+12	627
5.27E+12	628
5.27E+12	629
5.27E+12	630
5.27E+12	631
5.27E+12	632
5.27E+12	633
5.27E+12	634
5.27E+12	635
5.27E+12	636
5.27E+12	637
5.27E+12	638
5.27E+12	639
5.27E+12	640
5.27E+12	641
5.27E+12	642
5.27E+12	643
5.27E+12	644
5.27E+12	645
5.27E+12	646
5.27E+12	647
5.27E+12	648
5.27E+12	649
5.27E+12	650

5.27E+12	651
5.27E+12	652
5.27E+12	653
5.27E+12	654
5.27E+12	655
5.27E+12	656
5.27E+12	657
5.27E+12	658
5.27E+12	659
5.27E+12	660
5.27E+12	661
5.27E+12	662
5.27E+12	663
5.27E+12	664
5.27E+12	665
5.27E+12	666
5.27E+12	667
5.27E+12	668
5.55E+12	669
5.55E+12	670
5.55E+12	671
5.55E+12	672
5.55E+12	673
5.55E+12	674
5.55E+12	675
5.55E+12	676
5.55E+12	677
5.55E+12	678
5.55E+12	679
5.55E+12	680
5.55E+12	681
5.55E+12	682
5.55E+12	683
5.55E+12	684
5.55E+12	685
5.55E+12	686
5.55E+12	687
5.55E+12	688
5.55E+12	689
5.55E+12	690

5.55E+12	691
5.55E+12	692
5.55E+12	693
5.55E+12	694
5.55E+12	695
5.55E+12	696
5.55E+12	697
5.55E+12	698
5.55E+12	699
5.55E+12	700
5.55E+12	701
5.55E+12	702
5.55E+12	703
5.55E+12	704
5.55E+12	705
5.56E+12	706
5.56E+12	707
5.56E+12	708
5.56E+12	709
5.56E+12	710
5.56E+12	711
5.56E+12	712
5.56E+12	713
5.56E+12	714
5.56E+12	715
5.56E+12	716
5.56E+12	717
5.56E+12	718
5.56E+12	719
5.56E+12	720
5.56E+12	721
5.56E+12	722
5.56E+12	723
5.56E+12	724
5.56E+12	725
5.56E+12	726
5.56E+12	727
5.56E+12	728
5.56E+12	729
5.56E+12	730

5.56E+12	731
5.56E+12	732
5.56E+12	733
5.56E+12	734
5.56E+12	735
5.56E+12	736
5.56E+12	737
5.56E+12	738
5.56E+12	739
5.56E+12	740
5.56E+12	741
5.56E+12	742
5.56E+12	743
5.56E+12	744
5.56E+12	745
5.56E+12	746
5.56E+12	747
5.56E+12	748
5.56E+12	749
5.56E+12	750
5.56E+12	751
5.56E+12	752
5.56E+12	753
5.56E+12	754
5.56E+12	755
5.56E+12	756
5.56E+12	757
5.56E+12	758
5.56E+12	759
5.56E+12	760
5.56E+12	761
5.56E+12	762
5.56E+12	763
5.56E+12	764
5.56E+12	765
5.56E+12	766
5.56E+12	767
5.56E+12	768
5.56E+12	769
5.56E+12	770

5.56E+12	771
5.56E+12	772
5.56E+12	773
5.56E+12	774
5.56E+12	775
5.56E+12	776
5.56E+12	777
5.56E+12	778
5.56E+12	779
5.56E+12	780
5.56E+12	781
5.56E+12	782
5.56E+12	783
5.56E+12	784
5.56E+12	785
5.56E+12	786
5.56E+12	787
5.56E+12	788
5.56E+12	789
5.56E+12	790
5.56E+12	791
5.56E+12	792
5.56E+12	793
5.56E+12	794
5.56E+12	795
5.56E+12	796
5.56E+12	797
5.56E+12	798
5.56E+12	799
5.56E+12	800
5.56E+12	801
5.56E+12	802
5.56E+12	803
5.56E+12	804
5.56E+12	805
5.56E+12	806
5.56E+12	807
5.56E+12	808
5.56E+12	809
5.56E+12	810

5.56E+12	811
5.56E+12	812
5.56E+12	813
5.56E+12	814
5.56E+12	815
5.56E+12	816
5.56E+12	817
5.56E+12	818
5.56E+12	819
5.56E+12	820
5.56E+12	821
5.56E+12	822
5.56E+12	823
5.56E+12	824
5.56E+12	825
5.56E+12	826
5.56E+12	827
5.56E+12	828
5.56E+12	829
5.56E+12	830
5.56E+12	831
5.56E+12	832
5.56E+12	833
5.56E+12	834
5.56E+12	835
5.56E+12	836
5.56E+12	837
5.56E+12	838
5.56E+12	839
5.56E+12	840
5.56E+12	841
5.56E+12	842
5.56E+12	843
5.56E+12	844
5.56E+12	845
5.56E+12	846
5.56E+12	847
5.56E+12	848
5.56E+12	849
5.56E+12	850

5.56E+12	851
5.56E+12	852
5.56E+12	853
5.56E+12	854
5.56E+12	855
5.56E+12	856
5.56E+12	857
5.56E+12	858
5.56E+12	859
5.56E+12	860
5.56E+12	861
5.56E+12	862
5.56E+12	863
5.56E+12	864
5.56E+12	865
5.56E+12	866
5.56E+12	867
5.56E+12	868
5.56E+12	869
5.56E+12	870
5.56E+12	871
5.56E+12	872
5.56E+12	873
5.56E+12	874
5.56E+12	875
5.56E+12	876
5.56E+12	877
5.56E+12	878
5.56E+12	879
5.56E+12	880
5.56E+12	881
5.56E+12	882
5.56E+12	883
5.56E+12	884
5.56E+12	885
5.56E+12	886
5.56E+12	887
5.56E+12	888
5.56E+12	889
5.56E+12	890

5.56E+12	891	5.56E+12	931
5.56E+12	892	5.56E+12	932
5.56E+12	893	5.56E+12	933
5.56E+12	894	5.56E+12	934
5.56E+12	895	5.56E+12	935
5.56E+12	896	5.56E+12	936
5.56E+12	897	5.56E+12	937
5.56E+12	898	5.56E+12	938
5.56E+12	899	5.56E+12	939
5.56E+12	900	5.56E+12	940
5.56E+12	901	5.56E+12	941
5.56E+12	902	5.56E+12	942
5.56E+12	903	5.56E+12	943
5.56E+12	904	5.56E+12	944
5.56E+12	905	5.56E+12	945
5.56E+12	906	5.56E+12	946
5.56E+12	907	5.56E+12	947
5.56E+12	908	5.56E+12	948
5.56E+12	909	5.56E+12	949
5.56E+12	910	5.56E+12	950
5.56E+12	911	5.56E+12	951
5.56E+12	912	5.56E+12	952
5.56E+12	913	5.56E+12	953
5.56E+12	914	5.56E+12	954
5.56E+12	915	5.56E+12	955
5.56E+12	916	5.56E+12	956
5.56E+12	917	5.56E+12	957
5.56E+12	918	5.56E+12	958
5.56E+12	919	5.56E+12	959
5.56E+12	920	5.56E+12	960
5.56E+12	921	5.56E+12	961
5.56E+12	922	5.56E+12	962
5.56E+12	923	5.56E+12	963
5.56E+12	924	5.56E+12	964
5.56E+12	925	5.56E+12	965
5.56E+12	926	5.56E+12	966
5.56E+12	927	5.56E+12	967
5.56E+12	928	5.56E+12	968
5.56E+12	929	5.56E+12	969
5.56E+12	930	5.56E+12	970

5.56E+12	971
5.56E+12	972
5.56E+12	973
5.56E+12	974
5.56E+12	975
5.56E+12	976
5.56E+12	977
5.56E+12	978
5.56E+12	979
5.56E+12	980
5.56E+12	981
5.56E+12	982
5.56E+12	983
5.56E+12	984
5.56E+12	985
5.56E+12	986
5.56E+12	987
5.56E+12	988
5.56E+12	989
5.56E+12	990
5.56E+12	991
5.56E+12	992
5.56E+12	993
5.56E+12	994
5.56E+12	995
5.56E+12	996
5.56E+12	997
5.56E+12	998
5.56E+12	999
5.56E+12	1,000
5.56E+12	1,001
5.56E+12	1,002
5.56E+12	1,003
5.56E+12	1,004
5.56E+12	1,005
5.56E+12	1,006
5.56E+12	1,007
5.56E+12	1,008
5.56E+12	1,009
5.56E+12	1,010

5.56E+12	1,011
5.56E+12	1,012
5.56E+12	1,013
5.56E+12	1,014
5.56E+12	1,015
5.56E+12	1,016
5.56E+12	1,017
5.56E+12	1,018
5.56E+12	1,019
5.56E+12	1,020
5.56E+12	1,021
5.56E+12	1,022
5.56E+12	1,023
5.56E+12	1,024
5.56E+12	1,025
5.56E+12	1,026
5.56E+12	1,027
5.56E+12	1,028
5.57E+12	1,029
5.57E+12	1,030
5.58E+12	1,031
5.61E+12	1,032
5.61E+12	1,033
5.61E+12	1,034
5.61E+12	1,035
5.63E+12	1,036
5.64E+12	1,037
5.64E+12	1,038
5.64E+12	1,039
5.64E+12	1,040
5.64E+12	1,041
5.64E+12	1,042
5.64E+12	1,043
5.64E+12	1,044
5.64E+12	1,045
5.64E+12	1,046
5.64E+12	1,047
5.64E+12	1,048
5.64E+12	1,049

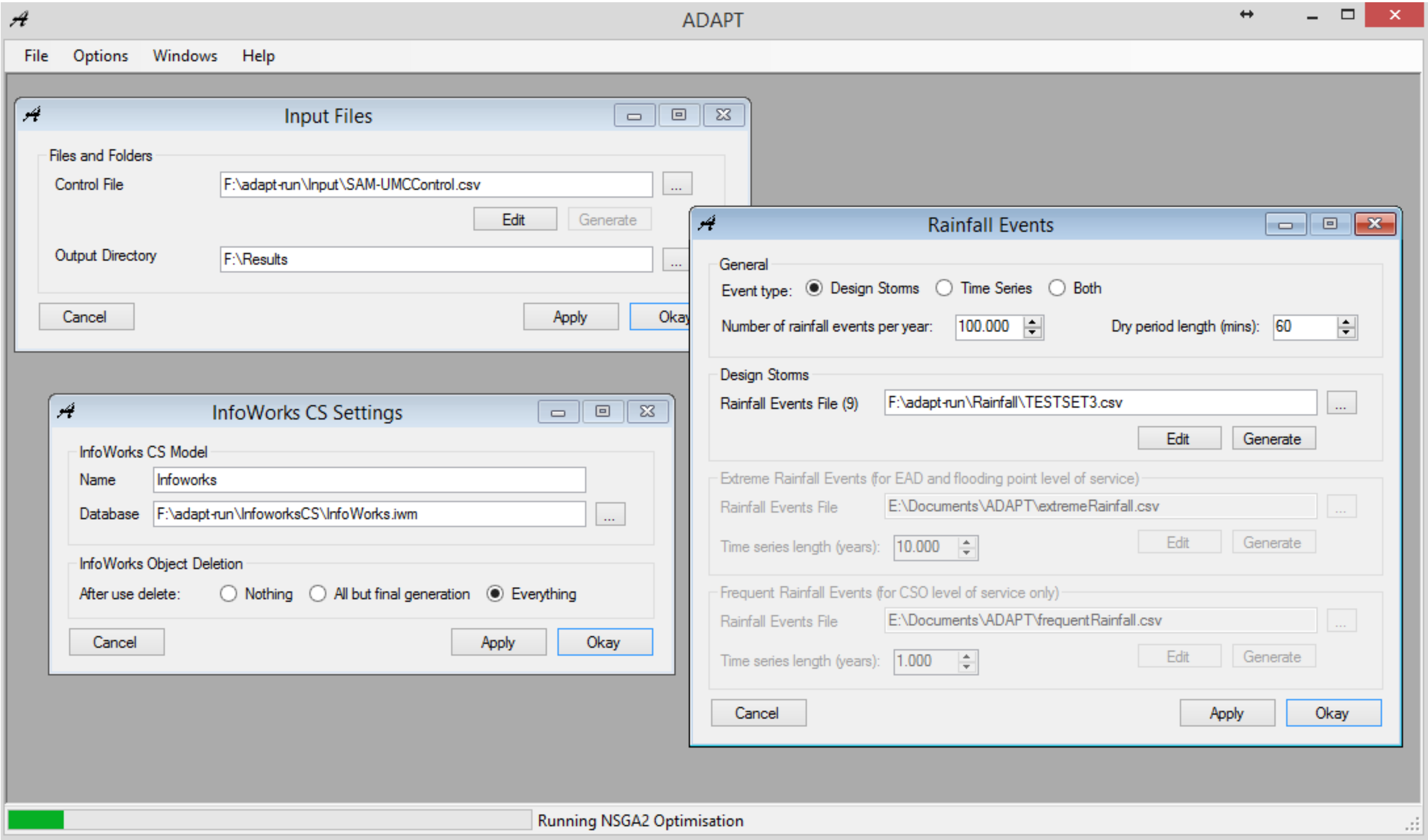
Appendix III – SAM-Risk Settings

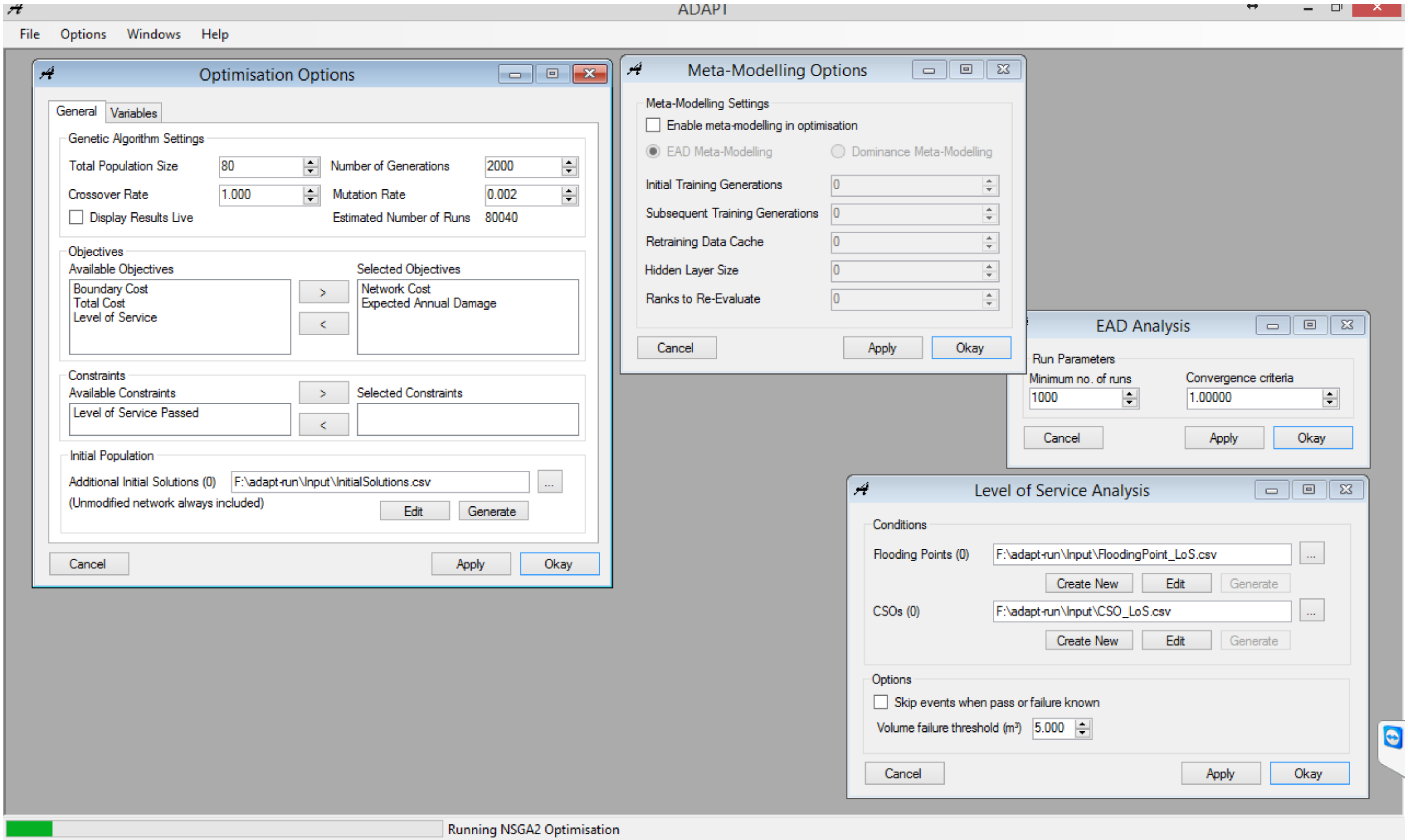
This appendix contains details of the exact settings used to run the Dalmarnock test problem using the ADAPT software developed as part of this thesis. The precise function of all settings is not described in this thesis, but relevant documentation will be cited to allow interested parties to investigate in-depth.

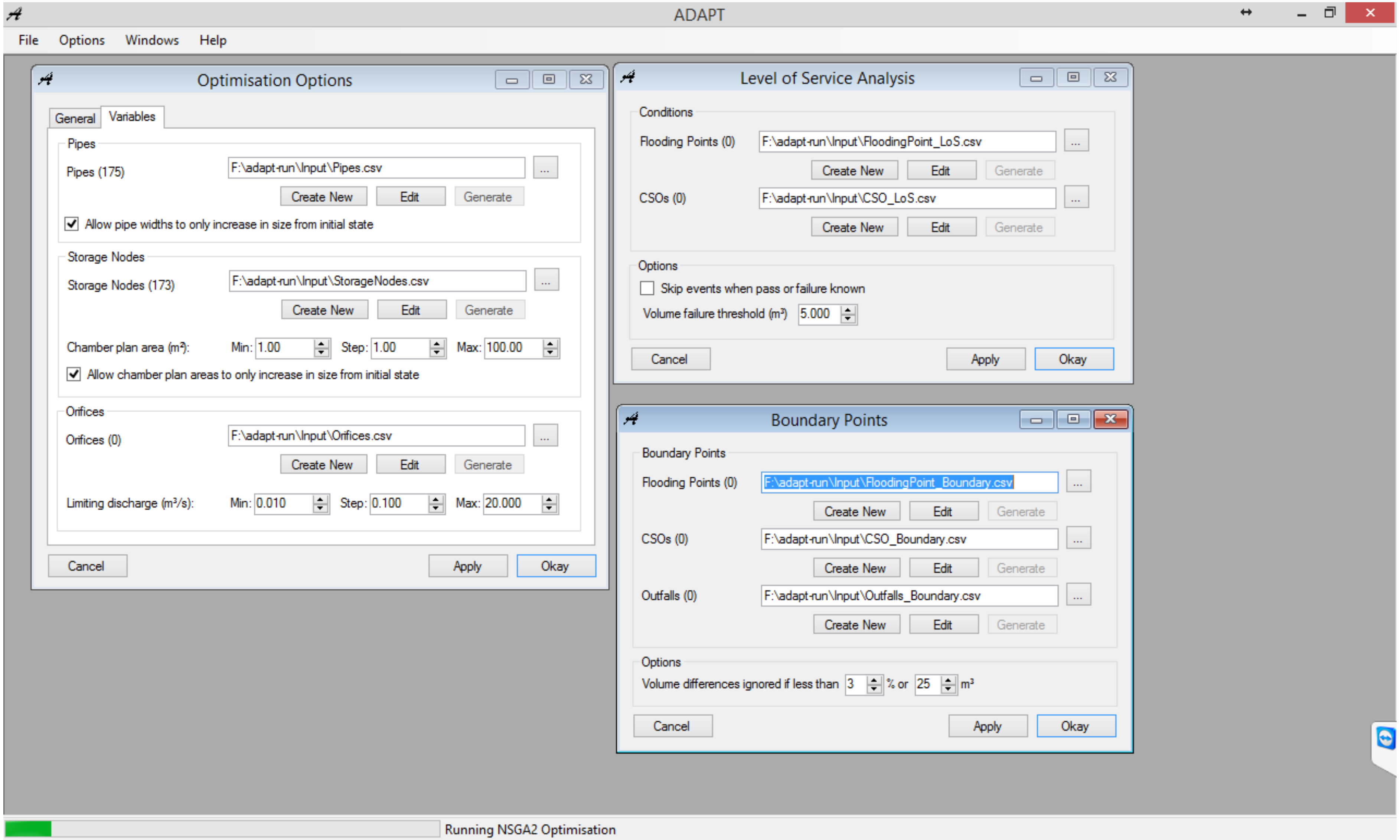
ADAPT Settings

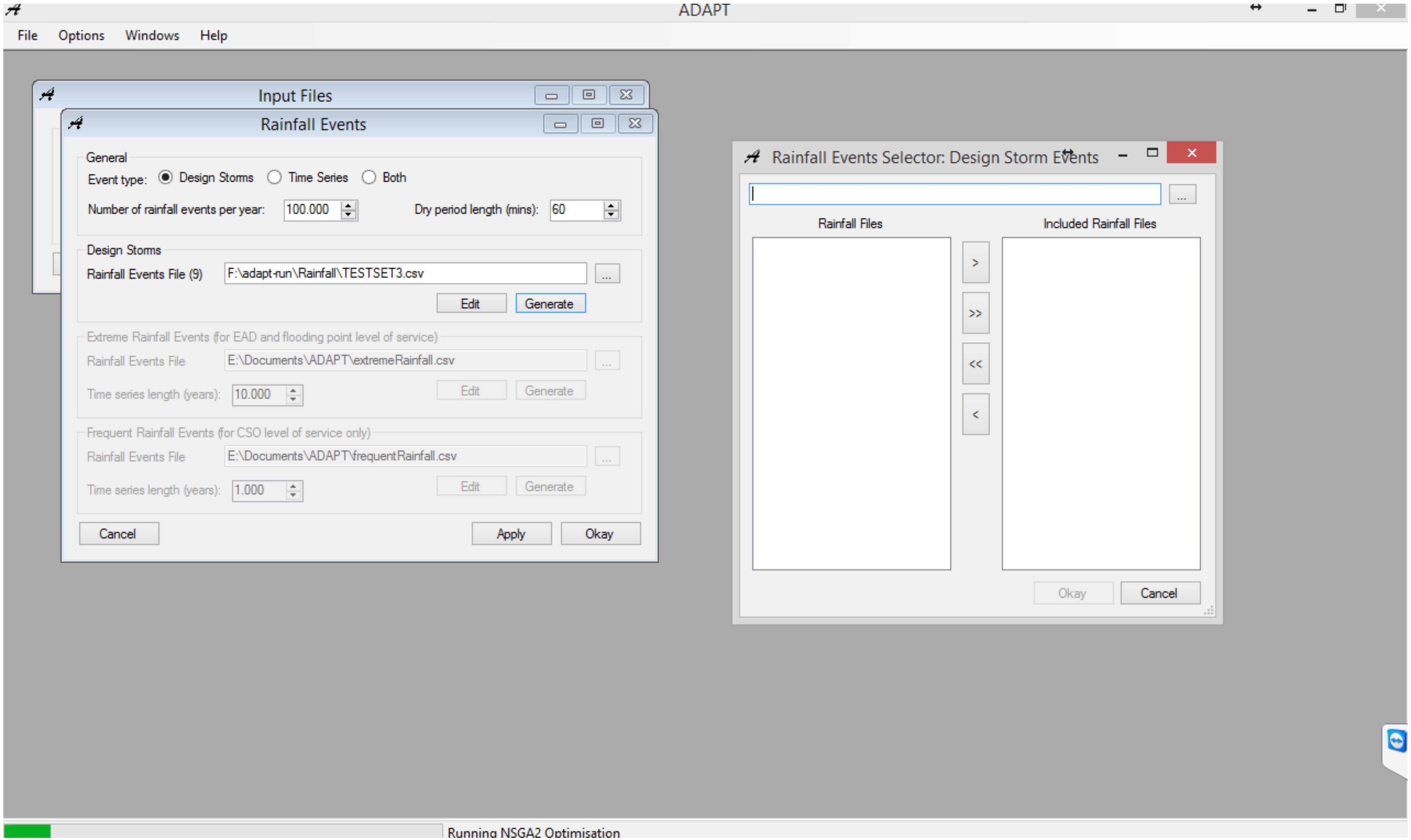
This section contains the settings used within the ADAPT application, some of which, but not all, are mirrored in the SAM-UMC settings files.

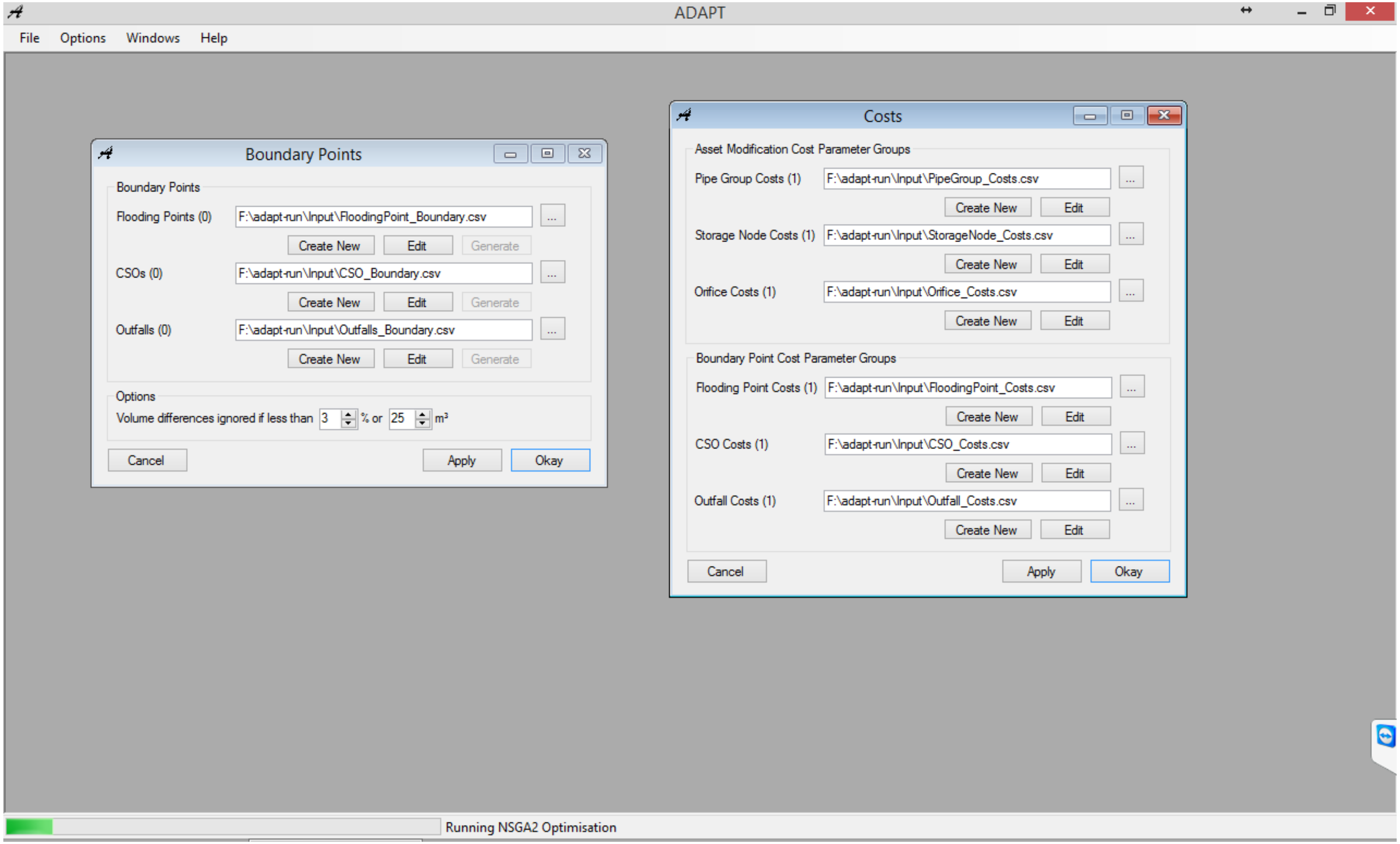
In order to easily present these settings without needing to undertake long and detailed descriptions of where different settings reside, this section is presented as a series of screenshots of the application running with the correct settings in place.

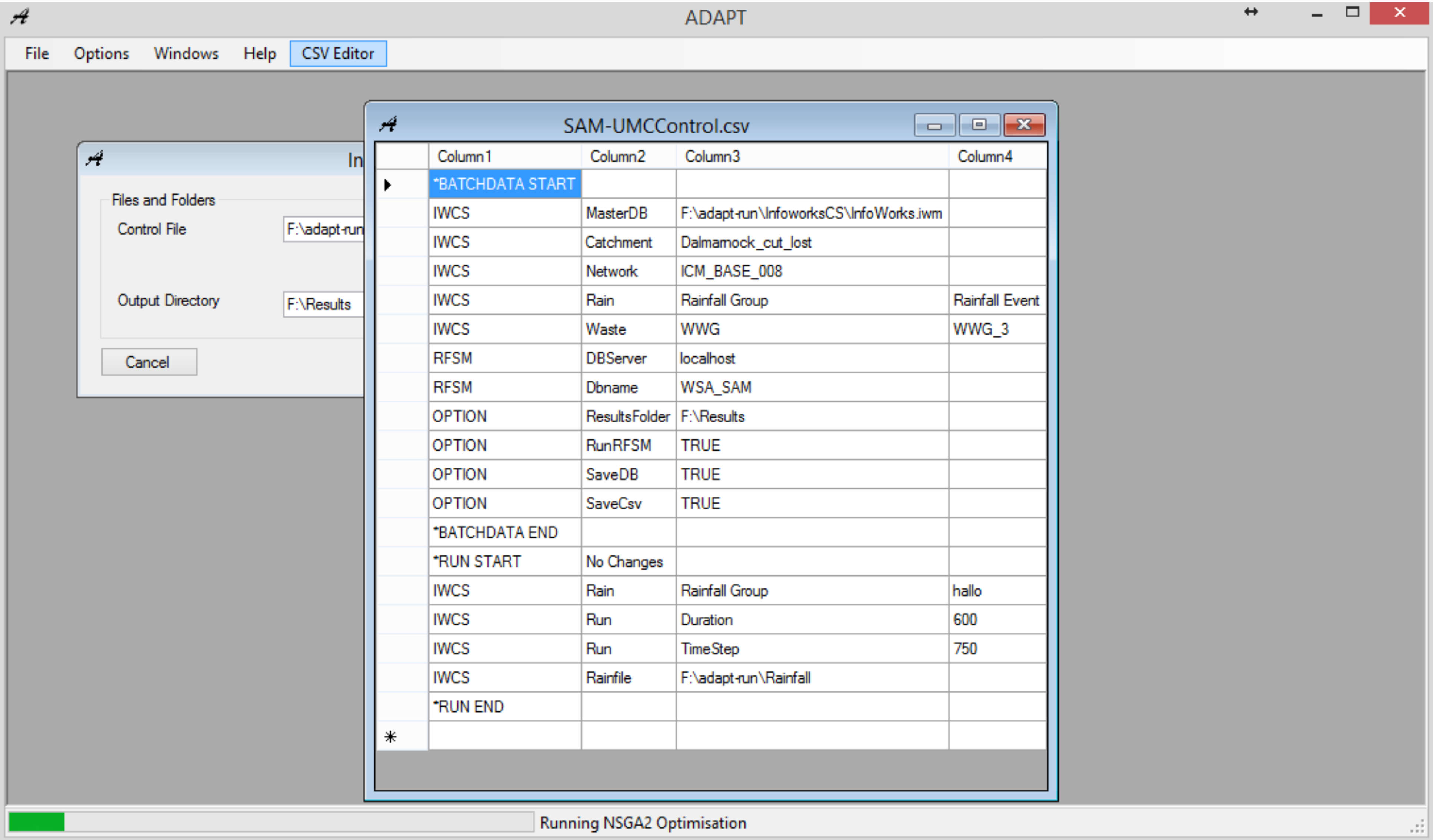

ADAPT
File
Options
Windows
Help
Input Files
Files and Folders
Control File
F:\adapt-run\Input\SAM-UMCControl.csv
Edit
Generate
Output Directory
F:\Results
Cancel
Apply
Okay
InfoWorks CS Settings
InfoWorks CS Model
Name
Infoworks
Database
F:\adapt-run\InfoworksCS\InfoWorks.iwm
InfoWorks Object Deletion
After use delete:
Nothing
All but final generation
Everything
Cancel
Apply
Okay
Rainfall Events
General
Event type:
Design Storms
Time Series
Both
Number of rainfall events per year:
100.000
Dry period length (mins):
60
Design Storms
Rainfall Events File (9)
F:\adapt-run\Rainfall\TESTSET3.csv
Edit
Generate
Extreme Rainfall Events (for EAD and flooding point level of service)
Rainfall Events File
E:\Documents\ADAPT\extremeRainfall.csv
Time series length (years):
10.000
Edit
Generate
Frequent Rainfall Events (for CSO level of service only)
Rainfall Events File
E:\Documents\ADAPT\frequentRainfall.csv
Time series length (years):
1.000
Edit
Generate
Cancel
Apply
Okay
Running NSGA2 Optimisation


ADAPT
File
Options
Windows
Help
Optimisation Options
General
Variables
Genetic Algorithm Settings
Total Population Size
80
Number of Generations
2000
Crossover Rate
1.000
Mutation Rate
0.002
Display Results Live
Estimated Number of Runs
80040
Objectives
Available Objectives
Boundary Cost
Total Cost
Level of Service
Selected Objectives
Network Cost
Expected Annual Damage
Constraints
Available Constraints
Level of Service Passed
Selected Constraints
Initial Population
Additional Initial Solutions (0)
F:\adapt-run\Input\InitialSolutions.csv
(Unmodified network always included)
Edit
Generate
Cancel
Apply
Okay
Meta-Modelling Options
Meta-Modelling Settings
Enable meta-modelling in optimisation
EAD Meta-Modelling
Dominance Meta-Modelling
Initial Training Generations
0
Subsequent Training Generations
0
Retraining Data Cache
0
Hidden Layer Size
0
Ranks to Re-Evaluate
0
Cancel
Apply
Okay
EAD Analysis
Run Parameters
Minimum no. of runs
1000
Convergence criteria
1.00000
Cancel
Apply
Okay
Level of Service Analysis
Conditions
Flooding Points (0)
F:\adapt-run\Input\FloodingPoint_LoS.csv
Create New
Edit
Generate
CSOs (0)
F:\adapt-run\Input\CSO_LoS.csv
Create New
Edit
Generate
Options
Skip events when pass or failure known
Volume failure threshold (m³)
5.000
Cancel
Apply
Okay
Running NSGA2 Optimisation


ADAPT
File
Options
Windows
Help
Optimisation Options
General
Variables
Pipes
Pipes (175)
F:\adapt-run\Input\Pipes.csv
Create New
Edit
Generate
Allow pipe widths to only increase in size from initial state
Storage Nodes
Storage Nodes (173)
F:\adapt-run\Input\StorageNodes.csv
Create New
Edit
Generate
Chamber plan area (m²):
Min: 1.00
Step: 1.00
Max: 100.00
Allow chamber plan areas to only increase in size from initial state
Orifices
Orifices (0)
F:\adapt-run\Input\Orifices.csv
Create New
Edit
Generate
Limiting discharge (m³/s):
Min: 0.010
Step: 0.100
Max: 20.000
Cancel
Apply
Okay
Level of Service Analysis
Conditions
Flooding Points (0)
F:\adapt-run\Input\FloodingPoint_LoS.csv
Create New
Edit
Generate
CSOs (0)
F:\adapt-run\Input\CSO_LoS.csv
Create New
Edit
Generate
Options
Skip events when pass or failure known
Volume failure threshold (m³) 5.000
Cancel
Apply
Okay
Boundary Points
Boundary Points
Flooding Points (0)
F:\adapt-run\Input\FloodingPoint_Boundary.csv
Create New
Edit
Generate
CSOs (0)
F:\adapt-run\Input\CSO_Boundary.csv
Create New
Edit
Generate
Outfalls (0)
F:\adapt-run\Input\Outfalls_Boundary.csv
Create New
Edit
Generate
Options
Volume differences ignored if less than 3 % or 25 m³
Cancel
Apply
Okay
Running NSGA2 Optimisation


ADAPT
File
Options
Windows
Help
Input Files
Rainfall Events
General
Event type:
Design Storms
Time Series
Both
Number of rainfall events per year:
100.000
Dry period length (mins):
60
Design Storms
Rainfall Events File (9)
F:\adapt-run\Rainfall\TESTSET3.csv
Edit
Generate
Extreme Rainfall Events (for EAD and flooding point level of service)
Rainfall Events File
E:\Documents\ADAPT\extremeRainfall.csv
Time series length (years):
10.000
Frequent Rainfall Events (for CSO level of service only)
E:\Documents\ADAPT\frequentRainfall.csv
1.000
Cancel
Apply
Okay
Rainfall Events Selector: Design Storm Events
Rainfall Files
Included Rainfall Files
Running NSGA2 Optimisation


ADAPT
File
Options
Windows
Help
Boundary Points
Boundary Points
Flooding Points (0)
F:\adapt-run\Input\FloodingPoint_Boundary.csv
Create New
Edit
Generate
CSOs (0)
F:\adapt-run\Input\CSO_Boundary.csv
Outfalls (0)
F:\adapt-run\Input\Outfalls_Boundary.csv
Options
Volume differences ignored if less than 3 % or 25 m³
Cancel
Apply
Okay
Costs
Asset Modification Cost Parameter Groups
Pipe Group Costs (1)
F:\adapt-run\Input\PipeGroup_Costs.csv
Storage Node Costs (1)
F:\adapt-run\Input\StorageNode_Costs.csv
Orifice Costs (1)
F:\adapt-run\Input\Orifice_Costs.csv
Boundary Point Cost Parameter Groups
Flooding Point Costs (1)
F:\adapt-run\Input\FloodingPoint_Costs.csv
CSO Costs (1)
F:\adapt-run\Input\CSO_Costs.csv
Outfall Costs (1)
F:\adapt-run\Input\Outfall_Costs.csv
Running NSGA2 Optimisation


ADAPT
File
Options
Windows
Help
CSV Editor
In
Files and Folders
Control File
F:\adapt-run
Output Directory
F:\Results
Cancel
SAM-UMCControl.csv
Column1 Column2 Column3 Column4
*BATCHDATA START
IWCS MasterDB F:\adapt-run\InfoworksCS\InfoWorks.iwm
IWCS Catchment Dalmarnock_cut_lost
IWCS Network ICM_BASE_008
IWCS Rain Rainfall Group Rainfall Event
IWCS Waste WWG WWG_3
RFSM DBServer localhost
RFSM Dbname WSA_SAM
OPTION ResultsFolder F:\Results
OPTION RunRFSM TRUE
OPTION SaveDB TRUE
OPTION SaveCsv TRUE
*BATCHDATA END
*RUN START No Changes
IWCS Rain Rainfall Group hallo
IWCS Run Duration 600
IWCS Run TimeStep 750
IWCS Rainfile F:\adapt-run\Rainfall
*RUN END
Running NSGA2 Optimisation

Settings Files

The listing of files necessary to allow for the proper functioning of SAM-UMC is as follows:

- CSO_Boundary.csv
- CSO_Costs.csv
- CSO_LoS.csv
- FloodingPoint_Boundary.csv
- FloodingPoint_Costs.csv
- FloodingPoint_LoS.csv
- InitialSolutions.csv
- Orifice_Costs.csv
- Orifices.csv
- Outfall_Costs.csv
- Outfalls_Boundary.csv
- PipeGroup_Costs.csv
- StorageNode_Costs.csv
- StorageNodes.csv
- Pipes.csv
- Sam_UMCControl.csv

Although it should be noted that not all files are used in the Dalmarnock optimisation performed for this thesis, it is necessary that all files are present and valid. Therefore, all files are identified and their contents during a Dalmarnock test run shown.

CSO_Boundary.csv

The CSO boundary file identifies combined drainage system overflow boundaries, and is unused for the Dalmarnock optimisation performed for this thesis.

US Node ID	Link Suffix	Cost Group

CSO_Costs.csv

The CSO costs file identifies costs associated with combined drainage system overflows. It is unused for the Dalmarnock optimisation performed for this thesis.

Cost Group	Mobilisation cost of new spills (£)	Cost per unit additional volume (£/m³)
1	1000	1000

CSO_LoS.csv

This file identifies a level of service that should be used against combined drainage system overflows when the level of service constraint is in place on the NSGA-II algorithm. This file is unused for the Dalmarnock optimisation performed for this thesis.

US Node ID	Link Suffix	Spill return period threshold (years)

FloodingPoint_Boundary.csv

This file identifies the flooding points boundaries, and is unused for the Dalmarnock optimisation performed for this thesis.

Node ID	Cost Group

FloodingPoint_Costs.csv

The flooding point costs file identifies the costs associated with various flooding points. This file is unused in the Dalmarnock optimisation performed for this thesis.

Cost Group	Mobilisation cost of new flooding (£)	Cost per unit additional volume (£/m³)
1	1000	1000

FloodingPoint_LoS.csv

The flooding point LoS file identifies the level of service required of given flooding points when the level of service restraint is being utilised during the optimisation. During the optimisation performed for this thesis, this restraint was not utilised.

Node ID	Flood return period threshold (years)

InitialSolutions.csv

This file is a utility file, which can be left empty (and is for the optimisation performed in this thesis). It allows the specification of certain initial solutions to include in the optimisation. This allows for pre-known promising solutions to be incorporated into the optimisation algorithms starting state.

This file is too large in terms of numbers of headers to easily incorporate into this document (and it contains no data anyway). Essentially it is of the form:

Pipe group 1 width (mm)	Pipe group 2 width (mm)

But the real file has a column for every pipe group specified in Pipes.csv.

Orifice_Costs.csv

This file contains the costs associated with various orifices within the model (i.e. how much it might cost to modify them. It is unused in the Dalmarnock optimisation.

Cost Group	Cost of altering orifice (£)
1	1000

Orifices.csv

The orifices file specifies all orifices that are within the model and to be used within an optimisation run. It is unused in the Dalmarnock optimisation.

US Node ID	Link Suffix	Cost Group

Outfall_Costs.csv

This file specifies the costs associated with various outfalls that may be within the model that is being run.

Cost Group	Mobilisation cost of volume increase (£)	Cost per unit additional volume (£/m³)
1	10000	1000

Outfalls_Boundary.csv

This file identifies the outfalls boundaries and is unused within the Dalmarnock optimisation performed for this thesis.

Node ID	Cost Group

PipeGroup_Costs.csv

This file identifies the costs associated with specific pipe groups to which the different pipe groups specified in pipes.csv can belong.

Cost Group	Base cost of altering pipe group (£)	Intervention cost per unit volume (£/m³)
1	1000	1000

StorageNode_Costs.csv

The storage node costs file identifies the costs associated with altering storage nodes during the optimisation run.

Cost Group	Base cost of altering node (£)	Intervention cost per unit plan area (£/m²)
1	10000	500

StorageNodes.csv

This file identifies the storage nodes present in the model that can be altered during the optimisation process.

Node ID	Cost Group
new mh11	1
newmh10	1
newmh12	1
newmh7	1
newmh8	1
newmh9	1
NS59644401	1
NS59645401	1
NS59645407	1
NS59645507	1
NS59645508	1

Node ID	Cost Group
NS59645510	1
NS59645512	1
NS59646501	1
NS59646511	1
NS59647505	1
NS59647556	1
NS59648404	1
NS59648407	1
NS59648410	1
NS59649302	1
NS60632801	1
NS60632802	1
NS60632804	1
NS60632901	1
NS60632902	1
NS60633401	1
NS60633501	1
NS60633502	1
NS60633601	1
NS60633602	1
NS60633705	1
NS60633708	1
NS60633801	1
NS60633802	1
NS60633803	1
NS60633806	1
NS60633901	1
NS60633902	1
NS60633905	1
NS60634201	1
NS60634301	1
NS60634302	1
NS60634303	1
NS60634401	1
NS60634403	1
NS60634404	1
NS60634405	1
NS60634409	1

Node ID	Cost Group
NS60634701	1
NS60634702	1
NS60634710	1
NS60634909	1
NS60635203	1
NS60635302	1
NS60635701	1
NS60635812	1
NS60635813	1
NS60635901	1
NS60635911	1
NS60635922	1
NS60635935	1
NS60635936	1
NS60635945	1
NS60636103	1
NS60636105	1
NS60636201	1
NS60636203	1
NS60636301	1
NS60636302	1
NS60636303	1
NS60636305	1
NS60636401	1
NS60636403	1
NS60636404	1
NS60636408	1
NS60636409	1
NS60636501	1
NS60636503	1
NS60636506	1
NS60636604	1
NS60636605	1
NS60636701	1
NS60636705	1
NS60636707	1
NS60636801	1
NS60636812	1

Node ID	Cost Group
NS60636815	1
NS60636901	1
NS60636902	1
NS60636909	1
NS60636911	1
NS60636916	1
NS60637001	1
NS60637005	1
NS60637101	1
NS60637102	1
NS60637103	1
NS60637104	1
NS60637106	1
NS60637108	1
NS60637110	1
NS60637201	1
NS60637202	1
NS60637207	1
NS60637301	1
NS60637501	1
NS60637506	1
NS60637705	1
NS60637907	1
NS60637916	1
NS60637920	1
NS60638002	1
NS60638101	1
NS60638102	1
NS60638107	1
NS60638204	1
NS60638206	1
NS60638208	1
NS60638209	1
NS60638301	1
NS60638303	1
NS60638406	1
NS60638501	1
NS60638504	1

Node ID	Cost Group
NS60638507	1
NS60638601	1
NS60638610	1
NS60638715	1
NS60639101	1
NS60639105	1
NS60639107	1
NS60639203	1
NS60639301	1
NS60639603	1
NS60639606	1
NS60639704	1
NS60641102	1
NS60641201	1
NS60641202	1
NS60641203	1
NS60641208	1
NS60641305	1
NS60641309	1
NS60641310	1
NS60642003	1
NS60642004	1
NS60642005	1
NS60642006	1
NS60642009	1
NS60642010	1
NS60642011	1
NS60642103	1
NS60643001	1
NS60643003	1
NS60643007	1
NS60645002	1
NS60645005	1
NS60645009	1
NS60646004	1
NS61620299	1
NS61620901	1
NS61630003	1

Node ID	Cost Group
NS61630102	1
NS61630105	1
NS61630202	1
NS61631001	1
NS61631002	1
NS61631102	1
NS61631104	1
NS61631105	1
NS61631106	1
NS61632101	1

SAM-UMCControl.csv

This control file is stored in comma-separated-value format (represented here as a table for easier reading). It consists of settings to enable SAM-UMC to run correctly. Only the “BATCHDATA” settings are used during the optimisation runs, as a correct “RUN” section is generated for each iteration.

Further information on what individual settings are for, how they can be set, and what additional settings are available, can be obtained by reading the SAM-UMC documentation (Wills, 2013).

*BATCHDATA START			
IWCS	MasterDB	F:\adapt-run\InfoworksCS\InfoWorks.iwm	
IWCS	Catchment	Dalmarnock_cut_lost	
IWCS	Network	ICM_BASE_008	
IWCS	Rain	Rainfall Group	Rainfall Event
IWCS	Waste	WWG	WWG_3
RFSM	DBServer	localhost	
RFSM	Dbname	WSA_SAM	
OPTION	ResultsFolder	F:\Results	
OPTION	RunRFSM	TRUE	
OPTION	SaveDB	TRUE	
OPTION	SaveCsv	TRUE	
*BATCHDATA END			
*RUN START	No Changes		
IWCS	Rain	Rainfall Group	hallo
IWCS	Run	Duration	600
IWCS	Run	TimeStep	750
IWCS	Rainfile	F:\adapt-run\Rainfall	
*RUN END			

Pipes.csv

The pipes file defines pipe groups comprised of pipes within the model. Each pipe group will be modified as one unified entity. I.e. if pipe group 1 is comprised of pipes A, B and C, then if A is increase to the next size up, B and C will be similarly increased. This allows for an Engineer to identify pipes that form one unified entity although identified as individual pipes within the model. For the Dalmarnock optimisation, each pipe that is being optimised is identified as a unique pipe group, therefore each pipe is treated individually within the optimisation.

US Node ID	Link Suffix	Pipe Group	Cost Group
new mh11	1	1	1
newmh10	1	2	1
newmh12	1	3	1
newmh7	1	4	1
newmh8	1	5	1
newmh9	1	6	1
NS59645401	1	7	1
NS59645407	1	8	1
NS59645507	1	9	1
NS59645508	1	10	1
NS59645510	1	11	1
NS59645512	1	12	1
NS59646501	1	13	1
NS59646511	1	14	1
NS59647505	1	15	1
NS59647556	1	16	1
NS59648404	1	17	1
NS59648407	1	18	1
NS59648410	1	19	1
NS59649302	1	20	1
NS60632801	1	21	1
NS60632802	1	22	1
NS60632804	1	23	1

US Node ID	Link Suffix	Pipe Group	Cost Group
NS60632901	1	24	1
NS60632902	1	25	1
NS60633401	1	26	1
NS60633501	1	27	1
NS60633502	1	28	1
NS60633601	1	29	1
NS60633602	1	30	1
NS60633705	1	31	1
NS60633708	1	32	1
NS60633801	1	33	1
NS60633802	1	34	1
NS60633802	2	35	1
NS60633803	1	36	1
NS60633806	1	37	1
NS60633901	1	38	1
NS60633902	1	39	1
NS60633905	1	40	1
NS60634201	1	41	1
NS60634301	1	42	1
NS60634302	1	43	1
NS60634303	1	44	1
NS60634401	1	45	1
NS60634403	1	46	1
NS60634404	1	47	1
NS60634405	1	48	1
NS60634409	1	49	1
NS60634701	1	50	1
NS60634702	1	51	1
NS60634710	1	52	1
NS60634909	1	53	1
NS60635203	1	54	1
NS60635302	1	55	1
NS60635701	1	56	1
NS60635812	1	57	1
NS60635813	1	58	1
NS60635901	1	59	1
NS60635911	1	60	1
NS60635922	1	61	1

US Node ID	Link Suffix	Pipe Group	Cost Group
NS60635935	1	62	1
NS60635936	1	63	1
NS60635945	1	64	1
NS60636103	1	65	1
NS60636105	1	66	1
NS60636201	1	67	1
NS60636203	1	68	1
NS60636301	1	69	1
NS60636302	1	70	1
NS60636303	1	71	1
NS60636305	1	72	1
NS60636401	1	73	1
NS60636403	1	74	1
NS60636404	1	75	1
NS60636408	1	76	1
NS60636408	2	77	1
NS60636409	1	78	1
NS60636501	1	79	1
NS60636503	1	80	1
NS60636506	1	81	1
NS60636604	1	82	1
NS60636605	1	83	1
NS60636701	1	84	1
NS60636705	1	85	1
NS60636707	1	86	1
NS60636801	1	87	1
NS60636812	1	88	1
NS60636815	1	89	1
NS60636901	1	90	1
NS60636902	1	91	1
NS60636909	1	92	1
NS60636911	2	93	1
NS60636916	1	94	1
NS60636916	2	95	1
NS60637001	1	96	1
NS60637005	1	97	1
NS60637101	1	98	1
NS60637102	1	99	1

US Node ID	Link Suffix	Pipe Group	Cost Group
NS60637103	1	100	1
NS60637104	1	101	1
NS60637106	1	102	1
NS60637108	1	103	1
NS60637110	1	104	1
NS60637201	1	105	1
NS60637202	1	106	1
NS60637207	1	107	1
NS60637301	1	108	1
NS60637501	1	109	1
NS60637506	1	110	1
NS60637705	1	111	1
NS60637907	1	112	1
NS60637916	1	113	1
NS60637920	1	114	1
NS60638002	1	115	1
NS60638101	1	116	1
NS60638102	1	117	1
NS60638107	1	118	1
NS60638204	1	119	1
NS60638206	1	120	1
NS60638208	1	121	1
NS60638209	1	122	1
NS60638209	2	123	1
NS60638301	1	124	1
NS60638303	1	125	1
NS60638406	1	126	1
NS60638501	1	127	1
NS60638504	1	128	1
NS60638507	1	129	1
NS60638601	1	130	1
NS60638610	1	131	1
NS60638715	1	132	1
NS60639101	1	133	1
NS60639105	1	134	1
NS60639107	1	135	1
NS60639203	1	136	1
NS60639301	1	137	1

US Node ID	Link Suffix	Pipe Group	Cost Group
NS60639603	1	138	1
NS60639606	1	139	1
NS60639704	1	140	1
NS60641102	1	141	1
NS60641201	1	142	1
NS60641202	1	143	1
NS60641203	1	144	1
NS60641208	1	145	1
NS60641305	1	146	1
NS60641309	1	147	1
NS60641310	1	148	1
NS60642003	1	149	1
NS60642004	1	150	1
NS60642005	1	151	1
NS60642006	1	152	1
NS60642009	1	153	1
NS60642010	1	154	1
NS60642011	1	155	1
NS60642103	1	156	1
NS60643001	1	157	1
NS60643003	1	158	1
NS60643007	1	159	1
NS60645002	1	160	1
NS60645005	1	161	1
NS60645009	1	162	1
NS60646004	1	163	1
NS61620901	1	164	1
NS61630003	1	165	1
NS61630102	1	166	1
NS61630105	1	167	1
NS61630202	1	168	1
NS61631001	1	169	1
NS61631002	1	170	1
NS61631102	1	171	1
NS61631104	1	172	1
NS61631105	1	173	1
NS61631106	1	174	1
NS61632101	1	175	1

Appendix IV – Decision Variable Details

This appendix contains information on the initial value of the decision variables used in our runs of the Dalmarnock system.

Initial Node Values

Node ID	Node Type	System Type	x (m)	y (m)	Ground Level (m AD)	Flood Level (m AD)	Chamber Floor Level (m AD)	Chamber Roof Level (m AD)	Chamber Plan Area (m2)	Shaft Plan Area (m2)
new mh11	Manhole	combined	261091	662498.2	7.12	7.12	1.027	2.627	4.4	4.4
newmh10	Manhole	combined	261095.6	662604.5	7.28	7.28	1.493	3.093	4.4	4.4
newmh12	Manhole	combined	261089.1	662395.5	7.29	7.29	0.578	2.178	4.4	4.4
newmh7	Manhole	combined	261126.1	662907.8	8.77	8.77	3.156	4.756	4.4	4.4
newmh8	Manhole	combined	261101.2	662812.6	8.44	8.44	3.042	4.642	4.4	4.4
newmh9	Manhole	combined	261102.1	662715.5	7.34	7.34	2.931	4.531	4.4	4.4
NS59644401	Outfall	storm	259504	664449.3	3.37	3.37	1.79	3.2	3.7	3.7
NS59645401	Manhole	storm	259516.8	664475.8	6.93	6.93	2.33	4.18	15.8	15.8
NS59645407	Manhole	storm	259510	664455	6.32	6.32	2.33	4.18	15.8	15.8
NS59645507	Manhole	storm	259534.7	664509.1	7.3	7.3	2.34	4.16	15.6	15.6
NS59645508	Manhole	storm	259578.4	664590.7	5.71	5.71	2.35	4.18	15.8	15.8
NS59645510	Manhole	storm	259578.3	664588.9	5.71	5.71	2.35	4.19	15.8	15.8
NS59645512	Manhole	storm	259588	664595	5.63	5.63	2.69	4.43	10	10
NS59646501	Manhole	storm	259606	664590	5.74	5.74	2.84	4.66	10	10

NS59646511	Manhole	storm	259666	664562	6	6	3.41	5.26	10	10
NS59647505	Manhole	storm	259742.4	664525.4	6.61	6.61	3.505	5.355	5.4	5.4
NS59647556	Manhole	storm	259762	664508	7.05	7.05	3.59	5.41	5.2	5.2
NS59648404	Manhole	storm	259837.5	664434.5	7.7	7.7	3.974	5.794	5.2	5.2
NS59648407	Manhole	storm	259864.8	664427.4	8.25	8.25	4.05	5.9	5.4	5.4
NS59648410	Manhole	storm	259823	664439	7.51	7.51	3.91	5.73	5.2	5.2
NS59649302	Manhole	storm	259942.7	664380.7	8.61	8.61	4.41	6.26	5.4	5.4
NS60632801	Manhole	combined	260290	663841	9.29	9.29	4.653	6.153	4	4
NS60632802	Manhole	combined	260277	663886	8.97	8.97	4.688	6.198	4	4
NS60632804	Manhole	combined	260297	663813	9.57	9.57	4.632	6.232	4.4	4.4
NS60632901	Manhole	combined	260259	663931	9.25	9.25	4.65	6.21	4	4
NS60632902	Manhole	combined	260254	663977	9.44	9.44	4.65	6.16	4	4
NS60633401	Manhole	combined	260396	663482	9.63	9.63	4.376	5.976	4.4	4.4
NS60633501	Manhole	combined	260381	663533	9.38	9.38	4.415	6.015	4.4	4.4
NS60633502	Manhole	combined	260371	663576	9.5	9.5	4.42	6.02	4.4	4.4
NS60633601	Manhole	combined	260357	663616	9.52	9.52	4.429	6.039	4.4	4.4
NS60633602	Manhole	combined	260338	663661	9.48	9.48	4.515	6.115	4.4	4.4
NS60633705	Manhole	combined	260302	663795	9.49	9.49	4.618	6.218	4.4	4.4
NS60633708	Manhole	combined	260316	663761	9.48	9.48	4.59	6.2	4.4	4.4
NS60633801	Manhole	combined	260385	663817	8.98	8.98	5.47	6.25	1.7	1.7
NS60633802	Manhole	combined	260380	663829	9	9	4.75	6.293	3	3
NS60633803	Manhole	combined	260362	663872	9.07	9.07	5.53	6.34	1.7	1.7
NS60633806	Manhole	combined	260383	663822	9.04	9.04	5.475	6.285	1.7	1.7
NS60633901	Manhole	combined	260350	663903	9.12	9.12	5.87	6.68	1.7	1.7
NS60633902	Manhole	combined	260341	663935	9.19	9.19	6	6.78	1.7	1.7

NS60633905	Manhole	combined	260398	663942	9.29	9.29	6.5	6.95	1.2	1.2
NS60634201	Manhole	combined	260483	663292	7.85	7.85	4.29	5.89	4.4	4.4
NS60634301	Manhole	combined	260440	663352	8.8	8.8	4.337	5.957	4.4	4.4
NS60634302	Manhole	combined	260461	663320	8.24	8.24	5.23	5.68	1.2	1.2
NS60634303	Manhole	combined	260460	663314	8.15	8.15	4.31	5.91	4.4	4.4
NS60634401	Manhole	combined	260463	663481	8.56	8.56	6.534	6.909	1	1
NS60634403	Manhole	combined	260442	663428	8.74	8.74	5.8	6.25	1.2	1.2
NS60634404	Manhole	combined	260458	663432	8.72	8.72	6.143	6.588	1	1
NS60634405	Manhole	combined	260470	663432	8.81	8.81	6.71	7.01	1	1
NS60634409	Manhole	combined	260418	663409	9.75	9.75	4.341	5.941	4.4	4.4
NS60634701	Manhole	combined	260406	663757	8.88	8.88	4.725	5.845	1.5	1.5
NS60634702	Manhole	combined	260403	663773	8.92	8.92	5.437	6.112	1.6	1.6
NS60634710	Manhole	combined	260475	663752	8.97	8.97	4.812	5.412	1.5	1.5
NS60634909	Manhole	storm	260490.5	663985.9	9.69	9.69	6.661	8.661	8.4	8.4
NS60635203	Manhole	combined	260559	663290	8.77	8.77	4.211	6.045	4.4	4.4
NS60635302	Manhole	combined	260559	663315	8.63	8.63	6.58	6.88	1	1
NS60635701	Manhole	combined	260525	663748	9.21	9.21	4.854	5.454	1.5	1.5
NS60635812	Manhole	combined	260563	663860	9.7	9.7	8.1	8.4	1	1
NS60635813	Manhole	combined	260510.8	663883.8	9.71	9.71	4.94	6.14	3	3
NS60635901	Manhole	foul	260551	663906	9.59	9.59	6.77	6.92	1	1
NS60635911	Manhole	storm	260571	663965	9.43	9.43	7.9	8.125	1	1
NS60635922	Manhole	combined	260554	663902	9.73	9.73	5.19	6.9	3	3
NS60635935	Manhole	storm	260568.5	663983.2	9.67	9.67	7.08	8.45	15.4	15.4
NS60635936	Manhole	storm	260577.3	663973.9	9.45	9.45	7.11	8.48	15.4	15.4
NS60635945	Manhole	storm	260522	663985	9.77	9.77	6.967	8.27	8.4	8.4

NS60636103	Manhole	combined	260660	663188	9	9	3.847	6.127	4.4	4.4
NS60636105	Manhole	combined	260666	663188	8.88	8.88	3.84	5.44	4.4	4.4
NS60636201	Manhole	combined	260656	663263	9.24	9.24	4.015	5.615	4.4	4.4
NS60636203	Manhole	combined	260640	663286	9.22	9.22	4.113	5.738	4.4	4.4
NS60636301	Manhole	combined	260662	663340	9.33	9.33	6.202	6.742	1.3	1.3
NS60636302	Manhole	combined	260663	663371	9.27	9.27	6.286	6.811	1.3	1.3
NS60636303	Manhole	combined	260661	663303	9.39	9.39	6.165	6.71	1.3	1.3
NS60636305	Manhole	combined	260691	663374	9.37	9.37	6.628	7.028	1	1
NS60636401	Manhole	combined	260668	663465	9.48	9.48	6.273	6.798	1.3	1.3
NS60636403	Manhole	combined	260665	663412	9.4	9.4	6.377	6.902	1.3	1.3
NS60636404	Manhole	combined	260643	663442	9.76	9.76	7.031	7.256	1	1
NS60636408	Manhole	combined	260667	663443	9.47	9.47	6.447	6.822	1	1
NS60636409	Manhole	combined	260669	663475	9.45	9.45	6.186	6.711	1.3	1.3
NS60636501	Manhole	combined	260665	663576	9.5	9.5	7.625	7.9	1	1
NS60636503	Manhole	combined	260603	663575	10.45	10.45	8.1	8.325	1	1
NS60636506	Manhole	combined	260678	663572	9.37	9.37	5.632	6.232	1.5	1.5
NS60636604	Manhole	combined	260676	663693	10.13	10.13	5.24	5.92	1.5	1.5
NS60636605	Manhole	combined	260674	663642	9.72	9.72	5.37	5.97	1.5	1.5
NS60636701	Manhole	combined	260677	663744	10.28	10.28	4.926	5.526	1.5	1.5
NS60636705	Manhole	combined	260686	663796	10.36	10.36	5.5	5.975	1.2	1.2
NS60636707	Manhole	combined	260681	663744	10.38	10.38	4.93	5.599	1.5	1.5
NS60636801	Manhole	combined	260604	663859	9.78	9.78	7.179	7.889	1.4	1.4
NS60636812	Manhole	combined	260690	663852	10.43	10.43	5.53	5.98	1.2	1.2
NS60636815	Manhole	combined	260604	663855	9.79	9.79	7.316	8.006	1.4	1.4
NS60636901	Manhole	storm	260607.7	663956.9	9.4	9.4	7.28	8.58	13	13

NS60636902	Manhole	foul	260610	663966	9.22	9.22	5.284	6.65	3	3
NS60636909	Manhole	combined	260600	663922	10.1	10.1	5.254	6.484	3	3
NS60636911	Manhole	storm	260681.2	663957.4	9.85	9.85	7.57	9.01	12.3	12.3
NS60636916	Manhole	storm	260614	663936	9.85	9.85	7.35	8.79	12.3	12.3
NS60637001	Manhole	combined	260789	663086	9.67	9.67	6.44	6.89	1.2	1.2
NS60637005	Manhole	combined	260784	663044	9.86	9.86	6.55	7	1.2	1.2
NS60637101	Manhole	combined	260780	663180	9.25	9.25	3.809	5.409	4.4	4.4
NS60637102	Manhole	combined	260726	663183	9.05	9.05	3.821	5.421	4.4	4.4
NS60637103	Manhole	combined	260791	663112	9.63	9.63	6.39	6.84	1.2	1.2
NS60637104	Manhole	combined	260794	663136	9.53	9.53	6.25	6.78	1.2	1.2
NS60637106	Manhole	combined	260795	663173	9.32	9.32	5.115	5.565	1.2	1.2
NS60637108	Manhole	combined	260794	663131	9.59	9.59	6.34	6.79	1.2	1.2
NS60637110	Manhole	combined	260796	663178	9.35	9.35	3.805	6.375	4.4	4.4
NS60637201	Manhole	combined	260797	663217	9.51	9.51	6.57	6.955	1	1
NS60637202	Manhole	combined	260799	663205	9.37	9.37	6.54	6.935	1	1
NS60637207	Manhole	combined	260796	663238	9.46	9.46	6.77	7.145	1	1
NS60637301	Manhole	combined	260751	663371	9.57	9.57	6.875	7.27	1	1
NS60637501	Manhole	combined	260775	663569	9.34	9.34	5.982	6.432	1.2	1.2
NS60637506	Manhole	combined	260749	663571	9.7	9.7	5.929	6.419	1.2	1.2
NS60637705	Manhole	combined	260795	663738	10.22	10.22	5.18	6.449	1.3	1.3
NS60637907	Manhole	combined	260700	663956	9.83	9.83	5.54	5.99	1.2	1.2
NS60637916	Manhole	storm	260779	663936.4	9.934	9.934	7.9	9.3	9.1	9.1
NS60637920	Manhole	storm	260739.1	663963.1	10.04	10.04	7.766	9.166	9.1	9.1
NS60638002	Manhole	combined	260829	663016	9.66	9.66	6.675	7.125	1.2	1.2
NS60638101	Manhole	combined	260892	663170	9.6	9.6	3.769	5.369	4.4	4.4

NS60638102	Manhole	combined	260833	663174	9.55	9.55	3.782	5.397	4.4	4.4
NS60638107	Manhole	combined	260880	663170	9.61	9.61	3.772	5.372	4.4	4.4
NS60638204	Manhole	combined	260881	663293	9.9	9.9	7.347	7.722	1	1
NS60638206	Manhole	combined	260831	663294	9.58	9.58	7.076	7.451	1	1
NS60638208	Manhole	combined	260801	663273	9.62	9.62	6.792	7.167	1	1
NS60638209	Manhole	combined	260802	663297	9.73	9.73	6.916	7.291	1	1
NS60638301	Manhole	combined	260805	663369	9.55	9.55	6.904	7.279	1	1
NS60638303	Manhole	combined	260803	663315	9.45	9.45	6.913	7.288	1	1
NS60638406	Manhole	combined	260806	663400	9.45	9.45	7.25	7.625	1	1
NS60638501	Manhole	combined	260812	663567	9.36	9.36	6.001	6.471	1.2	1.2
NS60638504	Manhole	combined	260868	663566	9.87	9.87	6.52	6.97	1.2	1.2
NS60638507	Manhole	combined	260813	663567	9.43	9.43	6.025	6.475	1.2	1.2
NS60638601	Manhole	combined	260817	663648	9.71	9.71	6.32	6.77	1.2	1.2
NS60638610	Manhole	combined	260814	663610	9.38	9.38	6.181	6.631	1.2	1.2
NS60638715	Manhole	combined	260802	663747	10.34	10.34	6.641	6.941	1	1
NS60639101	Manhole	combined	260952	663166	9.7	9.7	3.757	5.357	4.4	4.4
NS60639105	Manhole	combined	260936	663167	9.62	9.62	3.76	5.36	4.4	4.4
NS60639107	Manhole	foul	260997	663162	9.7	9.7	3.747	5.347	4.4	4.4
NS60639203	Manhole	combined	260960	663291	10.26	10.26	7.944	8.329	1	1
NS60639301	Manhole	combined	260938	663321	10.39	10.39	8.942	9.317	1	1
NS60639603	Manhole	storm	260933	663629	10	10	6.562	7.087	1.3	1.3
NS60639606	Manhole	combined	260918	663686	10.09	10.09	6.3	6.825	1.3	1.3
NS60639704	Manhole	combined	260907	663732	10.08	10.08	5.47	6.605	1.3	1.3
NS60641102	Manhole	combined	260176	664200	8.78	8.78	4.978	5.913	2.2	2.2
NS60641201	Manhole	combined	260150	664240	9.4	9.4	6.01	6.96	2.2	2.2

NS60641202	Manhole	combined	260132	664288	10.32	10.32	7.111	8.041	2.2	2.2
NS60641203	Manhole	combined	260135	664268	8.97	8.97	6.688	7.638	2.2	2.2
NS60641208	Manhole	storm	260118	664245	9.02	9.02	4.88	6.73	5.4	5.4
NS60641305	Manhole	combined	260135	664352	14.81	14.81	8.66	9.57	2.2	2.2
NS60641309	Manhole	combined	260146	664313	12.38	12.38	7.726	8.656	2.2	2.2
NS60641310	Manhole	combined	260161	664326	14.43	14.43	8.149	9.079	2.2	2.2
NS60642003	Manhole	combined	260275	664055	8.71	8.71	4.83	6.24	2.2	2.2
NS60642004	Manhole	combined	260250	664015	9.06	9.06	4.67	6.17	4	4
NS60642005	Manhole	combined	260247	664030	9.06	9.06	4.67	6.17	4	4
NS60642006	Manhole	combined	260279	664058	8.74	8.74	5.133	6.363	2.2	2.2
NS60642009	Manhole	storm	260236	664090	8.29	8.29	5.46	7.31	5.4	5.4
NS60642010	Manhole	storm	260271	664048	8.79	8.79	5.63	7.48	10	10
NS60642011	Manhole	storm	260263	664053	8.76	8.76	5.58	7.43	5.4	5.4
NS60642103	Manhole	combined	260230	664137.7	8.96	8.96	4.953	6.303	3.5	3.5
NS60643001	Manhole	storm	260311.4	664029.7	8.84	8.84	5.78	7.78	10	10
NS60643003	Manhole	combined	260306	664054	8.42	8.42	6.012	7.152	2.2	2.2
NS60643007	Manhole	storm	260365.6	664004.7	9.54	9.54	5.948	7.948	6	6
NS60645002	Manhole	combined	260548	664079	11.15	11.15	6.86	7.535	1.6	1.6
NS60645005	Manhole	combined	260598	664059	9.83	9.83	5.979	6.75	1.6	1.6
NS60645009	Manhole	foul	260584	664031	9.87	9.87	7.03	7.48	1	1
NS60646004	Manhole	combined	260645	664040	9.8	9.8	5.755	6.955	3	3
NS61620299	Outfall	combined	261084.5	662286.4	6.08	6.08	3.23	4.44	3	3
NS61620901	Manhole	combined	261029	662992	9.03	9.03	6.629	7.079	1.2	1.2
NS61630003	Manhole	combined	261036	663031	9.2	9.2	6.24	6.76	1.2	1.2
NS61630102	Manhole	combined	261054	663158	9.41	9.41	3.6	5.2	4.4	4.4

NS61630105	Manhole	combined	261063	663157	9.47	9.47	3.58	5.18	4.4	4.4
NS61630202	Manhole	combined	261000	663289	10.32	10.32	8.015	8.39	1	1
NS61631001	Manhole	combined	261161	663001	8.72	8.72	3.27	4.89	4.4	4.4
NS61631002	Manhole	combined	261176	663051	8.55	8.55	3.33	4.93	4.4	4.4
NS61631102	Manhole	combined	261154	663120	8.55	8.55	3.33	4.97	4.4	4.4
NS61631104	Manhole	combined	261115	663125	8.74	8.74	3.43	5.03	4.4	4.4
NS61631105	Manhole	combined	261194	663107	8.36	8.36	3.33	4.93	4.4	4.4
NS61631106	Manhole	combined	261100	663134	8.86	8.86	3.47	5.07	4.4	4.4
NS61632101	Manhole	combined	261207	663154	8.11	8.11	5.62	6.6	1.6	1.6

Initial Pipe Values

US Node ID	Link Suffix	DS Node ID	System Type	Length (m)	Shape ID	Width (mm)	Height (mm)	Conduit full capacity (m3/s)
new mh11	1	newmh12	combined	102.7	CIRC	1600	1600	4.905
newmh10	1	new mh11	combined	106.4	CIRC	1600	1600	4.905
newmh12	1	NS61620299	combined	109.2	CIRC	1600	1600	4.905
newmh7	1	newmh8	combined	98.5	CIRC	1600	1600	2.511
newmh8	1	newmh9	combined	97.1	CIRC	1600	1600	2.511
newmh9	1	newmh10	combined	111.2	CIRC	1600	1600	8.439
NS59645401	1	NS59645407	storm	21.9	FM424	3720	1850	0
NS59645407	1	NS59644401	storm	8.3	CIRC	1400	1400	13.223
NS59645507	1	NS59645401	storm	37.8	RECT	3700	1800	5.167
NS59645508	1	NS59645510	storm	1.8	FM433	3720	1830	-18.186
NS59645510	1	NS59645507	storm	90.9	AND22	3700	1820	2.132
NS59645512	1	NS59645508	storm	10.5	AND21	1300	1500	3.702
NS59646501	1	NS59645512	storm	18.7	AND20	2800	1740	12.861
NS59646511	1	NS59646501	storm	66.9	AND19	2800	1820	13.972
NS59647505	1	NS59646511	storm	86.3	AND17	1850	1850	3.579
NS59647556	1	NS59647505	storm	26.8	AND17	1800	1820	5.746
NS59648404	1	NS59648410	storm	15.2	AND17	1800	1820	6.612
NS59648407	1	NS59648404	storm	28.3	AND17	1800	1820	5.284
NS59648410	1	NS59647556	storm	95.7	AND17	1800	1820	5.894

NS59649302	1	NS59648407	storm	94.1	AND17	1850	1850	6.68
NS60632801	1	NS60632804	combined	28.9	CIRC	1500	1500	1.701
NS60632802	1	NS60632801	combined	46.8	CIRC	1500	1500	1.7
NS60632804	1	NS60633705	combined	18.7	CIRC	1600	1600	2.029
NS60632901	1	NS60632802	combined	48.5	CIRC	1500	1500	0.98
NS60632902	1	NS60632901	combined	46.3	CIRC	1500	1500	0
NS60633401	1	NS60634409	combined	76.4	CIRC	1600	1600	1.587
NS60633501	1	NS60633401	combined	53.2	CIRC	1600	1600	2.009
NS60633502	1	NS60633501	combined	44.1	CIRC	1600	1600	0.752
NS60633601	1	NS60633502	combined	42.4	CIRC	1600	1600	1.076
NS60633602	1	NS60633601	combined	48.8	CIRC	1600	1600	2.925
NS60633705	1	NS60633708	combined	36.8	CIRC	1600	1600	2.018
NS60633708	1	NS60633602	combined	102.5	CIRC	1600	1600	2.13
NS60633801	1	NS60634702	combined	47.5	CIRC	675	675	0.228
NS60633802	1	NS60632804	combined	84.5	CIRC	1200	1200	1.297
NS60633802	2	NS60633806	combined	7.6	EGG	690	810	0.306
NS60633803	1	NS60633802	combined	46.6	EGG	690	810	0.3
NS60633806	1	NS60633801	combined	5.4	EGG	690	780	0.276
NS60633901	1	NS60633803	combined	33.2	EGG	670	790	0.876
NS60633902	1	NS60633901	combined	33.2	EGG	690	780	0.498
NS60633905	1	NS60633902	combined	57.6	CIRC	450	450	0.202
NS60634201	1	NS60635203	combined	76	CIRC	1600	1600	2.383
NS60634301	1	NS60634303	combined	42.9	CIRC	1600	1600	1.856
NS60634302	1	NS60634303	combined	6.1	CIRC	450	450	0.986
NS60634303	1	NS60634201	combined	31.8	CIRC	1600	1600	1.855

NS60634401	1	NS60634404	combined	49.3	CIRC	375	375	0.129
NS60634403	1	NS60634409	combined	30.6	CIRC	450	450	0.425
NS60634404	1	NS60634403	combined	16.5	CIRC	300	300	0.117
NS60634405	1	NS60634404	combined	12	CIRC	300	300	0.181
NS60634409	1	NS60634301	combined	61.1	CIRC	1600	1600	-1.192
NS60634701	1	NS60633708	combined	90.1	CIRC	600	600	0.207
NS60634702	1	NS60634701	combined	16.3	CIRC	375	375	0.076
NS60634710	1	NS60634701	combined	69.2	CIRC	600	600	0.145
NS60634909	1	NS60643007	storm	126.8	CIRC	2000	2000	10.004
NS60635203	1	NS60636203	combined	81.1	CIRC	1600	1600	2.225
NS60635302	1	NS60635203	combined	25.1	CIRC	300	300	0.161
NS60635701	1	NS60634710	combined	50.2	CIRC	600	600	0.159
NS60635812	1	NS60636815	combined	41.3	CIRC	300	300	0.121
NS60635813	1	NS60633802	combined	141.8	CIRC	1200	1200	1.27
NS60635901	1	NS60635922	foul	5	CIRC	150	150	0.009
NS60635911	1	NS60635936	storm	10.9	CIRC	225	225	0.039
NS60635922	1	NS60635813	combined	46.9	CIRC	1200	1200	2.537
NS60635935	1	NS60635945	storm	46.5	ARCH	2500	1300	4.59
NS60635936	1	NS60635935	storm	12.9	RECT	3660	1370	10.248
NS60635945	1	NS60634909	storm	31.5	ARCH	2500	1300	9.313
NS60636103	1	NS60636105	combined	6	CIRC	1600	1600	2.516
NS60636105	1	NS60637102	combined	60.2	CIRC	1600	1600	1.318
NS60636201	1	NS60636103	combined	75.8	CIRC	1600	1600	3.494
NS60636203	1	NS60636201	combined	28	CIRC	1600	1600	4.378
NS60636301	1	NS60636303	combined	37	CIRC	525	525	0.083

NS60636302	1	NS60636301	combined	31	CIRC	525	525	0.184
NS60636303	1	NS60636103	combined	115	CIRC	525	525	0.273
NS60636305	1	NS60636302	combined	28.2	CIRC	375	375	0.176
NS60636401	1	NS60636409	combined	10	CIRC	525	525	0.362
NS60636403	1	NS60636302	combined	41	CIRC	525	525	0.184
NS60636404	1	NS60636408	combined	24	CIRC	225	225	0.064
NS60636408	1	NS60636401	combined	22	CIRC	375	375	0.141
NS60636408	2	NS60636403	combined	31.1	CIRC	375	375	0.075
NS60636409	1	NS60636506	combined	97.5	CIRC	375	375	0.12
NS60636501	1	NS60636506	combined	13.6	CIRC	225	225	0.156
NS60636503	1	NS60636501	combined	83.3	CIRC	225	225	0.029
NS60636506	1	NS60636605	combined	70.1	CIRC	600	600	0.339
NS60636604	1	NS60636707	combined	51.2	CIRC	600	600	0.381
NS60636605	1	NS60636604	combined	51	CIRC	600	600	0.173
NS60636701	1	NS60635701	combined	152.1	CIRC	600	600	0.12
NS60636705	1	NS60636707	combined	52.2	CIRC	450	450	0.247
NS60636707	1	NS60636701	combined	4	CIRC	600	600	0.175
NS60636801	1	NS60636909	combined	63.1	EGG	550	690	0.977
NS60636812	1	NS60636705	combined	56.1	CIRC	450	450	0.025
NS60636815	1	NS60636801	combined	4	EGG	550	690	0.958
NS60636901	1	NS60635936	storm	35	NARCH	3300	1300	9.625
NS60636902	1	NS60636909	foul	45.1	CIRC	1200	1200	0
NS60636909	1	NS60635922	foul	50.2	CIRC	1200	1200	1.24
NS60636911	2	NS60636916	storm	70.5	NARCH	3200	1440	8.656
NS60636916	1	NS60636901	storm	23.2	NARCH	2450	1150	4.482

NS60636916	2	NS60636902	foul		34	RECT	1200	0.05447
NS60637001	1	NS60637103	combined	DW	26.1	CIRC	450	0.00192
NS60637005	1	NS60637001	combined	DW	47	CIRC	450	0.00234
NS60637101	1	NS60637110	combined	DW	16.1	CIRC	1600	0.00022
NS60637102	1	NS60637101	combined	DW	54.1	CIRC	1600	0.00022
NS60637103	1	NS60637108	combined	DW	19.2	CIRC	450	0.0026
NS60637104	1	NS60637106	combined	DW	37	CIRC	450	0.03065
NS60637106	1	NS60637110	combined	DW	5.1	CIRC	450	0.03608
NS60637108	1	NS60637104	combined	DW	5	CIRC	450	0.002
NS60637110	1	NS60638102	combined	DW	37.2	CIRC	1600	0.00022
NS60637201	1	NS60637202	combined	DW	12.2	CIRC	375	0.00082
NS60637202	1	NS60637110	combined	DW	27.2	CIRC	375	0.01982
NS60637207	1	NS60637201	combined	DW	21	CIRC	375	0.00904
NS60637301	1	NS60636305	combined	DW	60.1	CIRC	375	0.0037
NS60637501	1	NS60637506	combined	DW	26.1	CIRC	450	0.00051
NS60637506	1	NS60636506	combined	DW	71	CIRC	450	0.00418
NS60637705	1	NS60636707	combined	DW	114.2	CIRC	525	0.00219
NS60637907	1	NS60636812	combined	DW	104.6	CIRC	450	0.00009
NS60637916	1	NS60637920	storm	DE	48	NARCH	2650	0.00278
NS60637920	1	NS60636911	storm		58.2	NARCH	2650	0.00338
NS60638002	1	NS60637005	combined	DW	53.2	CIRC	450	0.00235
NS60638101	1	NS60639105	combined	DW	44.1	CIRC	1600	0.00021
NS60638102	1	NS60638107	combined	DW	47.2	CIRC	1600	0.00021
NS60638107	1	NS60638101	combined	DW	12	CIRC	1600	0.00021
NS60638204	1	NS60638206	combined	DW	50	CIRC	375	0.00542

NS60638206	1	NS60638209	combined	DW	29.2	CIRC	375	0.00549
NS60638208	1	NS60637207	combined	DW	35.4	CIRC	375	0.00062
NS60638209	1	NS60638303	combined	DW	18	CIRC	375	0.00017
NS60638209	2	NS60638208	combined	DW	24	CIRC	375	0.00516
NS60638301	1	NS60637301	combined	DW	54	CIRC	375	0.00017
NS60638303	1	NS60638301	combined	DW	54.1	CIRC	375	0.00017
NS60638406	1	NS60638301	combined	DW	31	CIRC	375	0.01116
NS60638501	1	NS60637501	combined	DW	37.1	CIRC	450	0.00051
NS60638504	1	NS60638507	combined	DW	55	CIRC	450	0.009
NS60638507	1	NS60638501	combined	DW	1	CIRC	450	0.00365
NS60638601	1	NS60638610	combined	DW	38.1	CIRC	450	0.00364
NS60638610	1	NS60638507	combined	DW	43	CIRC	450	0.00365
NS60638715	1	NS60637705	foul	DW	11.4	CIRC	300	0.04315
NS60639101	1	NS60639107	combined	DW	45.2	CIRC	1600	0.00021
NS60639105	1	NS60639101	combined	DW	16	CIRC	1600	0.00021
NS60639107	1	NS61630102	combined	DW	57.2	CIRC	1600	0.00257
NS60639203	1	NS60638204	combined	DW	79	CIRC	375	0.00755
NS60639301	1	NS60638204	combined	DW	63.5	CIRC	375	0.02512
NS60639603	1	NS60639606	combined	DW	58.9	CIRC	525	0.00445
NS60639606	1	NS60639704	combined	DW	47.3	CIRC	525	0.00465
NS60639704	1	NS60637705	combined	DW	112.2	CIRC	525	0.00259
NS60641102	1	NS60642103	combined	DW	83.9	CIRC	920	0.00012
NS60641201	1	NS60641102	combined	DW	47.7	CIRC	920	0.02131
NS60641202	1	NS60641203	combined	DW	20.2	CIRC	890	0.02095
NS60641203	1	NS60641201	combined	DW	31.8	EGG	900	0.02134

NS60641208	1	NS59649302	storm	DE	222.7	AND17	1800	0.00211
NS60641305	1	NS60641310	combined	DW	36.8	CIRC	910	0.0139
NS60641309	1	NS60641202	combined	DW	28.7	CIRC	930	0.02144
NS60641310	1	NS60641309	combined	DW	19.8	CIRC	930	0.02134
NS60642003	1	NS60642005	combined	DW	37.5	EGG	900	0.00426
NS60642004	1	NS60632902	combined	DW	38.2	CIRC	1500	0.00026
NS60642005	1	NS60642004	combined	DW	15.3	CIRC	1500	0
NS60642006	1	NS60642003	combined	DW	5	EGG	930	0.0245
NS60642009	1	NS60641208	storm	DE	197.6	AND17	1850	0.00293
NS60642010	1	NS60642011	storm	DE	10.1	NARCH	1850	0.00496
NS60642011	1	NS60642009	storm	DE	46.1	AND16	1800	0.0026
NS60642103	1	NS60642005	combined	DW	119.8	CIRC	1350	0.00236
NS60643001	1	NS60642010	storm	DE	44.4	AND20	2800	0.00337
NS60643003	1	NS60642006	combined	DW	27.3	EGG	900	0.0322
NS60643007	1	NS60643001	storm	DE	59.7	CIRC	2000	0.00281
NS60645002	1	NS60645005	combined	DW	54.1	CIRC	675	0.01628
NS60645005	1	NS60646004	combined	DW	50.7	CIRC	450	0.00443
NS60645009	1	NS60645005	foul	DW	31.3	CIRC	450	0.02332
NS60646004	1	NS60636902	combined	DW	81.9	CIRC	1200	0.00449
NS61620901	1	NS61630003	combined	DW	39.6	CIRC	450	0.00805
NS61630003	1	NS61630105	combined	DW	128.9	CIRC	450	0.01981
NS61630102	1	NS61630105	combined	DW	9.1	CIRC	1600	0.00221
NS61630105	1	NS61631106	combined	DW	43.6	CIRC	1600	0.00252
NS61630202	1	NS60639203	combined	DW	40	CIRC	375	0.00152
NS61631001	1	newmh7	combined	DW	99.5	CIRC	1600	0.00115

NS61631002	1	NS61631001	combined	DW	52.2	CIRC	1600	0.00077
NS61631102	1	NS61631105	combined	DW	42.1	CIRC	1600	0
NS61631104	1	NS61631102	combined	DW	39.3	CIRC	1600	0.00153
NS61631105	1	NS61631002	combined	DW	58.8	CIRC	1600	0
NS61631106	1	NS61631104	combined	DW	17.5	CIRC	1600	0.00229
NS61632101	1	NS61631105	combined	DW	48.8	EGG	675	0.04159

A Node Values

Node ID	Chamber Plan Area (m2)
new mh11	14
newmh1	2.6
newmh10	14
newmh12	20
newmh14	1.5
newmh2	2.6
newmh3	2.6
newmh4	2.6
newmh5	2.6
newmh6	2.6
newmh7	18
newmh8	18
newmh9	21
NS59644401	20
NS59645401	24
NS59645407	24
NS59645507	22
NS59645508	32
NS59645510	22
NS59645512	23
NS59646501	20
NS59646511	22
NS59647505	11
NS59647556	21
NS59648404	21
NS59648407	18
NS59648410	21
NS59649302	18
NS59649604	3
NS59649705	3
NS60632801	13
NS60632802	19
NS60632804	11
NS60632901	16
NS60632902	19
NS60633401	20
NS60633501	20
NS60633502	21
NS60633601	10
NS60633602	17
NS60633705	14
NS60633708	17
NS60633801	10
NS60633802	8
NS60633803	8
NS60633806	12
NS60633901	18
NS60633902	8
NS60633905	9
NS60634201	20
NS60634301	11
NS60634302	17
NS60634303	14
NS60634401	7
NS60634403	17
NS60634404	16
NS60634405	18
NS60634409	17
NS60634701	15
NS60634702	18
NS60634710	10
NS60634909	24
NS60635203	20
NS60635302	13
NS60635701	9
NS60635812	17

NS60635813	10
NS60635901	14
NS60635911	17
NS60635922	20
NS60635935	21
NS60635936	26
NS60635945	15
NS60636103	20
NS60636105	18
NS60636201	19
NS60636203	17
NS60636301	18
NS60636302	14
NS60636303	17
NS60636305	7
NS60636401	17
NS60636403	11
NS60636404	8
NS60636408	7
NS60636409	14
NS60636501	6
NS60636503	11
NS60636506	14
NS60636604	18
NS60636605	9

NS60636701	11
NS60636705	16
NS60636707	14
NS60636801	11
NS60636812	14
NS60636815	17
NS60636901	27
NS60636902	11
NS60636909	12
NS60636911	28
NS60636916	25
NS60637001	10
NS60637005	14
NS60637101	19
NS60637102	21
NS60637103	11
NS60637104	13
NS60637106	17
NS60637108	11
NS60637110	12
NS60637201	17
NS60637202	2
NS60637207	17
NS60637301	13
NS60637501	14

NS60637506	12
NS60637705	17
NS60637907	11
NS60637916	22
NS60637920	25
NS60638002	11
NS60638101	20
NS60638102	10
NS60638107	19
NS60638204	17
NS60638206	17
NS60638208	17
NS60638209	11
NS60638301	10
NS60638303	17
NS60638406	14
NS60638501	17
NS60638504	15
NS60638507	17
NS60638601	17
NS60638610	17
NS60638715	18
NS60639101	14
NS60639105	12
NS60639107	9

NS60639203	17
NS60639301	17
NS60639603	9
NS60639606	14
NS60639704	13
NS60640603	3
NS60640606	3
NS60640609	1.2
NS60640701	1.2
NS60640707	2.7
NS60640708	2.7
NS60640709	3
NS60640804	1.2
NS60641102	8
NS60641201	17
NS60641202	18
NS60641203	17
NS60641208	21
NS60641305	20
NS60641309	17
NS60641310	18
NS60641502	1
NS60641503	1.2
NS60641605	2.2
NS60641702	1.2

NS60641704	1.5
NS60641705	1.8
NS60641706	2.2
NS60641708	1.5
NS60641710	1.5
NS60641803	1
NS60641805	1.2
NS60642003	17
NS60642004	12
NS60642005	18
NS60642006	8
NS60642009	20
NS60642010	16
NS60642011	9
NS60642103	18
NS60642501	1.7
NS60642506	1.9
NS60642507	3
NS60642512	3
NS60642605	1.5
NS60642701	1.8
NS60643001	22
NS60643003	12
NS60643007	19
NS60643401	3

NS60643403	1.9
NS60643405	2
NS60643410	3
NS60643412	3
NS60643414	2.6
NS60643604	1.5
NS60643612	1.2
NS60643715	1.7
NS60643716	1.8
NS60643902	3
NS60644502	1.8
NS60644503	1.8
NS60644504	2
NS60644506	3.2
NS60644602	1.2
NS60644603	1
NS60644604	1
NS60644611	1
NS60644614	1
NS60644622	3.2
NS60644624	3
NS60644704	1
NS60644804	3
NS60644903	1.8
NS60644904	3

NS60644906	3
NS60645002	18
NS60645005	12
NS60645009	17
NS60645711	1.8
NS60645715	1.5
NS60645793	1
NS60645794	1
NS60645804	3
NS60645902	1.2
NS60646004	19
NS60646407	2.4
NS60646597	2.4
NS60646598	2.4
NS60646599	1.8
NS60646701	2
NS60647108	4
NS60647203	3.7
NS60647217	1.8
NS60647218	3
NS60647301	3.7
NS60647306	3.7
NS60647404	1.9
NS60647406	1.9
NS60647407	3.7

NS60647503	1
NS60647504	1.7
NS60647601	1.6
NS60647602	1.9
NS60647603	1.8
NS60647702	2.2
NS60647704	2.2
NS60647705	1.6
NS60647706	2.2
NS60647713	2.2
NS60648501	1.7
NS60648601	1.6
NS60648602	1.6
NS60648606	1.7
NS60648710	1.2
NS60648713	1.2
NS60648718	1.7
NS60648719	1.8
NS60648720	1.8
NS60648801	2.2
NS60648901	2.2
NS60648902	1.5
NS60648905	2.2
NS60649601	1.6
NS60649603	1.7

NS60649604	1.7
NS60649712	1.3
NS60649802	1.6
NS60649901	1.5
NS60653004	3
NS60653109	3.2
NS60653110	3.2
NS60653114	2.1
NS60654102	2.1
NS60655003	1.6
NS60655104	1.6
NS60655107	2.1
NS60655112	1.6
NS60656001	1.9
NS60656002	1.9
NS60656101	1.2
NS60656103	1.2
NS60656111	1
NS60656116	1
NS60656118	1.5
NS60656201	1
NS60656213	1.5
NS60656215	1.5
NS60656305	1.4
NS60656306	1.5

NS60657001	2
NS60657002	2
NS60657008	1.5
NS60657107	1
NS60657109	1
NS60657114	1
NS60657115	1
NS60657302	1
NS60657402	1.4
NS60658001	1.4
NS60658006	1.4
NS60658007	1.4
NS60658102	1
NS60658402	1.2
NS60659008	1.4
NS60659101	1
NS61620299	20
NS61620901	18
NS61630003	17
NS61630102	18
NS61630105	11
NS61630202	6
NS61631001	13
NS61631002	20
NS61631102	20

NS61631104	17
NS61631105	19
NS61631106	19
NS61632101	13
NS61640603	1.6
NS61640605	1.5
NS61640803	1.5
NS61640905	1.6
NS61641603	1.2
NS61641609	1.5
NS61641701	1
NS61641707	1.5
NS61641801	1.5
NS61641804	1.5
NS61641999	1.5
NS61642801	1.4
NS61643806	1.5
NS61644903	1.3
NS61650001	1.8
NS61650003	1.3
NS61650007	1.5
NS61650009	1.4
NS61650010	1.8
NS61650011	1.8
NS61650012	1.3

NS61650102	1.3
NS61650301	1.3
NS61650302	1.3
NS61650402	1.3
NS61650403	1.2
NS61650501	1.2
NS61650503	1.2
NS61651001	1
NS61651002	1
NS61651006	1.8
NS61651007	1.8
NS61651008	1.7
NS61651103	1
NS61651109	1
NS61651111	1
NS61651113	1
NS61651513	1
NS61652005	1.7
NS61652101	1.2
NS61652102	1.2
NS61652301	1.2
NS61652305	1
NS61652306	1.2
NS61652402	1.2
NS61653002	1.7

NS61653102	1.2
NS61653304	1
NS61654001	1.7
NS61654101	1.2

NS61654302	1.2
NS61655005	1.7
NS61655101	1.4
NS61655103	1.4

NS61655201	1.4
NS61655301	1.4

A Pipe Values

US Node ID	Link Suffix	DS Node ID	Width (mm)
new mh11	1	newmh12	2100
newmh1	2	newmh3	1050
newmh10	1	new mh11	2200
newmh12	1	NS61620299	2500
newmh14	1	NS61640603	600
newmh2	1	newmh4	1050
newmh3	2	newmh5	1050
newmh4	1	newmh6	1050
newmh5	2	NS60642103	1050
newmh6	1	NS60642103	675
newmh7	1	newmh8	2400
newmh8	1	newmh9	2400
newmh9	1	newmh10	1600
NS59645401	1	NS59645407	4500
NS59645407	1	NS59644401	1950
NS59645507	1	NS59645401	3600
NS59645508	1	NS59645510	4000
NS59645510	1	NS59645507	3600
NS59645512	1	NS59645508	1950
NS59646501	1	NS59645512	4000
NS59646511	1	NS59646501	3200
NS59647505	1	NS59646511	2400
NS59647556	1	NS59647505	2400
NS59648404	1	NS59648410	2400
NS59648407	1	NS59648404	2400
NS59648410	1	NS59647556	2400
NS59649302	1	NS59648407	2200
NS59649604	1	NS60640606	1200
NS59649705	1	NS59649604	1200
NS60632801	1	NS60632804	2100
NS60632802	1	NS60632801	1950
NS60632804	1	NS60633705	1800
NS60632901	1	NS60632802	2100
NS60632902	1	NS60632901	2100
NS60633401	1	NS60634409	2550
NS60633501	1	NS60633401	2400
NS60633502	1	NS60633501	2400
NS60633601	1	NS60633502	2400
NS60633602	1	NS60633601	1600
NS60633705	1	NS60633708	2400
NS60633708	1	NS60633602	2400
NS60633801	1	NS60634702	975
NS60633802	1	NS60632804	1950
NS60633802	2	NS60633806	1200

NS60633803	1	NS60633802	1100
NS60633806	1	NS60633801	1100
NS60633901	1	NS60633803	750
NS60633902	1	NS60633901	975
NS60633905	1	NS60633902	975
NS60634201	1	NS60635203	2400
NS60634301	1	NS60634303	1950
NS60634302	1	NS60634303	900
NS60634303	1	NS60634201	2400
NS60634401	1	NS60634404	1200
NS60634403	1	NS60634409	750
NS60634404	1	NS60634403	675
NS60634405	1	NS60634404	800
NS60634409	1	NS60634301	2400
NS60634701	1	NS60633708	700
NS60634702	1	NS60634701	975
NS60634710	1	NS60634701	900
NS60634909	1	NS60643007	2500
NS60635203	1	NS60636203	2400
NS60635302	1	NS60635203	975
NS60635701	1	NS60634710	1300
NS60635812	1	NS60636815	600
NS60635813	1	NS60633802	1800
NS60635901	1	NS60635922	630
NS60635911	1	NS60635936	750
NS60635922	1	NS60635813	1800
NS60635935	1	NS60635945	3400
NS60635936	1	NS60635935	4000
NS60635945	1	NS60634909	3000
NS60636103	1	NS60636105	2500
NS60636105	1	NS60637102	2200
NS60636201	1	NS60636103	2200
NS60636203	1	NS60636201	2200
NS60636301	1	NS60636303	1050
NS60636302	1	NS60636301	1050
NS60636303	1	NS60636103	1300
NS60636305	1	NS60636302	600
NS60636401	1	NS60636409	975
NS60636403	1	NS60636302	800
NS60636404	1	NS60636408	500
NS60636408	1	NS60636401	600
NS60636408	2	NS60636403	600
NS60636409	1	NS60636506	1050
NS60636501	1	NS60636506	750
NS60636503	1	NS60636501	375
NS60636506	1	NS60636605	1100
NS60636604	1	NS60636707	1350
NS60636605	1	NS60636604	975
NS60636701	1	NS60635701	975
NS60636705	1	NS60636707	675

NS60636707	1	NS60636701	1100
NS60636801	1	NS60636909	800
NS60636812	1	NS60636705	1050
NS60636815	1	NS60636801	1300
NS60636901	1	NS60635936	4500
NS60636902	1	NS60636909	1800
NS60636909	1	NS60635922	1600
NS60636911	2	NS60636916	4500
NS60636916	1	NS60636901	3400
NS60636916	2	NS60636902	1250
NS60637001	1	NS60637103	975
NS60637005	1	NS60637001	800
NS60637101	1	NS60637110	2400
NS60637102	1	NS60637101	2400
NS60637103	1	NS60637108	800
NS60637104	1	NS60637106	600
NS60637106	1	NS60637110	900
NS60637108	1	NS60637104	1250
NS60637110	1	NS60638102	2400
NS60637201	1	NS60637202	600
NS60637202	1	NS60637110	675
NS60637207	1	NS60637201	750
NS60637301	1	NS60636305	450
NS60637501	1	NS60637506	1050
NS60637506	1	NS60636506	525

NS60637705	1	NS60636707	975
NS60637907	1	NS60636812	900
NS60637916	1	NS60637920	3200
NS60637920	1	NS60636911	3400
NS60638002	1	NS60637005	750
NS60638101	1	NS60639105	2400
NS60638102	1	NS60638107	2100
NS60638107	1	NS60638101	2500
NS60638204	1	NS60638206	600
NS60638206	1	NS60638209	675
NS60638208	1	NS60637207	975
NS60638209	1	NS60638303	450
NS60638209	2	NS60638208	675
NS60638301	1	NS60637301	800
NS60638303	1	NS60638301	525
NS60638406	1	NS60638301	1200
NS60638501	1	NS60637501	1200
NS60638504	1	NS60638507	1250
NS60638507	1	NS60638501	900
NS60638601	1	NS60638610	800
NS60638610	1	NS60638507	900
NS60638715	1	NS60637705	675
NS60639101	1	NS60639107	2400
NS60639105	1	NS60639101	1950
NS60639107	1	NS61630102	1950

NS60639203	1	NS60638204	600
NS60639301	1	NS60638204	1200
NS60639603	1	NS60639606	900
NS60639606	1	NS60639704	1050
NS60639704	1	NS60637705	975
NS60640603	1	NS60642512	1200
NS60640606	1	NS60640603	1200
NS60640609	1	NS60640708	450
NS60640701	1	NS60640707	450
NS60640707	1	NS60640708	1100
NS60640708	1	NS60640709	1100
NS60640709	1	NS59649705	1200
NS60640804	1	NS60640701	450
NS60641102	1	NS60642103	1600
NS60641201	1	NS60641102	1200
NS60641202	1	NS60641203	1800
NS60641203	1	NS60641201	1100
NS60641208	1	NS59649302	2400
NS60641305	1	NS60641310	1600
NS60641309	1	NS60641202	1200
NS60641310	1	NS60641309	1100
NS60641502	1	NS60641503	375
NS60641503	1	NS60642506	450
NS60641605	1	NS60641706	900
NS60641702	1	NS60641704	450

NS60641704	1	NS60641708	640
NS60641705	1	NS60641710	630
NS60641706	1	NS60641704	640
NS60641708	1	NS60640707	640
NS60641710	1	NS60641706	630
NS60641803	1	NS60641805	225
NS60641805	1	NS60641702	450
NS60642003	1	NS60642005	1350
NS60642004	1	NS60632902	2100
NS60642005	1	NS60642004	2100
NS60642006	1	NS60642003	1350
NS60642009	1	NS60641208	2400
NS60642010	1	NS60642011	2400
NS60642011	1	NS60642009	2400
NS60642103	1	NS60642005	1800
NS60642501	1	NS60642506	710
NS60642506	1	NS60643403	800
NS60642507	1	NS60643401	1200
NS60642512	1	NS60642507	1200
NS60642605	1	NS60641605	600
NS60642701	1	NS60641705	750
NS60643001	1	NS60642010	3200
NS60643003	1	NS60642006	1100
NS60643007	1	NS60643001	2500
NS60643401	1	NS60643410	1200

NS60643403	1	NS60643410	800
NS60643405	1	NS60643414	680
NS60643410	1	newmh2	1050
NS60643410	2	newmh1	1050
NS60643412	1	NS60643410	1200
NS60643414	2	NS60643412	1062
NS60643604	1	NS60642605	600
NS60643612	1	NS60643604	450
NS60643715	1	NS60643716	730
NS60643716	1	NS60642701	750
NS60643902	1	NS60644906	1200
NS60644502	1	NS60644503	745
NS60644503	1	NS60644504	750
NS60644504	1	NS60643405	830
NS60644506	1	NS60643412	1200
NS60644602	1	NS60644502	450
NS60644603	1	NS60644602	225
NS60644604	1	NS60644603	225
NS60644611	1	NS60644604	225
NS60644614	1	NS60644611	225
NS60644622	1	NS60644506	1250
NS60644624	1	NS60644622	1200
NS60644704	1	NS60644614	150
NS60644804	1	NS60645804	1200
NS60644903	1	NS60644906	770

NS60644904	1	NS60644804	1200
NS60644906	1	NS60644904	1200
NS60645002	1	NS60645005	900
NS60645005	1	NS60646004	900
NS60645009	1	NS60645005	800
NS60645711	1	NS60646701	750
NS60645715	1	NS60645711	600
NS60645793	1	NS60645794	150
NS60645804	1	NS60644624	1200
NS60645902	1	NS60644903	450
NS60646004	1	NS60636902	1500
NS60646407	1	NS60647407	900
NS60646597	1	NS60646407	1000
NS60646598	1	NS60646597	1000
NS60646599	1	NS60646598	740
NS60646701	1	NS60647702	825
NS60647217	1	NS60647218	710
NS60647218	1	NS60647108	1200
NS60647301	1	NS60647306	1400
NS60647306	1	NS60647203	1400
NS60647404	1	NS60647406	800
NS60647406	1	NS60647407	800
NS60647407	1	NS60647301	1400
NS60647503	1	NS60647404	375
NS60647504	1	NS60647404	700

NS60647601	1	NS60647702	675
NS60647602	1	NS60647603	740
NS60647603	1	NS60646599	740
NS60647702	1	NS60647602	800
NS60647704	1	NS60647713	920
NS60647705	1	NS60647704	650
NS60647706	1	NS60647702	920
NS60647713	1	NS60647706	920
NS60648501	1	NS60647504	700
NS60648601	1	NS60648602	675
NS60648602	1	NS60647601	675
NS60648606	1	NS60648501	700
NS60648710	1	NS60647705	450
NS60648713	1	NS60648710	450
NS60648718	1	NS60648713	450
NS60648719	1	NS60648718	720
NS60648720	1	NS60648719	770
NS60648801	1	NS60647704	920
NS60648901	1	NS60648905	920
NS60648902	1	NS60648901	600
NS60648905	1	NS60648801	920
NS60649601	1	NS60649603	675
NS60649603	1	NS60648601	675
NS60649604	1	NS60649603	690
NS60649712	1	NS60648720	500

NS60649802	1	NS60649901	600
NS60649901	1	NS60648902	600
NS60653004	1	NS60643902	1200
NS60653109	1	NS60653004	1200
NS60653110	1	NS60653109	1250
NS60653114	2	NS60653110	880
NS60654102	1	NS60653114	615
NS60655003	1	NS60655107	660
NS60655104	1	NS60655112	680
NS60655107	1	NS60654102	880
NS60655112	1	NS60655107	580
NS60656001	1	NS60655003	600
NS60656002	1	NS60656001	780
NS60656101	1	NS60656103	450
NS60656103	1	NS60656001	450
NS60656111	1	NS60656103	225
NS60656116	1	NS60656111	225
NS60656118	1	NS60655104	600
NS60656201	1	NS60656116	225
NS60656213	1	NS60656215	600
NS60656215	1	NS60656118	600
NS60656305	1	NS60656306	580
NS60656306	1	NS60656213	600
NS60657001	1	NS60657002	840
NS60657002	1	NS60656002	600

NS60657008	1	NS60657002	600
NS60657107	1	NS60657109	375
NS60657109	1	NS60657114	375
NS60657114	1	NS60657115	375
NS60657115	1	NS60656101	375
NS60657302	1	NS60656305	300
NS60657402	1	NS60656305	580
NS60658001	1	NS60658007	570
NS60658006	1	NS60657008	550
NS60658007	1	NS60658006	550
NS60658102	1	NS60657107	375
NS60658402	1	NS60657402	450
NS60659008	1	NS60658007	550
NS60659101	1	NS60658001	300
NS61620901	1	NS61630003	800
NS61630003	1	NS61630105	675
NS61630102	1	NS61630105	1950
NS61630105	1	NS61631106	2200
NS61630202	1	NS60639203	600
NS61631001	1	newmh7	2200
NS61631002	1	NS61631001	2400
NS61631102	1	NS61631105	2400
NS61631104	1	NS61631102	2700
NS61631105	1	NS61631002	2200
NS61631106	1	NS61631104	2400

NS61632101	1	NS61631105	1100
NS61640603	1	NS60649601	675
NS61640605	1	NS61640603	610
NS61640803	1	NS60649802	600
NS61640905	1	NS60649802	650
NS61641603	1	NS61640605	450
NS61641609	1	newmh14	600
NS61641701	1	NS61641603	375
NS61641707	1	NS61641609	600
NS61641801	1	NS61641804	600
NS61641804	1	NS61641707	600
NS61641999	1	NS61641801	600
NS61642801	1	NS61641801	580
NS61643806	1	NS61641804	610
NS61644903	1	NS61643806	530
NS61650001	1	NS61650010	770
NS61650003	1	NS61650012	520
NS61650007	1	NS61640905	635
NS61650009	1	NS60659008	550
NS61650010	1	NS61650009	550
NS61650010	2	NS61650007	620
NS61650011	1	NS61650010	760
NS61650012	1	NS61650001	520
NS61650102	1	NS61650003	500
NS61650301	1	NS61650302	500

NS61650302	1	NS61650102	460
NS61650402	1	NS61650301	500
NS61650403	1	NS61650402	450
NS61650501	1	NS61650403	450
NS61650503	1	NS61650501	450
NS61651001	1	NS61651007	375
NS61651002	1	NS61651001	375
NS61651006	1	NS61651007	750
NS61651006	2	NS61641999	600
NS61651007	1	NS61650011	770
NS61651008	1	NS61651006	700
NS61651103	1	NS61652101	375
NS61651109	1	NS61651111	300
NS61651111	1	NS61651002	375
NS61651513	1	NS61650503	375
NS61652005	1	NS61651008	690
NS61652101	1	NS61652102	450

NS61652102	1	NS61651008	450
NS61652301	1	NS61652306	450
NS61652305	1	NS61652306	300
NS61652306	1	NS61652101	450
NS61652402	1	NS61652301	450
NS61653002	1	NS61652005	690
NS61653102	1	NS61653002	450
NS61653304	1	NS61654302	375
NS61654001	1	NS61653002	720
NS61654101	1	NS61655103	450
NS61654302	1	NS61655301	450
NS61655005	1	NS61654001	720
NS61655101	1	NS61655103	580
NS61655103	1	NS61655005	570
NS61655201	1	NS61655101	580
NS61655301	1	NS61655201	580

B Node Values

Node ID	Chamber Plan Area (m2)
new mh11	4
newmh1	2.6
newmh10	4
newmh12	4
newmh14	1.5
newmh2	2.6
newmh3	2.6
newmh4	2.6
newmh5	2.6
newmh6	2.6
newmh7	4
newmh8	4
newmh9	5
NS59644401	4
NS59645401	16
NS59645407	16
NS59645507	16
NS59645508	16
NS59645510	16
NS59645512	10

NS59646501	10
NS59646511	10
NS59647505	5
NS59647556	5
NS59648404	6
NS59648407	5
NS59648410	6
NS59649302	5
NS59649604	3
NS59649705	3
NS60632801	4
NS60632802	6
NS60632804	4
NS60632901	4
NS60632902	4
NS60633401	4
NS60633501	4
NS60633502	5
NS60633601	4
NS60633602	4
NS60633705	4
NS60633708	4
NS60633801	2

NS60633802	3
NS60633803	2
NS60633806	2
NS60633901	2
NS60633902	2
NS60633905	1
NS60634201	4
NS60634301	4
NS60634302	1
NS60634303	4
NS60634401	1
NS60634403	1
NS60634404	1
NS60634405	4
NS60634409	4
NS60634701	1
NS60634702	2
NS60634710	1
NS60634909	8
NS60635203	4
NS60635302	7
NS60635701	1
NS60635812	1

NS60635813	3
NS60635901	1
NS60635911	2
NS60635922	4
NS60635935	15
NS60635936	15
NS60635945	9
NS60636103	4
NS60636105	4
NS60636201	6
NS60636203	4
NS60636301	2
NS60636302	1
NS60636303	1
NS60636305	1
NS60636401	1
NS60636403	1
NS60636404	1
NS60636408	1
NS60636409	1
NS60636501	1
NS60636503	1
NS60636506	1
NS60636604	2
NS60636605	1

NS60636701	1
NS60636705	1
NS60636707	1
NS60636801	1
NS60636812	1
NS60636815	2
NS60636901	13
NS60636902	3
NS60636909	3
NS60636911	12
NS60636916	12
NS60637001	1
NS60637005	1
NS60637101	4
NS60637102	5
NS60637103	1
NS60637104	1
NS60637106	2
NS60637108	1
NS60637110	4
NS60637201	1
NS60637202	1
NS60637207	1
NS60637301	1
NS60637501	1

NS60637506	1
NS60637705	1
NS60637907	9
NS60637916	9
NS60637920	9
NS60638002	1
NS60638101	4
NS60638102	4
NS60638107	4
NS60638204	2
NS60638206	1
NS60638208	1
NS60638209	1
NS60638301	1
NS60638303	1
NS60638406	1
NS60638501	1
NS60638504	1
NS60638507	1
NS60638601	1
NS60638610	1
NS60638715	2
NS60639101	4
NS60639105	4
NS60639107	4

NS60639203	2
NS60639301	2
NS60639603	1
NS60639606	1
NS60639704	1
NS60640603	3
NS60640606	3
NS60640609	1.2
NS60640701	1.2
NS60640707	2.7
NS60640708	2.7
NS60640709	3
NS60640804	1.2
NS60641102	2
NS60641201	2
NS60641202	2
NS60641203	2
NS60641208	5
NS60641305	9
NS60641309	3
NS60641310	2
NS60641502	1
NS60641503	1.2
NS60641605	2.2
NS60641702	1.2

NS60641704	1.5
NS60641705	1.8
NS60641706	2.2
NS60641708	1.5
NS60641710	1.5
NS60641803	1
NS60641805	1.2
NS60642003	2
NS60642004	4
NS60642005	4
NS60642006	2
NS60642009	5
NS60642010	10
NS60642011	6
NS60642103	4
NS60642501	1.7
NS60642506	1.9
NS60642507	3
NS60642512	3
NS60642605	1.5
NS60642701	1.8
NS60643001	10
NS60643003	2
NS60643007	6
NS60643401	3

NS60643403	1.9
NS60643405	2
NS60643410	3
NS60643412	3
NS60643414	2.6
NS60643604	1.5
NS60643612	1.2
NS60643715	1.7
NS60643716	1.8
NS60643902	3
NS60644502	1.8
NS60644503	1.8
NS60644504	2
NS60644506	3.2
NS60644602	1.2
NS60644603	1
NS60644604	1
NS60644611	1
NS60644614	1
NS60644622	3.2
NS60644624	3
NS60644704	1
NS60644804	3
NS60644903	1.8
NS60644904	3

NS60644906	3
NS60645002	2
NS60645005	2
NS60645009	1
NS60645711	1.8
NS60645715	1.5
NS60645793	1
NS60645794	1
NS60645804	3
NS60645902	1.2
NS60646004	3
NS60646407	2.4
NS60646597	2.4
NS60646598	2.4
NS60646599	1.8
NS60646701	2
NS60647108	4
NS60647203	3.7
NS60647217	1.8
NS60647218	3
NS60647301	3.7
NS60647306	3.7
NS60647404	1.9
NS60647406	1.9
NS60647407	3.7

NS60647503	1
NS60647504	1.7
NS60647601	1.6
NS60647602	1.9
NS60647603	1.8
NS60647702	2.2
NS60647704	2.2
NS60647705	1.6
NS60647706	2.2
NS60647713	2.2
NS60648501	1.7
NS60648601	1.6
NS60648602	1.6
NS60648606	1.7
NS60648710	1.2
NS60648713	1.2
NS60648718	1.7
NS60648719	1.8
NS60648720	1.8
NS60648801	2.2
NS60648901	2.2
NS60648902	1.5
NS60648905	2.2
NS60649601	1.6
NS60649603	1.7

NS60649604	1.7
NS60649712	1.3
NS60649802	1.6
NS60649901	1.5
NS60653004	3
NS60653109	3.2
NS60653110	3.2
NS60653114	2.1
NS60654102	2.1
NS60655003	1.6
NS60655104	1.6
NS60655107	2.1
NS60655112	1.6
NS60656001	1.9
NS60656002	1.9
NS60656101	1.2
NS60656103	1.2
NS60656111	1
NS60656116	1
NS60656118	1.5
NS60656201	1
NS60656213	1.5
NS60656215	1.5
NS60656305	1.4
NS60656306	1.5

NS60657001	2
NS60657002	2
NS60657008	1.5
NS60657107	1
NS60657109	1
NS60657114	1
NS60657115	1
NS60657302	1
NS60657402	1.4
NS60658001	1.4
NS60658006	1.4
NS60658007	1.4
NS60658102	1
NS60658402	1.2
NS60659008	1.4
NS60659101	1
NS61620299	4
NS61620901	2
NS61630003	1
NS61630102	4
NS61630105	4
NS61630202	1
NS61631001	4
NS61631002	4
NS61631102	8

NS61631104	4
NS61631105	9
NS61631106	4
NS61632101	2
NS61640603	1.6
NS61640605	1.5
NS61640803	1.5
NS61640905	1.6
NS61641603	1.2
NS61641609	1.5
NS61641701	1
NS61641707	1.5
NS61641801	1.5
NS61641804	1.5
NS61641999	1.5
NS61642801	1.4
NS61643806	1.5
NS61644903	1.3
NS61650001	1.8
NS61650003	1.3
NS61650007	1.5
NS61650009	1.4
NS61650010	1.8
NS61650011	1.8
NS61650012	1.3

NS61650102	1.3
NS61650301	1.3
NS61650302	1.3
NS61650402	1.3
NS61650403	1.2
NS61650501	1.2
NS61650503	1.2
NS61651001	1
NS61651002	1
NS61651006	1.8
NS61651007	1.8
NS61651008	1.7
NS61651103	1
NS61651109	1
NS61651111	1
NS61651113	1
NS61651513	1
NS61652005	1.7
NS61652101	1.2
NS61652102	1.2
NS61652301	1.2
NS61652305	1
NS61652306	1.2
NS61652402	1.2
NS61653002	1.7

NS61653102	1.2
NS61653304	1
NS61654001	1.7
NS61654101	1.2
NS61654302	1.2
NS61655005	1.7
NS61655101	1.4
NS61655103	1.4
NS61655201	1.4
NS61655301	1.4

B Pipe Values

US Node ID	Link Suffix	DS Node ID	Width (mm)
new mh11	1	newmh12	1600
newmh1	2	newmh3	1050
newmh10	1	new mh11	1600
newmh12	1	NS61620299	1650
newmh14	1	NS61640603	600
newmh2	1	newmh4	1050
newmh3	2	newmh5	1050
newmh4	1	newmh6	1050
newmh5	2	NS60642103	1050
newmh6	1	NS60642103	675
newmh7	1	newmh8	1600
newmh8	1	newmh9	1600
newmh9	1	newmh10	1600
NS59645401	1	NS59645407	3600
NS59645407	1	NS59644401	1400
NS59645507	1	NS59645401	3600
NS59645508	1	NS59645510	3600
NS59645510	1	NS59645507	3600
NS59645512	1	NS59645508	1400
NS59646501	1	NS59645512	2850
NS59646511	1	NS59646501	3200
NS59647505	1	NS59646511	2100
NS59647556	1	NS59647505	1800
NS59648404	1	NS59648410	1800
NS59648407	1	NS59648404	1800
NS59648410	1	NS59647556	1800
NS59649302	1	NS59648407	1950
NS59649604	1	NS60640606	1200
NS59649705	1	NS59649604	1200
NS60632801	1	NS60632804	1500
NS60632802	1	NS60632801	1500
NS60632804	1	NS60633705	1650
NS60632901	1	NS60632802	1500
NS60632902	1	NS60632901	1500
NS60633401	1	NS60634409	1800
NS60633501	1	NS60633401	1600
NS60633502	1	NS60633501	1600
NS60633601	1	NS60633502	1600
NS60633602	1	NS60633601	1600
NS60633705	1	NS60633708	1600
NS60633708	1	NS60633602	1600
NS60633801	1	NS60634702	675
NS60633802	1	NS60632804	1200
NS60633802	2	NS60633806	700

NS60633803	1	NS60633802	700
NS60633806	1	NS60633801	700
NS60633901	1	NS60633803	675
NS60633902	1	NS60633901	750
NS60633905	1	NS60633902	450
NS60634201	1	NS60635203	1650
NS60634301	1	NS60634303	1600
NS60634302	1	NS60634303	600
NS60634303	1	NS60634201	1600
NS60634401	1	NS60634404	400
NS60634403	1	NS60634409	500
NS60634404	1	NS60634403	400
NS60634405	1	NS60634404	300
NS60634409	1	NS60634301	1600
NS60634701	1	NS60633708	600
NS60634702	1	NS60634701	400
NS60634710	1	NS60634701	600
NS60634909	1	NS60643007	2000
NS60635203	1	NS60636203	1600
NS60635302	1	NS60635203	300
NS60635701	1	NS60634710	630
NS60635812	1	NS60636815	350
NS60635813	1	NS60633802	1200
NS60635901	1	NS60635922	150
NS60635911	1	NS60635936	275

NS60635922	1	NS60635813	1250
NS60635935	1	NS60635945	2600
NS60635936	1	NS60635935	3600
NS60635945	1	NS60634909	2700
NS60636103	1	NS60636105	1600
NS60636105	1	NS60637102	1600
NS60636201	1	NS60636103	1600
NS60636203	1	NS60636201	1800
NS60636301	1	NS60636303	525
NS60636302	1	NS60636301	525
NS60636303	1	NS60636103	525
NS60636305	1	NS60636302	375
NS60636401	1	NS60636409	525
NS60636403	1	NS60636302	525
NS60636404	1	NS60636408	225
NS60636408	1	NS60636401	400
NS60636408	2	NS60636403	450
NS60636409	1	NS60636506	500
NS60636501	1	NS60636506	300
NS60636503	1	NS60636501	225
NS60636506	1	NS60636605	630
NS60636604	1	NS60636707	750
NS60636605	1	NS60636604	600
NS60636701	1	NS60635701	600
NS60636705	1	NS60636707	450

NS60636707	1	NS60636701	630
NS60636801	1	NS60636909	525
NS60636812	1	NS60636705	450
NS60636815	1	NS60636801	630
NS60636901	1	NS60635936	3400
NS60636902	1	NS60636909	1200
NS60636909	1	NS60635922	1200
NS60636911	2	NS60636916	3500
NS60636916	1	NS60636901	2600
NS60636916	2	NS60636902	1200
NS60637001	1	NS60637103	450
NS60637005	1	NS60637001	450
NS60637101	1	NS60637110	1600
NS60637102	1	NS60637101	1650
NS60637103	1	NS60637108	450
NS60637104	1	NS60637106	450
NS60637106	1	NS60637110	450
NS60637108	1	NS60637104	450
NS60637110	1	NS60638102	1600
NS60637201	1	NS60637202	400
NS60637202	1	NS60637110	375
NS60637207	1	NS60637201	375
NS60637301	1	NS60636305	375
NS60637501	1	NS60637506	450
NS60637506	1	NS60636506	450

NS60637705	1	NS60636707	600
NS60637907	1	NS60636812	600
NS60637916	1	NS60637920	2700
NS60637920	1	NS60636911	2700
NS60638002	1	NS60637005	450
NS60638101	1	NS60639105	1600
NS60638102	1	NS60638107	1800
NS60638107	1	NS60638101	1650
NS60638204	1	NS60638206	375
NS60638206	1	NS60638209	375
NS60638208	1	NS60637207	400
NS60638209	1	NS60638303	375
NS60638209	2	NS60638208	375
NS60638301	1	NS60637301	400
NS60638303	1	NS60638301	375
NS60638406	1	NS60638301	400
NS60638501	1	NS60637501	450
NS60638504	1	NS60638507	525
NS60638507	1	NS60638501	450
NS60638601	1	NS60638610	500
NS60638610	1	NS60638507	450
NS60638715	1	NS60637705	300
NS60639101	1	NS60639107	1600
NS60639105	1	NS60639101	1600
NS60639107	1	NS61630102	1600

NS60639203	1	NS60638204	450
NS60639301	1	NS60638204	400
NS60639603	1	NS60639606	525
NS60639606	1	NS60639704	525
NS60639704	1	NS60637705	525
NS60640603	1	NS60642512	1200
NS60640606	1	NS60640603	1200
NS60640609	1	NS60640708	450
NS60640701	1	NS60640707	450
NS60640707	1	NS60640708	1100
NS60640708	1	NS60640709	1100
NS60640709	1	NS59649705	1200
NS60640804	1	NS60640701	450
NS60641102	1	NS60642103	900
NS60641201	1	NS60641102	900
NS60641202	1	NS60641203	900
NS60641203	1	NS60641201	900
NS60641208	1	NS59649302	1800
NS60641305	1	NS60641310	900
NS60641309	1	NS60641202	975
NS60641310	1	NS60641309	900
NS60641502	1	NS60641503	375
NS60641503	1	NS60642506	450
NS60641605	1	NS60641706	900
NS60641702	1	NS60641704	450

NS60641704	1	NS60641708	640
NS60641705	1	NS60641710	630
NS60641706	1	NS60641704	640
NS60641708	1	NS60640707	640
NS60641710	1	NS60641706	630
NS60641803	1	NS60641805	225
NS60641805	1	NS60641702	450
NS60642003	1	NS60642005	900
NS60642004	1	NS60632902	1600
NS60642005	1	NS60642004	1600
NS60642006	1	NS60642003	900
NS60642009	1	NS60641208	2000
NS60642010	1	NS60642011	2100
NS60642011	1	NS60642009	1800
NS60642103	1	NS60642005	1350
NS60642501	1	NS60642506	710
NS60642506	1	NS60643403	800
NS60642507	1	NS60643401	1200
NS60642512	1	NS60642507	1200
NS60642605	1	NS60641605	600
NS60642701	1	NS60641705	750
NS60643001	1	NS60642010	2800
NS60643003	1	NS60642006	900
NS60643007	1	NS60643001	2000
NS60643401	1	NS60643410	1200

NS60643403	1	NS60643410	800
NS60643405	1	NS60643414	680
NS60643410	1	newmh2	1050
NS60643410	2	newmh1	1050
NS60643412	1	NS60643410	1200
NS60643414	2	NS60643412	1062
NS60643604	1	NS60642605	600
NS60643612	1	NS60643604	450
NS60643715	1	NS60643716	730
NS60643716	1	NS60642701	750
NS60643902	1	NS60644906	1200
NS60644502	1	NS60644503	745
NS60644503	1	NS60644504	750
NS60644504	1	NS60643405	830
NS60644506	1	NS60643412	1200
NS60644602	1	NS60644502	450
NS60644603	1	NS60644602	225
NS60644604	1	NS60644603	225
NS60644611	1	NS60644604	225
NS60644614	1	NS60644611	225
NS60644622	1	NS60644506	1250
NS60644624	1	NS60644622	1200
NS60644704	1	NS60644614	150
NS60644804	1	NS60645804	1200
NS60644903	1	NS60644906	770

NS60644904	1	NS60644804	1200
NS60644906	1	NS60644904	1200
NS60645002	1	NS60645005	675
NS60645005	1	NS60646004	450
NS60645009	1	NS60645005	450
NS60645711	1	NS60646701	750
NS60645715	1	NS60645711	600
NS60645793	1	NS60645794	150
NS60645804	1	NS60644624	1200
NS60645902	1	NS60644903	450
NS60646004	1	NS60636902	1200
NS60646407	1	NS60647407	900
NS60646597	1	NS60646407	1000
NS60646598	1	NS60646597	1000
NS60646599	1	NS60646598	740
NS60646701	1	NS60647702	825
NS60647217	1	NS60647218	710
NS60647218	1	NS60647108	1200
NS60647301	1	NS60647306	1400
NS60647306	1	NS60647203	1400
NS60647404	1	NS60647406	800
NS60647406	1	NS60647407	800
NS60647407	1	NS60647301	1400
NS60647503	1	NS60647404	375
NS60647504	1	NS60647404	700

NS60647601	1	NS60647702	675
NS60647602	1	NS60647603	740
NS60647603	1	NS60646599	740
NS60647702	1	NS60647602	800
NS60647704	1	NS60647713	920
NS60647705	1	NS60647704	650
NS60647706	1	NS60647702	920
NS60647713	1	NS60647706	920
NS60648501	1	NS60647504	700
NS60648601	1	NS60648602	675
NS60648602	1	NS60647601	675
NS60648606	1	NS60648501	700
NS60648710	1	NS60647705	450
NS60648713	1	NS60648710	450
NS60648718	1	NS60648713	450
NS60648719	1	NS60648718	720
NS60648720	1	NS60648719	770
NS60648801	1	NS60647704	920
NS60648901	1	NS60648905	920
NS60648902	1	NS60648901	600
NS60648905	1	NS60648801	920
NS60649601	1	NS60649603	675
NS60649603	1	NS60648601	675
NS60649604	1	NS60649603	690
NS60649712	1	NS60648720	500

NS60649802	1	NS60649901	600
NS60649901	1	NS60648902	600
NS60653004	1	NS60643902	1200
NS60653109	1	NS60653004	1200
NS60653110	1	NS60653109	1250
NS60653114	2	NS60653110	880
NS60654102	1	NS60653114	615
NS60655003	1	NS60655107	660
NS60655104	1	NS60655112	680
NS60655107	1	NS60654102	880
NS60655112	1	NS60655107	580
NS60656001	1	NS60655003	600
NS60656002	1	NS60656001	780
NS60656101	1	NS60656103	450
NS60656103	1	NS60656001	450
NS60656111	1	NS60656103	225
NS60656116	1	NS60656111	225
NS60656118	1	NS60655104	600
NS60656201	1	NS60656116	225
NS60656213	1	NS60656215	600
NS60656215	1	NS60656118	600
NS60656305	1	NS60656306	580
NS60656306	1	NS60656213	600
NS60657001	1	NS60657002	840
NS60657002	1	NS60656002	600

NS60657008	1	NS60657002	600
NS60657107	1	NS60657109	375
NS60657109	1	NS60657114	375
NS60657114	1	NS60657115	375
NS60657115	1	NS60656101	375
NS60657302	1	NS60656305	300
NS60657402	1	NS60656305	580
NS60658001	1	NS60658007	570
NS60658006	1	NS60657008	550
NS60658007	1	NS60658006	550
NS60658102	1	NS60657107	375
NS60658402	1	NS60657402	450
NS60659008	1	NS60658007	550
NS60659101	1	NS60658001	300
NS61620901	1	NS61630003	450
NS61630003	1	NS61630105	450
NS61630102	1	NS61630105	1600
NS61630105	1	NS61631106	1600
NS61630202	1	NS60639203	375
NS61631001	1	newmh7	1600
NS61631002	1	NS61631001	1600
NS61631102	1	NS61631105	1800
NS61631104	1	NS61631102	1650
NS61631105	1	NS61631002	1600
NS61631106	1	NS61631104	1600

NS61632101	1	NS61631105	700
NS61640603	1	NS60649601	675
NS61640605	1	NS61640603	610
NS61640803	1	NS60649802	600
NS61640905	1	NS60649802	650
NS61641603	1	NS61640605	450
NS61641609	1	newmh14	600
NS61641701	1	NS61641603	375
NS61641707	1	NS61641609	600
NS61641801	1	NS61641804	600
NS61641804	1	NS61641707	600
NS61641999	1	NS61641801	600
NS61642801	1	NS61641801	580
NS61643806	1	NS61641804	610
NS61644903	1	NS61643806	530
NS61650001	1	NS61650010	770
NS61650003	1	NS61650012	520
NS61650007	1	NS61640905	635
NS61650009	1	NS60659008	550
NS61650010	1	NS61650009	550
NS61650010	2	NS61650007	620
NS61650011	1	NS61650010	760
NS61650012	1	NS61650001	520
NS61650102	1	NS61650003	500
NS61650301	1	NS61650302	500

NS61650302	1	NS61650102	460
NS61650402	1	NS61650301	500
NS61650403	1	NS61650402	450
NS61650501	1	NS61650403	450
NS61650503	1	NS61650501	450
NS61651001	1	NS61651007	375
NS61651002	1	NS61651001	375
NS61651006	1	NS61651007	750
NS61651006	2	NS61641999	600
NS61651007	1	NS61650011	770
NS61651008	1	NS61651006	700
NS61651103	1	NS61652101	375
NS61651109	1	NS61651111	300
NS61651111	1	NS61651002	375
NS61651513	1	NS61650503	375
NS61652005	1	NS61651008	690
NS61652101	1	NS61652102	450

NS61652102	1	NS61651008	450
NS61652301	1	NS61652306	450
NS61652305	1	NS61652306	300
NS61652306	1	NS61652101	450
NS61652402	1	NS61652301	450
NS61653002	1	NS61652005	690
NS61653102	1	NS61653002	450
NS61653304	1	NS61654302	375
NS61654001	1	NS61653002	720
NS61654101	1	NS61655103	450
NS61654302	1	NS61655301	450
NS61655005	1	NS61654001	720
NS61655101	1	NS61655103	580
NS61655103	1	NS61655005	570
NS61655201	1	NS61655101	580
NS61655301	1	NS61655201	580

C Node Values

Node ID	Chamber Plan Area (m2)
new mh11	4
newmh1	2.6
newmh10	4
newmh12	4
newmh14	1.5
newmh2	2.6
newmh3	2.6
newmh4	2.6
newmh5	2.6
newmh6	2.6
newmh7	4
newmh8	4
newmh9	4
NS59644401	4
NS59645401	16
NS59645407	16
NS59645507	16
NS59645508	16
NS59645510	16
NS59645512	10
NS59646501	10
NS59646511	10
NS59647505	5
NS59647556	5
NS59648404	5
NS59648407	5
NS59648410	5
NS59649302	5
NS59649604	3
NS59649705	3
NS60632801	4
NS60632802	4
NS60632804	4
NS60632901	4
NS60632902	4
NS60633401	4
NS60633501	4
NS60633502	4
NS60633601	4
NS60633602	4
NS60633705	4
NS60633708	4
NS60633801	2
NS60633802	3
NS60633803	2
NS60633806	2
NS60633901	2
NS60633902	2
NS60633905	1
NS60634201	4
NS60634301	4
NS60634302	1
NS60634303	4
NS60634401	1
NS60634403	1
NS60634404	1
NS60634405	1
NS60634409	4
NS60634701	1
NS60634702	2
NS60634710	1
NS60634909	8
NS60635203	4
NS60635302	1
NS60635701	1
NS60635812	1

NS60635813	3
NS60635901	1
NS60635911	1
NS60635922	3
NS60635935	15
NS60635936	15
NS60635945	8
NS60636103	4
NS60636105	4
NS60636201	4
NS60636203	4
NS60636301	1
NS60636302	1
NS60636303	1
NS60636305	1
NS60636401	1
NS60636403	1
NS60636404	1
NS60636408	1
NS60636409	1
NS60636501	1
NS60636503	1
NS60636506	1
NS60636604	1
NS60636605	1

NS60636701	1
NS60636705	1
NS60636707	1
NS60636801	1
NS60636812	1
NS60636815	1
NS60636901	13
NS60636902	3
NS60636909	3
NS60636911	12
NS60636916	12
NS60637001	1
NS60637005	1
NS60637101	4
NS60637102	4
NS60637103	1
NS60637104	1
NS60637106	1
NS60637108	1
NS60637110	4
NS60637201	1
NS60637202	1
NS60637207	1
NS60637301	1
NS60637501	1

NS60637506	1
NS60637705	1
NS60637907	1
NS60637916	9
NS60637920	9
NS60638002	1
NS60638101	4
NS60638102	4
NS60638107	4
NS60638204	1
NS60638206	1
NS60638208	1
NS60638209	1
NS60638301	1
NS60638303	1
NS60638406	1
NS60638501	1
NS60638504	1
NS60638507	1
NS60638601	1
NS60638610	1
NS60638715	1
NS60639101	4
NS60639105	4
NS60639107	4

NS60639203	1
NS60639301	1
NS60639603	1
NS60639606	1
NS60639704	1
NS60640603	3
NS60640606	3
NS60640609	1.2
NS60640701	1.2
NS60640707	2.7
NS60640708	2.7
NS60640709	3
NS60640804	1.2
NS60641102	2
NS60641201	2
NS60641202	2
NS60641203	2
NS60641208	5
NS60641305	2
NS60641309	2
NS60641310	2
NS60641502	1
NS60641503	1.2
NS60641605	2.2
NS60641702	1.2

NS60641704	1.5
NS60641705	1.8
NS60641706	2.2
NS60641708	1.5
NS60641710	1.5
NS60641803	1
NS60641805	1.2
NS60642003	2
NS60642004	4
NS60642005	4
NS60642006	2
NS60642009	5
NS60642010	10
NS60642011	5
NS60642103	4
NS60642501	1.7
NS60642506	1.9
NS60642507	3
NS60642512	3
NS60642605	1.5
NS60642701	1.8
NS60643001	10
NS60643003	2
NS60643007	6
NS60643401	3

NS60643403	1.9
NS60643405	2
NS60643410	3
NS60643412	3
NS60643414	2.6
NS60643604	1.5
NS60643612	1.2
NS60643715	1.7
NS60643716	1.8
NS60643902	3
NS60644502	1.8
NS60644503	1.8
NS60644504	2
NS60644506	3.2
NS60644602	1.2
NS60644603	1
NS60644604	1
NS60644611	1
NS60644614	1
NS60644622	3.2
NS60644624	3
NS60644704	1
NS60644804	3
NS60644903	1.8
NS60644904	3

NS60644906	3
NS60645002	2
NS60645005	2
NS60645009	1
NS60645711	1.8
NS60645715	1.5
NS60645793	1
NS60645794	1
NS60645804	3
NS60645902	1.2
NS60646004	3
NS60646407	2.4
NS60646597	2.4
NS60646598	2.4
NS60646599	1.8
NS60646701	2
NS60647108	4
NS60647203	3.7
NS60647217	1.8
NS60647218	3
NS60647301	3.7
NS60647306	3.7
NS60647404	1.9
NS60647406	1.9
NS60647407	3.7

NS60647503	1
NS60647504	1.7
NS60647601	1.6
NS60647602	1.9
NS60647603	1.8
NS60647702	2.2
NS60647704	2.2
NS60647705	1.6
NS60647706	2.2
NS60647713	2.2
NS60648501	1.7
NS60648601	1.6
NS60648602	1.6
NS60648606	1.7
NS60648710	1.2
NS60648713	1.2
NS60648718	1.7
NS60648719	1.8
NS60648720	1.8
NS60648801	2.2
NS60648901	2.2
NS60648902	1.5
NS60648905	2.2
NS60649601	1.6
NS60649603	1.7

NS60649604	1.7
NS60649712	1.3
NS60649802	1.6
NS60649901	1.5
NS60653004	3
NS60653109	3.2
NS60653110	3.2
NS60653114	2.1
NS60654102	2.1
NS60655003	1.6
NS60655104	1.6
NS60655107	2.1
NS60655112	1.6
NS60656001	1.9
NS60656002	1.9
NS60656101	1.2
NS60656103	1.2
NS60656111	1
NS60656116	1
NS60656118	1.5
NS60656201	1
NS60656213	1.5
NS60656215	1.5
NS60656305	1.4
NS60656306	1.5

NS60657001	2
NS60657002	2
NS60657008	1.5
NS60657107	1
NS60657109	1
NS60657114	1
NS60657115	1
NS60657302	1
NS60657402	1.4
NS60658001	1.4
NS60658006	1.4
NS60658007	1.4
NS60658102	1
NS60658402	1.2
NS60659008	1.4
NS60659101	1
NS61620299	3
NS61620901	1
NS61630003	1
NS61630102	4
NS61630105	4
NS61630202	1
NS61631001	4
NS61631002	4
NS61631102	4

NS61631104	4
NS61631105	4
NS61631106	4
NS61632101	2
NS61640603	1.6
NS61640605	1.5
NS61640803	1.5
NS61640905	1.6
NS61641603	1.2
NS61641609	1.5
NS61641701	1
NS61641707	1.5
NS61641801	1.5
NS61641804	1.5
NS61641999	1.5
NS61642801	1.4
NS61643806	1.5
NS61644903	1.3
NS61650001	1.8
NS61650003	1.3
NS61650007	1.5
NS61650009	1.4
NS61650010	1.8
NS61650011	1.8
NS61650012	1.3

NS61650102	1.3
NS61650301	1.3
NS61650302	1.3
NS61650402	1.3
NS61650403	1.2
NS61650501	1.2
NS61650503	1.2
NS61651001	1
NS61651002	1
NS61651006	1.8
NS61651007	1.8
NS61651008	1.7
NS61651103	1
NS61651109	1
NS61651111	1
NS61651113	1
NS61651513	1
NS61652005	1.7
NS61652101	1.2
NS61652102	1.2
NS61652301	1.2
NS61652305	1
NS61652306	1.2
NS61652402	1.2
NS61653002	1.7

NS61653102	1.2
NS61653304	1
NS61654001	1.7
NS61654101	1.2
NS61654302	1.2
NS61655005	1.7
NS61655101	1.4
NS61655103	1.4
NS61655201	1.4
NS61655301	1.4

C Pipe Values

US Node ID	Link Suffix	DS Node ID	Width (mm)
new mh11	1	newmh12	1600
newmh1	2	newmh3	1050
newmh10	1	new mh11	1600
newmh12	1	NS61620299	1600
newmh14	1	NS61640603	600
newmh2	1	newmh4	1050
newmh3	2	newmh5	1050
newmh4	1	newmh6	1050
newmh5	2	NS60642103	1050
newmh6	1	NS60642103	675
newmh7	1	newmh8	1600
newmh8	1	newmh9	1600
newmh9	1	newmh10	1600
NS59645401	1	NS59645407	3600
NS59645407	1	NS59644401	1400
NS59645507	1	NS59645401	3600
NS59645508	1	NS59645510	3600
NS59645510	1	NS59645507	3600
NS59645512	1	NS59645508	1300
NS59646501	1	NS59645512	2800
NS59646511	1	NS59646501	2800
NS59647505	1	NS59646511	1800
NS59647556	1	NS59647505	1800
NS59648404	1	NS59648410	1800
NS59648407	1	NS59648404	1800
NS59648410	1	NS59647556	1800
NS59649302	1	NS59648407	1800
NS59649604	1	NS60640606	1200
NS59649705	1	NS59649604	1200
NS60632801	1	NS60632804	1500
NS60632802	1	NS60632801	1500
NS60632804	1	NS60633705	1600
NS60632901	1	NS60632802	1500
NS60632902	1	NS60632901	1500
NS60633401	1	NS60634409	1600
NS60633501	1	NS60633401	1600
NS60633502	1	NS60633501	1600
NS60633601	1	NS60633502	1600
NS60633602	1	NS60633601	1600
NS60633705	1	NS60633708	1600
NS60633708	1	NS60633602	1600
NS60633801	1	NS60634702	675
NS60633802	1	NS60632804	1200
NS60633802	2	NS60633806	700

NS60633803	1	NS60633802	700
NS60633806	1	NS60633801	700
NS60633901	1	NS60633803	675
NS60633902	1	NS60633901	700
NS60633905	1	NS60633902	450
NS60634201	1	NS60635203	1600
NS60634301	1	NS60634303	1600
NS60634302	1	NS60634303	450
NS60634303	1	NS60634201	1600
NS60634401	1	NS60634404	375
NS60634403	1	NS60634409	450
NS60634404	1	NS60634403	300
NS60634405	1	NS60634404	300
NS60634409	1	NS60634301	1600
NS60634701	1	NS60633708	600
NS60634702	1	NS60634701	375
NS60634710	1	NS60634701	600
NS60634909	1	NS60643007	2000
NS60635203	1	NS60636203	1600
NS60635302	1	NS60635203	300
NS60635701	1	NS60634710	600
NS60635812	1	NS60636815	300
NS60635813	1	NS60633802	1200
NS60635901	1	NS60635922	150
NS60635911	1	NS60635936	225

NS60635922	1	NS60635813	1200
NS60635935	1	NS60635945	2500
NS60635936	1	NS60635935	3600
NS60635945	1	NS60634909	2500
NS60636103	1	NS60636105	1600
NS60636105	1	NS60637102	1600
NS60636201	1	NS60636103	1600
NS60636203	1	NS60636201	1600
NS60636301	1	NS60636303	525
NS60636302	1	NS60636301	525
NS60636303	1	NS60636103	525
NS60636305	1	NS60636302	375
NS60636401	1	NS60636409	525
NS60636403	1	NS60636302	525
NS60636404	1	NS60636408	225
NS60636408	1	NS60636401	375
NS60636408	2	NS60636403	375
NS60636409	1	NS60636506	375
NS60636501	1	NS60636506	225
NS60636503	1	NS60636501	225
NS60636506	1	NS60636605	600
NS60636604	1	NS60636707	600
NS60636605	1	NS60636604	600
NS60636701	1	NS60635701	600
NS60636705	1	NS60636707	450

NS60636707	1	NS60636701	600
NS60636801	1	NS60636909	525
NS60636812	1	NS60636705	450
NS60636815	1	NS60636801	525
NS60636901	1	NS60635936	3400
NS60636902	1	NS60636909	1200
NS60636909	1	NS60635922	1200
NS60636911	2	NS60636916	3200
NS60636916	1	NS60636901	2500
NS60636916	2	NS60636902	1200
NS60637001	1	NS60637103	450
NS60637005	1	NS60637001	450
NS60637101	1	NS60637110	1600
NS60637102	1	NS60637101	1600
NS60637103	1	NS60637108	450
NS60637104	1	NS60637106	450
NS60637106	1	NS60637110	450
NS60637108	1	NS60637104	450
NS60637110	1	NS60638102	1600
NS60637201	1	NS60637202	375
NS60637202	1	NS60637110	375
NS60637207	1	NS60637201	375
NS60637301	1	NS60636305	375
NS60637501	1	NS60637506	450
NS60637506	1	NS60636506	450

NS60637705	1	NS60636707	525
NS60637907	1	NS60636812	450
NS60637916	1	NS60637920	2700
NS60637920	1	NS60636911	2700
NS60638002	1	NS60637005	450
NS60638101	1	NS60639105	1600
NS60638102	1	NS60638107	1600
NS60638107	1	NS60638101	1600
NS60638204	1	NS60638206	375
NS60638206	1	NS60638209	375
NS60638208	1	NS60637207	375
NS60638209	1	NS60638303	375
NS60638209	2	NS60638208	375
NS60638301	1	NS60637301	375
NS60638303	1	NS60638301	375
NS60638406	1	NS60638301	375
NS60638501	1	NS60637501	450
NS60638504	1	NS60638507	450
NS60638507	1	NS60638501	450
NS60638601	1	NS60638610	450
NS60638610	1	NS60638507	450
NS60638715	1	NS60637705	300
NS60639101	1	NS60639107	1600
NS60639105	1	NS60639101	1600
NS60639107	1	NS61630102	1600

NS60639203	1	NS60638204	375
NS60639301	1	NS60638204	375
NS60639603	1	NS60639606	525
NS60639606	1	NS60639704	525
NS60639704	1	NS60637705	525
NS60640603	1	NS60642512	1200
NS60640606	1	NS60640603	1200
NS60640609	1	NS60640708	450
NS60640701	1	NS60640707	450
NS60640707	1	NS60640708	1100
NS60640708	1	NS60640709	1100
NS60640709	1	NS59649705	1200
NS60640804	1	NS60640701	450
NS60641102	1	NS60642103	900
NS60641201	1	NS60641102	900
NS60641202	1	NS60641203	900
NS60641203	1	NS60641201	900
NS60641208	1	NS59649302	1800
NS60641305	1	NS60641310	900
NS60641309	1	NS60641202	900
NS60641310	1	NS60641309	900
NS60641502	1	NS60641503	375
NS60641503	1	NS60642506	450
NS60641605	1	NS60641706	900
NS60641702	1	NS60641704	450

NS60641704	1	NS60641708	640
NS60641705	1	NS60641710	630
NS60641706	1	NS60641704	640
NS60641708	1	NS60640707	640
NS60641710	1	NS60641706	630
NS60641803	1	NS60641805	225
NS60641805	1	NS60641702	450
NS60642003	1	NS60642005	900
NS60642004	1	NS60632902	1500
NS60642005	1	NS60642004	1500
NS60642006	1	NS60642003	900
NS60642009	1	NS60641208	1800
NS60642010	1	NS60642011	1800
NS60642011	1	NS60642009	1800
NS60642103	1	NS60642005	1350
NS60642501	1	NS60642506	710
NS60642506	1	NS60643403	800
NS60642507	1	NS60643401	1200
NS60642512	1	NS60642507	1200
NS60642605	1	NS60641605	600
NS60642701	1	NS60641705	750
NS60643001	1	NS60642010	2800
NS60643003	1	NS60642006	900
NS60643007	1	NS60643001	2000
NS60643401	1	NS60643410	1200

NS60643403	1	NS60643410	800
NS60643405	1	NS60643414	680
NS60643410	1	newmh2	1050
NS60643410	2	newmh1	1050
NS60643412	1	NS60643410	1200
NS60643414	2	NS60643412	1062
NS60643604	1	NS60642605	600
NS60643612	1	NS60643604	450
NS60643715	1	NS60643716	730
NS60643716	1	NS60642701	750
NS60643902	1	NS60644906	1200
NS60644502	1	NS60644503	745
NS60644503	1	NS60644504	750
NS60644504	1	NS60643405	830
NS60644506	1	NS60643412	1200
NS60644602	1	NS60644502	450
NS60644603	1	NS60644602	225
NS60644604	1	NS60644603	225
NS60644611	1	NS60644604	225
NS60644614	1	NS60644611	225
NS60644622	1	NS60644506	1250
NS60644624	1	NS60644622	1200
NS60644704	1	NS60644614	150
NS60644804	1	NS60645804	1200
NS60644903	1	NS60644906	770

NS60644904	1	NS60644804	1200
NS60644906	1	NS60644904	1200
NS60645002	1	NS60645005	675
NS60645005	1	NS60646004	450
NS60645009	1	NS60645005	450
NS60645711	1	NS60646701	750
NS60645715	1	NS60645711	600
NS60645793	1	NS60645794	150
NS60645804	1	NS60644624	1200
NS60645902	1	NS60644903	450
NS60646004	1	NS60636902	1200
NS60646407	1	NS60647407	900
NS60646597	1	NS60646407	1000
NS60646598	1	NS60646597	1000
NS60646599	1	NS60646598	740
NS60646701	1	NS60647702	825
NS60647217	1	NS60647218	710
NS60647218	1	NS60647108	1200
NS60647301	1	NS60647306	1400
NS60647306	1	NS60647203	1400
NS60647404	1	NS60647406	800
NS60647406	1	NS60647407	800
NS60647407	1	NS60647301	1400
NS60647503	1	NS60647404	375
NS60647504	1	NS60647404	700

NS60647601	1	NS60647702	675
NS60647602	1	NS60647603	740
NS60647603	1	NS60646599	740
NS60647702	1	NS60647602	800
NS60647704	1	NS60647713	920
NS60647705	1	NS60647704	650
NS60647706	1	NS60647702	920
NS60647713	1	NS60647706	920
NS60648501	1	NS60647504	700
NS60648601	1	NS60648602	675
NS60648602	1	NS60647601	675
NS60648606	1	NS60648501	700
NS60648710	1	NS60647705	450
NS60648713	1	NS60648710	450
NS60648718	1	NS60648713	450
NS60648719	1	NS60648718	720
NS60648720	1	NS60648719	770
NS60648801	1	NS60647704	920
NS60648901	1	NS60648905	920
NS60648902	1	NS60648901	600
NS60648905	1	NS60648801	920
NS60649601	1	NS60649603	675
NS60649603	1	NS60648601	675
NS60649604	1	NS60649603	690
NS60649712	1	NS60648720	500

NS60649802	1	NS60649901	600
NS60649901	1	NS60648902	600
NS60653004	1	NS60643902	1200
NS60653109	1	NS60653004	1200
NS60653110	1	NS60653109	1250
NS60653114	2	NS60653110	880
NS60654102	1	NS60653114	615
NS60655003	1	NS60655107	660
NS60655104	1	NS60655112	680
NS60655107	1	NS60654102	880
NS60655112	1	NS60655107	580
NS60656001	1	NS60655003	600
NS60656002	1	NS60656001	780
NS60656101	1	NS60656103	450
NS60656103	1	NS60656001	450
NS60656111	1	NS60656103	225
NS60656116	1	NS60656111	225
NS60656118	1	NS60655104	600
NS60656201	1	NS60656116	225
NS60656213	1	NS60656215	600
NS60656215	1	NS60656118	600
NS60656305	1	NS60656306	580
NS60656306	1	NS60656213	600
NS60657001	1	NS60657002	840
NS60657002	1	NS60656002	600

NS60657008	1	NS60657002	600
NS60657107	1	NS60657109	375
NS60657109	1	NS60657114	375
NS60657114	1	NS60657115	375
NS60657115	1	NS60656101	375
NS60657302	1	NS60656305	300
NS60657402	1	NS60656305	580
NS60658001	1	NS60658007	570
NS60658006	1	NS60657008	550
NS60658007	1	NS60658006	550
NS60658102	1	NS60657107	375
NS60658402	1	NS60657402	450
NS60659008	1	NS60658007	550
NS60659101	1	NS60658001	300
NS61620901	1	NS61630003	450
NS61630003	1	NS61630105	450
NS61630102	1	NS61630105	1600
NS61630105	1	NS61631106	1600
NS61630202	1	NS60639203	375
NS61631001	1	newmh7	1600
NS61631002	1	NS61631001	1600
NS61631102	1	NS61631105	1600
NS61631104	1	NS61631102	1600
NS61631105	1	NS61631002	1600
NS61631106	1	NS61631104	1600

NS61632101	1	NS61631105	675
NS61640603	1	NS60649601	675
NS61640605	1	NS61640603	610
NS61640803	1	NS60649802	600
NS61640905	1	NS60649802	650
NS61641603	1	NS61640605	450
NS61641609	1	newmh14	600
NS61641701	1	NS61641603	375
NS61641707	1	NS61641609	600
NS61641801	1	NS61641804	600
NS61641804	1	NS61641707	600
NS61641999	1	NS61641801	600
NS61642801	1	NS61641801	580
NS61643806	1	NS61641804	610
NS61644903	1	NS61643806	530
NS61650001	1	NS61650010	770
NS61650003	1	NS61650012	520
NS61650007	1	NS61640905	635
NS61650009	1	NS60659008	550
NS61650010	1	NS61650009	550
NS61650010	2	NS61650007	620
NS61650011	1	NS61650010	760
NS61650012	1	NS61650001	520
NS61650102	1	NS61650003	500
NS61650301	1	NS61650302	500

NS61650302	1	NS61650102	460
NS61650402	1	NS61650301	500
NS61650403	1	NS61650402	450
NS61650501	1	NS61650403	450
NS61650503	1	NS61650501	450
NS61651001	1	NS61651007	375
NS61651002	1	NS61651001	375
NS61651006	1	NS61651007	750
NS61651006	2	NS61641999	600
NS61651007	1	NS61650011	770
NS61651008	1	NS61651006	700
NS61651103	1	NS61652101	375
NS61651109	1	NS61651111	300
NS61651111	1	NS61651002	375
NS61651513	1	NS61650503	375
NS61652005	1	NS61651008	690
NS61652101	1	NS61652102	450
NS61652102	1	NS61651008	450
NS61652301	1	NS61652306	450
NS61652305	1	NS61652306	300
NS61652306	1	NS61652101	450
NS61652402	1	NS61652301	450
NS61653002	1	NS61652005	690
NS61653102	1	NS61653002	450
NS61653304	1	NS61654302	375

NS61654001	1	NS61653002	720
NS61654101	1	NS61655103	450
NS61654302	1	NS61655301	450
NS61655005	1	NS61654001	720
NS61655101	1	NS61655103	580
NS61655103	1	NS61655005	570
NS61655201	1	NS61655101	580
NS61655301	1	NS61655201	580

Bibliography

Papers Presented by the Candidate

Sayers, W., Savić, D., Kapelan, Z., Kellagher, R., 2014. Artificial Intelligence Techniques for Flood Risk Management in Urban Environments. Procedia Eng. 70, 1505–1512. doi:10.1016/j.proeng.2014.02.165

List of References

Anonymous, 2000. In Praise of Bayes [WWW Document]. The Economist. URL http://www.economist.com/node/382968 (accessed 8.28.15).

Ashley, R., Blanksby, J., Cashman, A., Fewtrell, L., Jack, L., Wright, G., Packman, J., Maksimović, C., Kellagher, R., 2008. Adaptable Urban Drainage - Addressing Change in Intensity, Occurance, and Uncertainty of Stormwater (AUDACIOUS), Building Knowledge for a Changing Climate (BKCC). EPSRC/UKCIP.

Bach, P.M., Rauch, W., Mikkelsen, P.S., McCarthy, D.T., Deletic, A., 2014. A critical review of integrated urban water modelling - Urban drainage and beyond. Environ. Model. Softw. 54, 88–107. doi:10.1016/j.envsoft.2013.12.018

Back, T., 1996. Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press, US.

Baldi, P., 1995. Gradient descent learning algorithm overview: a general dynamical systems perspective. IEEE Trans. Neural Netw. 6, 182–195. doi:10.1109/72.363438

Bandyopadhyay, S., Saha, S., Maulik, U., Deb, K., 2008. A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA. IEEE Trans. Evol. Comput. 12, 269–283.

Batchelor, G.K., 1967. An Introduction to Fluid Dynamics. Cambridge University Press.

Bayer, P., Finkel, M., 2004. Evolutionary Algorithms for the Optimization of Advective Control of Contaminated Aquifer Zones. Water Resour. Res. 40.

Bayes, T., Price, R., 1763. An Essay Towards Solving a Problem in the Doctrine of Chance. Philos. Trans. R. Soc. Lond. 0, 370–418.

Behzadian, K., Kapelan, Z., Savić, D.A., Ardeshir, A., 2009. Stochastic Sampling Design using Multiobjective Genetic Algorithm and Adaptive Neural Networks. Environ. Model. Softw. 24, 530–541.

Bekele, E.G., Nicklow, J., 2007. Multi-objective automatic calibration of SWAT using NSGA-II. J. Hydrol. 341, 165–176.

Bertsekas, D.P., 1999. Nonlinear Programming, 2nd ed. Athena Scientific, Cambridge, MA, USA.

Biswajeet, P., Lee, S., Buchroithner, M.F., 2010. A GIS-based back-propagation neural network model and its cross-application and validation for landslide susceptibility analyses. Comput. Environ. Urban Syst. 34, 216–235.

Boelee, L., Kellagher, R., 2015. ADAPT: A Drainage Analysis Planning Tool (No. D43.2), TRUST: Transitions to the Urban Water Services of Tomorrow. HR Wallingford.

Broad, D.R., Dandy, G.C., Maier, H.R., 2005. Water Distribution System Optimization Using Metamodels. J. Water Resour. Plan. Manag. 131, 172–180.

Brownlee, J., 2012. Swarm Algorithms, in: Clever Algorithms: Nature-Inspired Programming Recipes. lulu.com.

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., Hullender, G., 2005. Learning to rank using gradient descent. ACM, Bonn, Germany.

Butler, D., Davies, J., 2010. Urban Drainage, 3rd ed. CRC Press.

Centre for Ecology & Hydrology, 2014. The Recent Storms and Floods in the UK.

Charnes, A., Cooper, W.W., 1961. Management Models and Industrial Applications of Linear Programming. John Wiley, New York.

Chen, A.S., Evans, B., Djordjević, S., Savić, D.A., 2012a. Multi-layered coarse grid modelling in 2D urban flood simulations. J. Hydrol. 470-471, 1–11.

Chen, A.S., Evans, B., Djordjević, S., Savić, D.A., 2012b. A course-grid approach to representing building blockage effects in 2D urban flood modelling. J. Hydrol. 426-427, 1–16. doi:10.1016/j.jhydrol.2012.01.007

Coates, A., Carpenter, B., Case, C., Satheesh, S., Suresh, B., Wang, T., Wu, D.J., Ng, A.Y., 2011. Text Detection and Character Recognition in Scene Images with Unsupervised Feature Learning.

Coello, C.A.C., 2005. Twenty Years of Evolutionary Multiobjective Optimization: A Historical Overview of the Field. IEEE Comput. Intell. Mag. 1, 28–36.

Coello, C.A.C., 1999. A Comprehensive Survey of Evolutionary Multiobjective Optimization Techniques. Knowl. Inf. Syst. 1, 269–308.

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Math. Control Signals Syst. 2, 303–314.

Darwin, C., 1859. On the Origin of the Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life. John Murray, London.

Deb, K., Agrawal, R.B., 1994. Simulated Binary Crossover For Continuous Search Space (Technical No. IITK/ME/SMD-94027). Deptartment of Mechanical Engineering, Indian Institute of Technology, Kanpur, India.

Deb, K., Agrawal, S., Pratab, A., Meyarivan, T., 2000. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II, in: et Al., M.S. (Ed.), . Springer, Paris, France, pp. 849–858.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6, 182–197.

De Jong, K.A., 1975. An Analysis of the Behaviour of a Class of Genetic Adaptive Systems. University of Michigan.

Department for Communities and Local Government, 2010. Planning Policy Statement 25: Development and Flood Risk.

de Werra, D., Hertz, A., 1989. Tabu Search Techniques: A Tutorial and an Application to Neural Networks. Spektrum 11, 131–141.

di Pierro, F., Khu, S.-T., Savić, D., Berardi, L., 2009. Efficient multi-objective optimal design of water distribution networks on a budget of simulations using hybrid algorithms. Environ. Model. Softw. 24, 202–213. doi:10.1016/j.envsoft.2008.06.008

Djordjević, S., Prodanović, D., Maksimović, C., 1999. An approach to simulation of dual drainage. Water Sci. Technol. 39, 95–103.

Dorigo, M., 1992. Optimization, Learning and Natural Algorithms. Politecno di Milano, Milan, Italy.

Dorigo, M., Blum, C., 2005. Ant Colony Optimization Theory: A Survey. Theor. Comput. Sci. 344, 243–278.

Dorigo, M., Maniezzo, V., Colorni, A., 1997. Ant System: Optimization by a Colony of Cooperating Agents. IEEE Trans. Man Syst. Cybern. - Part B 26, 29–41.

Dorigo, M., Maniezzo, V., Colorni, A., 1991. Positive Feedback as a Search Strategy. Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy.

Dorigo, M., Stutzle, T., 2004. Ant Colony Optimization. MIT Press, Cambridge, Massachusetts.

Dorigo, M., Stutzle, T., 2002. A Short Convergence Proof for a class of ACO Algorithms. IEEE Trans. Evol. Comput. 6, 358–365.

Environment Agency, 2009. Flooding in England: A National Assessment of Flood Risk.

Fedeski, M., Gwilliam, J., 2007. Urban sustainability in the presence of flood and geological hazards: The development of a GIS-based vulnerability and risk assessment methodology. Landsc. Urban Plan. 83, 50–61.

Fewtrell, L., Kay, D., 2008. An attempt to quantify the health impacts of flooding in the UK using an urban case study. Public Health 122, 446–451.

Fonseca, C.M., Fleming, P.J., 1993. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, in: Forrest, S. (Ed.), . Morgan Kauffman Publishers, San Mateo, California.

Fonseca, C.M., Paquete, L., López-Ibáñez, M., 2006. An improved dimension-sweep algorithm for the hypervolume indicator, in: IEEE Congress on Evolutionary Computation. IEEE, Vancouver, BC, Canada, pp. 1157–1163. doi:10.1109/CEC.2006.1688440

Foord, M., Muirhead, C., 2009. IronPython in Action. Manning Publications Co., Greenwich, CT, USA.

Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A., Parizeau, M., Gagné, C., 2012. DEAP: Evolutionary Algorithms Made Easy. J. Mach. Learn. Res. 13, 2171–2175.

Fu, G., Kapelan, Z., 2010. Embedding Neural Networks in Multiobjective Genetic Algorithms for Water Distribution System Design. Tucson, Arizona, USA.

Garcìa, L., Barreiro-Gomez, J., Escobar, E., Téllez, D., Quijano, N., Ocampo-Martinez, C., 2015. Modeling and Real-Time Control of Urban Drainage Systems: A Review. Adv. Water Resour. doi:10.1016/j.advwatres.2015.08.007

Garey, M.R., Johnson, D.S., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness, A Series of Books in the Mathematical Sciences. W. H. Freeman and Company, United States.

Geman, S., Geman, D., 1984. Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. Mach. Intell. 6, 721–741.

Gendreau, M., Potvin, J.Y., 2005. Tabu Search, in: Burke, E.K., Kendall, G. (Eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques. Springer, pp. 165–186.

Ghimire, B., Chen, A.S., Guidolin, M., Keedwell, E.C., Slobodan, D., Savić, D.A., 2013. Formulation of a fast 2D urban pluvial flood model using a cellular automata approach. J. Hydroinformatics 15, 676–686.

Gires, A., Onof, C., Maksimović, Ć., Schertzer, D., Tchiguirinskaia, I., Simoes, N., 2012. Quantifying the impact of small scale unmeasured rainfall variability on urban runoff through multifractal downscaling: A case study. J. Hydrol. 442-443, 117–128. doi:10.1016/j.jhydrol.2012.04.005

Glover, F., Laguna, M., 1997. Tabu Search. Kluwer Academic Publishers, Boston.

Glover, F., Taillard, E.D., de Werra, D., 1993. A Users Guide to Tabu Search. Ann. Oper. Res. 41, 3–28.

Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, Massachusetts.

Goldberg, D.E., Wang, L., 1997. Adaptive Niching via Coevolutionary Sharing. Genet. Algorithms Evol. Strategy Eng. Comput. Sci. 21–38.

Gutjahr, W.J., 2002. ACO Algorithms with Guaranteed Convergence to the Optimal Solution. Inf. Process. Lett. 82, 145–153.

Gutjahr, W.J., 2000. A Graph-Based Ant System and its Convergence. Future Gener. Comput. Syst. 16, 873–888.

Hartman, E.J., Keeler, J.D., Kowalski, J.M., 1990. Layered neural networks with gaussian hidden units as universal approximations. Neural Comput. 2, 210–215.

Heaton, J., 2014. Encog 3.3: Development Guide.

Hertz, A., de Werra, D., 1991. The Tabu Search Metaheuristics: How We Used It. Ann. Math. Artif. Intell. 1, 111–121.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems. University of Michigan Press, Michigan, USA.

Holland, J.H., 1962. Outline for a Logical Theory of Adaptive Systems. J. Assoc. Comput. Mach. 3, 297–314.

Hornik, K., 1991. Approximation Capabilities of Mutlilayer Feedforward Networks. Neural Netw. 4, 251–257.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedfordward networks are universal approximators. Neural Netw. 2, 359–366.

Horn, J., Nafpliotis, N., Goldberg, D.E., 1994. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. IEEE Service Center, Piscataway, New Jersey.

Hromadka, T.V., Whitley, R.J., 1988. The design storm concept in flood control design and planning. Stoch. Hydrol. Hydraul. 2, 213–239.

HR Wallingford, D.I.H. Barr, 2006. Tables for the Hydraulic Design of Pipes, Sewers and Channels, 8th ed. ICE Publishing.

Huang, C.Y., Chen, L.H., Chen, Y.L., Chang, F.M., 2009. Evaluating the Process of a Genetic Algorithm to Improve the Back-Propagation Network: A Monte Carlo Study. Expert Syst. Appl. 1459–1465.

Igel, C., Hüsken, M., 2000. Improving the Rprop Learning Algorithm, in: Proceedings of the Second International Symposium on Neural Computation.

Ijirii, Y., 1965. Management Goals and Accounting for Control. North Holland, Amsterdam.

Innovyze, 2011. Infoworks Collection Systems Technical Review.

Innovyze, 2007. The InfoWorks/InfoNet COM Interface. Innovyze, Oxford, UK.

Ito, K., Akagi, S., Nishikawa, M., 1983. A Multiobjective Optimization Approach to a Design Problem of Heat Insulation for Thermal Distribution Piping Network Systems. J. Mech. Transm. Autom. Des. 105, 206–213.

JBA Risk Management Limited, Met Office UK, 2012. UK Flooding April to July 2012.

Jones, G., Brown, R.D., Clark, D.E., Willett, P., Glen, R.C., 1993. Searching Databases of Two-Dimensional and Three-Dimensional Chemical Structures using Genetic Algorithms. Morgan Kaufmann, San Mateo, California, USA.

Jourdan, L., Corne, D., Savić, D., Walters, G., 2005. Preliminary Investigation of the `Learnable Evolution Model' for Faster/Better Multiobjective Water Systems Design. Evol. Multi-Criterion Optim. 841–855.

Jourdan, L., Corne, D., Savić, D., Walters, G., 2004. Hybridising Rule Induction and Multi-Objective Evolutionary Search for Optimising Water Distribution Systems, in: Hybrid Intelligent Systems, 2004. HIS '04. Fourth International Conference on. Presented at the Fourth International Conference on Hybrid Intelligent Systems, IEEE, pp. 434–439. doi:10.1109/ICHIS.2004.58

Kalteh, A.M., Hjorth, P., Berndtsson, R., 2008. Review of the self-organizing map (SOM) approach in water resources: Analysis, modelling and application. Environ. Model. Softw. 23, 835–845.

Kannan, S., Baskar, S., McCalley, J.D., Murugan, P., 2009. Application of NSGA-II Algorithm to Generation Expansion Planning. IEEE Trans. Power Syst. 24, 454–461.

Kapelan, Z., Savić, D., Walters, G., 2005. Decision-support tools for sustainable urban development. Eng. Sustain. 158, 135–142.

Kellagher, R.B.B., 2010. Discussion on the subject of SAM Risk 1 versus SAM Risk 2.

Kellagher, R.B.B., Cesses, Y., Counsell, C.J., 2008. Management of urban flood risks-the development of a risk-based method. Manchester University.

Kellagher, R.B.B., Cesses, Y., Sayers, P., 2009. SAM -System-based analysis and management of urban flood risks (No. SR700). HR Wallingford.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by Simulated Annealing. Science 220, 671–680.

Knowles, J.D., Corne, D.W., 2000. Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. Evol. Comput. 8, 149–172.

Kotsiantis, S.B., 2007. Supervised Machine Learning: A Review of Classification Techniques, in: Maglogiannis, I., Karpouzis, K., Wallace, B.A., Soldatos, J. (Eds.), . IOS Press, pp. 3–24.

Kubal, C., Haase, D., Meyer, V., Scheuer, S., 2009. Integrated urban flood risk assessment - adapting a multicriteria approach to a city. Nat. Hazards Earth Syst. Sci. 9, 1881–1895.

Laplace, P.S., 1986. Memoir on the Probability of the Causes of Events. Stat. Sci. 1, 364–378.

Leandro, J., Djordjević, S., Chen, A.S., Savić, D., Stanić, M., 2011. Calibration of a 1D/1D urban flood model using 1D/2D model results in the absence of field data. Water Sci. Technol. 64, 1016–1024.

Lhomme, J., Sayers, P., Gouldby, B., Samuels, P., Wills, M., Mulet-Marti, J., 2008. Recent Development and Application of a Rapid Flood Spreading Method. HR Wallingford, Keble College, Oxford, UK.

Lippman, R.P., 1987. An introduction to computing with neural nets.

Liu, X., Begg, D.W., Fishwick, R.I., 1998. Genetic Approach to Optimal Topology/Controller Design of Adaptive Structures. Int. J. Numer. Methods Eng.

Liu, X., Gong, D., 2011. A comparative study of A-star algorithms for search and rescue in perfect maze. Wuhan.

Long, R., 2008. Water Wastewater Infrastructure: Likelihood-of-failure modelling. Mouchel Group.

López-Ibáñez, M., Paquete, L., Stutzle, T., 2004. On the design of ACO for the biojective quadratic assignment problem. Lect. Notes Comput. Sci. 3172, 214–225.

López-Ibáñez, M., Stützle, T., 2012. The automatic design of multi-objective ant colony optimization algorithms. IEEE Trans. Evol. Comput. 16, 861–875.

López-Ibáñez, M., Stutzle, T., 2010. Automatic configuration of multi-objective ant colony optimization algorithms. Lect. Notes Comput. Sci. 6234, 95–106.

Magnier, L., Haghighat, F., 2010. Multiobjective optimization of building design using TRNSYS simulations, genetic algorithm, and Artificial Neural Network. Build. Environ. 45, 739–746. doi:10.1016/j.buildenv.2009.08.016

Maksimović, C., Prodanović, D., 2001. Modelling of Urban Flooding - Breakthrough or Recycling of Outdated Concepts, in: Urban Drainage Modeling. Presented at the Speciality Symposium on Urban Drainage Modeling at the World Water and Environmental Resources Congress, American Society of Civil Engineers, Orlando, Florida, US.

Marsalek, J., Maksimović, C., Zeman, E., 1998. Hydroinformatics tools for planning, design, operation and rehabilitation of sewer systems. Springer Science and Business Media.

Marsh, T., Hannaford, J., 2007. The summer 2007 floods in England & Wales - a hydrological appraisal.

Masters, T., 1993. Practical Neural Network Recipes in C++. Morgan Kauffman Publishers.

McCulloch, W., Pitts, W., 1943. A Logical Calculus of the Ideas Immanent in Nervous Activity. Bull. Math. Biol. 5, 115–133.

Michalski, R.S., Esposito, F., Saitta, L., 2000. Learning Evolution model: Evolutionary Processes Guided by Machine Learning. Mach. Learn. 38, 9–40.

Miller, B.L., Shaw, M.J., 1996. Genetic Algorithms with Dynamic Niche Sharing for Multimodal Function Optimization. New York, USA, pp. 786–791.

Montana, D.J., Davis, L., 1989. Training Feedforward Neural Networks using Genetic Algorithms. Morgan Kaufmann Publishers Inc., Detroit, Michigan, USA.

Morimoto, T., Takeuchi, T., Hashimoto, Y., 1993. Growth optimization of plant by means of the hybrid system of genetic algorithm and neural network, in: Neural Networks, Proceedings of 1993 International Joint Conference on. IEEE, Nagoya, Japan, pp. 2979–2982. doi:10.1109/IJCNN.1993.714348

Neal, R.M., 1995. Bayesian Learning for Neural Networks.

Nicklow, J., Reed, P., Savić, D., Dessalegne, T., Harrell, L., Chan-Hilton, A., Karamouz, M., Minsker, B., Ostfeld, A., Singh, A., Zechman, E., 2010. State of the Art for Genetic Algorithms and Beyond in Water Resources Planning and Management. J. Water Resour. Plan. Manag. 136, 412–432.

Nissen, S., 2012. Fast Artificial Neural Network Library (FANN) [WWW Document]. Ref. Man. FANN 220. URL http://leenissen.dk/fann/html/files/fann-h.html (accessed 9.18.15).

Nissen, S., 2011. Native C# .NET wrapper for FANN released [WWW Document]. URL http://leenissen.dk/fann/wp/2011/02/native-c-net-wrapper-for-fann-released/

Pareto, V., 1896. Cours D'Economie Politique Vol. I & II. F. Rouge, Lausanne.

Parker, D.B., 1982. Learning Logic. Stanford University, Office of Technology Licensing.

Park, J., Sandberg, I.W., 1991. Universal approximation using radial basis function networks. Neural Comput. 3, 246–257.

Parliamentary Office of Science and Technology, 2007. Urban Flooding. Parliamentary Office of Science and Technology.

Penning-Rowsell, E., Johnson, C., Tunstall, S., Tapsell, S., Morris, J., Chatterton, J., Green, C., 2005. The benefits of flood and coastal risk management. A handbook of assessment techniques. (The Multi-Coloured Manual). The Flood Hazard Research Centre, Middlesex University.

Pialorsi, P., Russo, M., 2007. Introducing Microsoft® LINQ. Microsoft Press, Redmond, WA, USA.

Prieto, M.S., Allen, A.R., 2009. Using self-organising maps in the detection and recognition of road signs. Image Vis. Comput. 27, 673–683.

Quagliarella, D., Vicini, A., 1997. Coupling Genetic Algorithms and Gradient Based Optimization Techniques, in: Quagliarella, D., Périaux, J., Poloni, C., Winter, G. (Eds.), . John Wiley & Sons, West Sussex, England, pp. 289–309; 14.

Ranjithan, S., Eheart, J.W., Liebman, J.C., 1992. Incorporating Fixed-Cost Component of Pumping into Stochastic Groundwater Management: A Genetic Algorithm-Based Optimization Approach. EOS Trans. AGU 73.

Reed, P., Minsker, B.S., Goldberg, D.E., 2000. Designing a Competent Simple Genetic Algorithm for Search and Optimization. Water Resour. Res. 36, 3757–3761.

Riedmiller, M., Braun, H., 1993. A direct adaptive method for faster backpropagation learning: The rprop algorithm. Proc. IEEE Int. Conf. Neural Netw. 586–591.

Ritzel, B.J., Eheart, J.W., Ranjithan, S., 1994. Using Genetic Algorithms to Solve a Multiple Objective Groundwater Pollution Containment Problem. Water Resour. Res. 30, 1589–1603.

Rodríguez, J.P., McIntyre, N., Díaz-Granados, M., Quijano, J.P., Maksimović, Ć., 2013. Monitoring and modelling to support wastewater system management in developing mega-cities. Sci. Total Environ. 445-446, 79–93. doi:10.1016/j.scitotenv.2012.12.022

Rosenberg, R.S., 1967. Simulation of genetic populations with bio-chemical properties. University of Michigan, Michigan.

Rosenblatt, F., 1958. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. Psychol. Rev. 65, 386–408.

Rossman, L.A., 2000. EPANET 2 Users Manual. United States Environmental Protection Agency, Cincinnati, OH, USA.

Rowley, H.A., Baluja, S., Kanade, T., 1998. Neural network-based face detection. IEEE Trans. Pattern Anal. Mach. Intell. 23–38.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning Internal Representations by Error Propagation. MIT Press, Cambridge, MA, USA.

Sandgren, E., 1994. Multicriteria Design Optimization by Goal Programming, in: Adeli, H. (Ed.), . Chapman & Hall, London, pp. 225–265.

Santarelli, S., Yu, T.L., Goldberg, D.E., Altshuler, E., O'Donnel, T., Southall, H., Mailloux, R., 2006. Military Antenna Design using Simple and Competent Genetic Algorithms. Math. Comput. Model. 990–1022.

Sayers, P.B., Hall, J.W., Meadowcroft, I.C., 2002. Towards risk-based flood hazard management in the UK. Proc. Inst. Civ. Eng. - Civ. Eng. 36–42.

Sayers, W., Savić, D., Kapelan, Z., Kellagher, R., 2014. Artificial Intelligence Techniques for Flood Risk Management in Urban Environments. Procedia Eng. 70, 1505–1512. doi:10.1016/j.proeng.2014.02.165

Schaffer, J.D., 1985. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. pp. 93–100.

Schaffer, J.D., 1984. Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms. Vanderbilt University, Nashville.

Schrijver, A., 1998. Theory of Linear and Integer Programming. John Wiley & Sons.

Scully, T., Brown, K.N., 2009. Wireless LAN Load Balancing with Genetic Algorithms. Knowl.-Based Syst. 529–534.

Smith, K.I., Savić, D., 2006. A Study of Simulated Annealing Techniques for Multi-Objective Optimisation. University of Exeter, Exeter.

Smythe, M., 2013. Cumbria Floods November 2009.

Soriano, P., Gendreau, M., 1996. Diversification Strategies in Tabu Search Algorithms for the Maximum Clique Problems. Ann. Oper. Res. 63, 189–207.

Souza, C., 2015. Accord Framework .NET.

Srinivas, N., Deb, K., 1994. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. Evol. Comput. 2, 221–248.

Svozil, D., Kvasnička, V., Pospíchal, J., 1997. Introduction to multi-layer feed-forward neural networks. Chemom. Intell. Lab. Syst. 39, 43–62.

Syswerda, G., Palmucci, J., 1991. The Application of Genetic Algorithms to Resource Scheduling. Morgan Kaufmann, San Mateo, California, USA.

Tanaka, M., Tanino, T., 1992. Global Optimization by the Genetic Algorithm in a Multiobjective Decision Support System. pp. 261–270.

TxDOT, 2014. Hydraulic Design Manual. Texas Department of Transportation (TxDOT), Bridge Division, Austin, Texas.

Wallingford, H.R., 2009. SAM - System-based analysis and management of urban flood risks. Oxford.

Wang, Q., Guidolin, M., Savić, D., Kapelan, Z., 2014. Two-Objective Design of Benchmark Problems of a Water Distribution System via MOEAs: Towards the Best-Known Approximation of the True Pareto Front. J. Water Resour. Plan. Manag.

Ward, P.J., de Moel, H., Aerts, J.C.J.H., 2011. How are flood risk estimates affected by the choice of return-periods? Nat. Hazards Earth Syst. Sci. 11, 3181–3195.

Werbos, P., 1974. Beyond Regression: New Tools for Predictions and Analysis in the Behavioural Sciences. Harvard.

Wessing, S., 2010. HyperVolume. TU Dortmund University.

Wienke, P.B., Lucasius, C., Kateman, G., 1992. Multicriteria Target Optimization of Analytical Procedures using a Genetic Algorithm. Anal. Chim. Acta 265, 211–225.

Wills, M., 2013. SAM-UMC 1.1.5. HR Wallingford, Oxford, UK.

Wilson, P.B., Macleod, M.D., 1993. Low Implementation Cost IIR Digital Filter Design using Genetic Algorithms. IEEIEEE Workshop Nat. Algorithms Signal Process. 4/1–4/8.

Wojtusiak, J., Michalski, R., 2006. The LEM3 Implementation of learnable evolution model and its testing on complex function optimization problems, in: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation. Presented at the 8th Annual Conference on Genetic and Evolutionary Computation, ACM, pp. 1281–1288. doi:10.1145/1143997.1144197

Woodward, M., 2012. The use of real options and multi-objective optimisation in flood risk management (PhD Thesis). University of Exeter, Exeter, UK.

Woodward, M., Gouldby, B., Kapelan, Z., Hames, D., 2013a. Multiobjective optimisation for improved management of flood risk. J. Water Resour. Plan. Manag. ASCE 2, 201–215.

Woodward, M., Kapelan, Z., Gouldby, B., 2013b. Adaptive Flood Risk management under Climate Change Uncertainty using Real Options and Optimisation. Risk Anal. 1, 75–92.

Yang, X., Gen, M., 1994. Evolution Program for Bicriteria Transportation Problem. Ashikaga, Japan.

Yoon, J., Shoemaker, C., 2001. An Improved Real-Coded GA for Groundwater Bioremediation. J. Comput. Civ. Eng. 15, 224–231.

Zhou, L., Haghighat, F., 2009. Optimization of ventilation system design and operation in office environment, Part I: Methodology. Build. Environ. 44, 651–656. doi:10.1016/j.buildenv.2008.05.009

Zitzler, E., Laumanns, M., Thiele, L., 2002. SPEA2: Improving the Strength Pareto Evolutionary Algorithm, in: Giannakoglou, et al K. (Ed.), . Athens, Greece, pp. 95–100.

Zitzler, E., Thiele, L., 1999. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. IEEE Trans. Evol. Comput. 3, 257–271.

Zitzler, E., Thiele, L., 1998. Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Study, in: Eiben, A.E. (Ed.), . Springer-Verlag, Amsterdam, pp. 292–301.