IMPROVING THE PENALTY-FREE MULTI-OBJECTIVE EVOLUTIONARY DESIGN OPTIMIZATION OF WATER DISTRIBUTION SYSTEMS

EMILY KAMBALAME

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Civil Engineering.

Johannesburg, 2023

DECLARATION

I declare that this dissertation is my own unaided work. It is being submitted to the Degree of Master of Science to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other University.

EMILY KAMBALAME

(Signature of Candidate)

3rd day of December year 2023.

ABSTRACT

Water distribution networks necessitate many investments for construction, prompting researchers to seek cost reduction and efficient design solutions. Optimization techniques are employed in this regard to address these challenges. In this context, the penalty-free multi-objective evolutionary algorithm (PFMOEA) coupled with pressure-dependent analysis (PDA) was utilized to develop a multi-objective evolutionary search for the optimization of water distribution systems (WDSs).

The aim of this research was to find out if the computational efficiency of the PFMOEA for WDS optimization could be enhanced. This was done by applying real coding representation and retaining different percentages of feasible and infeasible solutions close to the Pareto front in the elitism step of the optimization. Two benchmark network problems, namely the Two-looped and Hanoi networks, were utilized in the study. A comparative analysis was then conducted to assess the performance of the real-coded PFMOEA in relation to other approaches described in the literature. The algorithm demonstrated competitive performance for the two benchmark networks by implementing real coding. The real-coded PFMOEA achieved the novel best-known solutions ($419,000 and $6.081 million) and a zero-pressure deficit for the two networks, requiring fewer function evaluations than the binary-coded PFMOEA.

In previous PFMOEA studies, elitism applied a default retention of 30% of the least cost-feasible solutions while excluding all infeasible solutions. It was found in this study that by replacing 10% and 15% of the feasible solutions with infeasible ones that are close to the Pareto front with minimal pressure deficit violations, the computational efficiency of the PFMOEA was significantly

enhanced. The configuration of 15% feasible and 15% infeasible solutions outperformed other retention allocations by identifying the optimal solution with the fewest function evaluations.

ACKNOWLEDGEMENTS

I am profoundly grateful to God for His significant assistance in finishing my MSc Dissertation. I would not have gotten this far without His direction.

I would like to thank Prof. Ndiritu, my research supervisor, for his constant patience and guidance during every stage of my research project. Additionally, I acknowledge Prof Tiku Tanyimboh for being there during my research development stages. His continuous passion for learning has encouraged me to give my best in my profession.

Furthermore, I would like to thank the Agriculture Transformation Initiative Fellowship and Scholarship Fund (ATI FSF) for obtaining a grant from the Foundation for a Smoke-Free World (FSFW) to fund my master's dissertation, as well as the Institute of International Education (IIE) for administering the scholarship. Moreover, I thank Mr Dylan for being empathetic and serving as my scholarship adviser.

My deep appreciation goes to my dear husband, Wisdom Mwale, and my family for their prayers, and encouragement during my master's dissertation journey. Lastly, I thank my colleagues - Dr. Nathan Lungu, Aaron Kapata, Dr. Williams Dunu, Dr. Vincent Siwawa, Abideen Owolabi, Mercy Shafe, Dr. Elna Owembanazi and Liteboho Rantso - for establishing a friendly, pleasant working atmosphere that enhanced my studies. Their continuous support was genuinely priceless.

DEDICATION

This dissertation is dedicated to my late grandfather, David Kambalame, my late uncle, Peter Kambalame, and all the victims of Cyclone Freddy in Malawi.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter Two

Chapter Three

Chapter Four

LIST OF TABLES

LIST OF APPENDICES

Appendix A: Data Input and Supplementary Results for Two-loop Network

Appendix B: Data Input and Supplementary Results for Hanoi Network

LIST OF SYMBOLS

Blended crossover	BLX
Demand Driven Analysis	DDA
Dynamic link library	DLL
Elitist Multi-Objective evolutionary algorithm	EMOEA
Evolution Algorithm	EA
Extended period simulation	EPS
Genetic Algorithm	GA
Global Gradient Method	GGM
Harmony search	HS
Heuristic Crossover	HX
Multi-objective elitist genetic algorithm	MOEGA
Multi-Objective Evolutionary Algorithm	MOEA
Multi-Objective Particle Swarm Optimization	MOPSO
Non-Dominated Sorting Genetic Algorithm II	NSGA II
Number of function evaluations	NFEs
Particle Swarm Optimization	PSO
Penalty-Free Multi-Objective Evolutionary Algorithm	PFMOEA

Pipe Index	PI
Polynomial Mutation	PM
Pressure-dependent analysis	PDA
Simulated Annealing	SA
Simulated Binary Crossover	SBX
Strength Pareto Evolutionary Algorithm 2	SPEA2
Surrogate Assisted Evolutionary	SADE
United States Environmental Protection Agency	USEPA
Water Distribution Network	WDN
Water Distribution System	WDS
World Health Organization	WHO

LIST OF ACRONYMS AND NOMENCLATURE

actual pressure at node i	Pr_i
available head at node i	H_i
Chezy-Manning dimensionless conversion factor (10.29 in SI)	η
child solution	C_k
cost of the pipe	c(D)
cost per unit length of pipe x with diameter D_x	$c(D_x)$
corrected flow rate	Q_{ij}^k
corrected head	H_{ij}^k
Darcy-Weisbach friction factor	f
dependent variable of u and η	δ_k
desired head at node n	H_n^{des}
diameter of the pipe	*D*
flow exponent is equivalent to 1.852	n_f
flow rate in pipe	Q
gravitation due to gravity	*G*
Hazen-Williams conversion factor	Ω

head loss in a pipe	h
Inflow in a pipe	Q_{in}
inflows and outflow of the node *i*	q_i
length of pipe *x*	L_x
lower limits of the parent component	p_k^l
maximum allowable pressure heads at a node	$H_{n.max}$
minimum allowable pressure heads at node	$H_{n.min}$
Number of pipes or population size	N
Outflow	Q_{out}
parent solution	p_k
next iteration number	kth
pipe resistance coefficient	K
Polynomial Mutation	PM
resistance coefficient	R
required demand or supply at node i	Q_i^{req}
required head at node	H_i^{req}
roughness coefficient	*C*
uniform random number	u

upper limit of the parent component	p_k^u
Diameter of pipe x	D_x

1 INTRODUCTION

1.1 Background

Water Distribution Systems (WDSs) are essential in modern society, and their value cannot be overstated. These systems serve as passive infrastructure for transporting potable water from treatment plants to consumers, satisfying residential, industrial, commercial, and firefighting requirements (WHO, 2014). Nonetheless, it is worth noting that the construction, operation, and maintenance costs associated with WDS are usually very high (Sangroula *et al.*, 2022). According to Sangroula *et al.* (2022), the WDS accounts for approximately 80% of the entire cost of a water supply project. Therefore, designing a cost-effective and sustainable Water Distribution Network (WDN) is vital.

Previously, engineers relied on trial-and-error techniques to find low-cost WDS solutions (Zarei *et al*., 2022). However, engineers have been doing it for a few decades using optimization techniques to select cost-effective WDS that meet the required water demands while maintaining adequate pressure (Zarei *et al.,* 2022). One distinctive aspect of engineering optimization is the use of multiple yet sometimes conflicting decision-making criteria alongside constraints (Deb *et al*., 2002; Mala-Jetmarova *et al*., 2018). Such problems have optimal solutions in the constraint boundary region, whereas the infeasible and feasible solutions are located on both sides of the boundary region (Siew and Tanyimboh, 2012a).

The computational complexity of optimizing WDNs is high due to the numerous physical components and operational requirements inherently associated with the systems (Sangroula *et al*., 2022). As a result, traditional optimization approaches such as enumerative and calculus have seen a major decrease in recent decades (Savic and Walters, 1997; Tran, 2005). Therefore, the development of random guided search techniques known as metaheuristics enables a trade-off between computational time and quality Mala-Jetmarova *et al*., 2018). These search techniques include hill climbing, Tabu search, and Evolutionary Algorithms (EAs) (Savic and Walters, 1997; Tran, 2005). Fang (2007) stated that EAs solve complex multi-objective optimization problems by mimicking biological theories. EAs are integrated into hydraulic models like EPANET, and the resulting nodal head values are assessed using pressure-related constraints (Zarei *et al*., 2022). Over the past decade, various techniques falling under the umbrella of EAs have been developed and employed (Mala-Jetmarova *et al*., 2018). One prominently used approach within the array of EAs are the Genetic Algorithms (GAs), which has gained recognition and broad usage (Deb *et al*., 2002; Tran, 2005).

One significant drawback of EAs is their inability to deal with constraints directly (Siew *et al.*, 2014). Most WDS optimization studies use the conventional Demand Driven Analysis (DDA) method as a hydraulic solver (Kadu *et al.*, 2008). However, DDA cannot simulate pressure-deficient conditions due to the assumption that all nodal demands are fully supplied regardless of pressure (Siew and Tanyimboh, 2012b). Therefore, many past studies (Kadu *et al*., 2008; Abdy Sayyed *et al.,* 2019; Sangroula *et al.,* 2022; Zarei *et al.,* 2022) utilized penalty functions or special techniques to handle pressure constraint violations. In this context, infeasible solutions under the cost-driven objective function incur extra penalty costs based on its current state of deficiency (Siew and Tanyimboh, 2012a). The extent of this shortfall is directly proportional to the corresponding penalty cost (Kadu *et al*., 2008). This implies that as the level of deficiency increases, so does the penalty cost (Kadu *et al*., 2008). However, the disadvantage of this approach is that additional expertise is required to calibrate the penalty parameters (Siew and Tanyimboh, 2012a). Furthermore, the method is not simple and is fine-tuned on a case basis (Mala-Jetmarova *et al.,* 2018).

Siew and Tanyimboh (2012a) developed the Penalty-Free Multi-Objective Evolutionary Algorithm (PFMOEA) approach to optimize WDSs to address the drawback mentioned earlier. The approach creates a multi-objective evolutionary search, eliminating the requirement for ad hoc penalty functions, additional "boundary search" parameters, or specialized constraint handling procedures (Siew *et al*., 2014). It achieves this through the use of pressure-dependent analysis (PDA) (Siew *et al*., 2014). PDA can simulate both normal and pressure-deficient networks, enabling accurate and rapid identification of feasible solutions in the solution space (Siew and Tanyimboh, 2012b).

In earlier PFMOEA research, solutions were encoded and decoded using binary representation between the solution and coding spaces (Siew and Tanyimboh, 2012a; Seyoum and Tanyimboh., 2016; Tanyimboh and Seyoum, 2020). Although binary coding is simple to use, some researchers (Siew, 2011; Siew *et al.,* 2014) claim that it generates redundant codes. These codes do not match any members of the finite discrete set to which the encoded parameter belongs (Siew, 2011; Siew *et al.,* 2014). Thus, real coding has been suggested by Tanyimboh (2021) as a suitable substitute for redundant codes and to increase the GA's computational efficiency.

Previous studies (Siew *et al*., 2014, 2016; Seyoum *et al*., 2016) have characterized the PFMOEA as a bias-free approach that does not impose penalties on infeasible solutions. However, a closer examination reveals that this perception is not entirely accurate. The PFMOEA incorporates an elitism mechanism that involves preserving 30% of the population, consisting of the least cost-effective feasible solutions, within each generation (Siew and Tanyimboh, 2012a). Concurrently, the remaining 70% of solutions undergo selection using the crowding distance operator, which fills the remaining population slots (Siew and Tanyimboh, 2012a). Consequently, the preference for preserving feasible solutions in the PFMOEA leads the solutions to primarily approach the active constraint boundary from the feasible side (Siew and Tanyimboh, 2012a). This behaviour limits exploration of both the infeasible and feasible sides (Siew *et al*., 2014). However, Ray *et al.* (2009) observed that maintaining a small percentage of infeasible solutions close to the constraint boundary during optimization enhances the GA's convergence rate. Hence, refining the elitism process of the PFMOEA becomes crucial to improve the approach performance (Ray *et al.*, 2009).

1.2 Aims and Objectives of the Research

This study aims at finding out if the computational efficiency of the Penalty-Free Multi-Objective Evolutionary Algorithm (PFMOEA) design optimization of water distribution systems could be improved by:

1) Incorporating real coding representation in PFMOEA of WDN optimization process.

2) Altering the elitism concept of PFMOEA that retains 30% of the best least-cost feasible solutions in each generation.

1.3 Main Steps of the Methodology

The proposed approach employed real-coded Non-Dominated Sorting Genetic Algorithm II (NSGA II) with GA operators such as Polynomial Mutation (PM), Simulated Binary Crossover (SBX), and simple tournament selection. This was done to develop a multi-objective framework within the PFMOEA approach. Furthermore, the implementation was conducted in a MATLAB environment and EPANET 2 was used to simulate system hydraulics. The results were compared with the previous binary-coded PFMOEA and other approaches in the literature. This was done to evaluate the real-coded PFMOEA's efficiency and effectiveness. The second part of the study investigated the elitism concept in PFMOEA using different retention percentages of infeasible and feasible solutions in each generation. The Two-looped and Hanoi networks were utilised to achieve the objectives of the study.

1.4 Layout of the Thesis

The dissertation is composed of seven chapters. Following the introduction chapter, the dissertation has the following structure:

Chapter 2 provides a literature review of the analysis of water distribution systems (WDSs), exploring the fundamental elements integral to their formulation and modelling. This chapter has also tackled the applications, standard procedures, operators, solution representation, boundary search techniques, and constraint handling methods of Genetic Algorithms (GA). In addition, the PFMOEA was also reviewed in detail.

Chapter 3 presents the proposed research methodology, as summarized in section 1.3.

Chapter 4 provides the results and discussions of the study. Much emphasis is on the results achieved using real-coded PFMOEA and the retention of different percentages of feasible and infeasible solutions on the active constraint region.

Chapter 5 concludes the study and presents suggestions for further research.

Chapter 6 contains a list of references utilized in the present research.

Chapter 7 presents the input data used in this study.

2 HYDRAULIC ANALYSIS AND DESIGN OPTIMIZATION OF WATER DISTRIBUTION SYSTEMS USING PENALTY-FREE MULTI-OBJECTIVE OPTIMIZATION APPROACH

The objective of this chapter is to review the water distribution network analysis, including its governing hydraulic equations and types of simulations. Additionally, it discusses the design optimization of WDSs using Genetic Algorithms (GA) and the Penalty-Free Multi-Objective Evolutionary Algorithm (PFMOEA).

2.1 Review of Water Distribution Network Analysis

Water distribution models have evolved into fundamental engineering tools, vital to numerous applications within water utilities, including design, calibration, rehabilitation, and operation (Bello *et al*., 2019). These models have the capability to simulate system dynamics across an array of scenarios, such as pipe failure, peak demands, fire flows, and pipes bursts (Bhave and Gupta, 2006). Through the utilization of WDN models, it becomes possible to predict operational challenges and evaluate potential solutions before allocating resources to real-world projects (Bello *et al*., 2019). These models provide valuable insights that help engineers make informed decisions about investment capital and materials for Water Distribution Systems (WDS) (Bhave and Gupta, 2006).

The WDS may be simulated using either extended-period or steady-state simulation (Siew and Tanyimboh, 2012a). A steady-state analysis simulates a WDS in a single time snapshot under stable or normal conditions (Ezzeldin and Djebedjian, 2020; Khalifeh *et al.*, 2020). In contrast, an extended period analysis simulates the performance of a water distribution system more realistically over a specified time duration (Siew, 2011). This makes it appropriate for modelling tank filling and draining, controlling valve operations, and variations in pressure and flow rates owing to demand fluctuation (Khalifeh *et al.,* 2020).

WDN analysis model employs a node-link formulation regulated by energy conservation around hydraulic linkages and mass conservation at nodes (Khalifeh *et al.,* 2020). Numerical methods are employed to solve the governing hydraulic equations, yielding flow rates and nodal heads within the connections (Ezzeldin and Djebedjian, 2020). The resultant model outcomes undergo comparison against minimal performance requirements to ensure the solution is feasible (Bello *et al*., 2019).

2.1.1 Governing hydraulic equations

WDNs are designed so that one element's hydraulic condition complements the neighboring element's hydraulic state (Rossman *et al*., 2020). Mass conservation and energy conservation concepts are utilized to determine if a WDN is hydraulically balanced (Ilemobade and Stephension, 2006). The continuity equation represents the mass conservation principle, met when a network's sum of inflows and outflows is zero (Zarei *et al.,* 2022). This occurs when the mass entering any pipe equals the mass leaving the pipe. In network modelling, outflows are taken at network junctions. The continuity equation is written as follows:

$$\sum_{i=1}^{n} q_i = 0 \tag{2.1}$$

where i is the node, q_i is the nodal inflows and outflow, and n is the number of pipes connected at the node.

The conservation of energy is stated using the head loss equation (Zarei *et al.,* 2022). This occurs when the total head losses in the pipes that constitute each loop are equal to zero as expressed in Equation 2.2.

$$\sum_{j=1}^{m} h_j = 0 \tag{2.2}$$

where m is the number of pipes in the loop, and h_j is the head loss in each pipe j.

In a hydraulic system, fluid possesses kinetic energy (due to movement), potential (elevation) and pressure energy (Haghighi *et al*., 2011). The reduction in water pressure or potential energy as it flows through a WDS is referred to as head loss (Haghighi *et al*., 2011). Much of the head loss in a pipeline system is caused by friction between the fluid and the walls of a pipe (Bhave and Gupta, 2006). Additionally, it is also as a result of the fluid particles as they move relative to one another (Bhave and Gupta, 2006). Darcy-Weisbach (Equation 2.3), Hazen-Williams (Equation 2.4), and Chezy-Manning's (Equation 2.5) equations can be used to calculate pipe head loss (Bhave and Gupta, 2006).

$$h = \frac{8fLQ^2}{gD^5\pi^2} \tag{2.3}$$

$$h = \frac{\omega LQ^{1.852}}{C^{1.852}D^{4.871}} \tag{2.4}$$

$$h = \frac{\eta L(CQ)^2}{D^{5.33}} \tag{2.5}$$

where h, L, and D are the pipe's head loss, length, and diameter in metres. f is the pipe’s Darcy-Weisbach friction factor, Q is the flow rate in a pipe in cubic meters per second, g is the gravitation due to gravity, C is the roughness coefficient, and η is the Chezy-Manning dimensionless conversion factor (10.29 in SI).

The unit conversion factor ω is determined by the units used in Equation (2.4). As a result, the calculation of head loss in Equation (2.4) is strongly dependent on the specific value assigned to ω. By choosing a greater ω value, the head loss in the pipe increases, necessitating larger pipe sizes and increasing the network's overall cost (Palod *et al*., 2021). A lower ω value, on the other hand, minimizes head loss, allowing for smaller pipe diameters and thereby lowering network costs (Palod *et al.,* 2021). As a result, it is not appropriate to directly compare the costs of different networks from different studies since researchers utilize different values of ω (Palod *et al.,* 2021). Siew and Tanyimboh (2012a), as well as Savic and Walters (1997) used similar ω values of 10.5088 and 10.9031. On the other hand, Cunha and Sousa (2001) solely employed a ω value of 10.5088 while Ilemobade and Stephenson (2006) used a ω value of 10.7. EPANET 2 hydraulic simulator uses a Hazen-Williams equation with ω value of 10.667 (Rossman, 2000). Thus, the efficiency of the optimization approach can be evaluated by comparing the maximum number of function evaluations (NFEs) (Palod *et al.,* 2021). As they are required to obtain an optimal cost across different ω values (Abdy Sayyed *et al.,* 2019; Palod *et al.,* 2021).

In addition to the equations 2.3 to 2.5, the resistance coefficient for the pipe is used to calculate the head loss h, as shown in equation 2.6.

$$h = KQ^{n_f} \tag{2.6}$$

where the pipe resistance coefficient is represented by K; the flow exponent n_f is equivalent to 2 for Darcy-Weisbach and Chezy-Manning, and 1.852 for Hazen-Williams.

At each demand node, the minimum required heads are often specified to fully evaluate the WDS's capacity to supply nodal requirements. This constraint is briefly written as

$$H_i \geq H_i^{req} \tag{2.7}$$

where H_i is the actual pressure head at the node, and H_i^{req} is the required pressure head at the node. The required pressure head is the minimum pressure at a node above which all demands are met. H_i is determined by hydraulic simulation.

2.1.2 Types of hydraulic simulations

Steady state simulation

The hydraulic simulation in steady state analysis is done in a single period under the premise that reservoir water levels and nodal demands are constant (Khezzar *et al*., 2001; Qiu and Ostfeld, 2021). This assumption does not accurately represent real world network performance since the nodal requirements and reservoir water levels do not stay constant (Khezzar *et al*., 2001). Engineers often utilize steady state simulation for WDN design purposes since it does not need much data and can be used to determine the appropriate network layout and pipe diameters (Qiu and Ostfeld, 2021). Nevertheless, there exist numerous scenarios where steady-state simulation proves to be insufficient, such as when assessing tank volumes or examining valve operations (Qiu and Ostfeld, 2021). In these instances, it becomes imperative to simulate WDNs over a specified time period (Qiu and Ostfeld, 2021).

Extended period simulation

The extended period simulation (EPS) is used to evaluate changes in demand, pump performance, or tank water level over a longer duration (Saleh, 2013; Bello *et al.,* 2019). Due to significant daily swings in demand, the EPS is often done over 24 hours (Czajkoska, 2016).

2.1.3 Network Analysis

The WDN is designed to deliver water to every node in the system with appropriate pressure to satisfy the needs of all users (WHO, 2014). Thus, network analysis methods that include the head-driven and demand-driven analysis are used to simulate and evaluate water flow (Siew, 2011).

Demand-driven network analysis

The conventional approach for analyzing WDNs is demand-driven analysis (DDA). It assumes that nodal demands are satisfied in full irrespective of the nodal pressure (Siew, 2011). In real world, satisfying nodal demands becomes challenging due to the available pressure falling below the required pressure (Seyoum, 2015). Several factors, including increased pipe demand, network expansion, and leakage through joints and pipes, induce pressure drops (Seyoum, 2015). In such operating situations, it is more feasible to apply pressure-dependent analysis, in which nodal supplies are based on nodal pressure, and to evaluate the pressure-outflow connection (Siew and Tanyimboh, 2012b).

Despite its limitations, the DDA method continues to be commonly used in the water industry due to its simplicity. Four distinct approaches exist for addressing the DDA problem. These methods encompass the Hardy-Cross method (Martin and Peters, 1963), Linear Theory (Wood and Charles, 1972), and the Global Gradient approach (Todini and Pilati, 1988). This study presents the Global Gradient method explicitly.

Global Gradient Method

Todini and Pilati (1988) used the Newton-Raphson iterative numerical technique in the Global Gradient Method (GGM) to calculate pipe nodal heads and flow rates. Specifically, the GGM replaces the corrected values for nodal heads and pipe flow rates in the head loss equation as $H_i^k = H_i^k + \Delta H_i^k$ and $Q_{ij}^k = Q_{ij}^k + \Delta Q_{ij}^k$ respectively.

Furthermore, the GGM, like other linear theory methods, does not need solving the continuity equations at all nodes to begin the process (Czajkoska, 2016). The unknowns in the Q-H equations are the nodal heads and pipe flow rates (Czajkoska, 2016). Nonlinear functions are linearized using Taylor's series (Czajkoska, 2016). Therefore, the kth iteration for the non-linear head loss equations can be expressed as

$$\left(H_i^k + \Delta H_i^k\right) - \left(H_j^k + \Delta H_j^k\right) = K(Q_{ij}^k)^{n_f} + n_f K_{ij} \left|Q_{ij}^k\right|^{n_f - 1} \Delta Q_{ij}^k \qquad \forall_{ij} \qquad (2.8)$$

By stating the nodal heads H_i^{k+i} and H_j^{k+i} in corrected terms, Equation 2.8 can be rearranged and rewritten as follows.

$$H_i^{k+i} - H_j^{k+i} - n_f K_{ij} \left| Q_{ij}^k \right|^{n_f - 1} \Delta Q_{ij}^k = K_{ij} (Q_{ij}^k)^{n_f} \qquad \forall_{ij} \qquad (2.9)$$

In the above equation, $H_i^{k+i} = H_i^k + \Delta H_i^k$ and $H_j^{k+i} = H_j^k + \Delta H_j^k$. Substituting the corrected pipe flow rate Q_{ij}^{k+1} for $Q_{ij}^k + \Delta Q_{ij}^k$ in Equation 2.9 results in the following

$$H_i^{k+i} - H_j^{k+i} - n_f K_{ij} \left| Q_{ij}^k \right|^{n_f - 1} Q_{ij}^k = (1 - n_f) K (Q_{ij}^k)^{n_f} \qquad \forall_{ij} \qquad (2.10)$$

Equation 2.10 provides a sequence of linear equations for combining the adjusted nodal heads and pipe flow rates. Because of the linearity of the continuity equation, the relationship may be expressed in adjusted pipe flow rates as

$$\sum_{ij \in j} Q_{ij}^{k+1} + Q_j = 0 \qquad j, \dots, N - 1 \qquad (2.11)$$

Equations 2.10 and 2.11 are solved simultaneously to establish the values of pipe flow rates and nodal heads. Initially, the values of Q_{ij}^k (pipe flow rates) might be given to one or any arbitrary number for the first iteration. As the iteration process continues, the changes in nodal head and pipe flow rates become insignificant (Czajkoska, 2016).

The United States Environmental Protection Agency (USEPA) developed EPANET 2.0 (Rossman, 2000), which has been the industry standard for modelling WDNs for the last twenty years. EPANET 2.0 is a DDA simulator that simulates networks of any size using a hydraulic solver based on the fast GGA

(Rossman, 2000). It also includes extended period simulation and steady state analysis (Rossman, 2000; Qiu and Ostfeld, 2021). EPANET 2.0 calculates friction head loss in pipes using one of the three well-known formulas (Darcy-Weisbach, Chezy-Manning, or Hazen-Williams) (Saleh, 2013). The equations include minor head losses for fittings and bends. Besides that, EPANET 2 computes pumping energy and cost (Rossman, 2000; Rossman *et al*., 2020). The pumps are modelled using a head-flow curve, which may represent various types of pumps with variable or constant speed (Rossman *et al*., 2020). EPANET 2 additionally has tank size functionality whereby the tank height is dependent on the diameter (Rossman, 2000).

EPANET 2.0 provides a wide variety of water quality modelling tools in addition to hydraulic modelling (Rossman, 2000). It allows users to track the age of water in a system and analysis the flow of reactive and non-reactive constituents in the system over time (Rossman, 2000). A global reaction rate coefficient that may be adjusted pipe by pipe is also included (Saleh, 2013). Users may monitor the percentage of flow from one node to another over time, and growth or decay reactions can continue until the limiting concentration is reached (Rossman, 2000; Saleh, 2013). The software is available as an independent external application and an open-source Programmer Toolkit. The Programmer Toolkit comprises over 50 functions and is a dynamic link library (DLL) that empowers researchers to modify EPANET to suit their specific requirements (Saleh, 2013).

The newest version of EPANET 2.0 released by the USEPA is EPANET 2.2 (Rossman *et al*., 2020). It provides the option of implementing PDA or DDA solver for hydraulic analysis (Rossman *et al*., 2020). However, Gorev *et al*. (2021) observed some problems with using the PDA option. EPANET 2.2 assumes that all nodes have the same minimum and required pressures and that each node has a single type of demand (Gorev *et al.*, 2021). This may not reflect real-world WDNs, where the minimum and required pressures can vary between nodes, and some nodes may have fixed fire. Therefore, alternative PDA methods are needed to accurately simulate networks with different pressure requirements or multiple demand types.

Siew and Tanyimboh (2012b) developed an EPANET 2 model extension called EPANET-PDX, incorporating a pressure-dependent analysis feature. EPANET-PDX maintains all the features of EPANET 2, such as demand-driven analysis, extended period simulation, hydraulics, and water quality (Siew and Tanyimboh, 2012b). Unlike EPANET 2.2, EPANET-PDX employs the Tanyimboh and Templeman (2010) logit pressure-dependent function (Siew and Tanyimboh, 2012b). The function aids in the generation of correct results for network simulations with pressure deficits (Seyoum *et al*., 2011; Seyoum and Tanyimboh, 2016).

Pressure Dependent Analysis

The Pressure Dependent Analysis (PDA) takes into account of the relationship between pressures at nodes and the demands in a network (Tanyimboh and Templeman, 2010; Seyoum *et al.,* 2011; Seyoum, 2015). This provides a more precise depiction of network deficiencies, offering improved accuracy (Siew and Tanyimboh, 2012b). The development of PDA models can assist engineers in water companies to assess WDSs performance under various pressure conditions (Gorev *et al.*, 2021). Moreover, these models allow for the identification of pressure-deficient nodes where nodal demand may not be fully met (Seyoum *et al.,* 2011; Seyoum, 2015). In contrast to DDA, PDA provides realistic nodal head values, particularly in scenarios characterized by pressure insufficiency (Siew and Tanyimboh, 2012b). Over time, various functions have emerged to account for the pressure-dependency of nodal consumption. These functions are based on the concept of pressure-outflow relationships (Seyoum *et al.,* 2011). Presented herein are a few examples of PDA functions.

A parabolic equation for nodal pressure and outflow was proposed by Wagner *et al*. (1988) and Chandapillai (1991), which can be written as

$$H_i^{des} = H_i^{min} + R\left(Q_i^{req}\right)^{n_e} \tag{2.12}$$

$$Q_i = Q_i^{req} \qquad\qquad H_i = H_i^{req} \tag{2.13}$$

$$Q_i = Q_i^{req}\left[\frac{H_i - H_i^{min}}{H_i^{req} - H_i^{min}}\right]^{\frac{1}{n_e}} \qquad\qquad H_i^{min} \le H_i \le H_i^{req} \tag{2.14}$$

$$Q_i = 0 \qquad\qquad H_i = H_i^{min} \tag{2.15}$$

where R denotes the resistance coefficient, and the exponent parameter, n_e, can be modified from 1.5 to 2 (Gupta and Bhave, 1996).

Q_i refers to the nodal i outflow that can be provided by the system, while Q_i^{req} is the required demand at node i. H_i is the actual head at node i, and H_i^{min} denotes the nodal head at node i that results in zero outflows, which is the node elevation in practice. Finally, H_i^{req} is the head required at node i to fully meet the demand.

Germanopoulos (1985) developed an equation that is used to approximate the nodal flow in a pressure-deficient network as

$$Q_i = Q_i^{req}\left[1 - b_i e^{-c_i\left(\frac{Pr_i}{Pr_i^*}\right)}\right] \qquad (2.16)$$

where Pr_i is the actual pressure at the node i while Pr_i^* is the part of the pressure at which the required demand of node i is met. The coefficients of a node to be calibrated are given as b_i and c_i. If field data is not obtainable b_i and c_i are assumed to be 10 and 5, respectively. Similarly, Pr_i^* is also assumed as a head to meet 93,2% of the required node demand. Equation 2.16 has a drawback as it tends to give $Q_i \neq 0$ at $H_i = H_i^{min}$ and $Q_i \neq Q_i^{req}$ at $H_i = H_i^{req}$ (Czajkoska, 2016). As a reiteration, Gupta and Bhave (1996) developed an upgraded version of Equation 2.16 that addresses the setback stated earlier.

$$Q_i = Q_i^{req}\left[1 - b_i e^{-c_i\left(\frac{H_i - H_i^{min}}{H_i^{req} - H_i^{min}}\right)}\right] \qquad (2.17)$$

Fujiwara and Ganesharajah (1993) incorporated pressure-dependent outflow in their approach. Although the function of the approach can analyze any WDN, it has a complex form which can be written as

$$Q_i = Q_i^{req}\frac{\int_{H_i^{min}}^{H_i}\left(H - H_i^{min}\right)\left(H_i^{req} - H\right)dH}{\int_{H_i^{min}}^{H_i^{req}}\left(H - H_i^{min}\right)\left(H_i^{req} - H\right)dH} \qquad H_i^{min} \leq H_i \leq H_i^{req} \qquad (2.18)$$

Tanyimboh and Templeman (2010) proposed a nodal outflow function based on a Logit function. It permits a seamless transition from zero to partial nodal outflow, as well as from partial to full demand satisfaction. Thus, when one likens it to other head flow functions, there are no discontinuities in the derivatives or function. Presented herein is Tanyimboh and Templeman (2010) nodal outflow equation.

$$Q_i = Q_i^{req} \frac{\exp(\alpha_i + \beta_i H_i)}{1 + \exp(\alpha_i + \beta_i H_i)} \quad (2.19)$$

The parameters calibrated to control the flow in the field and the shape of the function curve are α_i and β_i, respectively. In case the field data is not obtainable, Tanyimboh and Templeman, (2010) proposed replacing the Demand Satisfaction Ratio (DSR) value with 0.01 in scenarios where $H_i < H_i^{min}$ and 0.999 when, $H_i = H_i^{req}$. When Equation 2.19 is adjusted, the equation will be as presented.

$$\frac{Q_i}{Q_i^{req}} = \frac{exp(\alpha_i + \beta_i H_i)}{1 + exp(\alpha_i + \beta_i H_i)} \quad (2.20)$$

Where the node DSR is Q_i/Q_i^{req} and is associated with values between 0 and 1.0. The correlation of nodal head and DSR values is given below.

$H_i \geq H_i^{req}$ If DSR=1 (2.21)

$H_i \leq H_i^{min}$ If DSR=0 (2.22)

$H_i^{min} \leq H_i \leq H_i^{req}$ If 0<DSR<1 (2.23)

In Equations 2.21 to 2.23, the node demand is fully met when the DSR is equivalent to 1.0 (Czajkoska, 2016). The demand is partially satisfied when the DSR is associated with a value between 1 and 0. In addition, when the DSR is 0, no outflow is achieved.

2.2 Review of Design Optimisation of Water Distribution Systems

The use of EAs to address the challenges of WDN designs have drastically increased over past years. EAs encompass a range of methods including GAs, simulated annealing, harmony search, differential evolution and self-adaptive differential evolution (Czajkoska, 2016; Sirisant and Janga, 2018). These algorithms have consistently demonstrated their efficacy over time, effectively navigating both small and large networks to identify optimal solutions for the WDN design (Saleh, 2013). However, it is worth noting that the dependability and reliability of these algorithms are closely tied to specific parameters, necessitating substantial effort for parameter tuning (Sirisant and Janga, 2018). Hence, GAs are commonly used as they possess built-in mechanisms that promote reliability, robustness, and adaptability across diverse problem domains (Deb *et al*., 2002; Tran, 2005). This reduces the necessity for extensive parameter fine-tuning (Tran, 2005).

2.2.1 Genetic algorithms in WDS

In the early 1960s and 1970s, Holland developed the GA concept (Deb *et al*., 2002). GAs stand as metaheuristic optimization techniques founded on the principles of natural evolutionary processes that is natural selection and genetic inheritance (Savic and Walters, 1997). The core concept of natural selection is based on the idea of evolution (Deb *et al*., 2002). In which individuals with larger fitness levels persist and reproduce, while weaker individuals experience a decline (Deb *et al*., 2002). On the other hand, genetic inheritance dictates that offspring retain fundamental traits inherited from their parents, while random mutations introduce novel attributes (Deb, 2000; Fang, 2007). Should these new traits prove advantageous, they are retained and inherited by subsequent generations (Deb, 2000). Consequently, the average fitness of individuals increases with each successive generation (Deb, 2000). Previous research has shown that GA is reliable and can solve design, rehabilitation, operation scheduling, tank sitting, and water quality optimization problems (Siew, 2011; Czajkoska, 2016). Most studies observed that GAs are not usually trapped in local optimum and can confidently undertake a global search (Konak *et al.*, 2006; Mishra, 2017; Katoch *et al.,* 2021).

Classification of *g*enetic algorithms in water distribution systems

Genetic Algorithms are categorized into two main types, real-coded and binary-coded, based on how they represent solutions in the search space (Katoch *et al.,* 2021). Consequently, when a GA is employed for an optimization task, the potential solutions are expressed within the phenotypic (solution) space and/or the genotype (coding) space (Seyoum, 2015). However, the choice between these two types depends on the nature of the problem being solved and how the solutions are best represented in the algorithm (Katoch *et al.,* 2021). According to various studies (Holland, 1975; Kumar, 2013; Seyoum, 2015; Czajkoska, 2016), proper representation of the phenotypic and genotypic spaces significantly impacts the performance of the GA.

Binary-coded GAs

This is one of the most straightforward and widely used representations in GAs. In this type of representation, the genotype uses bit combinations of 0 and 1 to denote problem variables (Siew, 2011; Seyoum, 2015). Many prior researchers (Barlow, 2009; Seyoum, 2015; Czajkoska, 2016) argue that binary encoding proves highly advantageous for the optimization of WDS design. Particularly, this is due to the discrete nature of decision variables such as pipe diameters (Saleh, 2013). Nonetheless, certain encodings may lead to redundancy when employing binary representations for parameters belonging to a finite discrete set (Siew, 2011; Seyoum, 2015). For instance, in a network with six potential pipe diameter sizes, a 3-bit substring would be required, yielding eight substrings (i.e., 2^3) (Czajkoska, 2016). Consequently, the decision variable would encompass six substrings, with the remaining two demonstrating redundancy within this context (Czajkoska, 2016). As a countermeasure to this issue, Tanyimboh (2021) proposed the adoption of real coding as a viable substitute for these redundant representations, resulting in an enhancement of the computational efficiency of the GA.

Real-coded GAs

In real coding, the actual values represent variables in solutions (Seyoum, 2015; Czajkoska, 2016). Essentially, the genotype space has the same structure as the phenotypic space (Barlow, 2009; Deep *et al*., 2009; Seyoum, 2015). The real coding is advantageous, as it encompasses a continuous search space and accommodate numerous design variables within the framework of an optimization problem (Deep *et al*., 2009; Seyoum, 2015). In contrast, an approach involving bit-string representation would be inefficient, as it would be represented by excessively lengthy chromosomes (Tanyimboh, 2021). Real-coded GAs are also commonly used in automatic control, combination optimization, planning and design (Wang *et al.*, 2019).

Genetic operators

Genetic Algorithms (GAs) employ several operators to navigate the search process effectively. The essential genetic operators, namely selection, crossover, and mutation, constitute pivotal components of this process (Deb *et al al.*, 2000). Both binary-coded and real-coded Genetic Algorithms (GAs) employ identical selection methodologies, as the whole selection mechanism relies on the assessment of fitness values (Panteleev and Metlitskaya, 2011). However, the crossover and mutation types employed by real-coded GAs differ from binary-coded GAs (Jebari and Madiafi, 2013; Marques *et al*., 2015). This difference is due to the distinct encoding schemes employed by each type of algorithm (Marques *et al*., 2015). The reason behind this distinction lies in the nature of the solution space (see section 2.2.1.1). Therefore, to ensure the effective exploration of the continuous solution space, real-coded GAs require crossover and mutation operators that are tailored to handle real-valued representations (Deb *et al*., 2002; Deep *et al*., 2009; Panteleev and Metlitskaya, 2011). These modified operators enable the exploration and exploitation of potential solutions more appropriately, considering the continuous nature of the problem domain (Panteleev and Metlitskaya, 2011).

Selection operator

The selection operator is an important step in reproduction in which the fittest genes are chosen to carry on to the next generation (Barlow, 2009; Deb, 2000). This operator enables the elimination of weak genes while maintaining the population size constant (Deb *et al*., 2002). This operation effectively guides the GA search process toward a region within the search space that holds potential for productive exploration (Deb *et al*., 2002). Nevertheless, caution must be exercised to avoid a single, incredibly fit solution consuming the entire population within a few generations (Jebari and Madiafi, 2013). This would result in the solutions being like one another in the solution space, reducing their diversity (Jebari and Madiafi, 2013).

Furthermore, the selection pressure affects the convergence rate of GA (Katoch, 2021). Premature convergence or the engulfment of the entire population by a single extremely fit solution is an undesirable condition in a GA (Katoch, 2021). Malik and Wadhwa (2014) recommended that premature convergence should be avoided using a low selection pressure (i.e., a reduced degree to which better solutions are chosen). This is done to maintain diversity within the population and facilitate thorough exploration across the search space (Malik and Wadhwa, 2014). Considering this, it is advisable to employ a high selection pressure as the genetic search approaches completion, effectively refining the search space (Katoch, 2021). In essence, the key to solving optimization problems successfully lies in striking the right balance between exploration and exploitation (Katoch, 2021). This equilibrium can be maintained by carefully applying selection pressure, as noted by Hussain and Muhammad (2020).

There are several ways to incorporate the selection operator into GAs. According to researchers (Siew, 2011; Xavier *et al.,* 2013; Seyoum, 2015; Czajkoska, 2016), the commonly used forms of selection are the tournament and roulette wheel. Holland (1975) invented the roulette wheel/proportionate selection, whereby a slot on the roulette wheel represents each person in the population. The wheel is revolved while being fastened at the location indicated on its circumference (Xavier *et al*., 2013).

It is evident that solutions with more fitness have bigger areas/pies on the wheel (Xavier *et al.*, 2013). There is a greater likelihood of landing in front of the fixed point when the wheel is rotated (Xavier *et al.*, 2013). This is because the area filled by a solution and the parent's fittness correlates (Xavier *et al.*, 2013). As a result, a person's fitness directly impacts their chance of being selected (Xavier *et al*., 2013; Seyoum, 2015). However, according to Goldberg and Deb (1991), spinning the roulette wheel to create fitter individuals leads the population to lose genetic variability. This results in the algorithm's premature convergence to a less-than-optimal solution (Goldberg and Deb, 1991).

Another form of selection which is the tournament was suggested by Goldberg and Deb (1991). Within this selection method, a specific number of individuals are selected from the population, and their fitness levels are assessed (Deb, 2000). Typically, a tournament comprises a group size of 2 (Siew, 2011; Seyoum, 2015). The winner is determined by physical fitness, and they are chosen to be a member of the mating pool (Seyoum, 2015). This competitive process is reiterated with varying participants until the mating pool reaches an optimal capacity, as demonstrated Seyoum, (2015). Within the context of tournament selection, adjustments to the selection pressure can be made in accordance with the tournament size (Siew, 2011). Notably, the intensity of selection pressure escalates as the size of the tournament increases (Siew, 2011). Each tournament's winner (the one with the greatest fitness) is chosen for crossover.

Tournament selection allows all individuals to be picked, preserving diversity (Miller and Goldberg, 1995; Filipović, 2003). Although this is the case, the convergence rate is slowed when the tournament size is small (Siew, 2011). This is also applicable as the tournament size increases; weak individuals tend to have a lower chance of being chosen, resulting in a loss of diversity. However, if a tournament has many competitors, for instance four, it will yield an average mating pool where the tournament is limited to just two competitors (Czajkoska, 2016). This pool tends to encompass more fit solutions compared to a situation where the tournament consists of only two competitors (Czajkoska, 2016). Using tournament selection, Miller and Goldberg (1995) created a model that can objectively estimate the selection pressure and convergence rates of GAs. The study results showed that tournament selection can be efficiently coded and that the level of selection pressure can be easily modified.

Crossover operator

The crossover operator controls the GA reproduction stage to produce children by partially exchanging genetic information between the parents (Elaoud *et al*., 2010; Kora and Yadlapalli, 2017; Zainuddin and Samad, 2020). The aim is to produce offspring with superior characteristics to their parents (Zainuddin and Samad, 2020), thereby making the GA search more exploitable (Seyoum, 2015). In this context, exploitation which refers to using successful existing solutions to generate even better and improved ones is implemented (Seyoum, 2015). As a result, the search space continuously narrows while the method converges faster.

In binary-coded GAs, the crossover operators are characterized by a single-point crossover, two-point crossover, or uniform crossover (Elaoud *et al*., 2010; Zainuddin and Samad, 2020). Within the single-point crossover, a designated point along a chromosome is chosen, leading to an exchange of all bits situated on one side of this point (Konak *et al*., 2006; Reca *et al*., 2017). In a two-point crossover, the central half of the parent's genetic material is transferred between the crossover points to generate new offspring (Mishra, 2017). Uniform crossover allows an equal probability of two parents' genes being swapped by allowing a 50% chance of exchanging the first parent's genes, followed by an extra half of the genes from the next parent's genes (Fang, 2007). (Katoch *et al*., 2021) observed that the single-point crossover is straightforward to implement, while uniform crossover is suitable for large solutions. The crossover types have been illustrated in **Figure 2.1.**

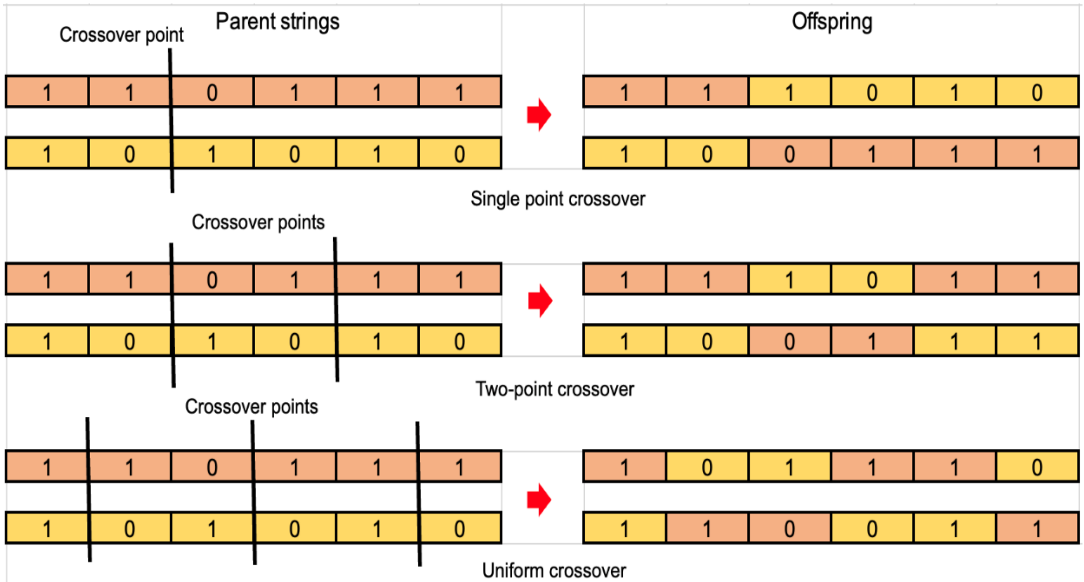

Figure 2. 1 Operation of single-point, two-point, and uniform crossover

According to Deb *et al.* (2002), the crossover operator significantly influences the search for good solutions. This is so, as it impacts the odds of creating better solutions. Furthermore, Deb *et al*. (2002) discovered that as the number of cross sites in the crossover operator increases, string preservation decreases until it approaches a minimum in the case of a uniform crossover operator. Some of the crossover types used in real-coded GAs include heuristic crossover (HX), blend crossover (BLX-α), and simulated binary crossover (SBX) (Deb and Bhushan, 1995; Raghuwanshi *et al*., 2008). For the seek of this research simulated binary crossover (SBX) will be described in details.

Real-coded Genetic Algorithms (GAs) incorporate the Simulated Binary Crossover (SBX) as a fundamental crossover operator (Deb and Bhushan, 1995; Mokhade and Kakde, 2014). The selection of SBX was influenced by the effective application of single-point crossover in binary-coded GAs tailored for problems with discrete search spaces (Sorsa *et al.*, 2008). The SBX operator offers advantages, demonstrating performance on par with or surpassing that of binary-coded GAs using single-point crossovers (Mokhade and Kakde, 2014). Furthermore, SBX operator becomes pronounced in scenarios containing multiple optimal solutions within a confined global basin (Raghuwanshi *et al*., 2008). This also occurs in instances where the predefined lower and upper bounds of global optimum are not available beforehand (Raghuwanshi *et al*., 2008).

To perform SBX, the following procedure is followed:

Step 1: Select a uniform random number, $u \in (0,1)$

Step 2: Utilize Equation 2. 24 to determine the spread factor.

$$\gamma = \begin{cases} 2u^{\frac{1}{(\eta+1)}} & \text{If } u \leqslant 0.5 \\ \dfrac{}{[2(1-u)]^{\frac{1}{(\eta+1)}}} & \text{otherwise} \end{cases} \tag{2.24}$$

The non-negative real number η serves as the distribution index. The probability of producing near-parent solutions increases for large values of η. The selection of distant solutions as children's solutions occurs when η has lower values.

Step 3: Let $p_{1,k}$ and $p_{2,k}$ be two parent solutions who are mated. Children's solutions may then be calculated using Equations (2.25) and (2.26).

$$C_{1,k} = \frac{1}{2}\left[(1-\gamma) * p_{1,k} + (1+\gamma) * p_{2,k}\right] \tag{2.25}$$

$$C_{2,k} = \frac{1}{2}\left[(1+\gamma) * p_{1,k} + (1-\gamma) * p_{2,k}\right] \tag{2.26}$$

Mutation operator

Contrary to crossover, mutation broadens space exploration by introducing random variation inside chromosomes at the gene level (Fang, 2007; Katoch *et al*., 2021). In essence, this involves the stochastic modification of a particular bit value within a string (Fang, 2007). This helps the algorithm explore new areas and prevents it from getting stuck in local optima (Malik and Wadhwa, 2014). The mutation is often used with an extremely low probability (Loewe and Hill, 2010). Introducing a high probability of mutation could result in frequent disruptions to valuable genetic material, consequently impeding the algorithm's ability to rapidly converge (Fang, 2007; Katoch *et al*., 2021).

The polynomial mutation is a variation operator often utilized in real-coded GA optimization problems (Deb and Bhushan, 1995; Sorsa *et al*., 2008). Given that the parents$p_{1,k}$ and $p_{2,k}$ are identical, the polynomial mutation is carried out as follows:

$$C_k = p_k + \left(p_k^u - p_k^l\right) * \delta_k \qquad (2.27)$$

In the Equation 2.27, C_k represents the child solution while p_k the parent solution. p_k^u and p_k^l represent the lower and upper limits of the parent component, respectively, and δ_k is dependent on u and η as depicted in equation (1.28).

$$\delta_k = \begin{cases} (2u)^{\frac{1}{(\eta_m+1)}} & \text{If } u \leqslant 0.5 \text{ otherwise} \\ 1 - [2(1-u)]^{\frac{1}{(\eta_m+1)}} & \end{cases} \qquad (2.28)$$

In this case, the mutation distribution index is applied in conjunction with a uniformly distributed random number ranging between 0 and 1.

A review of the effects of solution representation strategies on GA operators

Panteleev and Metlitskaya (2011) suggested a methodology aimed at tackling the challenge of approximating suboptimal control synthesis for nonlinear deterministic systems. This was accomplished through the utilization of EAs within a framework of conditional optimization, employing both binary and real coding strategies (Panteleev and Metlitskaya, 2011). The investigation resulted in the development of an algorithm, providing the flexibility for users to adjust its parameters and assess its efficacy. The outcomes of the study demonstrated that GAs employing real coding exhibit enhanced speed compared to their binary-coded counterparts (Panteleev and Metlitskaya, 2011). This is due to the elimination of encoding and decoding procedures.

Conventional Genetic Algorithm Procedure

As described in the preceding sections, GAs employ three fundamental biological operations: selection, crossover, and mutation (Fang, 2007). These operators are applied to the population in a cyclic and systematic manner, simulating the evolutionary process of the initial population (Savic and Walters, 1997; Fang, 2007). The first step involves the establishment of decision variables (Zainuddin and Samad, 2020). Subsequently, a random population of solutions is generated using the GA methodology (Zainuddin and Samad, 2020). After fitness values have been assigned to each individual, the chromosomes of the parent population are evaluated in alignment with the optimization objectives (Zainuddin and Samad, 2020).

The selection of the solutions in a mating pool signifies the start of the reproduction phase (Fang, 2007; Mishra, 2017). Following this step, the crossover operator selects the fitter parents for reproduction (Fang, 2007). Only a small percentage of the children undergoes genetic mutation for the population to remain diverse (Mishra, 2017). The fittest members of the offspring population are then picked to make up the next generation after the offspring population is assigned a fitness level (Mishra, 2017). If a solution is not found, the population is replaced with a new one and undergoes the genetic operators once more (Mishra, 2017). Achieving an approximate solution to the problem becomes attainable if one specified termination criteria of the algorithm are met (Mishra, 2017). The individual from the preceding population with the highest fitness function value is then selected (Mishra, 2017).

Elitism preservation strategy in GA

The elitism approach is one technique to prevent losing the best solutions in the population during the GA search process (Deb, 2000). The method described by Purshouse and Fleming (2002) assures that top-performing individuals, known as "elites," are passed down to the next generation, preventing them from being lost or altered throughout the evolution process. The advantage of adopting elitism in evolutionary algorithms is that it provides a reference for desirable areas of the search space throughout generations, allowing for continuous exploration of these areas and perhaps leading to the discovery of local or global optima (Tran, 2005). However, using elitism might reduce diversity within a population and hinder an algorithm's exploratory potential (Purshouse and Fleming, 2002). As a result, elite groups usually comprise a small number of individuals, calculated as a percentage of the entire population (Jebari and Madiafi, 2013). Moreover, determining the optimal configuration for elitism within evolutionary algorithms is contingent upon the specific problem at hand and the characteristics inherent to the search space prediction (Jebari and Madiafi, 2013). This configuration remains elusive to accurate prediction.

Recent development of water distribution systems optimization using Genetic Algorithms

a) Constraint handling and boundary search methods

Evolutionary Algorithms (EAs) are known to have difficulty handling constraints effectively. This is because they were formulated for unconstrained search spaces (Siew *et al*., 2016). The inclusion of constraints makes the search space more complex and the decision space more multi-modal and discontinuous (Deb, 2000). Thus, making it harder for EAs to find optimal or near-optimal solutions (Siew and Tanyimboh, 2012a). Researchers (Savic and Walters, 1997; Deb *et al.,* 2002; Kadu *et al*., 2008; Siew et al., 2012; Abdy Sayyed *et al.,* 2019; Zarei *et al.,* 2022) have taken on this challenge and proposed several constraint-handling methods which focus on retaining near-feasible solutions. As depicted in **Figure 2.2**, these techniques focus on the exploration at the active constraint boundaries that demarcate feasible solutions from infeasible ones. It is in these regions that optimal or close-to-optimal solutions are situated (Dandy *et al*., 1996; Siew, 2011). The progression of techniques to handle constraints and explore boundaries effectively is an active and evolving area of research in WDS optimization (Siew, 2011).

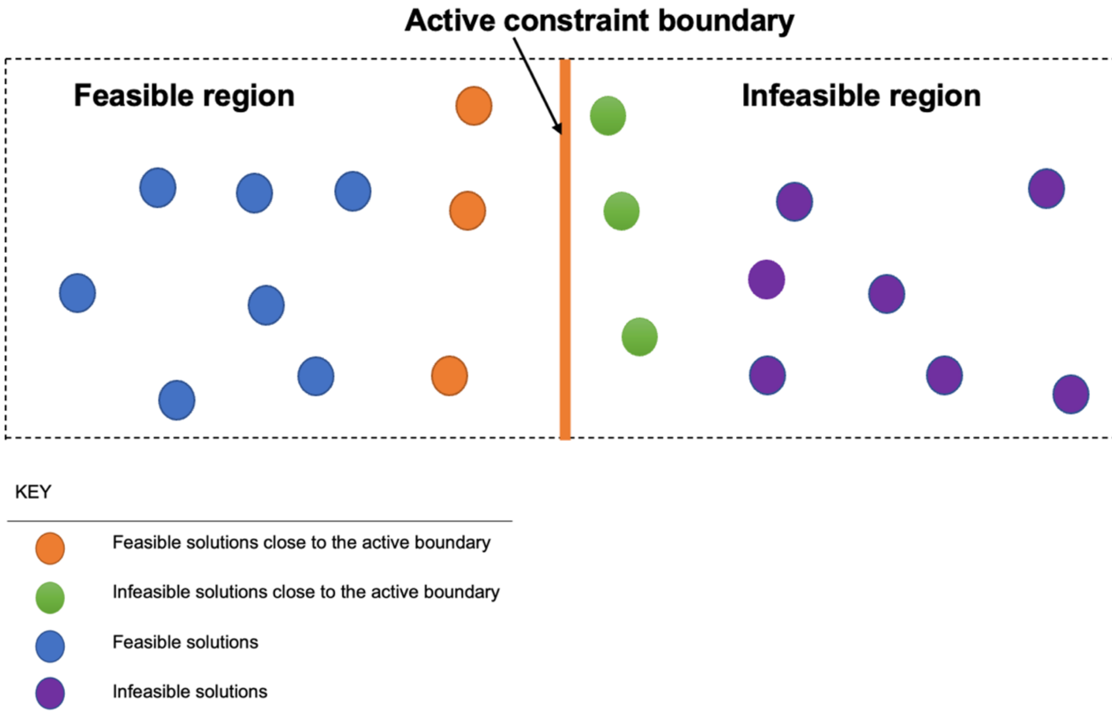

Figure 2. 2 Allocation of solutions on the active boundary region

In the process of designing a WDS through the utilization of an EA, trial solutions are subjected to simulation using a hydraulic simulator (Savic and Walters, 1997). These solutions are subsequently assessed based on pressure constraints, and those that fail to adhere to these constraints are deemed infeasible (Ray *et al*., 2009). Nevertheless, a limitation inherent to EAs is their inability to handle constraints directly, which poses challenges in effectively differentiating between feasible and infeasible solutions (Siew, 2011). In response to this challenge, most studies on WDS optimization using EAs involves the utilization of penalty functions (Savic and Walters, 1997; Deb *et al.,* 2002; Kadu *et al*., 2008; Abdy Sayyed *et al.,* 2019; Sangroula *et al.,* 2022; Zarei *et al.,* 2022). This approach introduces an additional cost for infeasible solutions, proportional to the extent of constraint violation (Zarei *et al.,* 2022). The effectiveness of the penalty function approach depends on the specific structure of the penalties applied (Zarei *et al.,* 2022). Employing a substantial penalty cost can confine the EA's search to the region of feasible solutions, potentially yielding costly and redundant results (Abdy Sayyed *et al.,* 2019). Conversely, a low penalty cost prompt the EA to regard infeasible solutions as viable alternative, causing it to explore the infeasible region (Abdy Sayyed *et al.,* 2019; Sangroula *et al.,* 2022; Zarei *et al.,* 2022).

Selecting appropriate penalty functions and parameters is a complex procedure (Siew and Tanyimboh, 2012a). It frequently demands iterative experimentation to identify the most suitable configurations guiding the exploration towards feasible solutions (Zarei *et al.,* 2022). This process can be time-consuming and may require fine-tuning (Kadu *et al*., 2008). Furthermore, the impact of penalty parameters can vary depending on the specific optimization problem at hand (Abdy Sayyed *et al.,* 2019). Poorly chosen parameters can negatively influence both the optimality of the final solution and the convergence rate (Abdy Sayyed *et al.,* 2019). Moreover, the unconstrained optimization problems formulated through the application of penalty functions encompass a broader objective function space compared to the original problem (De*b,* 2000). As a result, the chances of finding the optimal solution decreases (Deb*,* 2000).

Many studies have attempted to address the challenges associated with EAs optimisation for WDSs. Further details into the advancements in EAs can be found in the works of Wu and Simpson, 2002, Wu and Walski 2005, Khu and Keedwell, 2005, and Afshar and Moeini, 2008.

b) Advances in representation scheme and genetic operators

Savic and Walters (1997) and Dandy *et al*. (1996) introduced the application of Gray coding as a substitute for binary coding. Specifically, this was done to address the convergence effect associated with the hamming cliff. While Vairavamoorthy and Ali (2000) and Kadu *et al.* (2008) utilised real coding to prevent the redundancy problem associated with binary coding. In addition the approach also addressed the stress linked to encoding and decoding diameter variables. In another research by Pattanaik *et al*. (2020), real and binary coding were employed in the WDS optimization using GA. It was observed that in scenarios with a large number of decision variables, real coding is preferred over binary coding. This is due to the fact that binary coding lead to extended chromosome lengths.

Tanyimboh (2021) explored redundant binary codes within the context of optimizing WDNs. This research arose due to the substantial impact of mapping schemes on the solutions' quality in both the phenotype and objective spaces (Tanyimboh, 2021). Thus it was an integral of the computational resources available.The adoption of several fixed mapping strategies produced substantially different solutions within the search space (Tanyimboh, 2021).

c) Search space reduction

The efficiency and efficacy of GA explorations are significantly influenced by the dimensions of the network and the quantity of candidate pipes under consideration (Kadu *et al*., 2008). Researchers have found that larger networks and more candidate pipes lead to a broader search space (Kadu *et al*., 2008; Tanyimboh and Seyoum, 2020). This, in turn, increases the computational time required for convergence to be achieved and reduces the probability of attaining the global optimum solution (Siew *et al*., 2011). It is recommendable to limit the number of candidate pipe diameters to reduce the size of the solution space (Kadu *et al*., 2008). However, this must be done carefully, as choosing inappropriate candidate pipes could result in sub-optimal solutions (Kadu *et al*., 2008).

One approach to reducing the search space is using the pipe index (PI) as suggested by Vairavamoorthy and Ali (2005). The PI assesses a pipe's influence on the hydraulic performance of the network and its relative significance within the entire network (Vairavamoorthy and Ali, 2005). An alternative approach to limiting the search space involves the application of the critical path principle in the design of water distribution networks (Kadu *et al*., 2008). This method assumes that the shortest path from the source to the demand node is the cheapest option for delivering water (Kadu *et al*., 2008). In a separate approach, Haghighi *et al*. (2011) suggested a hybrid optimization scheme that combines the GA with Integer-Linear Programming (ILP) to reduce the search space. The network is transformed into a quasi-branched network, and a single pipe from each loop is excluded and optimized by the GA (Haghighi *et al*., 2011).

Multiple-objective Genetic Algorithm applications in WDNs

WDN optimisation is a difficult task that frequently includes numerous competing objectives (Fang, 2007). As a result, multi-objective evolutionary algorithms (MOEAs) are commonly used to find optimum solutions (Siew and Tanyimboh, 2012a; Siew *et al*. 2014). Therefore, MOEAs has been classified as: non-elitist algorithms and elitist algorithms. Both algorithms work towards finding Pareto-optimal solutions, but they differ in the selection of solutions (Deb, 2000). As a result, the algorithm that is employed is determined by the situation at hand.

A) Non-Elitist Multi-Objective Evolutionary Algorithms

These algorithms do not maintain a set of non-dominated solutions throughout the search process but instead use selection schemes that favour diverse and high-quality solutions (Deb, 2000). The two significant examples of Non-Elitist Multi-Objective Evolutionary Algorithms used in WDN optimization, include: the Multi-Objective Particle Swarm Optimization (MOPSO) algorithm and the Strength Pareto Evolutionary Algorithm (SPEA2) (Deb, 2000).

i) Multi-Objective Particle Swarm Optimization (MOPSO)

The particle swarm optimisation (PSO) technique developed by Kennedy and Eberhart (1995) is among the widely used meta-heuristic optimisation algorithms employed in WDSs. This algorithm draws inspiration from swarm intelligence phenomena observed in nature such as fish schools and flocks of birds (Kennedy and Eberhart, 1995). Therefore, the algorithm aligns with the optimal solution in the context of optimization. Building upon this, Coello and Lechuga (2002) extended the PSO algorithm's capabilities to give rise to the multi-objective PSO (MOPSO) algorithm. This enhancement included the integration of an external archive designed to store non-dominated solutions (Coello and Lechuga, 2002). Thereby, effectively approximating the Pareto set throughout the optimization procedure (Coello and Lechuga, 2002).

ii) Strength Pareto Evolutionary Algorithm (SPEA) and Strength Pareto Evolutionary Algorithm 2 (SPEA2)

The Strength Pareto Evolutionary Algorithm (SPEA) was formulated with the primary objective of identifying solutions that represent the Pareto front within a multi-objective optimization context (Zitzler and Thiele, 1999). Although SPEA boasts inherent advantages, it encounters certain challenges (Zitzler *et al.*, 2001). One such challenge is the lack of density estimation within the objective space, resulting in crowding phenomena and premature convergence towards a limited subset of solutions (Zitzler *et al.*, 2001). Additionally, SPEA utilizes a binary tournament selection method, which occasionally fails to strike a reasonably equilibrium between selection pressure and diversity retention (Zitzler *et al.*, 2001). The algorithm's archiving technique is also rather basic and uses a fixed size (Zitzler *et al.*, 2001). Thereby, it poses challenges to determine which solutions to retain and discard, because the archive does not accurately record the changing pareto front (Zitzler *et al.*, 2001). Furthermore, SPEA necessitates manual parameter adjustment for diverse problem domains (Zitzler *et al.*, 2001). A process that proves time-consuming and complex due to the varying parameter demands of different problems for optimal performance (Zitzler and Thiele, 1999).

It is against the aforementioned challenges that SPEA2 was developed. SPEA2 was improved over SPEA in order to handle a wider variety of problem types and achieve higher convergence, diversity maintenance, and solution quality (Guimarães and Walters, 2011).

B) Multi-Objective Elitist Genetic Algorithms (MOEGAs)

As the name suggest, Multi-Objective Elitist Genetic Algorithms (MOEGAs) are used for optimising WDNs with multiple objectives. These algorithms maintain non-dominated solutions throughout the search process and use selection schemes favouring high-quality and diverse solutions (Deb, 2000).

i) Rudolph's Elitist Multi-Objective evolutionary algorithm (EMOEA)

Rudolph's MOEGAs is based on the principles of MOEAs (Rudolph, 1999). It has demonstrated to be an effective technique for tackling multi-objective optimization problems. Specially, It emphasize on elitism, non-dominated sorting, and crowding distance (Rudolph, 1999; Rudolph *et al*., 2000). This has led to its wide application and integration into various problem domains (Rudolph, 1999). The approach has been particularly valuable as it maintains a diverse set of solutions on the Pareto front through non-dominated sorting and crowding distance assignments (Rudolph *et al*., 2000). Additionally, by balancing exploration and exploitation, the approach generates a well-distributed set of solutions that capture the trade-offs between conflicting objectives (Rudolph, 1999; Rudolph *et al*., 2000). However, the limitation of the Rudolph's MOEGA lies in its scalability to handle high-dimensional and complex problems (Rudolph *et al*., 2000). As a result, the computational complexity can become a challenge when dealing with problems that have numerous decision variables and complex fitness landscapes (Rudolph *et al*., 2000).

ii) Distance-Based Pareto Genetic Algorithms (DBPGA)

The principles of Distance-Based Pareto Genetic Algorithms (DBPGA) exhibit a substantial reliance on the concepts underlined by SPEA and SPEA2 (Schädler *et al*., 2016). However, in contrast to SPEA and SPEA2, which cannot effectively manage infeasible solutions, DBPGA retains these solutions within its population (Schädler *et al*., 2016). This strategic decision is made to prevent any hindrance to the evolutionary procedure (Deb, 2000). Additionary, DBPGA differ from other MOEGAs in terms of the fitness assignment and parent selection employed (Schädler *et al*., 2016). Instead of using traditional fitness functions, DBPGAs calculate a fitness value for each solution based on the distance to its k-nearest neighbors in the objective space (Deb *et al*., 2002). The distance is computed using metrics like Euclidean distance or other appropriate distance measures (Deb, 2000). Furthermore, DBPGAs use a distance-based approach to select parent solutions for mating (Deb, 2000).

Solutions that are closer to each other in the objective space have a higher chance of being selected as parents, thereby, promoting diversity in the population (Schädler *et al*., 2016).

iii) Elitist Non-Dominated Sorting Genetic Algorithm (NSGA II)

Deb *et al.* (2002) suggested the Non-Dominated Sorting Genetic Algorithm (NSGA II) for optimizing water distribution systems. NSGA II addresses the weaknesses of classical optimization methods (Deb *et al.,* 2002; Siew, 2011; Ciro *et al.,* 2016). These weaknesses include: computational complexity, lack of elitism, and the requirement for a predefined sharing parameter (Deb *et al.,* 2002). The philosophy of NSGA-II is based on four main principles, which are: Non-Dominated Sorting, Elite Preserving Operator, Crowding Distance and Selection Operator (Deb *et al*., 2002).

The non-dominated sorting procedure involves sorting population members based on Pareto dominance. It starts by assigning the first rank to non-dominated individuals, forming the first front as illustrated in **Figure 2.3 (a)**. The process continues until all population members are placed in different fronts based on their ranks (Siew, 2011). The elite preserving operator then ensures that the best solutions in a population are carried forward directly to the next generation. This means that the non-dominated solutions identified in each generation continue into subsequent generations until they are surpassed by some other solutions (Siew, 2011).

The crowding distance in NSGA-II is a metric used to measure the density of solutions within a front (a set of non-dominated solutions) in the objective space (Deb *et al.* (2002). It's calculated for each individual based on its proximity to its neighbours within the same front as depicted in **Figure 2.3 (b)**. When comparing solutions with varying crowding distances, a larger crowding distance indicates the existence of a solution in a less densely populated region (Deb *et al*., 2002). This distance for an individual solution is determined by the average side length of a cuboid, as illustrated in **Figure 2.3 (b)**. In mathematical terms, if 'f_j^i' represents the 'j^{th}' value of an objective function for the 'i^{th}' individual, and 'f_j^{max}' and 'f_j^{min}' are the maximum and minimum values respectively of the 'j^{th}' objective function among all individuals. Then the crowding distance of the 'i^{th}' individual is calculated as the average distance between the two nearest solutions on either side, as in Equation 2.29.

$$cd\ (i) = \sum_{i=1}^{k} \frac{f_j^{i+1} - f_j^{i-1}}{f_j^{max} - f_j^{min}} \tag{2.29}$$

where k is the number of objective functions.

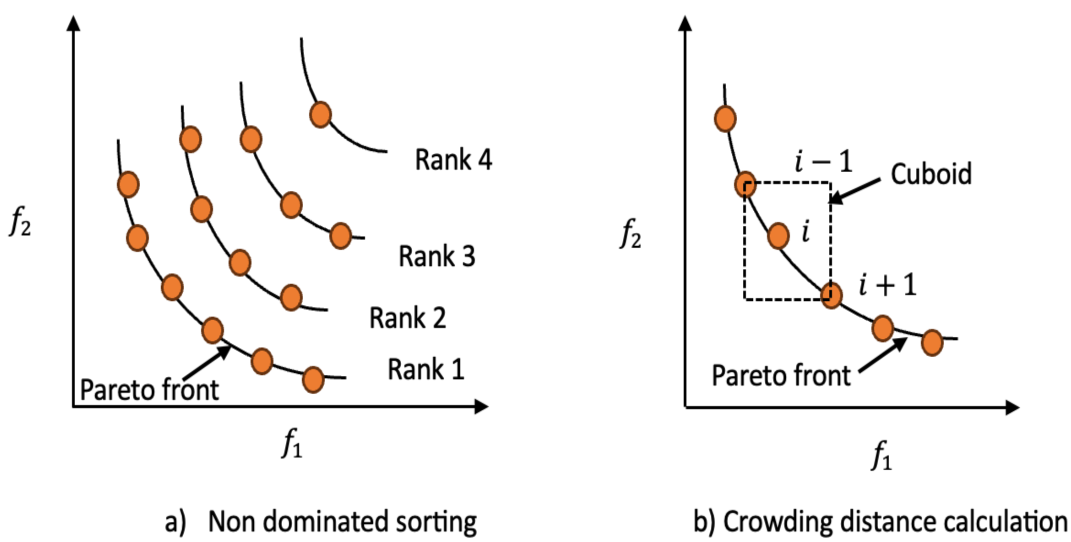

Figure 2. 3 Non- dominated sorting procedure and Crowding distance (Deb *et al*., 2002)

Furthermore, the next generation's population is chosen using a crowded tournament selection operator, utilizing both the rank and crowding distances of the population members for selection (Deb *et al*., 2002). The selection rule is as follows: (i) If two population members have different ranks, the one with the superior rank is chosen for the next generation(Deb *et al*., 2002). (ii) If both population members have the same rank, the one with the greater crowding distance is selected for the next generation (Deb *et al*., 2002).

Figure 2.4 depicts the primary structure of the NSGA-II algorithm. In which the parent population P_t produces offspring population Q_t, and the two populations merge to generate an offspring population R_t of size 2N (Deb *et al.,* 2002; Siew, 2011). The non-dominated sorting is then used to classify the entire population R_t into mutually exclusive non-dominated sets (Deb *et al*., 2002). Any member from the same set is regarded to have the same scale of importance with respect to the objectives (Tran, 2005; Deb *et al.*, 2002). After sorting filling of solutions is done in fronts starting with the best solutions. Following the sorting process, solutions are allocated across consecutive fronts, beginning with the best solutions (Tran, 2005). Subsequently, this procedure extends to the solutions constituting the second non-dominated front, and continues to fill the remaining least non-dominated fronts (Zarei *et al*., 2002). Researchers have observed that incorporating the 2N overall population within the N available slots in the new population is not feasible (Ciro *et al*., 2016; Tran, 2005; Siew *et al.,* 2012). As a result, a niching approach is used to define the final front, which is positioned within the least crowded region of that particular front (Zarei *et al*., 2002).

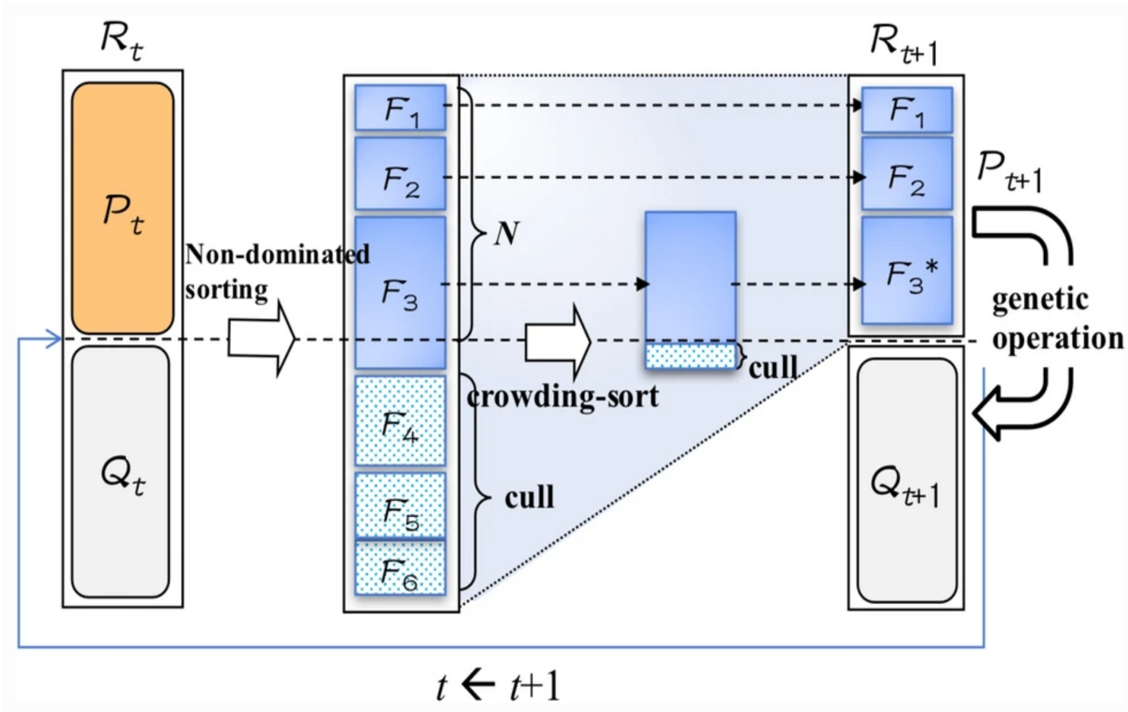

Figure 2. 4 Generating a new population through crowded tournament (Sato and Sato, 2022)

NSGA-II has been extensively applied in water engineering, particularly in the MOEA of WDSs. Zarei *et al*. (2022) coupled NSGA II to EPANET hydraulic simulator in a MATLAB environment. The outcome demonstrated the algorithms strong capacity for identifying the best solutions. Similarly, the NSGA II was used by Jayaram and Srinivasan (2008) to address the challenge of optimal performance-based WDS design and rehabilitation. The algorithm's application in this context demonstrated its effectiveness in generating solutions that enhance system performance. In the domain of drinking WDS security, Preis and Ostfeld (2008) applied NSGA II to address two optimisation objectives. These include: the consumed contaminant mass and the number of operational activities needed to manage and eliminate the contaminant from the system (Preis and Ostfeld, 2008). This application highlighted the algorithm's utility in enhancing the security and reliability of water networks (Preis and Ostfeld, 2008).

Furthermore, Prasad (2010) and Siew *et al*. (2014) utilized the NSGA II to optimize the design and upgrading of the "Anytown" network. This encompassed optimising factors like multi-operating conditions, pump scheduling, and tank sizing and placement (Prasad, 2010; Siew *et al*. 2014). Farmani *et al.* (2006) improved the algorithm's efficacy by integrating network reliability and water quality considerations into their solution for the "Anytown" problem. This additional dimension illustrated the algorithm's flexibility in accommodating multiple objectives. Moreover, Siew and Tanyimboh (2012a) also employed the NSGA II in the framework of its MOEA. The approach was called the Penalty-Free Multi-Objective Evolutionary Algorithms (PFMOEA). Siew *et al.* (2014) extended this approach to consider initial cost, rehabilitation, upgrading cost, repairs, and pipe failures, as well as deterioration over time. Overall, NSGA-II has been successfully applied to the optimisation problems using PFMOEA and the other aforementioned approach. Its ability to handle multiple objectives and constraints simultaneously, as well as its efficiency and robustness, have made it a popular choice for WDS optimization.

2.3 Penalty-Free Multi-Objective Evolutionary Algorithm (PFMOEA)

PFMOEA was developed to solve the multi-objective evolutionary optimization algorithm for WDS (Siew and Tanyimboh 2012a; Siew *et al*., 2014). This penalty-free approach seeks to avoid the usage of penalty functions or other specific constraint-handling strategies within the MOEA search (Siew and Tanyimboh 2012a; Saleh and Tanyimboh, 2013). Thus, its use has shown to be effective and robust in guiding the GA search to optimum or close to optimal solutions (Siew and Tanyimboh 2012a). Additionally, penalty-free techniques offer the benefit of retaining infeasible solutions that hold useful decision variables, which might not be prevalent among feasible solutions (Siew and Tanyimboh 2012a). This inclusion, therefore, serves to preserve diversity within the genetic pool (Siew and Tanyimboh 2012a).

The multi-objective optimization technique employed for the PFMOEA is the elitist-preserving non-dominated sorting genetic algorithm (NSGA-II) (Siew and Tanyimboh, 2012a). PFMOEA evaluates solutions based on performance function, and then ranks them using pareto dominance that considers both cost functions and performance (Siew and Tanyimboh, 2012a). In contrast to NSGA II (Deb *et al*., 2002), PFMOEA strictly ranks solutions as being feasible and infeasible on the basis of their cost-effectiveness (Siew and Tanyimboh, 2012a; Siew *et al*., 2014). All feasible and infeasible non-dominated solutions surpass feasible and infeasible dominated solutions (Siew and Tanyimboh, 2012a). Consequently, non-dominated infeasible solutions are preferred to dominated feasible ones (Siew and Tanyimboh, 2012a).

Prior PFMOEA studies linked NSGA II programme with a hydraulic simulator in C++ environment and the code was written in binary coding (Siew and Tanyimboh, 2012a). While binary coding is widely used, its drawback is the occurrence of redundant codes (Czajkoska, 2016). This occurs when the encoded parameter is part of a finite discrete set whose basic number is not a power of two (Czajkoska, 2016). According to Saleh and Tanyimboh (2014), redundant codes result in the early loss of vital information and beneficial genes, lowering the GA's efficiency. Thus, Tanyimboh (2021) proposed real coding as a viable replacement for redundant codes and as a method of improving the GA's computational efficiency. Therefore, there is a need to incorporate real coding in the PFMOEA, as a way of addressing redundant codes.

Siew and Tanyimboh (2012a) incorporated the pressure-driven hydraulic simulator EPANET-PDX in the PFMOEA. This hydraulic simulator takes into account the pressure conditions at demand nodes (refer to section 2.1.3). On the other hand, Saleh and Tanyimboh (2013, 2014, 2016) employed the EPANET 2 hydraulic simulator to demonstrate the efficacy of the penalty free approach. Demand driven analysis is the term used to describe the hydraulic modelling approach in EPANET 2 (Rossman, 2000; Seyoum and Tanyimboh, 2011; Siew and Tanyimboh, 2012b). It is based on the assumption that flows at nodes are consistently satisfied irrespective of the existing pressures (Seyoum and Tanyimboh, 2011). It should be noted that the PFMOEA approach involved the basic hydraulic constraints on minimum pressure, conservation of energy, and conservation of mass (Siew and Tanyimboh, 2012a; Siew *et al.,* 2014). The aforementioned hydraulic simulators, EPANET 2 (Rossman, 2000), and EPANET-PDX (Siew and Tanyimboh, 2012b), were used to manage these constraints and ensures that they are satisfied internally. For more explanation of EPANET 2.0 and EPANET PDX refer to section 2.1.3.

The past studies on PFMOEA, used the Elitism concept to retain 30% of the population, encompassing the least cost feasible solutions within every generation, as shown in **Figure 2.5**. This subset is distinguished by allocating much higher crowding distance values to each solution (Siew *et al*., 2014). Subsequently, the remaining solutions undergoes selection through the crowding distance operator to occupy the remaining slots within the population (Siew and Tanyimboh, 2012). However, Ray *et al.* (2009) claims that searching both the infeasible and feasible regions improves an approach's convergence rate, as opposed to searching exclusively via the feasible region.

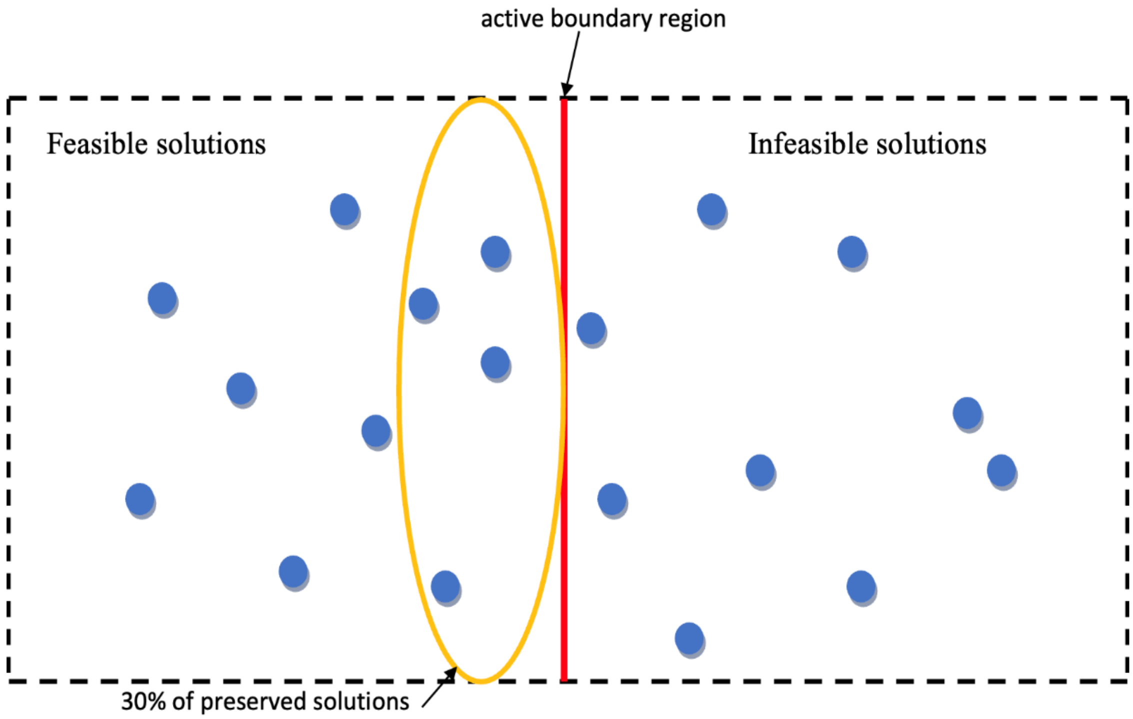

Figure 2. 5 Schematic diagram of the elitism criteria in PFMOEA

2.3.1 The relative merits of PFMOEA

Previous research on PFMOEA has shown that the technique is very reliable and efficient since it is applicable to a spectrum of optimization problems (Siew and Tanyimboh, 2012a). These range from small networks (such as the two-looped network with 8 pipes) to big networks (such as the 251-pipe network) (Siew and Tanyimboh, 2012a; Seyoum and Tanyimboh, 2016). In addition, the approach can be applied to complex network optimization problems. For example, from just initial design or capital cost minimization problems to subsequent rehabilitation, upgrading and capacity expansion network (Siew *et al.,* 2016; Seyoum and Tanyimboh, 2016). Therefore, the initial design problems of relatively simple networks with a few pipes were done on the following networks: the two-loop, Hanoi, Kadu, New York Tunnels and 251-pipe network (Siew and Tanyimboh, 2012a; Seyoum and Tanyimboh, 2016).

Furthermore, Siew *et al.*, (2014) used the PFMOEA to solve the Wobulenzi network optimization problem. As a result, they obtained the cheapest solution of $3,814,298 (Siew *et al.*, 2014). This signifies a 3.5% reduction from the optimal solution of $3,953,663 obtained by Tanyimboh and Kalungi (2008) using the linear programming method. In another similar study, Siew *et al*., (2016) obtained a least total cost of $10.31 million on a complex design and upgrade problem of the Anytown network. The study's results were not only competitive but also demonstrated full feasibility, highlighting its effectiveness of being practical and applicable across various problems (Siew *et al*., 2016).

The other penalty-free approaches previously used (Saleh and Tanyimboh, 2014, 2016) are relatively difficult to use due to their level of complexity. The approach by Tanyimboh and Czajkowska (2018, 2021) introduced a reliability-based algorithm that maximized flow entropy as a substitute for resilience while minimizing the initial construction cost. Entropy is a scale of uncertainty that assesses the resilience of WDS (Saleh and Tanyimboh, 2014). Unlike the past studies that investigated the flow entropy function using one operating condition, Tanyimboh and Czajkowska (2018, 2021) maximized the total entropy value under multiple operating conditions. As a result, compared to prior studies, there was a considerable increase of 274% in the number of nondominated solutions obtained (Tanyimboh and Czajkowska, 2018). In another study, Saleh and Tanyimboh (2014) factored in network connectivity, statistical entropy, and hydraulic feasibility throughout the optimisation process of a penalty-free approach. It is in this light, that the approach succeeded to achieve a whole spectrum of solutions for completely branched, moderately, and completely looped networks (Saleh and Tanyimboh, 2016; Siew *et al.,* 2016).

2.4 Summary of Literature Review

Overall, this chapter reviewed the water distribution network analysis, including its governing hydraulic equations and types of simulations. Additionally, it discussed the design optimization of WDSs using Genetic Algorithms (GA) and the Penalty-Free Multi-Objective Evolutionary Algorithm (PFMOEA) approach The literature review has highlighted the advantages of the PFMOEA and also revealed that this method has applied binary coding to date while real number coding could enhance its efficiency. The literature review further revealed that the PFMOEA applications have subjectively applied 30% of the feasible least cost solutions in implementing elitism. While straddling across the bound of feasibility to include efficient but infeasible solutions close to this boundary could probably also improve the efficiency of optimization. These findings form the basis of the aims and objectives of the study stated in Section 1.2. The next chapter will look into the methodology of the research.

3 METHODOLOGY

The penalty free multi-objective evolutionary algorithm (PFMOEA) developed by Siew and Tanyimboh (2012a) was adopted for this research. This methodology provides a detailed explanation of the materials and procedures employed to implement the study's objectives. These include the utilization of real coding in PFMOEA, and the alteration of the retention percentage in implementing elitism. Additionally, the research includes details about the two benchmark problems employed in the study.

3.1 Incorporating Real Coding in PFMOEA

In pursuit of this research objective, real-coded NSGA II source code was linked to EPANET model in MATLAB environment. NSGA II (Deb *et al.*, 2002) was utilized as a computational tool because it exhibits robustness and adaptability by handling complex, multi-objective, and constraint-based optimisation problems. On the other hand, MATLAB was chosen due to its user-friendly interface, making it accessible to researchers and engineers with limited programming experience (Khalifeh *et al.,* 2020). EPANET 2.0 was used for network modelling and hydraulic simulation. The code was modified to reflect the PFMOEA approach, and GA operators such as tournament selection, simulated binary crossover (SBX) and polynomial mutation (PM) were employed. **Figure 3.1** depicts how the computational tools were coupled to enable a seamless modelling and optimization process.

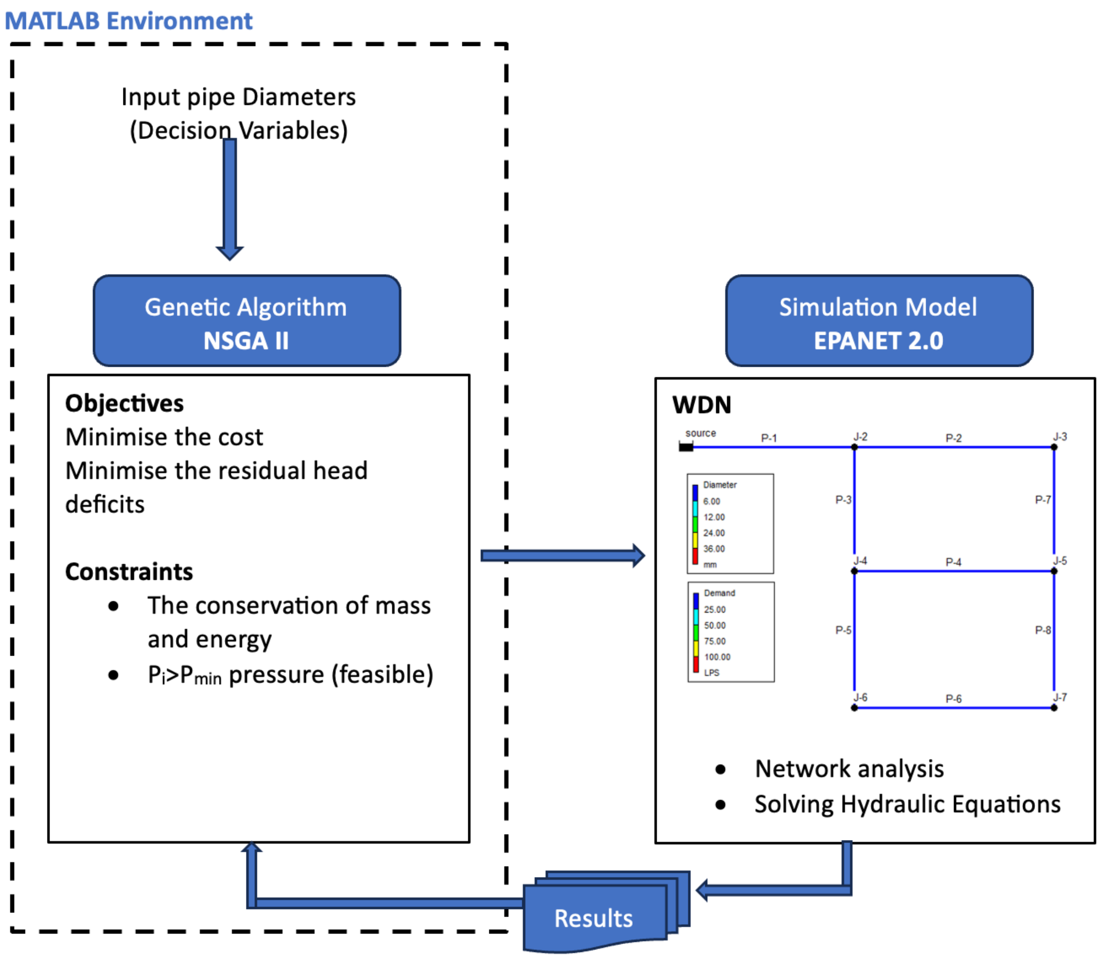

Figure 3. 1 Coupling of computational tools in the study

3.1.1 Design optimization problem formulation

Objective function

The design optimization model aimed to achieve two main objectives: minimizing the cost of the initial construction and minimizing the pressure deficit at the critical demand node i.e., characterized by a significant shortfall in the required head. The cost minimization objective was expressed as function, f_1 (Equation 3.1) and was straightforward to implement as it simply involved solving the corresponding equation. It is worth observing that the optimization design focused solely on reducing the initial construction cost. Although operational, rehabilitation, and expansion costs contribute to the total cost of the WDS (Shirajuddin, 2023), this study did not consider their inclusion within its scope.

The head deficit objective was signified by function, f_2 (Equation 3.2) and involved converting head shortfall values to positive values, while surplus heads were set to zero. Feasible solutions were identified as those with zero values of f_2, indicating that the nodal head was fully satisfied, meeting or exceeding the desired head value. Infeasible solutions, on the other hand, were associated with positive values of f_2, indicating that the nodal head did not meet the desired requirements.

Minimise the cost $$f_1 = \sum_{x=1}^{X} c(D_x)\, L_x \quad (3.1)$$

Minimise the residual head deficits $$f_2 = max(H_n^{des} - H_n, 0) \quad (3.2)$$

where $c(D_x)$ is the cost per unit length of pipe x with diameter D_x, L_x is the length of pipe *x*, and X is the number of pipes in a network. In Equation 3.2, the nodal pressure deficit at node n was obtained by subtracting the available head (H_n) from the desired head (H_n^{des}) at node n.

Constraints

The optimisation objectives were subject to the following hydraulic constraints.

1) The conservation of mass and energy as indicated by Equations 2.1 and 2.2, respectively, were satisfied by integrating EPANET 2 simulator into

NSGA II. The present research also utilized the Hazen Williams formula for calculating friction head loss in pipes as illustrated in Equation 2.4.

2) The optimization problems involved the selection and sizing of discrete pipe diameters from a predefined set of commercially available sizes. Thus the aim was to determine the most suitable diameters for the pipes (Tanyimboh and Czajkowska, 2018).

3) The minimum node pressure constraints were addressed in the second optimisation objective that deals with head deficit conditions. Demands were aggregated and allocated at the nodes even though they occur along the pipes in real world. Equation 2.7 (chapter 2, section 2.1.1) was included as one of the governing equations for assessing the hydraulic performance of WDS.

3.1.2 Elitism criteria in PFMOEA

The real-coded PFMOEA approach employed a solution allocation and retention procedure that involved preserving 30% of the population, comprising of the best feasible solutions in each generation. These retained solutions were selected by being given a very high crowding distance value. Then the remaining solutions were evaluated using the crowding distance operator to fill the remaining population slots. This process aimed to preserve feasible solutions that were close to the boundary region, while also maintaining a certain level of diversity among the population members. The PFMOEA procedure utilized in this research is illustrated in **Figure 3.2**.

3.1.3 Application of the real-coded PFMOEA approach

In this research, the applicability of real-coded PFMOEA was assessed through two benchmark optimization problems namely, the Two-loop network (Alperovits and Shamir, 1977), and Hanoi network (Fujiwara and Khang, 1990). While these networks do not completely represent the real-world situations of WDSs, they have been extensively examined in prior research including the previous binary-coded PFMOEA research. In this study, the effectiveness of the real-coded PFMOEA approach was assessed by considering both the optimal cost and the number of function evaluations. Additionally, a comparative analysis was conducted against other approaches with the smallest number of function evaluations in the literature. Upon conducting the evaluation and comparing the results, the approach that demonstrated the least number of function evaluations was identified as the best-performing approach.

The optimisation experiments were conducted on a desktop computer with an Intel ® Xeon ® W-2223 at 3.60 GHz and an installed storage of 16.0GB (15.7 GB usable). An Uninterruptible Power Supply (UPS) was utilized as a backup for the computer. The parameters for crossover distribution index and mutation distribution index were set to η_c=20 and η_m=20, respectively, following the methodology outlined by Deb *et al.* (2002). For both networks, 20 runs were permitted. Each run began with a randomly generated population with a total of 200,000 function evaluations which effectively corresponds to 1,000 generations for a population size of 200. The objective was to find pipe diameters that minimised total cost while satisfying demand and system constraints at every node.

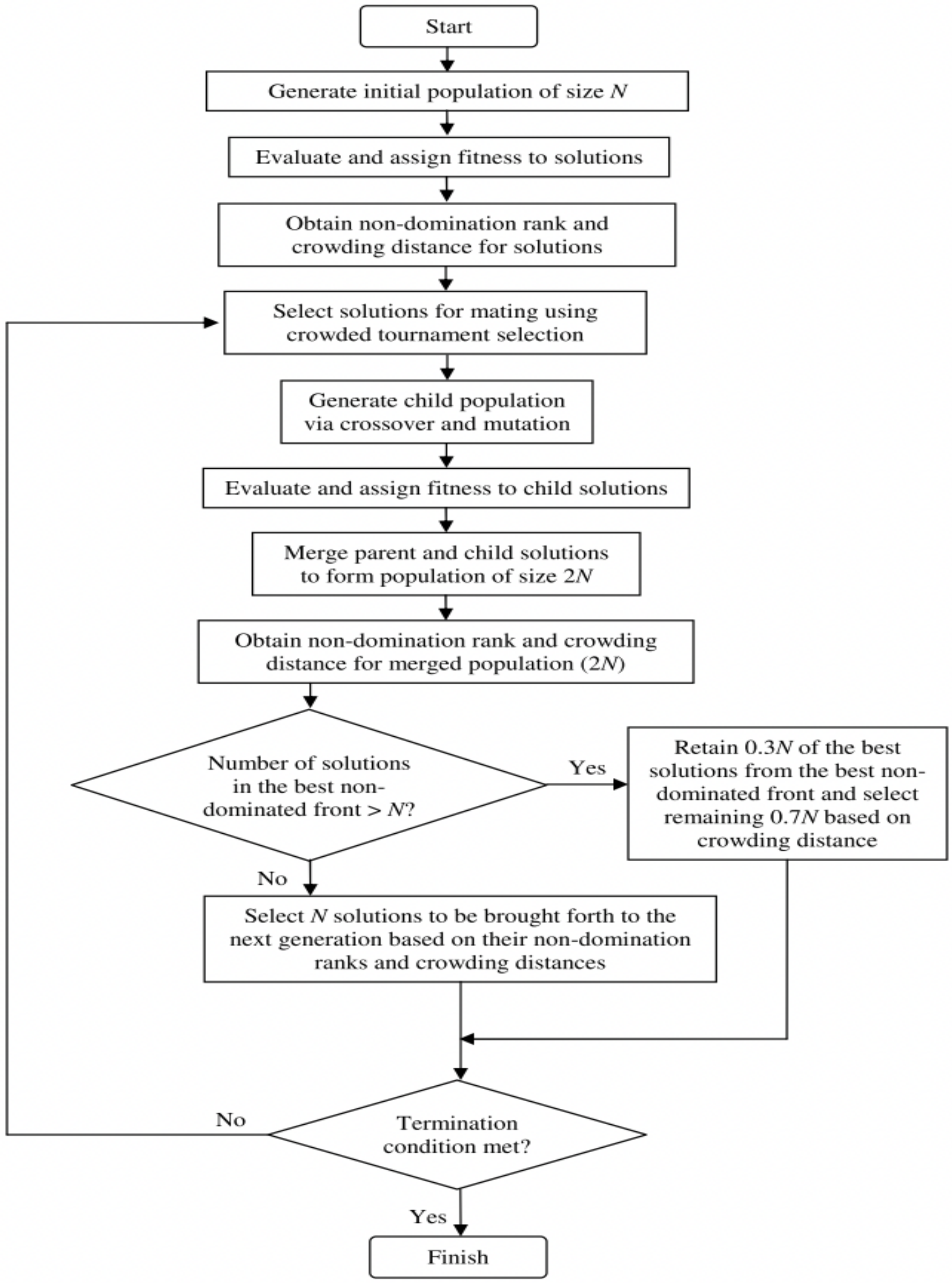

Figure 3. 2 Flowchart of PFMOEA (Siew and Tanyimboh, 2012a)

3.1.4 Altering the elitism retention of PFMOEA

The performance of PFMOEA approach was investigated by altering the allocation percentages of the least infeasible and feasible solutions at the boundary regions shown in **Figure 3.3**. Three different allocation percentages were employed for this study, and they were labelled as PFMOEA-A, PFMOEA-B, and PFMOEA-C. These allocations comprised of 15% feasible and 15% infeasible solutions, 20% feasible and 10% infeasible solutions, and 30% feasible and 0% infeasible solutions, respectively. Each solution within the population underwent individual evaluation based on the optimization objectives (Equation 3.1 and Equation 3.2). Solutions violating the pressure constraint were identified as infeasible, while feasible solutions were those exhibiting zero-deficit pressure.

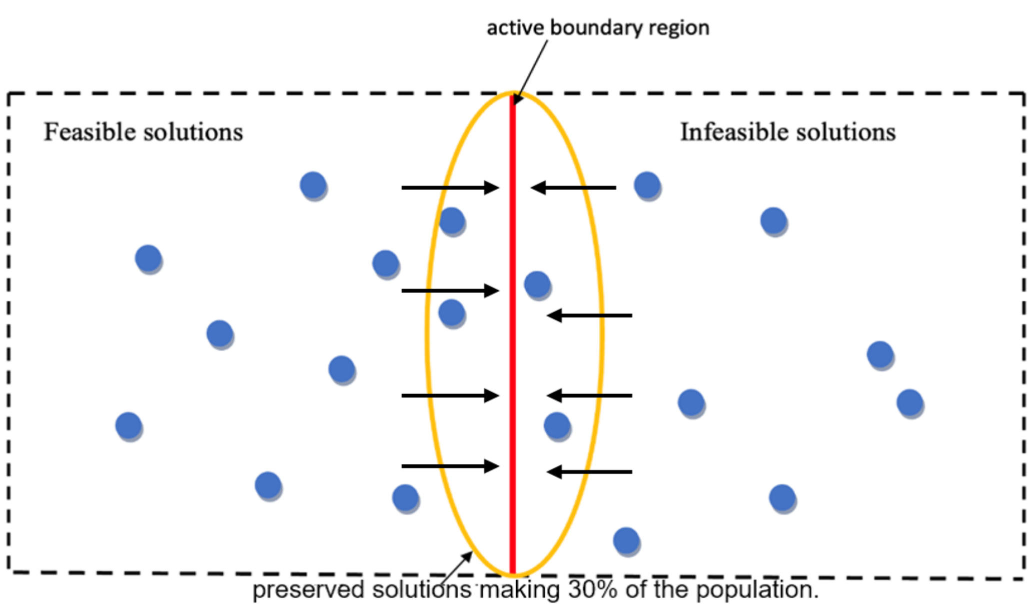

Figure 3. 3 Schematic diagram of the altered elitism criteria in PFMOEA

The elitism retention for PFMOEA was implemented as follows:

1. A random initial population (P_t) was created with a size of N.

2. The initial population was sorted and ranked based on the two objectives (i.e. cost and head deficit).
3. Selection, combination, and mutation operators were applied to P_t to generate an offspring population (Q_t) with a size of N.
4. After producing the first generation, encompassing parent and offspring chromosomes, the new population was generated in the following manner:

- Combinations of P_t parent chromosomes and Q_t offspring chromosomes produced the second offspring (R_t) generation with a size of 2N.
- R_t generation was sorted based on the non-dominated classification method. This involved the identification and classification of non-dominated fronts (F1, F2,…) as described in section 2.3, **Figure 2.4**.
- The parent generation (P_{t+1}) for the subsequent iteration, comprised of size N elements and was produced for **PFMOEA-C** using the approach detailed in section 2.3, **Figure 2.5**.
- Operations including selection, combination, and mutation were applied to the new parent generation (P_{t+1}), giving rise to the offspring generation (Q_{t+1}) comprising N elements as in **Figure 2.4**.

5. The process continued iteratively from step 4 until reaching the termination condition of **PFMOEA-C**.

PFMOEA-A implemented steps 1 through 3 as described above while step 4 involving the generation of parent population (P_{t+1}) with size N, was altered. This alteration retained 15% of the least costly feasible solutions. These solutions were obtained by ranking the feasible solutions on the basis of the cost and selecting the 15% with the least cost. Simultaneously, 15% of the best infeasible solutions were selected on the basis of head deficit. Ranking these infeasible solutions involved assessing their head deficits, and the 15% with the lowest deficits were preserved in each generation. It was presumed that selected solutions were dominated by those near the boundary of feasibility although no tests were undertaken to inform if this was the case. The remaining 70% of N solutions underwent a selection process using the crowding distance operator, occupying the remaining population slots.

Subsequently, step 5 was iteratively executed until the termination condition designated for **PFMOEA-A** was satisfied. This iterative approach ensured the continual refinement and enhancement of the solution set until the convergence criteria were satisfied. Subsequently, this process was replicated for **PFMOEA-B** with an altered retention percentage in step 4.

3.2 Description of Case Study Networks

3.2.1 Two-looped network

Alperovits and Shamir (1977) proposed the two-looped network which consists of a single reservoir and eight pipes that are 1000 m long. A Hazen-Williams coefficient of 130 was assumed for new pipes. The minimum pressure required to satisfy the nodal demands in full was 30 m. This design optimization problem was based on a set of 14 commercially available pipe diameters, leading to a search of 14^8 which is approximately 1.4 X 10^9 solutions. The Two-looped network's layout is depicted in **Figure 3.4**, whereas other nodal and pipe data can be found in Appendix A (section A-1 to A-3).

3.2.2 Hanoi network

The Hanoi network (**Figure 3.5**) is composed of 34 pipes and 32 nodes, with a reservoir located at an elevation of 100m (Fujiwara and Khang, 1990). The nodal demands required a minimum head of 30m to ensure sufficient flow. All new pipes were assumed to have a Hazen-Williams friction coefficient of 130. The nodes were placed at ground level, at an elevation of 0.0m. The diameters of the pipes utilized in designing the system are listed in Appendix B (section B-1 to B-3).

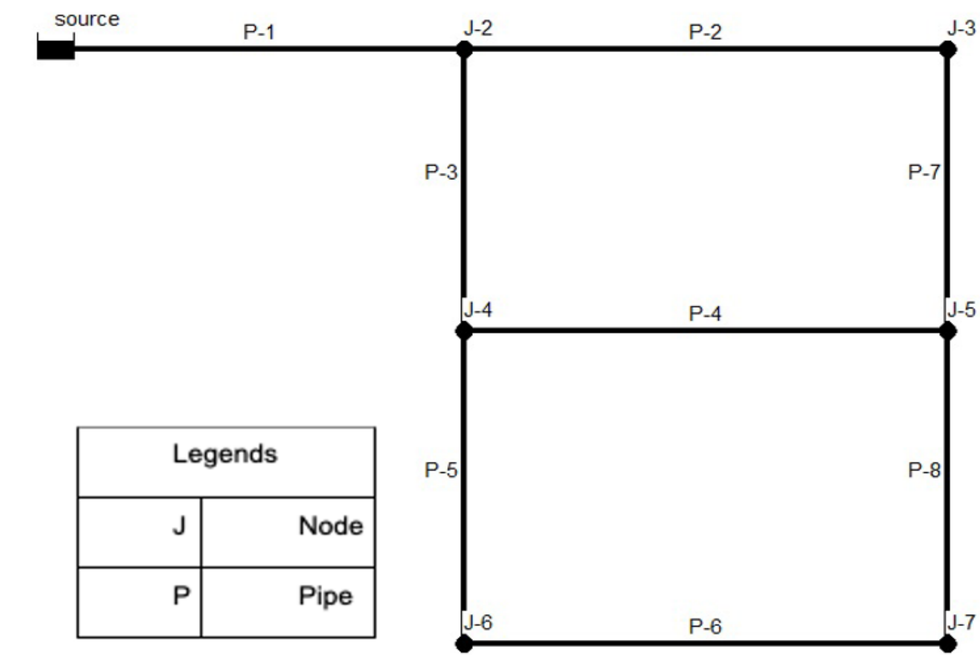

Figure 3. 4 Layout of Two-Loop network (Alperovits and Shamir, 1977)

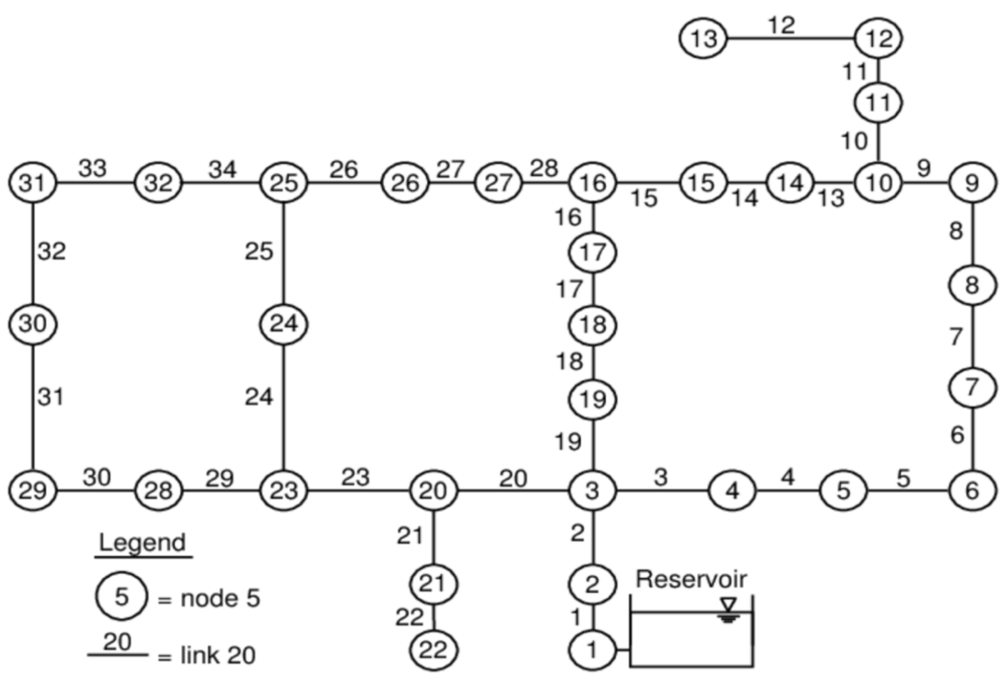

Figure 3. 5 Layout of Hanoi network (Fujiwara and Khang, 1990)

4 RESULTS AND DISCUSSION

This section contains results obtained from implementing real coding representation and modifying the solution allocation in PFMOEA. Two benchmark optimization problems were utilized, namely, the Two-looped network and Hanoi network. The results of this implementation are discussed in detail, providing valuable insights into the effectiveness of real coding and solution allocation alterations.

4.1 Effectiveness of Real Coding in PFMOEA

4.1.1 Two-looped network optimisation

The real-coded PFMOEA approach implemented on the Two-looped network in this study identified an optimum solution of $419,000 and a pressure deficit of zero within 2,000 function evaluations. This represents a fraction of 1.35×10^{-4} % of the total solution space (i.e., $14^8 \approx 1.48 \times 10^9$). Additionally, the average CPU time required to perform a single optimization run comprising 200,000 function evaluations was approximately 48 hrs. The pressure heads and deficits for the optimal solution are depicted in **Figure 4.1** with a consistent 30m minimum head requirement across all network nodes. As depicted in the figure, the actual pressures obtained at the nodes surpass the minimum pressure heads specified for each node. Furthermore, there is no shortfall in pressure as evident by a zero-pressure deficit at all nodes. It is assumed that these analyses uses the 30%-70% split as shown in Figure 3.2 and then only you will change the selection of the population.

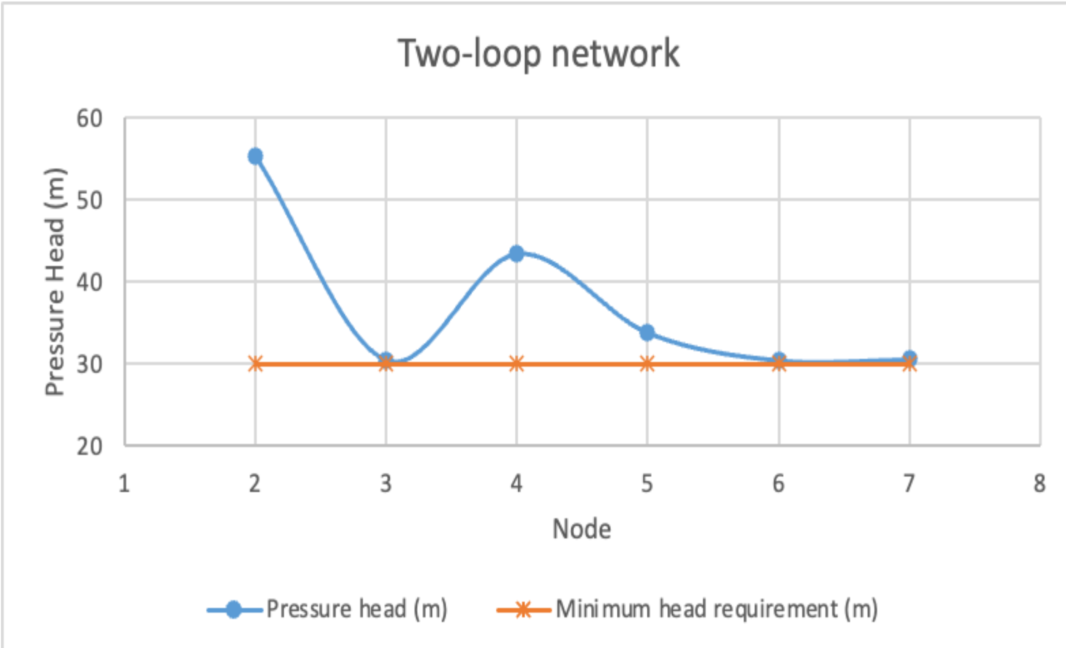

Figure 4. 1 The Two-loop network nodal pressure heads

Figure 4.2 illustrates the Pareto optimal front obtained for the Two-looped network with a parameter value of ω=10.667, based on 20 random runs. The pareto front of cost against head deficit showcases a range of solutions that achieve different levels of cost and head deficit, highlighting the trade-off between these two objectives. Moreover, in **Figure 4.2**, it is evident that several hydraulically feasible designs exist in close proximity to the least expensive feasible solution, valued at $419,000. This indicates that the non-domination sorting process guarantees the survival of only the most economical feasible design at the boundary of feasibility (Siew *et al.,* 2014)

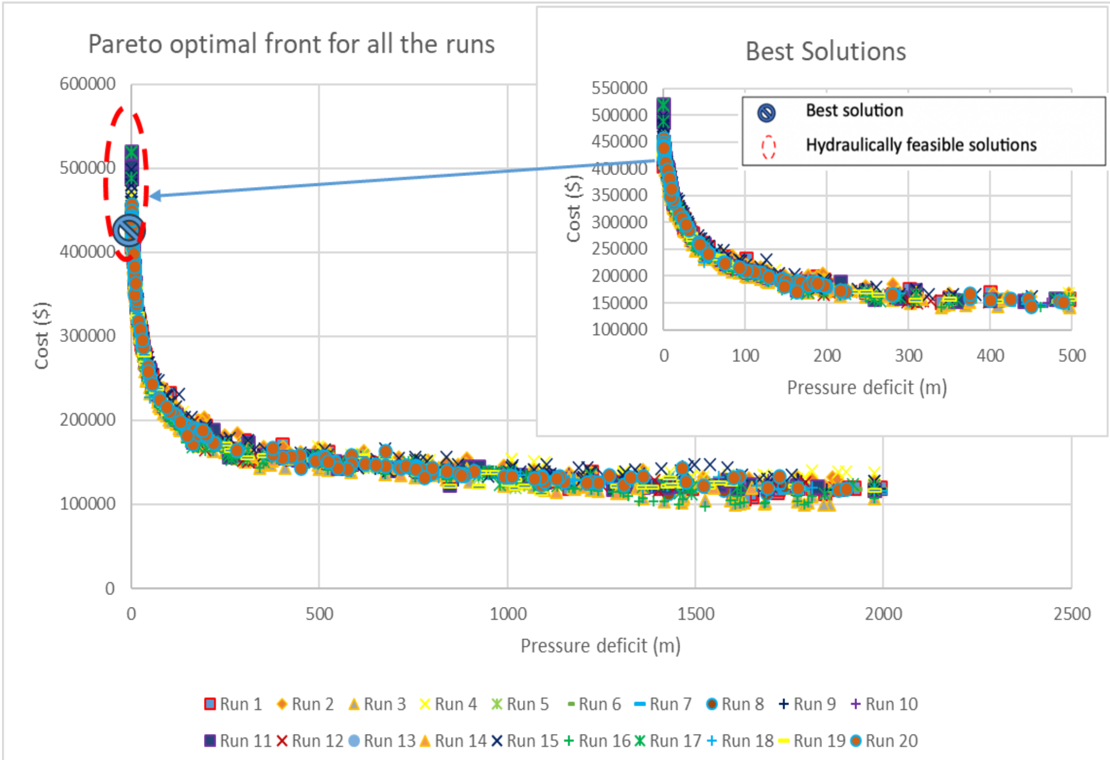

Figure 4. 2 Pareto fronts for the Two-loop Network

Previously binary-coded PFMOEA approach by Siew and Tanyimboh (2012a) identified the optimal solution of $419,000 and $420,000 within 2,200 and 2,600 function evaluations, respectively. The optimal solution was obtained in a single run out of 10 conducted runs. Similarly, the real-coded PFMOEA in the present study achieved the same optimal solution ($419,000) within lesser (2,000) functional evaluations. The optimal value was achieved three times out of 20 runs which were conducted (see **Figure 4.3**).

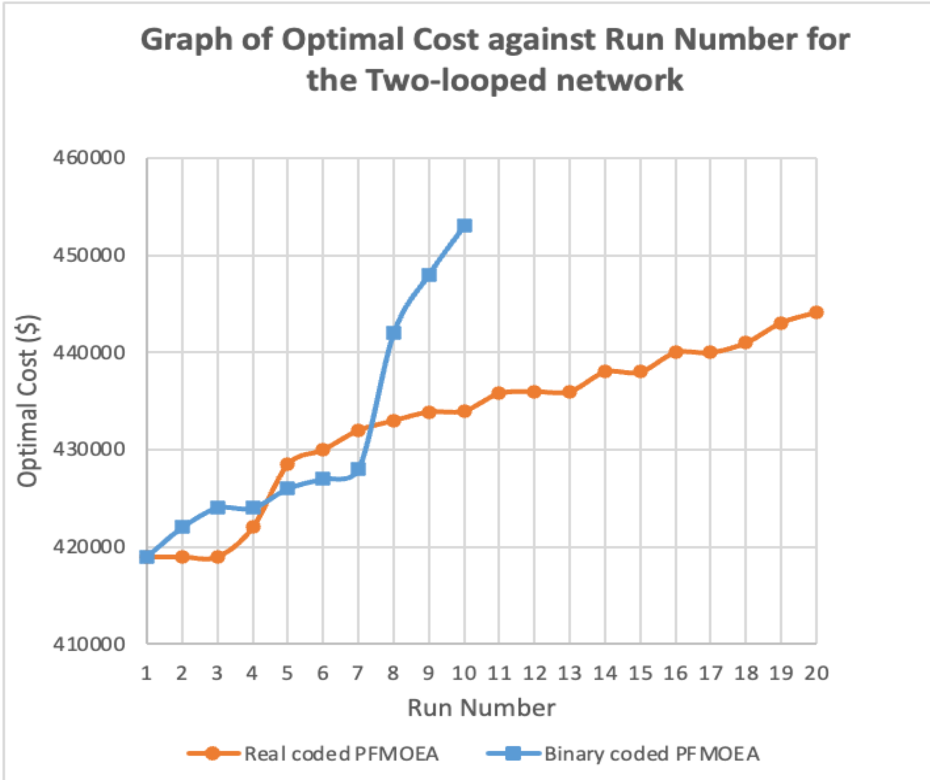

Figure 4. 3 Comparison of the optimal cost against the run number for the binary-coded and real-coded PFMOEA using the Two-looped network

Figure 4.4 shows a graph of function evaluation against the run number displaying fluctuations in both the binary-coded and real-coded PFMOEA. It was observed that the real-coded PFMOEA fluctuated on a lower side than the binary-coded PFMOEA curve. In addition, the average number of function evaluations of 3,797 which was obtained for the real-coded PFMOEA was smaller than the average of 5,290 obtained by the binary-coded PFMOEA. This represents a 28.21% reduction in the number of function evaluations and indicates that the real-coded PFMOEA outperformed the binary-coded PFMOEA in computational efficiency. Therefore, the real-coded PFMOEA may be a more effective and efficient optimization approach than the binary-coded PFMOEA.

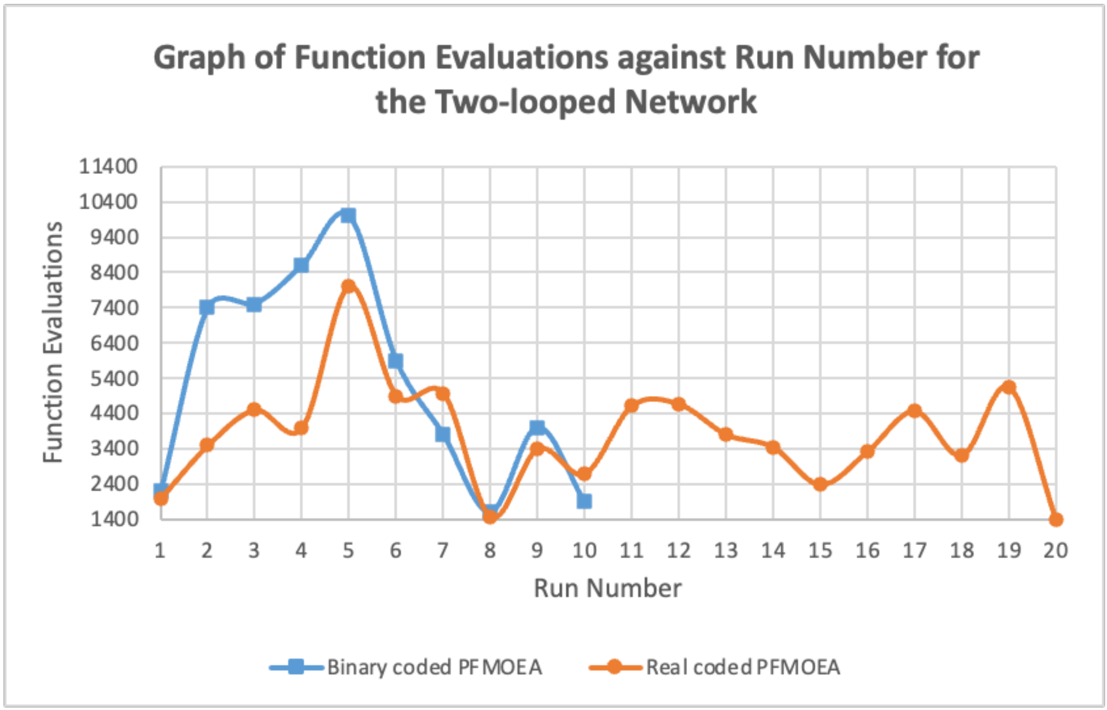

Figure 4. 4 Comparison of the function evaluation against the run number for the binary-coded and real-coded PFMOEA using the Two-looped network

Table 4.1 presents a comparison of optimal costs and the number of function evaluations for the Two-looped network from various studies. To facilitate easy comparison, the pipe diameters in the table have been provided in SI units. Particularly, the real-coded PFMOEA surpassed other studies by achieving an optimal cost of $419,000 with just 2,000 function evaluations. This is in contrast to the respective function evaluations of 7467 (Wu *et al.,* 2001), 25,000 (Cunha and Sousa, 1999), and 9500 (Sirisant and Janga, 2018). However, these results were not compared based on CPU time due to its heavy dependence on the specific computer configuration used (Palod *et al*., 2021).

Table 4. 1 Solutions for the Two-looped network

Pipe number	Diameter of pipes (mm)							
	ω =10.508				ω =10.9031		ω =10.667	
	Savic and Walters (1997)	Cunha and Sousa (1999)	Wu *et al*, (2001)	Siew and Tanyimboh (2012a)	Savic and Walters (1997)	Siew and Tanyimboh (2012a)	Sirisant and Janga (2018)	Present work
1	457.2	457.2	457.2	457.2	508	508	457.2	457.2
2	254	254	254	254	254	254	254	254
3	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4
4	101.6	101.6	101.6	101.6	25.4	25.4	101.6	101.6
5	406.4	406.4	406.4	406.4	355.6	355.6	406.4	406.4
6	254	254	254	254	254	254	254	254
7	254	254	254	254	254	254	254	254
8	25.4	25.4	25.4	25.4	25.4	25.4	25.4	25.4
Optimization method	GA	SA	GA	Binary-coded PFMOEA	GA	Binary-coded PFMOEA	SADE	Real-coded PFMOEA
Total cost ($)	419,000	419,000	419,000	419,000	420,000	420,000	419,000	419,000
Maximum Function Evaluation	250,000	25,000	7,467	2,200	250,000	2,600	9,500	2,000

4.1.2 Hanoi network optimisation

The real-coded PFMOEA identified the least cost feasible solution of \$6.081 million, and a pressure deficit of zero using ω=10.667. This was achieved four times out of 20 runs in which the best run took 48,000 function evaluations. The solution, therefore, corresponds to 1.67×10^{-20} % of the entire search space ($6^{34} \approx 2.865 \times 10^{26}$). In all those runs, the minimum required pressure of 30 m was met for all nodes as demonstrated in **Figure 4.5** for the least cost feasible solution. Moreover, it took approximately 24 hours on average for a single optimization run comprising of 200,000 function evaluations to be completed.

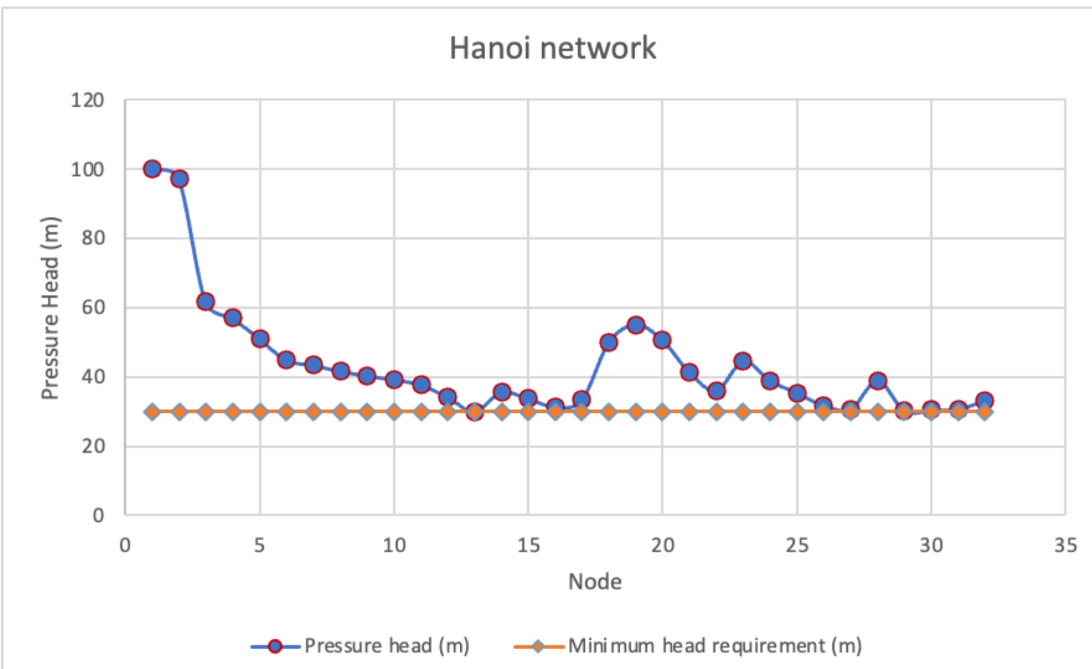

Figure 4. 5 Pressure heads in the optimized Hanoi network

Figure 4.6 depicts the pareto-optimal fronts from the 20 runs of the real-coded PFMOEA approach. It is evident that the PFMOEA consistently performs well, as all the fronts presented in the figure exhibit a remarkable level of similarity.

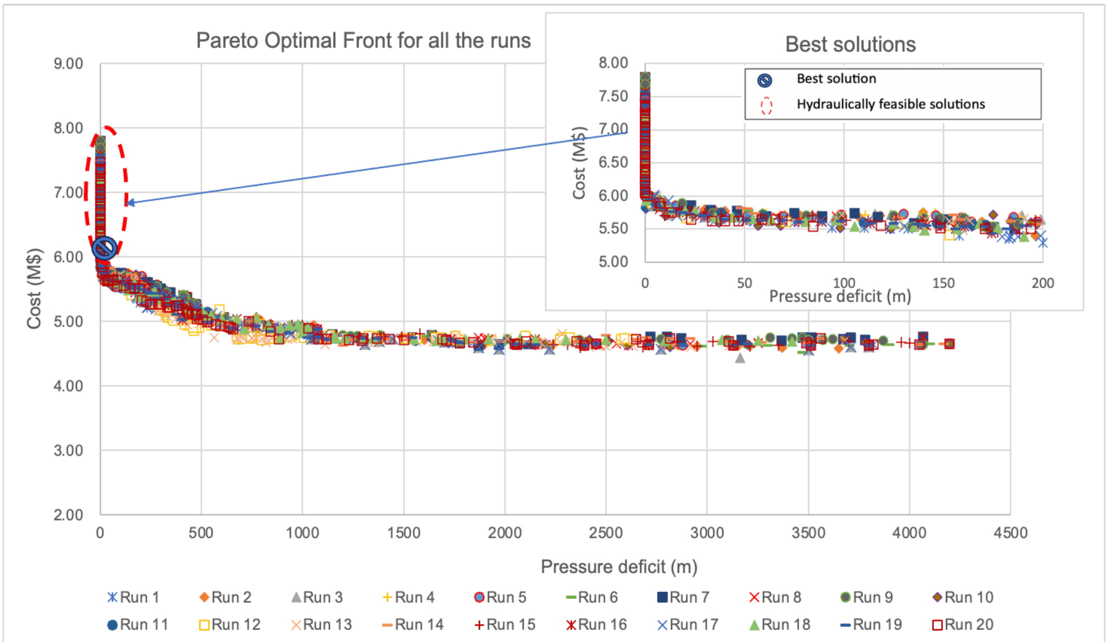

Figure 4. 6 Pareto-Optimal Fronts of the 20 Real-coded PFMOEA Runs for the Hanoi Network

The Hanoi optimization problem which was solved using the real-coded PFMOEA (in the present study) slightly outperformed the binary-coded PFMOEA (Siew and Tanyimboh, 2012a). This was based on the number of function evaluations needed to reach the optimum solution. In contrast to Siew and Tanyimboh (2012a) who attained the least cost design of $6.056 million in 51,000 function evaluations, the present research achieved the optimal solution of $6.081 million within 48,000 function evaluations. This was attained four times out of 20 runs while the binary-coded PFMOEA obtained the optimal solution eleven times out of 60 runs (see **Figure 4.7**). Additionally, the average costs for the 20 runs of real-coded PFMOEA were found to be $6.17 million. Conversely, the 60 runs of binary-coded PFMOEA had an average cost of $6.16 million (Siew and Tanyimboh, 2012a). The two approaches are therefore reasonably close in effectiveness based on the ability to achieve the global optimum.

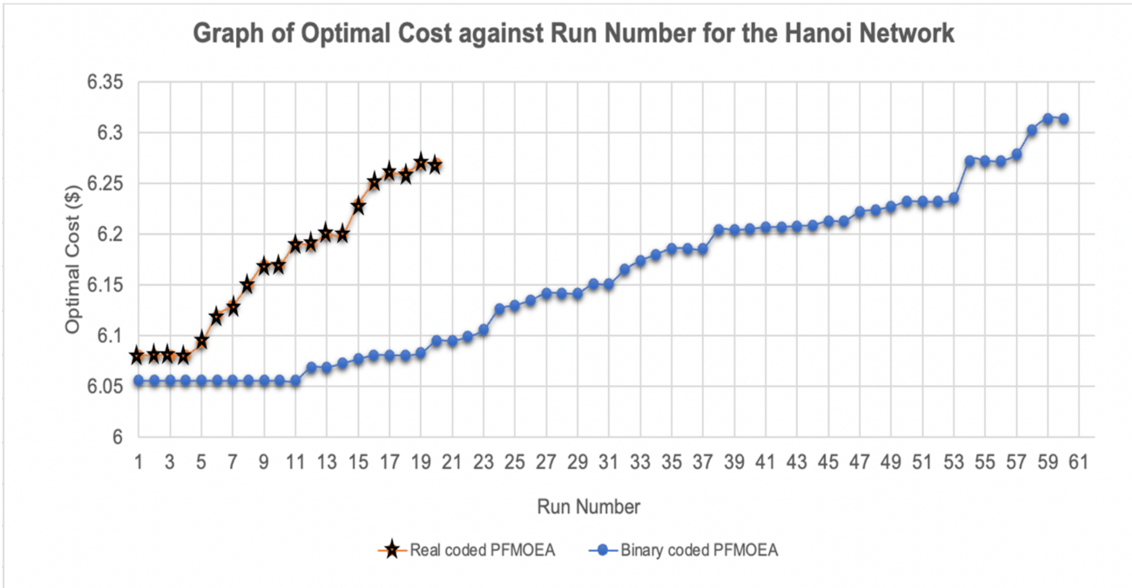

Figure 4. 7 Comparison of the optimal cost against the run number for the binary-coded and real-coded PFMOEA using Hanoi Network

Figure 4.8 shows that the real-coded PFMOEA applied to the Hanoi network, converged in fewer function evaluations in many more runs than the binary-coded PFMOEA. On average, the real-coded PFMOEA obtained 82,281 function evaluations. This represents a 34.38% decrease from the average of 125,398 achieved for the binary-coded PFMOEA. These results are comparable to those of the Two-looped network optimisation problem presented in Section 4.1.1. This indicates that the real-coded PFMOEA has superior computational efficiency over the binary-coded PFMOEA.

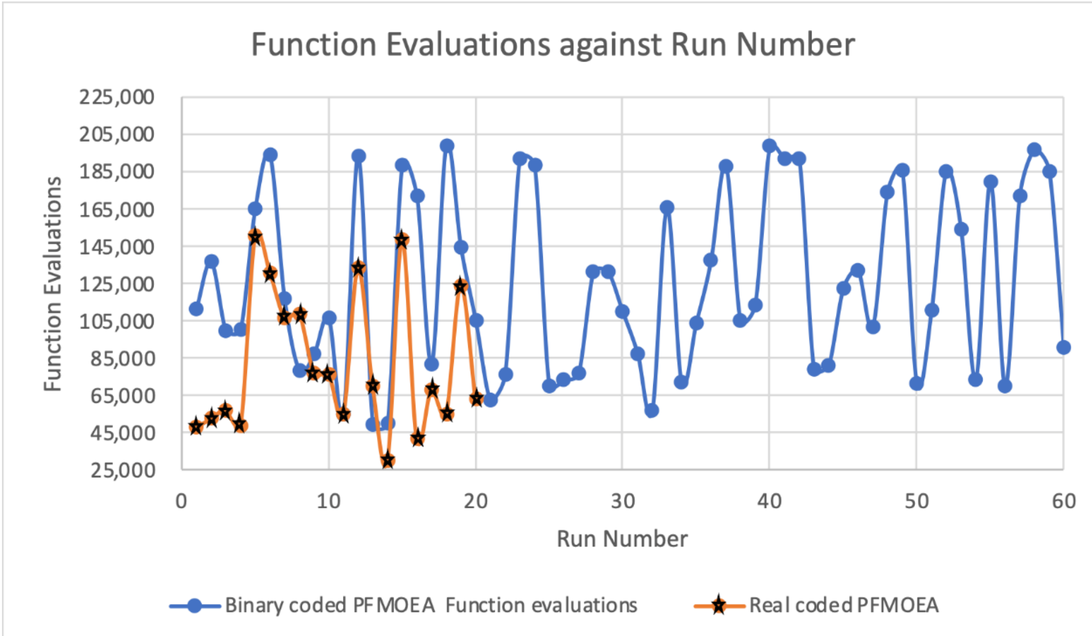

Figure 4. 8 Comparison of the function evaluation against the run number for the binary-coded and real-coded PFMOEA

Table 4.2 presents the optimal diameters, optimal costs, and their corresponding maximum function evaluations for various approaches including the real-coded PFMOEA. This study's optimal cost of $6.081 million at ω=10.667 is consistent with the findings of Suribabu (2010). However, it differs from the results of other researchers (Cunha and Sousa, 1999; Vairavamoorthy and Ali, 2000; Wu and Walski, 2005; Geem, 2006; Kadu *et al*., 2008; Siew and Tanyimboh, 2012a) who obtained their optimum cost at a higher function evaluation as indicated in **Table 4.2.**

The variations in optimal costs in **Table 4.2** can be attributed to the different ω values employed in the head-loss equation (refer to **Equation 2.4**) and the reduction in the search space. Kadu *et al.* (2008) utilized a search space reduction technique, where only a subset of candidate pipe diameters from the

6 commercial pipe sizes were considered. As a result, the GA search was performed on a reduced search space of 2.351×10^{19} which accounts for approximately 8.2×10^{-6} % of the entire solution space (i.e., $6^{34} = 2.865 \times 10^{26}$). In contrast to Kadu *et al.* (2008), this current research explored the entire solution space.

Table 4. 2 Historic and current solution for the Hanoi network

	Diameter of pipes (mm)									
ω	10.508	10.508	10.508	10.509	10.508	10.508	10.903	10.9031	10.667	10.667
Pipe number	Cunha and Sousa (1999)	Vairavamoorthy and Ali (2000)	Wu and Walski (2005)	Geem (2006)	Kadu *et al.*, (2008)	Siew and Tanyimboh (2012a)	Kadu *et al.*, (2008)	Siew and Tanyimboh (2012a)	Suribabu (2010)	Present study
1–8	1016	1016	1016	1016	1016	1016	1016	1016	1016	1016
9	1016	1016	1016	1016	1016	1016	762	1016	1016	1016
10	762	762	762	762	762	762	762	762	762	762
11	609.6	609.6	609.6	609.6	609.6	609.6	762	609.6	609.6	609.6
12	609.6	609.6	609.6	609.6	609.6	609.6	609.6	609.6	609.7	609.6
13	508	508	508	508	508	508	304.8	304.8	508	508
14	406.4	406.4	406.4	406.4	406.4	406.4	304.8	304.8	406.4	406.4
15	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8
16	304.8	304.8	304.8	304.8	304.8	304.8	406.4	304.8	304.8	304.8
17	406.4	406.4	406.4	406.4	406.4	406.4	508	508	406.4	406.4
18	508	508	508	508	508	508	609.6	609.6	609.6	508
19	508	508	508	508	508	508	609.6	609.6	508	508
20	1016	1016	1016	1016	1016	1016	1016	1016	1016	1016
21	508	508	508	508	508	508	508	508	508	508
22	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8

23	1016	1016	1016	1016	1016	1016	1016	1016	1016	1016
24–25	762	762	762	762	762	762	762	762	762	762
26	508	508	508	508	508	508	508	609.6	762	508
27–28	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8
29	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4
30	304.8	304.8	304.8	304.8	304.8	304.8	304.8	406.4	304.8	304.8
31	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8	304.8
32	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4	406.4
33	406.4	406.4	406.4	406.4	406.4	406.4	508	406.4	406.4	406.4
34	609.6	609.6	609.6	609.6	609.6	609.6	609.6	609.6	609.6	609.6
Optimization method	SA	GA	GA	HS	GA	Binary-coded PFMOEA	GA	Binary-coded PFMOEA	Differential evolution	Real-coded PFMOEA
Total cost ($M)	6.056	6.056	6.056	6.056	6.056	6.056	6.19	6.182	6.081	6.081
Maximum Function Evaluation	53000	160000	150000	200000	18000	51000	18000	100000	48724	48000

4.2 Effect of Altering the Allocation of Solutions Near the Active Constraint Boundary Region in Real-coded PFMOEA.

The progress in achieving the optimal solution across function evaluations for different retention percentages in real coded PFMOEA are presented in **Figure 4.9**.

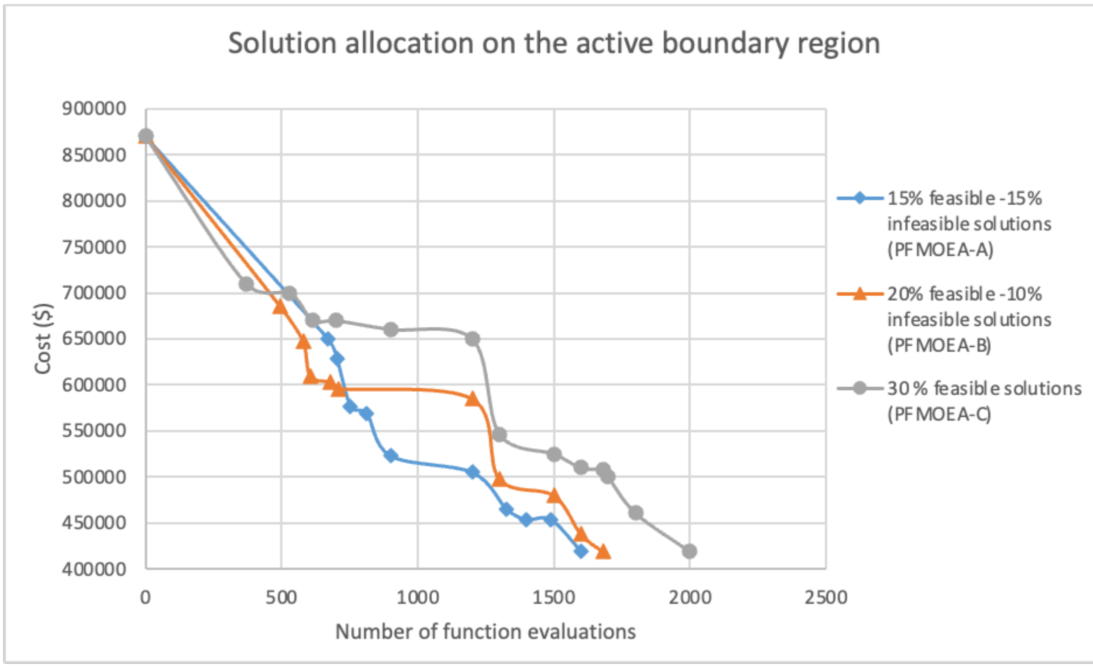

Figure 4. 9 Progress of the Real-coded PFMOEA optimization with varied percentages of retention in the boundary region for the Two-looped network

In the early stages of the evolutionary process, PFMOEA-A, PFMOEA-B, and PFMOEA-C exhibited rapid improvements in minimizing the network cost (**Figure 4.9**). Remarkably, the progress made by PFMOEA-A with a 15% allocation on both sides of the boundary region surpassed the progress of the other percentage allocations. Additional runs not reported here also revealed the same. During optimization process, the progress lines for PFMOEA-A, PFMOEA-B, and PFMOEA-C continued to advance, resulting in the attainment of the optimal solution at 1,600, 1,680, and 2,000 function evaluations, respectively. The results of the computational effort required by PFMOEA-A was 20% lower than that of PFMOEA-C. Notably, none of the allocations encountered a plateau before reaching the optimal solution ($419,000) which indicates efficient progress.

Furthermore, the findings indicate that the 15% feasible-15% infeasible percentage allocation of PFMOEA-A is superior to the other two. This allocation imposes greater pressure on the search to stay in proximity to the boundary, where both good feasible solutions and the least costly infeasible solutions are situated. By preserving infeasible solutions near the boundary region, it becomes easier to transform them into feasible and optimal solutions (Ray *et al*., 2009). This strategy improves diversity among population members while satisfying constraints (Ray *et al*., 2009).

The results presented in **Figure 4.10** are for the three percentage retentions in elitism for the Hanoi WDS. The figure demonstrates a similarity to the findings for the Two-looped network. PFMOEA-A, PFMOEA-B and PFMOEA-C attained the optimal solution ($M 6.081) with 39,700, 41,500 and 48,000 generations. This confirms that the 15% feasible-15% infeasible percentage allocation employed by PFMOEA-A outperformed PFMOEA-B (20% feasible-10% infeasible solutions) and PFMOEA-C (30% feasible solutions).

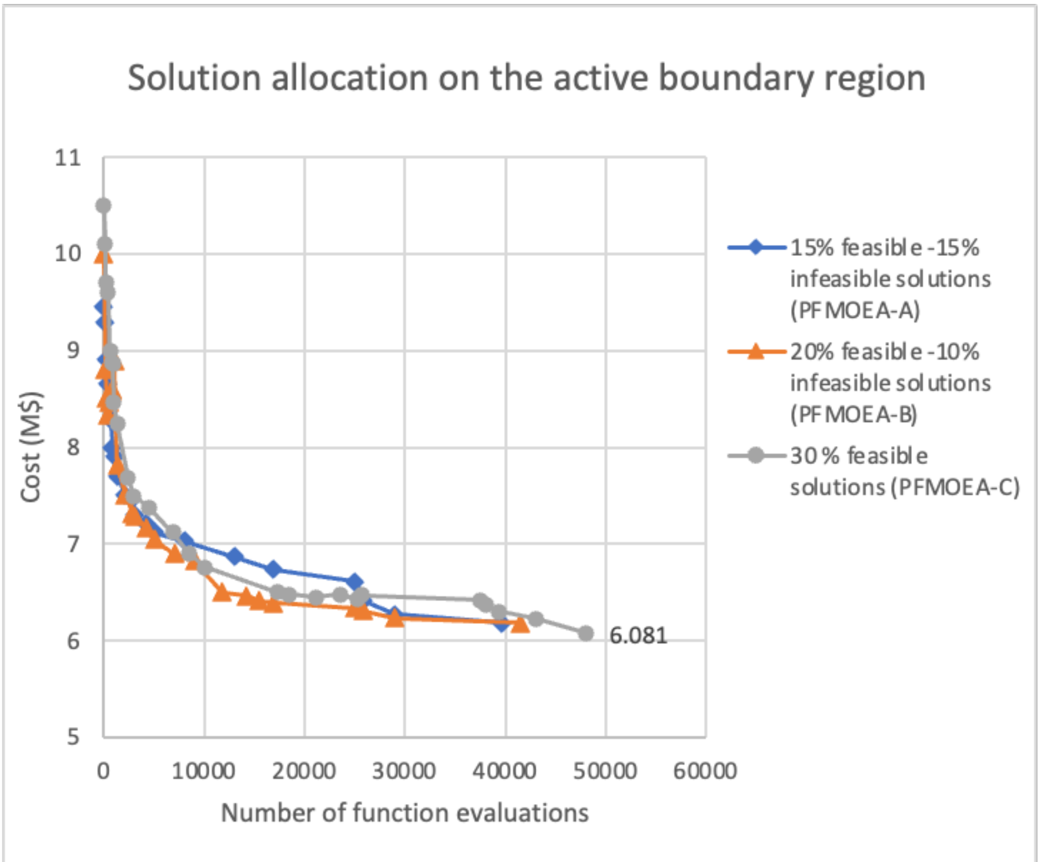

Figure 4. 10 Progress of the real-coded PFMOEA optimization with varied percentage allocations on the constraint boundary region for the Hanoi Network (ω=10.667).

5 CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

This study aimed to find out if the PFMOEA for WDS could be improved by incorporating real coding representation and alterating the elitism concept in the approach. Previous PFMOEA research utilized binary representation to encode and decode solutions. Although binary coding is simple to use, it tends to generate redundant codes that do not match any members of the finite discrete set to which the encoded parameter belongs. Hence the need to find out if using real coding could improve PFMOEA performance. The study applied two benchmark networks; Two-looped and Hanoi networks that have been applied in previous studies using binary-coded PFMOEA and other optimization methods.

The real-coded PFMOEA succeed in identifying the least cost feasible solution of $419,000 and $6.081 million for the Two-looped and Hanoi networks, respectively. Hence, the reduction in the average number of function evaluations required to attain the optimal solution. Specifically, for the Two-looped network, there was a decrease of 28.21% while for the Hanoi network, the reduction amounted to 34.38%. As a result, the real-coded PFMOEA outperformed the binary-coded PFOMEA in terms of computational efficiency as it converged to the optimal solutions faster. This might be attributed to a more effective exploration of the solution space, as real coding does not generate redundant genes unlike binary coding.

In previous PFMOEA studies, 30% of the least-cost feasible solutions were retained in each generation with no inclusion of any infeasible solutions. Therefore the second objective of this research, was to find out if replacing various percentages of both feasible and infeasible solution close to the active constraint boundary could improve the PFMOEA performance. The solution allocation of 15% feasible and 15% infeasible solutions achieved the best results. As it just required 1600 function evaluations to achieve the optimal solution using the Two-looped network. In comparison, allocating 20% feasible and 10% infeasible solutions required 1680 function evaluations, while allocating 30% feasible and 0% infeasible solutions required 2000 function evaluations. This implies that the first solution allocation (15% feasible and 15% infeasible) outperformed the other two in terms of efficiency.

In the case of the Hanoi optimisation problem, the solution allocation comprising 15% feasible and 15% infeasible solutions also required the fewest number of function evaluations of 39,700 to obtain the best solution. Allocating 20% feasible and 10% infeasible solutions involved 41,500 function evaluations, whereas allocating 30% feasible and 0% infeasible solutions needed 48,000 function evaluations. Thus, including infeasible solutions close to the feasible least-cost boundary significantly enhanced the computational efficiency of the PFMOEA.

The significant limitation of this study was the time constraint in coupling EPANET 2 with NSGA II, and efforts involved in testing the coupled model and aligning the algorithm to incorporate the principles of PFMOEA. Moreover, the extensive CPU time required for optimizing the water distribution network (WDN) within the duration of the master's program prevented further exploration of additional WDNs. Most of the optimizations were conducted on a single desktop computer due to limitations of and constraints on computational facilities. This constraint restricted the timing, scalability, and parallelization of the optimization process. Thereby, also limiting the number of networks explored and the assessment of effect of retention allocation percentages on optimization performance.

5.2 Recommendation

This study achieved its aims and objectives successfully, improving the computational efficiency of the PFMOEA. Nevertheless, there are opportunities for future investigations and improvements to build upon the current findings. Therefore, there is a need to use a wider range of WDNs with diverse configurations and characteristics. This broader analysis would enhance the generalizability and applicability of the real-coded PFMOEA findings to real-world situations. Furthermore, exploring a spectrum of retention percentages in the PFMOEA would be appropiate. This is so as previous researchers employed a default of 30% as a total retention percentage. However, there was no basis to how the 30% was selected. Therefore, it is subjective, and further research would be recommended.

Additionally, this study used three different retention precentages of feasible and infeasible solutions totaling to 30%. Hence, exploring a wider range of retention percentages would be beneficial for future research. The selection of infeasible solutions for retention in this study was based on head deficit only and other approaches for this selection could be investigated in future studies. Moreover, Matlab was used in the present study as it is user-friendly. Nevertheless, it tends to utilize a lot of CPU time in the optimization process. Thus, future studies on real-coded PFMOEA should consider utilizing other computer languages like C++ which might be computationally more efficient.

6 REFERENCES

Abdy Sayyed, M.A.H., Gupta, R. and Tanyimboh, T.T. (2019) "Combined flow and pressure deficit-based penalty in GA for optimal design of water distribution network." *ISH Journal of Hydraulic Engineering*, 00(00), pp. 1–11. https://doi.org/10.1080/09715010.2019.1604180.

Afshar, M.H. and Moeini, R. (2008) "Partially and fully constrained ant algorithms for the optimal solution of large-scale reservoir operation problems." *Water Resources Management*, 22(12), pp. 1835–1857. https://doi.org/10.1007/s11269-008-9256-6.

Alperovits, E., and Shamir, U. (1977). "Design of optimal water distribution systems." *Water Resource Research*, 13, pp. 885-900. https://doi.org/10.1029/WR013i006p00885.

Barlow, E. (2009) "An investigation into the response of a bubble and an ultrasound contrast agent under chirp excitation." *University of Strathclyde*, Thesis [Preprint], pp 1-238. https://doi.org/10.48730/4ew1-w181.

Bello, O., Abu-Mahfouz, A.M., Hamam, Y., Page, P.R., Adedeji, K.B. and Piller, O. (2019) "Solving management problems in water distribution networks: A survey of approaches and mathematical models." *Water*, 11(3), pp. 562. https://doi.org/10.3390/w11030562.

Bhave, R.P. and Gupta, R. (2006) "Analysis of Water Distribution Networks." UK, Alpha Science International Ltd.

Chandapillai, J. (1991) "Realistic simulation of water distribution systems." *Journal of Transportation Engineering*, 117 (2), pp. 258-263. https://doi.org/10.1061/(ASCE)0733-947X(1991)117:2(258).

Ciro, C., Dugardin, F., Yalaoui, F. and Russel, K. (2016) "A NSGA-II and NSGA-III comparison for solving an open shop scheduling problem with resource constraints." *IFAC-PapersOnLine,* 49(12), pp. 1272–1277. https://doi.org/10.1016/j.ifacol.2016.07.690.

Coello, C. A. and Lechuga, M. S. (2002) “Mopso: A Proposal for Multiple Objective Particle Swarm Optimization,” *IEEE Congress on Evolutionary Computation*, 2, pp. 1051-1056.

Cunha, M. C., and Sousa, J. (2001) “Hydraulic infrastructures design using simulated annealing.” *Journal of Infrastructure Systems,* 7(1), pp. 32–39. http://dx.doi.org/10.1061/(ASCE)1076-0342(2001)7:1(32).

Cunha, M.D.C. and Sousa, J. (1999) “Water distribution network design optimization: simulated annealing approach.” *Journal of water resources planning and management, 125*(4), pp. 215-221.

Czajkoska, A. M. (2016) “Maximum entropy based evolutionary optimization of water distribution networks under multiple operating conditions and self-adaptive search space reduction method.” *University of Strathclyde*, Thesis [Preprint], pp. 1-260. https://doi.org/10.48730/fv7v-qa64.

Dandy, G.C., Simpson, A.R., and Murphy, L.J. (1996) “An improved genetic algorithm for pipe network optimization.” *Water Resources Research*, 32(2), pp. 449-458.

Deb, K. (2000) “An efficient constraint handling method for genetic algorithms.” *Computer Methods in Applied Mechanics and Engineering*, 186(2), pp. 311-338.

Deb, K. and Bhushan, A. R. (1995) “Simulated Binary Crossover for Continuous Search Space.” *Complex Systems*, 9, pp. 115–148.

Deb, K., Pratap, A., Agarwal, S. and Mewarivan, T. (2002) “A fast and elitist multi-objective genetic algorithm: NSGA-II.” *IEEE Transactions on Evolutionary Computation*, 6(2), pp. 182–197. https://doi.org/10.1109/4235.996017.

Deep, K., Singh, K. P., Kansal, M. L. and Mohan, C. (2009) “A real coded genetic algorithm for solving integer and mixed integer optimization problems.” *Applied Mathematics and Computation*, 212(2), pp. 505–518. https://doi.org/10.1016/j.amc.2009.02.044.

Elaoud, S., Teghem, J. and Loukil, T. (2010) “Multiple crossover genetic algorithm for the multiobjective traveling salesman problem.” *Electronic Notes in Discrete Mathematics*, 36(C), pp. 939–946. https://doi.org/10.1016/j.endm.2010.05.119.

Eshleman, L.J. and Schaffer, J.D. (1993) “Real-coded genetic algorithms and interval- schemata.” *Foundations of Genetic* Algorithms, 2, pp. 187-202.

Ezzeldin, R. M. and Djebedjian, B. (2020) “Optimal design of water distribution networks using whale optimization algorithm.” *Urban Water Journal*, 17(1), pp. 14–22. https://doi.org/10.1080/1573062X.2020.1734635.

Fang, X. (2007) “Engineering design using genetic algorithms.” *IOWA state university*, pp. 101. https://doi.org/10.31274/rtd-180813-17142.

Farmani, R., Walters, G.A. and Savic, D.A. (2005) “Trade-off between Total Cost and Reliability for Anytown Water Distribution Network.” *Journal of Water Resources Planning and Management*, 131(3), pp. 161–171. https://doi.org/10.1061/(asce)0733-9496(2005)131:3(161).

Filipović, V. (2003) “Fine-grained tournament selection operator in genetic algorithms.” *Computing and Informatics*, 22(2), pp. 143–161.

Fujiwara, O. and Ganesharajah, T. (1993) “Reliability assessment of water supply systems with storage and distribution networks.” *Water Resources Research*, 29(8), pp. 2917-2924. https://doi.org/10.1016/j.proeng.2017.03.234.

Fujiwara, O. and Khang, D.B. (1990) “A two-phrase decomposition method for optimal design of looped water distribution networks.” *Water Resources Research*, 26(4), pp. 539–549. https://doi.org/10.1029/WR026i004p00539.

Geem, Z. W. (2006) "Optimal cost design of water distribution networks using harmony search." *Engineering Optimization*, 38(3), pp. 259-280. https://doi.org/10.1080/03052150500467430.

Germanopoulos, G. (1985) “A technical note on the inclusion of pressure dependent demand and leakage terms in water supply network models.”

Civil Engineering Systems, 2, pp. 171-179. https://doi.org/10.1080/02630258508970401.

Goldberg, D. E. and Deb, K. (1991) "A comparative Analysis of Selection Schemes Used in Genetic Algorithm." *Foundations of Genetic Algorithms*, 1, pp. 69-93. https://doi.org/10.1016/B978-0-08-050684-5.50008-2.

Gorev, N. B., Gorev, V. N., Kodzhespirova, I. F., Shedlovsky, I. A. and Sivakumar, P. (2021) "Technique for the pressure-analysis of water distribution networks with flow-and pressure-regulating valves." *Journal of Water Resources Planning and Management (ASCE)*, 147 (5), 06021005, https://doi.org/10.1061/(ASCE)WR.1943-5452.0001357.

Gupta, R. and Bhave, P. R. (1996). "Comparison of methods for predicting deficient network performance." *Journal of Water Resource Planning and Management*, 122(3), pp. 214-217. https://doi.org/10.1061/(ASCE)0733-9496(1996)122:3(214).

Haghighi, A., Samani, H.M.V. and Samani, Z.M.V. (2011) "GA-ILP Method for Optimization of Water Distribution Networks." *Water Resources Management*, 25(7), pp. 1791–1808. https://doi.org/10.1007/s11269-011-9775-4.

Holland J. (1975) "Adaptation in Natural and Artificial Systems." *Michigan: University of Michigan Press.*

https://doi.org/10.1109/CEC.2002.1004388

Hussain, A. and Muhammad, Y.S. (2020) "Trade-off between exploration and exploitation with genetic algorithm using a novel selection operator." *Complex and Intelligent Systems*, 6(1), pp. 1–14. https://doi.org/10.1007/s40747-019-0102-7.

Ilemobade, A. A and Stephenson, D. (2006) "Application of a constrained non-linear hydraulic gradient design tool to water reticulation network upgrade." *Urban Water Journal*, 3(4), 203-204. https://doi.org/10.1080/15730620601060213.

Jebari, K. and Madiafi, M. (2013) "Selection Methods for Genetic Algorithms." *International Journal of Emerging Sciences*, 3(4), pp. 333–344.

Kadu, M.S., Gupta, R. and Bhave, P.R. (2008) "Optimal Design of Water Networks Using a Modified Genetic Algorithm with Reduction in Search Space." *Journal of Water Resources Planning and Management*, 134(2), pp. 147–160. https://doi.org/10.1061/(asce)0733-9496(2008)134:2(147).

Katoch, S., Chauhan, S.S. and Kumar, V. (2021) "A review on genetic algorithm: past, present, and future." *Multimed Tools Applications*, 80, pp. 8091–8126. https://doi.org/10.1007/s11042-020-10139-6.

Kennedy, J. and Eberhart, R. (1995) "Particle swarm optimization." International Conference on Neural Networks, 4, pp. 1942-1948. https://doi.org/10.1109/ICNN.1995.488968.

Khalifeh, S., Akbarifard, S., Khalifeh, V. and Zallaghi, E. (2020) "Optimization of water distribution of network systems using the Harris Hawks optimization algorithm (Case study: Homashahr city)." *MethodsX*, 7, pp. 100948. https://doi.org/10.1016/j.mex.2020.100948.

Khezzar, L., Harous, S. and Benayoune, M. (2001) "Steady-state analysis of water distribution networks including pressure-reducing valves." *Computer-Aided Civil and Infrastructure Engineering*, 16(4), pp. 259–267. https://doi.org/10.1111/0885-9507.00231.

Khu, S.T. and Keedwell, E. (2005) "Introducing more choices (flexibility) in the upgrading of water distribution networks: The New York city tunnel network example." *Engineering Optimization*, 37(3), pp. 291–305. https://doi.org/10.1080/03052150512331303445.

Konak, A., Coit, D.W. and Smith, A.E. (2006) "multi-objective optimization using genetic algorithms: A tutorial." *Reliability Engineering and System Safety*, 91(9), pp. 992–1007. https://doi.org/10.1016/j.ress.2005.11.018.

Kora, P. and Yadlapalli, P. (2017) “Crossover Operators in Genetic Algorithms: A Review.” *International Journal of Computer Applications*, 162(10), pp. 34–36. https://doi.org/10.5120/ijca2017913370.

Kumar, B. R. (2013) “Introduction to genetic algorithms.” *Indian Institute of Technology Guwahati*, pp. 1–89. Kumar, A (2013) “Encoding schemes in genetic algorithm.” *International Journal of Advanced Research in IT and Engineering,* 2(3), pp. 1–7.

Loewe, L. and Hill, W.G. (2010) “The population genetics of mutations: Good, bad and indifferent.” *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1544), pp. 1153–1167. https://doi.org/10.1098/rstb.2009.0317.

Mala-Jetmarova, H., Sultanova, N. and Savic, D. (2018) “Lost in optimisation of water distribution systems? A literature review of system design.” *Water (Switzerland)*, 10(3). https://doi.org/10.3390/w10030307.

Malik, S. and Wadhwa, S. (2014) “Preventing Premature Convergence in Genetic Algorithm Using DGCA and Elitist Technique.” *International Journal of Advanced Research in Computer Science and Software Engineering*, 4(6), pp. 410–418.

Marques, J., Cunha, M. and Savić, D.A. (2015) “Multi-objective optimization of water distribution systems based on a real options approach.” *Environmental Modelling and Software*, 63, pp. 1–13. https://doi.org/10.1016/j.envsoft.2014.09.014.

Martin, D. W., and Peters, G. (1963) “The application of newton’s method to network analysis by digital computer.” *Journal of the Institute of Water Engineers*, 17, pp 115- 129.

Miller, B.L. and Goldberg, D.E. (1995) “Genetic algorithms, tournament selection, and the effects of noise.” *Complex Systems*, 9(3), pp. 193–212.

Mishra, S. (2017) "Genetic algorithm: an efficient tool for global optimization." *Advances in computational sciences and technology*, 10(8), pp. 2201–2211.

Mokhade, A.S. and Kakde, O.G. (2014) "Overview of selection schemes in real-coded genetic algorithms and their applications." *Journal of Industrial and Intelligent Information*, 2(1), pp. 71–77. https://doi.org/10.12720/jiii.2.1.71-77.

Palod, N., Prasad, V. and Khare, R. (2021) "Redefining the application of an evolutionary algorithm for the optimal pipe sizing problem." *Journal of Water and Climate Change*, 12(6), pp. 2299–2313. https://doi.org/10.2166/wcc.2021.288.

Panteleev, A. V. and Metlitskaya, D. V. (2011) "An application of genetic algorithms with binary and real coding for approximate synthesis of suboptimal control in deterministic systems." *Automation and Remote Control*, 72(11), pp. 2328–2338. https://doi.org/10.1134/S0005117911110075.

Pattanaik, J. K., Basu, M. and Dash, D. P. (2020) "Improved real-coded genetic algorithm for fixed head hydrothermal power system." *IETE Journal of Research,* 68(6), pp. 3984-3993. https://doi.org/10.1080/03772063.2020.1785339.

Prasad, T. D. (2010) "Design of pumped water distribution networks with storage." Journal of Water Resources Planning and Management, 136, pp. 129-132. http://dx.doi.org/10.1061/(ASCE)0733-9496(2010)136:1(129).

Prasad, T. D., and Park N. S. (2004) "Mmulti-objective genetic algorithms for design of water distribution networks." Journal of Water Resources Planning and Management, 130(1), pp. 73-82. https://doi.org/10.1061/(ASCE)0733-9496(2004)130:1(73).

Preis, A., and Ostfeld, A. (2008). "Multiobjective contaminant sensor network design for water distribution systems." Journal of Water Resource

Planning Management, 134(4), pp. 366- 377. https://doi.org/10.1061/(ASCE)0733-9496(2008)134:4(366).

Purshouse, R.C. and Fleming, P.J. (2002) “Why use Elitism and Sharing in a MultiObjective Genetic Algorithm?” *Proceedings of the 4th Annual Conference on Genetic and Evolutionary computation*, pp. 520–527.

Qiu, M. and Ostfeld, A. (2021) “A head formulation for the steady-state analysis of water distribution systems using an explicit and exact expression of the colebrook–white equation.” *Water (Switzerland)*, 13(9). https://doi.org/10.3390/w13091163.

Raghuwanshi, M. M., Singru, P., Kale, U. and Kakde, O. (2008) “Simulated binary crossover with lognormal distribution.” *Complexity International*, 12, pp. 1–10.

Ray, T., Singh, H. K., Isaacs, A. and Smith, W. (2009) “Infeasibility driven evolutionary algorithm for constrained optimization.” *Studies in Computational Intelligence*, 198, pp. 145–165. https://doi.org/10.1007/978-3-642-00619-7_7.

Reca, J., Martínez, J. and López, R. (2017) “A hybrid water distribution networks design optimization method based on a search space reduction approach and a genetic algorithm.” *Water*, 9(11), pp. 845. https://doi.org/10.3390/w9110845.

Reed, P., Minsker, B. and Goldberg, D.E., (2003) “Simplifying multi-objective optimization: an automated design methodology for the non-dominated sorted genetic algorithm-II.” *Water Resource Research*, 2003, 39(7), pp. 1196. http: //doi.org/10.1029/2002WR001483.

Rossman, L. A., Woo, H., Tryby, M., Shang, F., Janke, R. and Haxton, T. (2020) “EPANET 2.2 User Manual.” *U.S. Environmental Protection Agency,* [Preprint], pp. 1-118.

Rossman, L.A. (2000) “EPANET 2 users manual.” *U.S. Environmental Protection Agency,* [Preprint], pp. 1-198.

Rudolph, G. (1999) “Evolutionary search under partially ordered sets.” *Department of Computer Science*/LS11, pp. 67/99.

Rudolph, G. and Agapie, A. (2000) "Convergence properties of some multi-objective evolutionary algorithms." *Congress on Evolutionary Computation,* 2, pp. 1010-1016. http://doi.org/10.1109/CEC.2000.870756.

Saleh, S. and Tanyimboh, T.T. (2016) “Multi-Directional Maximum-Entropy Approach to the Evolutionary Design Optimization of Water Distribution Systems”, *Water Resource Management,* 30, pp.1885-1901.

Saleh, S. H. and Tanyimboh, T. T. (2013) “Coupled topology and pipe size optimization of water distribution systems.” *Water Resource Management*, 27(14), pp. 4795-4814. http://doi.org/10.1007/s11269-013-0439-4.

Saleh, S.H. and Tanyimboh, T.T. (2014) “Optimal design of water distribution systems based on entropy and topology.” *Water Resource Management*, 28(11), pp. 3555-3575. https://doi.org/10.1007/s11269-014-0687-y.

Saleh, S.H. (2013) “The integration of topology and entropy-based reliability into the optimal design of water distribution systems.” *University of Strathclyde*, thesis [preprint], pp.1-200. https://doi.org/10.48730/jhbn-gv34

Sangroula, U., Han, K. H., Koo, K. M., Gnawali, K. and Yum, K. T. (2022) “Optimization of Water Distribution Networks Using Genetic Algorithm Based SOP–WDN Program.” *Water (Switzerland)*, 14(6), pp. 851. https://doi.org/10.3390/w14060851.

Sato, Y. and Sato, M. (2022) “Using Dominated Solutions at Edges to the Diversity and the Uniformity of Non-dominated Solution Distributions in NSGA-II.” *SN COMPUT. SCI.,* **3**, pp. 432. https://doi.org/10.1007/s42979-022-01303-w.

Savic, D.A. and Walters, G.A. (1997) “Genetic Algorithms for Least-Cost Design of Water Distribution Networks.” *Journal of Water Resources Planning and Management*, 123(2), pp. 67–77. https://doi.org/10.1061/(asce)0733-9496(1997)123:2(67).

Schädler, P., Berdugo, J. D., Hanne, T. and Dornberger, R. (2016) "A distance-based pareto evolutionary algorithm based on SPEA for combinatorial problems." *4th International Symposium on Computational and Business Intelligence (ISCBI)*, Olten, Switzerland, pp. 112-117. http://doi.org/10.1109/ISCBI.2016.7743268.

Seyoum, A.G. (2015) “Head-dependent modelling and optimization of water distribution systems.” *University of Strathclyde*, Thesis [Preprint], pp. 2-142. https://doi.org/10.48730/mf0j-st83.

Seyoum, A.G. and Tanyimboh, T.T. (2016) ‘Investigation into the Pressure-Driven Extension of the EPANET Hydraulic Simulation Model for Water Distribution Systems’, *Water Resources Management*, 30(14), pp. 5351–5367. Available at: https://doi.org/10.1007/s11269-016-1492-6.

Seyoum, A.G., Tanyimboh, T.T. and Siew, C. (2011) “Comparison of demand driven and pressure dependent hydraulic approaches for modelling water quality in distribution networks.” UK, *Urban Water Management: Challenges and Opportunities - 11th International Conference on Computing and Control for the Water Industry,* pp. 3–9.

Seyoum, A.G., Tanyimboh, T.T. and Siew, C. (2016) “Practical Application of Penalty-Free Evolutionary Multi-Objective Optimization of Water Distribution Systems.” *Water Utility Journal*, (12), pp. 49–55.

Shirajuddin, T. M., Muhammad, N.S. and Abdullah, J. (2023) “Optimization problems in water distribution systems using Non-dominated Sorting Genetic Algorithm II: An overview.” *Ain Shams Engineering Journal*, 14(4), pp. 101932. https://doi.org/10.1016/j.asej.2022.101932.

Siew, C. and Tanyimboh, T.T (2012a) “Penalty-Free Feasibility Boundary Convergent Multi-Objective Evolutionary Algorithm for the Optimization of Water Distribution Systems." *Water Resources Management*, 26(15), pp. 4485–4507. https://doi.org/10.1007/s11269-012-0158-2.

Siew, C. and Tanyimboh, T.T. (2012b) "Pressure-Dependent EPANET Extension." *Water Resources Management*, 26(6), pp. 1477–1498. https://doi.org/10.1007/s11269-011-9968-x.

Siew, C. (2011) "A penalty-free multi-objective evolutionary optimization approach for the design and rehabilitation of water." *University of Strathclyde*, Thesis [Preprint], pp.1-200. https://doi.org/10.48730/13ay-fp44.

Siew, C., Tanyimboh, T.T. and Seyoum, A.G. (2016) "Penalty-Free Multi-Objective Evolutionary Approach to Optimization of Anytown Water Distribution Network." *Water Resources Management*, pp. 3671–3688. https://doi.org/10.1007/s11269-016-1371-1.

Siew, C., Tanyimboh, T.T. and Seyoum, A.G. (2014) "Assessment of Penalty-Free Multi-Objective Evolutionary Optimization Approach for the Design and Rehabilitation of Water Distribution Systems." *Water Resources Management*, 28(2), pp. 373–389. https://doi.org/10.1007/s11269-013-0488-8.

Sirisant, S. and Janga, R.M. (2018) "Reliability-based design of water distribution networks using self-adaptive differential evolution algorithm." *ISH Journal of Hydraulic Engineering*, 24 (2), pp. 198–212. http://doi.org/10.1080/09715010.2017.1408038.

Sorsa, A., Peltokangas, R. and Leiviskä, K. (2008) "Real-coded genetic algorithms and nonlinear parameter identification." *IEEE Xplore,* 2, pp. https://doi.org/10.1109/IS.2008.4670495.

Suribabu, C.R. (2010) "Differential evolution algorithm for optimal design of water distribution networks" *Journal of Hydroinformatics,* 12(1), pp. 66–82. https://doi.org/10.2166/hydro.2010.014.

Tanyimboh, T.T. and Kalungi, P. (2008) “Optimal long-term design, rehabilitation and upgrading of water distribution networks.” *Engineering Optimization*, 40(7), pp. 637-654. https://doi.org/10.1080/03052150801981915.

Tanyimboh, T.T. (2021) “Redundant binary codes in genetic algorithms: multi-objective design optimization of water distribution networks.” *Water Science and Technology: Water Supply*, 21(1), pp. 444–457. https://doi.org/10.2166/ws.2020.329.

Tanyimboh, T.T. and Czajkowska, A.M. (2021) “Entropy maximizing evolutionary design optimization of water distribution networks under multiple operating conditions.” *Environment Systems and Decisions*, 41, pp. 267-285. https://doi.org/10.1007/s10669-021-09807-1.

Tanyimboh, T.T. and Czajkowska, M. (2018) “Joint Entropy Based Multi-Objective Evolutionary Optimization of Water Distribution Networks.” *Water Resource Management*, 32(8), pp. 2569–2584. https://doi.org/10.1007/s11269-017-1888-y.

Tanyimboh, T.T. and Seyoum, A.G. (2020) “Design optimization of water distribution networks: Real-world case study with penalty-free multi-objective genetic algorithm using pressure-driven simulation.” *Water SA*, 46(3), pp. 465–475. https://doi.org/10.17159/wsa/2020.v46.i3.8657.

Tanyimboh, T.T. and Templeman, A.B. (2010) “Seamless pressure-deficient water distribution system model.” *Proceedings of the Institution of Civil Engineers: Water Management*, 163(8), pp. 389–396. https://doi.org/10.1680/wama.900013.

Todini, E. and Pilati, S. (1988) “A gradient algorithm for the analysis of pipe networks.” Computer Science, 1, pp. 1-20.

Tran, K.D. (2005) “Elitist Non-Dominated Sorting GA-II (NSGA-II) as a Parameter-less Multi-Objective Genetic Algorithm.” *IEEE,* pp. 359–367. https:doi.org/10.1109/SECON.2005.1423273.

Vairavamoorthy, K. and Ali, M. (2000). “Optimal design of water distribution systems using genetic algorithms.” *Computer-aided Civil and*

Infrastructure Engineering, 15, 374-382. https://doi.org/10.1111/0885-9507.00201.

Vairavamoorthy, K. and Ali, M. (2005). “Pipe index vector: a method to improve genetic- algorithm-based pipe optimization.” *Journal of hydraulic Engineering*,131(12), pp. 1117-1125. https://doi.org/10.1061/(ASCE)0733-9429(2005)131:12(1117).

Wagner, J.M. Shamir, U. and Marks, D.H. (1988) “Water distribution reliability: simulation methods.” *Journal of Water Resources Planning and Management*, 114(3), pp. 276-294. https://doi.org/10.1061/(ASCE)0733-9496(1988)114:3(276).

Wang, Q., Wang, L., Huang, W., Wang, Z., Liu, S. and Savić, D.A. (2019) “Parameterization of NSGA-II for the Optimal Design of Water Distribution Systems.” *Water (Switzerland)*, *11*, pp. 971. https://doi.org/10.3390/w11050971.

WHO (World Health Organization) (2014) “Water safety in distribution systems.” *WHO Document Production Services, Geneva, Switzerland*, pp. 153.

Wood, D. and Charles, C. (1972) “Hydraulic network analysis using linear theory.” *Journal of Hydraulics Engineering*, 98, pp. 1157-1170.

Wu, Z. Y. and Simpson, A. R. (2002) “A self-adaptive boundary search genetic algorithm and its application to water distribution systems.” *Journal of Hydraulic Research*, pp. 191-203. https://doi.org/10.1080/00221680209499862.

Wu, Z. Y. and Walski, T. (2005) “Self-adaptive penalty approach compared with other constraint-handling techniques for pipeline optimization.” *Journal of Water Resource Planning Management,* 131 (3), pp. 181–192. https://doi.org/10.1061/(ASCE)0733-9496(2005)131:3(181).

Wu, Z.Y., Boulos, P. F., Orr, C.H. and Ro, J.J. (2001). “Using genetic algorithm to rehabilitate distribution systems.” *Journal American Water Works Association*, 93(11), 74-85. https://doi.org/10.1002/j.1551-8833.2001.tb09335.x.

Xavier, C., Amorim, E., Vieira, V. and Santos, R. (2013) "Genetic Algorithm for the History Matching Problem." *Procedia Computer Science*. 18. Pp. 946–955. https://doi.org/10.1016/j.procs.2013.05.260.

Zainuddin, F.A. and Samad, M.F.A. (2020) "A review of crossover methods and problem representation of genetic algorithm in recent engineering applications." *International Journal of Advanced Science and Technology*, 29(6), pp. 759–769.

Zarei, N., Azari, A. and Heidari, M.M. (2022) "Improvement of the performance of NSGA-II and MOPSO algorithms in multi-objective optimization of urban water distribution networks based on modification of decision space." *Applied Water Science*, 12(6), pp. 1–12. https://doi.org/10.1007/s13201-022-01610-w.

Zitzler, E. and Thiele, L. (1999) "Multi-objective evolutionary algorithms: A comparative case study and the strength pareto approach." *IEEE Transactions on Evolutionary Computation, 3*(4), pp. 257–271.

Zitzler, E., Deb, K. and Thiele, L., (2000) "Comparison of multi-objective evolutionary algorithms: empirical results." *Evolutionary Computation*, 8(2), pp. 125–148.

Zitzler, E., Laumanns, M. and Thiele, L. (2001) " SPEA2: Improving the strength pareto evolutionary algorithm." *ETH Zurich*, 103, pp. 1-17. https://doi.org/10.3929/ethz-a-004284029.

7 APPENDIX

Appendix A: Data Input and Supplementary Results for Two-Loop Network

Table A- 1 Two-loop network node data

Node No.	Elevation (m)	Demand (l/s)	Minimum required pressure head (m)
2	150	27.78	30
3	160	27.78	30
4	155	33.33	30
5	150	75.00	30
6	165	91.66	30
7	160	55.56	30

Table A- 2 Reservoir data for the Two-loop network

Reservoir No.	Head (m)
1	210.00

Table A- 3 Available commercial pipe diameters options and their corresponding costs for the Two-loop network

Diameter (in)	Diameter (mm)	Cost (units)
1	25.4	2
2	50.8	5
3	76.2	8
4	101.6	11
6	152.4	16
8	203.2	23
10	254.0	32
12	304.8	50
14	355.6	60
16	406.4	90
18	457.2	130
20	508.0	170
22	558.8	300
24	609.6	500

Appendix B: Data Input and Supplementary Results for Hanoi Network

Table B- 1 Hanoi network nodal data

Node No.	Elevation (m)	Demand (m^3/h)	Minimum required pressure head (m)
2	0	890	247.22
3	0	850	236.11
4	0	130	36.11
5	0	725	201.39
6	0	1,005	279.17
7	0	1,350	375.00
8	0	550	152.78
9	0	525	145.83
10	0	525	145.83
11	0	500	138.89
12	0	560	155.56
13	0	940	261.11
14	0	615	170.83
15	0	280	77.78
16	0	310	86.11
17	0	865	240.28
18	0	1,345	373.61
19	0	60	16.67

20	0	1,275	354.17
21	0	930	258.33
22	0	485	134.72
23	0	1,045	290.28
24	0	820	227.78
25	0	170	47.22
26	0	900	250.00
27	0	370	102.78
28	0	290	80.56
29	0	36	10.00
30	0	360	100.00
31	0	105	29.17
32	0	805	223.61

Table B- 2 Reservoir data for the Hanoi network

Reservoir	Total head (m)
1	100

Table B- 3 Available commercial pipe diameters options and their corresponding costs for the Hanoi network

Diameter (mm)	Cost ($/m)
304.80	45.73
406.40	70.40
508.00	98.38
609.60	129.33
762.00	180.80
1,016.00	278.30

Table B- 4 Pipe data for the Hanoi network

Pipe	Start node	End Node	Length (m)	H-W Coefficient
1	1	2	100	130
2	2	3	1,350	130
3	3	4	900	130
4	4	5	1,150	130
5	5	6	1,450	130
6	6	7	450	130

7	7	8	850	130
8	8	9	850	130
9	9	10	800	130
10	10	11	950	130
11	11	12	1,200	130
12	12	13	3,500	130
13	10	14	800	130
14	14	15	500	130
15	15	16	550	130
16	17	16	2,730	130
17	18	17	1,750	130
18	19	18	800	130
19	3	19	400	130
20	3	20	2,200	130
21	20	21	1,500	130
22	21	22	500	130
23	20	23	2,650	130
24	23	24	1,230	130
25	24	25	1,300	130

26	26	25	850	130
27	27	26	300	130
28	16	27	750	130
29	23	28	1,500	130
30	28	29	2,000	130
31	29	30	1,600	130
32	30	31	150	130
33	32	31	860	130
34	25	32	950	130