Sensitivity of algorithm parameters and objective function scaling in multi-objective optimisation of water distribution systems

Helena Mala-Jetmarova, Andrew Barton and Adil Bagirov

ABSTRACT

This paper presents an extensive analysis of the sensitivity of multi-objective algorithm parameters and objective function scaling tested on a large number of parameter setting combinations for a water distribution system optimisation problem. The optimisation model comprises two operational objectives minimised concurrently, the pump energy costs and deviations of constituent concentrations as a water quality measure. This optimisation model is applied to a regional non-drinking water distribution system, and solved using the optimisation software GANetXL incorporating the NSGA-II linked with the network analysis software EPANet. The sensitivity analysis employs a set of performance metrics, which were designed to capture the overall quality of the computed Pareto fronts. The performance and sensitivity of NSGA-II parameters using those metrics is evaluated. The results demonstrate that NSGA-II is sensitive to different parameter settings, and unlike in the single-objective problems, a range of parameter setting combinations appears to be required to reach a Pareto front of optimal solutions. Additionally, inadequately scaled objective functions cause the NSGA-II bias towards the second objective. Lastly, the methodology for performance and sensitivity analysis may be used for calibration of algorithm parameters.

Key words | algorithm parameters, multi-objective optimisation, performance metrics, scaling, sensitivity, water distribution systems

NOTATION

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_i$</td>
<td>area determined by the solutions in $PF_{known}$ and the reference point</td>
</tr>
<tr>
<td>$b_{mi}$</td>
<td>binary variable describing the status of pump $m$ during a time interval $i$</td>
</tr>
<tr>
<td>$c$</td>
<td>constituent (water quality parameter)</td>
</tr>
<tr>
<td>$C$</td>
<td>number of constituents (water quality parameters) $c$</td>
</tr>
<tr>
<td>$c_{adj}$</td>
<td>actual (modelled) concentration of a constituent $c$ at the customer node $d$ during a time interval $j$</td>
</tr>
<tr>
<td>$c_{max}$</td>
<td>maximum allowed concentration of a constituent $c$ at the customer node $d$</td>
</tr>
<tr>
<td>$c_{min}$</td>
<td>minimum required concentration of a constituent $c$ at the customer node $d$</td>
</tr>
<tr>
<td>CRS</td>
<td>crossover operator</td>
</tr>
<tr>
<td>$d$</td>
<td>customer demand node</td>
</tr>
<tr>
<td>$D$</td>
<td>number of customer demand nodes $d$</td>
</tr>
<tr>
<td>$def_{i}^{max}$</td>
<td>maximum allowed volume deficit in storage tank $s$ at the end of the simulation period $T$</td>
</tr>
<tr>
<td>$E$</td>
<td>metric error ratio</td>
</tr>
<tr>
<td>$EC_{i}$</td>
<td>electricity tariff during a time interval $i$</td>
</tr>
<tr>
<td>$EX$</td>
<td>metric extent</td>
</tr>
<tr>
<td>$FI(b)$</td>
<td>economic objective function as pump energy costs, over the simulation period $T$</td>
</tr>
<tr>
<td>$FII(b)$</td>
<td>water quality objective function as squared deviations of actual constituent concentrations from required values, over the simulation period $T$</td>
</tr>
</tbody>
</table>

doi: 10.2166/hydro.2015.062
INTRODUCTION

Optimisation of water distribution systems (WDSs), which generally covers optimum system design, rehabilitation and operation, has been a subject of intensive research over the past several decades. Although these problems inherently involve multiple objectives, they were initially approached as single-objective due to the computational complexity involved. Over the past two decades, nonetheless, multi-objective approaches have been increasingly applied to analyse trade-offs between competing objectives; for example system and operational costs, service quality, system reliability and water quality. Regarding optimum design and rehabilitation, the minimisation of the network costs is often defined as the first objective, with other competing objectives being the maximisation of network reliability (Prasad & Park 2004; Alvisi & Franchini 2006; Farmani et al. 2006, Farmani et al. 2005b; Giustolisi & Berardi 2009;
network robustness (Kapelan et al. 2005; Babayan et al. 2007) or network performance (Halhal et al. 1999; Walters et al. 1999; Vamvakeridou-Lyroudia et al. 2005). Additionally, network design and/or rehabilitation costs were traded off against service quality measures such as pressure head deficit (Farmani et al. 2005a; Atiquzzaman et al. 2006; Keedwell & Khu 2006; Jin et al. 2008). Optimum operation of WDSs has been mainly concerned with the minimisation of energy costs for pumping as the first objective and pump maintenance as the second objective (Savic et al. 1997; Sotelo & Baran 2001; Kelner & Leonard 2003; Lopez-Ibanez et al. 2005). Most recently, tradeoffs between pump energy costs and greenhouse gas emissions (Wu et al. 2010) or water quality measures (Kurek & Ostfeld 2013, 2014) have also been investigated.

Scaling of the objective and constraint functions is important, because it may have a significant influence on the performance of optimisation algorithms. It represents down-scaling or up-scaling of these functions, so that they cover a similar range of numerical values. However, the majority of the above literature did not consider scaling of the objective functions. Inadequate scaling of the objective functions may lead, according to Bhattacharjya (2007), to the algorithm being biased towards one objective, specifically an objective with larger objective function values. It may affect spacing between the solutions with concentrations of solutions either increasing or decreasing along the Pareto front and may cause discontinuity of the Pareto front (Bhattacharjya 2007). It may also impact the location of the Pareto front with solutions being away from the true Pareto front (Bhattacharjya 2007). More detailed information on scaling of optimisation problems can be found in Gill et al. (1981). Work presented within this paper addresses those scaling issues by comparing a broad range of results with non-scaled and scaled objective functions for a multi-objective problem of the optimal operation of WDSs.

To solve multi-objective optimisation problems in WDSs, multiple objective evolutionary algorithms (MOEAs) have been mostly applied. Those MOEAs include non-dominated sorting genetic algorithm (NSGA) (Srinivas & Deb 1994), NSGA-II (Deb et al. 2002), strength Pareto evolutionary algorithm (SPEA) (Zitzler & Thiele 1999), SPEA2 (Zitzler et al. 2001) and others. These algorithms use certain parameters being population size, probability of mutation and crossover, which need to be set up (i.e. calibrated) for each individual problem, so the algorithm delivers optimal solution (Gibbs et al. 2010; Younis & Dandy 2012; Marchi et al. 2014). Nevertheless, very limited attention has been dedicated to date to analyse sensitivity of algorithm parameters and their performance in multi-objective optimisation of WDSs. In the majority of literature reviewed, the selection of parameter settings is either not discussed (Halhal et al. 1999; Walters et al. 1999; Kelner & Leonard 2003; Farmani et al. 2006; Babayan et al. 2007; Raad et al. 2010) or is based on literature values (Farmani et al. 2005a; Alvisi & Franchini 2006; Alfonso et al. 2010) with no further rationale provided. There are also cases where the same algorithm with the same parameter settings is applied to different test networks (Farmani et al. 2005a; Kurek & Ostfeld 2014), they did not present such analysis as part of their research. This current work explores the sensitivity of multi-objective algorithm parameters, evaluates the performance of a wide range of parameter settings and establishes the influence of these settings on solutions for the optimal operation of WDSs including two objectives. It utilises a set of performance metrics to do so.

Performance metrics were proposed in Van Veldhuizen (1999) and Zitzler (1999) and are normally used to compare performances of different multi-objective algorithms (Van Veldhuizen & Lamont 2000b; Zitzler et al. 2000; Kollat & Reed 2006; Kollat et al. 2008). According to Zitzler et al. (2005), the comparison of solutions in multi-objective optimisation is substantially more complex than in single-objective, because there is no single performance metric which is both compatible and complete. Zitzler et al. (2000) suggest that such a comparison evaluates (i) distance of the obtained Pareto front from the Pareto front of optimal solutions (i.e. true Pareto front), (ii) uniformity of distribution of the solutions in the Pareto front and (iii) the extent of the obtained Pareto front to ensure that a wide range of objective values is covered. In WDS optimisation, performance metrics were applied in optimal system design (Keedwell & Khu...
2006; Raad et al. 2010; Kanta et al. 2012) and operation (Baran et al. 2005) to compare performances of multiple MOEAs. To the authors’ knowledge, there are no WDS-focused studies which have analysed the sensitivity of algorithm parameters within one MOEA. Early studies were applied to general test cases (Hadka & Reed 2012) or general water resources applications (Reed et al. 2012), and used performance metrics and Sobol’s global variance decomposition to evaluate parameter sensitivity.

The purpose of this current work is to comprehensively explore the sensitivity of multi-objective algorithm parameters and objective function scaling. This work is believed to be the first dealing with the performance and sensitivity of algorithm parameters within one MOEA in WDS optimisation. Algorithm parameters considered include population size, number of generations, probability of crossover (subsequently referred to as crossover) and probability of mutation (subsequently referred to as mutation). The test network adopted from the literature (Ostfeld & Salomons 2004; Ostfeld et al. 2011) represents a regional non-drinking WDS, where two operational objectives, economic and water quality, are formulated to be optimised. It was identified in both urban drinking WDSs (Kurek & Ostfeld 2014) and regional non-drinking WDSs (Mala-Jetmarova et al. 2014) that a relationship between the economic objective (i.e. pump energy costs) and water quality objective is conflicting. The reason for this relationship is that, with more relaxed water quality requirements, increased flexibility exists for pumps to operate more effectively over a broader objective space (Mala-Jetmarova et al. 2014).

OPTIMISATION MODEL

Objective functions

Economic objective

The economic objective is represented by pump energy costs and is often referred to in the literature as a pump scheduling problem (Lopez-Ibanez et al. 2008). Explicit pump schedules are used, which specify the time when a pump operates. For this purpose, simulation period $T$ is divided into equal time intervals $i$, in which the pump $m$ adopts a binary variable $b_{mi}$ of either 0 or 1 to describe its status as being off or on, respectively. The economic objective function for the pump scheduling problem is written as:

$$FI(b) = \sum_{m=1}^{M} \sum_{i=1}^{L} EC_i \frac{kH_{mi} Q_{mi}}{\eta_m} \Delta t_i b_{mi}$$

where $M$ ($\cdot$) is the number of pumps, $m=1, \ldots, M$; $L$ ($\cdot$) is the number of equal time intervals $i$ for a hydraulic simulation within the simulation period $T$, $i=1, \ldots, L$; $EC_i$ ($$/kWh$$) is the electricity tariff during a time interval $i$; $k$ ($\cdot$) is a unit conversion coefficient; $H_{mi}$ ($m$) is the total dynamic head supplied by pump $m$ during a time interval $i$; $Q_{mi}$ ($L/s$) is the flow through pump $m$ during a time interval $i$; $\eta_m$ ($\cdot$) is the efficiency of pump $m$; $\Delta t_i$ ($s$) is the length of a time interval $i$; $b_{mi}$ ($\cdot$), $b_{mi}=0, 1$ is the binary variable describing the status of pump $m$ as being off or on, respectively, during a time interval $i$.

Water quality objective

The water quality objective aims to meet the water quality requirements of various customer groups. This objective is defined as deviations of actual constituent concentrations from required values, which are normally prescribed by health and other regulatory guidelines. Water quality requirements of various customer groups, who are represented as demand nodes, are specified by lower and upper bounds of constituent concentrations. Each of those customer groups has different water quality requirements. The constituents considered are conservative (i.e. non-reactive). The objective function is defined as the square of deviations of actual (modelled) constituent concentrations from required minimum and maximum values as follows (adapted from Sakarya & Mays (2003)):

$$FII(b) = \sum_{c=1}^{C} \sum_{d=1}^{D} \sum_{j=1}^{L} \left\{ \min\{0, \min(c_{cdj} - c_{cd}^{\min}, c_{cd}^{\max} - c_{cdj})\} \right\}^2$$

where $C$ ($\cdot$) is the number of constituents (water quality parameters) $c$, $c=1, \ldots, C$; $D$ ($\cdot$) is the number of customer...
demand nodes $d$, $d=1, \ldots, D$; $L$ is the number of equal time intervals $j$ for water quality simulation within the simulation period $T$, $j=1, \ldots, L$; $c_{min}^d$ and $c_{max}^d$ (mg/L) are the minimum required and maximum allowed concentrations of a constituent $c$ at the customer node $d$, respectively; $c_{cdj}^d$ (mg/L) is the actual (modelled) concentration of a constituent $c$ at the customer node $d$ during a time interval $j$.

**Constraints and decision variables**

The constraints to the optimisation problem are, firstly, explicit system constraints such as conservation of mass of flow, conservation of energy and conservation of mass of constituent, which are controlled by network analysis software EPANet. These can be found in Rossman (2000), thus are not listed here. Secondly, there are implicit bound constraints, which include the minimum pressure at customer nodes (3), the water level limits at storage tanks (4) and the volume deficit at storage tanks at the end of the simulation period (5) as follows:

\[ H_{di} \geq H_{dmin} \]  
\[ y_{smin} \leq y_{si} \leq y_{smax} \]  
\[ \frac{\Delta V_{ST}}{V_s} \times 100 \leq def_{smax} \]

where $H_{di}$ (m) is pressure at demand node $d$ during a time interval $i$; $H_{dmin}^i$ (m) is the minimum required pressure at demand node $d$; $y_{si}$ (m) is the water level in storage tank $s$ during a time interval $i$; $y_{smin}^i$ (m) and $y_{smax}^i$ (m) is the minimum and maximum water levels in storage tank $s$, respectively; $\Delta V_{ST}$ ($m^3$) is volume deficit (difference between the initial and final volume) in storage tank $s$ at the end of the simulation period $T$; $V_s$ ($m^3$) is volume of storage tank $s$ constrained by minimum and maximum water levels; $def_{smax}$ (%) is the maximum allowed volume deficit in storage tank $s$ at the end of the simulation period $T$.

The optimisation problem includes one type of decision variable, which is the binary variable $b_m \in \{0, 1\}$ describing the status of pump $m$ (off or on, respectively) during a time interval $i$.

**Multi-objective optimisation problem**

The multi-objective optimisation problem for the minimisation of pump energy costs and optimisation of water quality is written as:

\[ \text{Minimise} \left(FI(b), FII(b)\right) \]

subject to conservation of mass of flow and constituent, conservation of energy and (3)–(5). Conservation of mass of flow and constituent, conservation of energy and constraint (4) are managed by the EPANet, while constraints (3) and (5) are included in the optimisation problem using the penalty functions. These penalty functions are implemented using an ‘infeasibility cell’ within the Excel spreadsheet of software GANetXL (CWS 2011), described further in the Solution scheme section.

**MODEL APPLICATION**

The optimisation model is applied to a network shown in Figure 1, which represents a regional WDS, where non-drinking water is supplied to various customer groups, for example, agricultural, domestic and industrial. Requirements for water quality vary across these customer groups. This network was adapted from Ostfeld & Salomons (2004) with minor modifications from Ostfeld et al. (2011). It consists of 11 pipes ($p_1$–$p_{11}$), six customer demand nodes ($d_1$–$d_6$), three non-demand nodes ($n_1$–$n_3$), eight pumps ($m_1$–$m_8$) located at three pump stations ($P_1$–$P_3$) and an elevated...
storage tank \( (s_1) \). The system is supplied from three distinct water sources (reservoirs) \( (r_1-r_3) \) of different qualities.

Water quality is represented by one non-specified conservative (i.e. non-reactive) water quality parameter with constituent concentration (mg/L), which differs from reservoir to reservoir, but is constant over time. Each customer group \( (d_1-d_6) \) has different water quality requirements characterised by the maximum value of the concentration. Data for the network can be found in Mala-Jetmarova et al. (2013).

The optimisation is undertaken for an extended period simulation (EPS) with the length of the simulation period being 24 hours, and the simulation time step for both hydraulic and water quality analysis being 1 hour. Because the emphasis of this work is not concerned with detailed system operations, but rather sensitivity of the NSGA-II parameters, these settings are deemed acceptable for the purpose of this work.

**SOLUTION SCHEME**

The solution methodology integrates a network analysis simulator with an optimisation model (Figure 2). Network analysis software EPANet (Rossman 2000) version 2.0 is used to perform hydraulic and water quality extended period simulations (EPS). EPANet is a widely accepted open-source software package, which is frequently used in conjunction with optimisation research for WDSs.

**GANETXL**

Software GANetXL (CWS 2011), available from the Centre for Water Systems (CWS), University of Exeter, UK, is used as the optimisation tool. GANetXL, which was developed by the CWS as a Microsoft Excel add-in, is a generic optimisation engine with spreadsheet based interface for solving both single and multi-objective optimisation problems (Savic et al. 2011). The foremost advantage of the GANetXL is that it enables easy integration with the EPANet via visual basic for applications (VBA) code and also enables direct access to evolutionary single and multi-objective algorithms. From the family of MOEAs, GANetXL incorporates the NSGA-II (Deb et al. 2002). This work confirmed that GANetXL is well suited for solving multi-objective optimisation problems in WDS operation, which incorporate water quality aspects with conservative constituents.

**NSGA-II**

The NSGA-II is a population based multi-objective optimisation algorithm, which was introduced by Deb et al. (2000).
Since then, this algorithm has been successfully applied to multi-objective optimisation problems of the design and rehabilitation (Prasad & Park 2004; Farmani et al. 2005b; Atiquzzaman et al. 2006; Jin et al. 2008; Kanta et al. 2012), and operation (Baran et al. 2005) of WDSs. The NSGA-II principally works as follows. An initial population $P_0$ of random solutions of size $N$ is generated and each solution assigned a fitness (or rank). Applying binary tournament selection, and crossover and mutation operators, the offspring population $Q_0$ of size $N$ is created. Populations $P_0$ and $Q_0$ are combined into the population $R_0 = P_0 \cup Q_0$ of the size $2N$. Non-dominated sorting is then applied, where the population $R_0$ is sorted into sets of non-dominated fronts or sets of solutions which are dominated by the same number of Pareto fronts. From these sets $F_1, F_2, F_3, \ldots, F_n$ of the combined population $R_0$, the set $F_1$ includes the best and $F_n$ worst solutions. If the size of the set $F_i$ is smaller than $N$, all members of this set are selected for the new population $P_t$. The remaining places in the population $P_t$ are filled in subsequently from sets $F_2, F_3$ and so on until the size $N$ is reached, using both the rank and crowding distance. The solutions in the set, which as the last set contributes to the new population $P_b$, are sorted using the crowded-comparison operator and the best solutions are selected to fill in the last places in $P_t$. The offspring population $Q_t$ is then created from $P_t$ applying selection, crossover and mutation operators, where selection is based on crowded-comparison operator. The process is repeated by creating the population $R_t = P_t \cup Q_t$ of the size $2N$, forming non-dominated sets $F_1, F_2, F_3, \ldots, F_n$ and creating the new population $P_{t+1}$ and so on, until stopping criteria is met. A detailed description of the NSGA-II can be found in Deb et al. (2000, 2002).

### METHODOLOGY FOR PERFORMANCE AND SENSITIVITY ANALYSIS

The value of this work lies chiefly in the extensive performance and sensitivity analysis of the NSGA-II parameters and objective function scaling, which is tested on a wide range of parameter settings. The NSGA-II parameters considered are: (i) population size, (ii) the number of generations, (iii) crossover and (iv) mutation. A reasonable range for each of these parameters was identified based on the recommended values from within literature and also preliminary testing, and is included in Table 1. Mutation and crossovers can also adopt different types. The type selected for mutation is ‘simple by gene’, which is set as the default within GANetXL. Two types were tested for crossover, being ‘simple one point’ (S1P) and ‘simple multi-point’ (SXP). These mutation and crossover types are described in more detail in CWS (2011).

Regarding scaling of the objective function, two alternatives are considered. Firstly, objective functions are not scaled. Secondly, the water quality objective function is linearly scaled down to ensure that the two objective functions have a similar range of numerical values. As an example, the range of the water quality objective as deviations of constituent concentrations is 0–100,000 (mg/L)$^2$ and the range of the economic objective as pump energy costs is 150–210 $/\text{day}$. Hence, the water quality objective is multiplied by 0.01 to obtain a similar order to the economic objective. The first case is referred to subsequently in the text as non-scaled objective functions, and the second case as scaled objective functions.

NSGA-II parameters settings were systematically arranged into 76 parameter setting combinations with either

<table>
<thead>
<tr>
<th>Table 1</th>
<th>NSGA-II parameter settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
<td>Parameter setting</td>
</tr>
<tr>
<td>Population size</td>
<td>50; 100; 150; 200; 300; 400</td>
</tr>
<tr>
<td>Number of generations</td>
<td>500; 1,000; 2,000</td>
</tr>
<tr>
<td>Mutation operator</td>
<td>0.002; 0.005; 0.01; 0.1</td>
</tr>
<tr>
<td>Crossover operator</td>
<td>0.6; 0.7; 0.8; 0.9; 1.0</td>
</tr>
</tbody>
</table>
non-scaled or scaled objective functions (Table 2), so the total number of analyses performed was 152 (i.e. $2 \times 76$). Due to the large number of modelling combinations the optimisation runs were undertaken as a two stage process, being sensitivity analysis 1 (SA1) and sensitivity analysis 2 (SA2). In the first stage (SA1), sensitivity of mutation and crossover was tested with a constant population size and constant number of generations. Those constant values for the population size and number of generations were cautiously selected based on preliminary testing, which was used initially to identify a reasonable range for the algorithm parameters (as previously reported in Table 1). The results obtained were visually compared with the well performing mutation and crossover operators then selected. In the second stage (SA2), these operators were applied for the sensitivity analysis of the population size and the number of generations. Additionally, all parameter setting combinations were analysed with both non-scaled and scaled objective functions.

A single run was performed for each parameter setting combination from an initial population of random solutions. Therefore, the sensitivity of seed value (i.e. convergence of an algorithm from different initial populations of random solutions) was not investigated as part of this work and has been left for further research. A seed value of 1 was used for all analyses conducted.

Generally, to evaluate the sensitivity of a particular model parameter (i.e. model input), the following question needs to be answered: How sensitive is the solution (i.e. model output such as Pareto front) to the change of this particular parameter, when other parameters are kept constant? Analysing sensitivity of individual parameters, authors strive to answer this question. It is paramount to note that the sensitivity analysis was undertaken as performance evaluation. For example, if a particular parameter (adapting various values, when other parameters are kept constant) delivers ‘good’ solutions for some of its values and ‘poor’ solutions for others, a solution is very sensitive towards this parameter, and consequently the parameter is very sensitive. Vice versa, when a parameter keeps delivering ‘good’ solutions (while it adapts various values and other parameters are kept constant), it is not very sensitive.

Two measures were used for performance evaluation of parameter setting combinations; first, a visual comparison and second, a set of performance metrics. Due to the large amount of results, where often the Pareto fronts are non-parallel and intersecting, the visual comparison in isolation would become a very time consuming and difficult task. For this reason, the set of metrics supplemented the visual comparison to provide a faster, less subjective, and more quantitative means for performance evaluation.

Only Pareto fronts, in which all solutions are feasible (i.e. comply with constraints and thus are useful for operators), can be used for performance evaluation. The exclusion of Pareto fronts containing infeasible solutions is to ensure that a parameter setting combination, which delivered a Pareto front with infeasible solutions, cannot score better than another parameter setting combination, which produced only feasible solutions.

### Table 2 Parameter setting combinations for sensitivity analysis

<table>
<thead>
<tr>
<th>Analysis name</th>
<th>Sensitivity analysis of</th>
<th>Setting of constant parameters</th>
<th>Setting of variable parameters*</th>
<th>No. of parameter setting combinations</th>
<th>No. of analyses conducted</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA1</td>
<td>• Crossover</td>
<td>POP: 100</td>
<td>MUT: 0.002; 0.005; 0.01; 0.1</td>
<td>40</td>
<td>40 (non-scaled&lt;sup&gt;**&lt;/sup&gt;)</td>
</tr>
<tr>
<td></td>
<td>• Mutation</td>
<td>GEN: 2,000</td>
<td>CRS: 0.6; 0.7; 0.8; 0.9; 1.0</td>
<td></td>
<td>40 (scaled&lt;sup&gt;**&lt;/sup&gt;)</td>
</tr>
<tr>
<td>SA2</td>
<td>• Population size</td>
<td>MUT: 0.005</td>
<td>POP: 50; 100; 150; 200; 300; 400</td>
<td>36</td>
<td>36 (non-scaled&lt;sup&gt;**&lt;/sup&gt;)</td>
</tr>
<tr>
<td></td>
<td>• Number of generations</td>
<td>CRS: 0.7 SXP; 1.0 S1P</td>
<td>GEN: 500; 1,000; 2,000</td>
<td></td>
<td>36 (scaled&lt;sup&gt;**&lt;/sup&gt;)</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td>76</td>
<td>152</td>
</tr>
</tbody>
</table>

POP – population size, GEN – number of generations, MUT – mutation, CRS – crossover.

*All possible combinations were analysed.

**Objective functions either non-scaled or scaled.
Performance metrics

Performance metrics can measure algorithm effectiveness and efficiency (Van Veldhuizen 1999). In this work, effectiveness (or quality of solution) is the main aim as to evaluate how parameter settings affect the final solution. In other words, the quality of the Pareto front is the forefront interest, thus objective function values are being evaluated only. In order to keep consistent terminology with previous research, the following terms are adapted from Van Veldhuizen & Lamont (2000a):

- A known Pareto front (PF known), which is the final computed Pareto front returned by the NSGA-II at termination, for the particular parameter setting combination.
- The true Pareto front (PF true), similarly described as a true Pareto set, ‘is not explicitly known for problems of any difficulty … is implicitly defined by the functions composing an MOP (multi-objective optimisation problem) – it is fixed and does not change’ (Van Veldhuizen & Lamont 2000a).

Similar to Baran et al. (2005), a very good approximation to the PF true is obtained with the union of all computed PF known. More specifically, PF true used for this work is the union of the best solutions found in all analyses conducted for all parameter setting combinations, including both alternatives for objective function scaling.

No single metric can entirely capture total algorithm performance (Van Veldhuizen 1999; Deb et al. 2002). For this reason, a set of eight metrics is applied, which were adapted mainly from Van Veldhuizen (1999), Zitzler (1999) and Zitzler et al. (2000, 2003). Metrics were selected with the intent of being able to capture the overall quality of the PF known. Some of these metrics compare similarity and proximity of a PF known with the PF true (i.e. how distant they are), whereas others evaluate the quality of a PF known (i.e. the number of non-dominated solutions) or the quality of its individual solutions (i.e. spacing between these solutions). As PF known is a result of a parameter setting combination, it is worth noting that by assessing the quality of the PF known, the performance of the parameter setting combination can be evaluated.

Note that two major modifications were made to the original performance metrics of Van Veldhuizen (1999), Zitzler (1999) and Zitzler et al. (2000, 2003). Firstly, the original Van Veldhuizen and Zitzler metrics were designed for n-dimensional objective space. As two objectives are considered in this work, the metrics were converted to the (simpler) form relevant to two-dimensional objective space. Secondly, some metrics were transformed, where possible, into a normalised or relative value (i.e. per cent) which is suited better for the subsequent analysis.

Non-dominated number

The metric non-dominated number (NN) represents the percentage of total number of non-dominated solutions in a PF known. It is a modification of Van Veldhuizen’s metric overall non-dominated vector generation (ONVG) with its original definition as (Van Veldhuizen 1999):

\[
\text{ONVG} = \frac{|\text{PF known}|}{|\text{POP}|} \times 100\%
\]

where \(|\text{PF known}|\) is the cardinality of PF known, expressing the total number of solutions in PF known.

It is obvious that ONVG is dependent upon the population size, because it is expected that a certain percentage, if not all initial random solutions, will evolve into final non-dominated solutions. If all analyses were run with the same population size, ONVG would give a fair comparison. However, this is not the case as different population sizes are used. To achieve a fair comparison, NN is converted into a relative value as follows:

\[
\text{NN} = \frac{|\text{PF known}|}{\text{POP}} \times 100\%
\]

where \(\text{POP}\) is the population size used to obtain the PF known.

It is expected that the NSGA-II with calibrated parameter settings is able to evolve the entire initial population of random solutions into the same number of non-dominated optimal solutions. It is, therefore, desirable that \(\text{NN}=100\%\), which means that the final number of non-dominated solutions equals to the population size.
The metric unique non-dominated number (UN) is introduced in this work to report the percentage of unique non-dominated solutions in a PF\textsubscript{known}. A unique solution represents a solution which has a different pump schedule from other solutions. The metric UN is defined as:

$$UN = \frac{\sum_{i=1}^{NN'} \sigma_i}{POP} \times 100\% \quad \sigma_i = \begin{cases} 1 & \text{if } \eta_i \text{ is unique,} \\ 0 & \text{otherwise.} \end{cases}$$

where $\eta_i$ is a solution in PF\textsubscript{known}, $NN'$ is the number of non-dominated solutions in PF\textsubscript{known}.

The reason for this metric is that NSGA-II may not always preserve the diversity of population, thus it may return some solutions in PF\textsubscript{known} which are identical (i.e. having the same pump schedules). This means that some members of the initial population of random solutions evolved into the identical final solutions. Having identical solutions in the PF\textsubscript{known} is not desirable, nonetheless, as diverse solutions are required to be available to a decision maker. In an ideal case, all solutions in PF\textsubscript{known} are unique (i.e. UN=100%).

**True number**

The metric true number (TN) measures the percentage of solutions in PF\textsubscript{known}, which are members of the PF\textsubscript{true}. This metric is a variation of the original Van Veldhuizen’s metric error ratio (E) reporting the number of solutions in PF\textsubscript{known}, which are not members of the PF\textsubscript{true} (Van Veldhuizen 1999) as:

$$E = \frac{\sum_{i=1}^{NN'} \sigma_i}{NN'} \quad \sigma_i = \begin{cases} 1 & \text{if } \eta_i, \ i = (1, \ldots, NN') \in PF_{true}, \\ 0 & \text{otherwise.} \end{cases}$$

A result $E=0$ (an ideal situation) indicates that all solutions in PF\textsubscript{known} are contained in the PF\textsubscript{true}, while $E=1$ indicates that there are none. For easier interpretation of how many solutions in PF\textsubscript{known} are part of the PF\textsubscript{true}, TN simply reports the percentage of such solutions as:

$$TN = \frac{\sum_{i=1}^{NN'} \sigma_i}{NN'} \times 100\%,$$

where $\sigma_i$ is defined as:

$$\sigma_i = \begin{cases} 1 & \text{if } \eta_i, \ i = (1, \ldots, NN') \in PF_{true}, \\ 0 & \text{otherwise.} \end{cases}$$

In an ideal situation, when all solutions in PF\textsubscript{known} are members of PF\textsubscript{true}, TN=100%. The higher the percentage of PF\textsubscript{known} solutions are members of PF\textsubscript{true}, the better the PF\textsubscript{known} is.

**Generational distance**

The metric generational distance (GD) evaluates how close a PF\textsubscript{known} to the PF\textsubscript{true} is, by measuring the distance between those two Pareto fronts. It is adapted from Van Veldhuizen (1999) as follows:

$$GD = \sqrt{\frac{\sum_{i=1}^{NN'} \delta_i^2}{NN'}}$$

where $\delta_i$ is the Euclidean distance between the $i$th solution in PF\textsubscript{known} and the nearest solution in PF\textsubscript{true}. For example, the Euclidean distance between the two points $A=(x_A, y_A)$ and $B=(x_B, y_B)$ in 2-dimensional space $x$-$y$ is calculated as $\delta_{AB} = [(x_A-x_B)^2 + (y_A-y_B)^2]^{1/2}$.

The GD metric is kept as an absolute value consistent with the original Van Veldhuizen’s formula (Van Veldhuizen 1999). It is obvious that the smaller the distance, the closer the PF\textsubscript{known} is to the PF\textsubscript{true}. A result GD=0 indicates that PF\textsubscript{known}=PF\textsubscript{true} in terms of location, any other value indicates that PF\textsubscript{known} deviates from the PF\textsubscript{true} (Van Veldhuizen 1999), so zero is a desired value.

**$\varepsilon$-indicator**

The metric $\varepsilon$-indicator (IE) measures ‘the smallest distance that an approximation set [PF\textsubscript{known}] must be translated in order to completely dominate a reference set [PF\textsubscript{true}]’ (Kollat et al. 2008). This metric is adapted from Zitzler
et al. (2003) as:

\[ IE(PF_{true}, PF_{known}) = \max_{z^{1} \in PF_{true}} \min_{z^{2} \in PF_{known}} \max_{i} \frac{z^{1}_i}{z^{2}_i} \]  

(13)

where \( z^{1}_i \) is the value of the \( i \)th objective in \( PF_{known} \), \( z^{2}_i \) is the value of the \( i \)th objective in \( PF_{true} \).

The IE metric adopts values equal or bigger than 1. A result \( IE = 1 \) indicates that \( PF_{known} = PF_{true} \) in terms of location, the bigger IE value represents the larger distance from the \( PF_{true} \), so 1 is a desired value.

**S-metric**

The S-metric (SM) is defined for 2-dimensional objective space (Zitzler & Thiele 1998), an equivalent metric for \( n \)-dimensional objective space is called hypervolume (Fleischer 2003; Zitzler et al. 2005). This metric measures the area or volume, respectively, covered by the \( PF_{known} \) from the worst possible solution (the reference point). The SM is defined as follows:

\[ SM = \bigcup_{i} a_i \]  

(14)

where \( a_i \) is the area determined by the solutions in \( PF_{known} \) and the reference point.

The SM is kept as an absolute value. A \( PF_{known} \) with the largest SM value is closest to the \( PF_{true} \). Therefore, the larger value of SM, which expresses the closer proximity to the \( PF_{true} \), is desirable.

**Extent**

The metric extent (EX) represents the spread or extent of \( PF_{known} \) across the objective space. This metric is a modification of Zitzler’s metric (M3) calculated as (Zitzler 1999; Zitzler et al. 2000):

\[ M_3 = \frac{e_{outer}}{e_{PFtrue}} \]  

(15)

where \( e_{outer} \) is the Euclidean distance between the objective function values of two outer solutions in \( PF_{known} \).

To enable better comparison for the purpose of sensitivity analysis, Equation (15) is modified to express the percentage of the extent of a \( PF_{known} \) in relation to the \( PF_{true} \) as follows:

\[ EX = \frac{e_{outer}}{e_{PFtrue}} \times 100\% \]  

(16)

where \( e_{PFtrue} \) is the Euclidean distance between the objective function values of two outer solutions in the \( PF_{true} \).

The extent of \( PF_{known} \) should be maximised, i.e. for each objective a wide range of values should be covered by the non-dominated solutions (Zitzler et al. 2000), so a wide range of solutions is available to a decision maker. Equally, the greater value of EX expressing the larger the extent of \( PF_{known} \) is desirable. It is worth noting that the extent of a \( PF_{known} > 100\% \) indicates that the \( PF_{known} \) has the larger extent than the \( PF_{true} \) (i.e. \( EX_{PFtrue} = 100\% \)).

**Spacing**

The metric spacing (SC) evaluates the spread or distribution of solutions in \( PF_{known} \) and is adapted from Van Veldhuizen (1999) as follows:

\[ SC = \sqrt{\frac{1}{NN} \sum_{i=1}^{NN} (\bar{e} - e_i)^2} \]  

(17)

where \( e_i \) is the Euclidean distance between the \( i \)th solution and its closest neighbour in \( PF_{known} \), calculated in the objective space; \( \bar{e} \) is the mean of all \( e_i \).

The SC metric is kept as an absolute value according to the original Van Veldhuizen’s (1999) formula. To ensure that the results are not affected by identical solutions in the \( PF_{known} \), only unique solutions were included into SC calculations. The more uniformly distributed solutions within the \( PF_{known} \) the better the \( PF_{known} \) is. A result of \( SC = 0 \) indicates an equidistant spacing between solutions in the \( PF_{known} \), which is an ideal situation.

An overview of the performance metrics, including their expected range and desired values, is shown in Table 3. Some of these metrics require knowledge about the \( PF_{true} \), which is also indicated in Table 3.
Performance and sensitivity evaluation

There are two possible approaches to evaluate performance using the metrics. The first approach examines the results which were directly obtained from the metrics and compares them. This approach provides, on the one hand, very detailed information about performance of a parameter setting combination in each aspect of the quality of the Pareto front. On the other hand, the quantity of results (eight results for each parameter setting combination of 152 total combinations) would cause difficulties in analysing and presenting them. Hence, this approach was not used as it is more suitable for low quantities of results. It was used, for example, by Baran et al. (2005) to compare six multi-objective evolutionary algorithms.

The second approach, which is developed in the following paragraphs and applied for this work, combines the metrics together with the aim to obtain an overall performance for a parameter setting combination. This approach requires, besides setting the performance criteria for each of the metric with the corresponding performance scores, also the assignment of a weighting factor to each metric (Table 4). Based on the performance criteria, the value for each metric is replaced by means of linear interpolation with a corresponding performance score of 0.0–5.0 to provide a common basis for comparison. This step would not be required if all metrics were assessed as normalised or relative values. Weighting factors are then determined to provide a desired level of balance across all metrics according to the user's preference. These weighting factors

Table 3 | Metric overview

<table>
<thead>
<tr>
<th>Metric</th>
<th>Abbreviation</th>
<th>Expected range</th>
<th>Desired value</th>
<th>PFtrue required?</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-dominated number</td>
<td>NN</td>
<td>0–100%</td>
<td>100%</td>
<td>No</td>
<td>Percentage of the population size</td>
</tr>
<tr>
<td>Unique non-dominated number</td>
<td>UN</td>
<td>0–100%</td>
<td>100%</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>True number</td>
<td>TN</td>
<td>0–100%</td>
<td>100%</td>
<td>Yes</td>
<td>Percentage of the total number of non-dominated solutions in PFknown</td>
</tr>
<tr>
<td>Generational distance</td>
<td>GD</td>
<td>≥0</td>
<td>0</td>
<td>Yes</td>
<td>Absolute value</td>
</tr>
<tr>
<td>ε-indicator</td>
<td>IE</td>
<td>≥1</td>
<td>1</td>
<td>Yes (no)*</td>
<td>Absolute value</td>
</tr>
<tr>
<td>S-metric</td>
<td>SM</td>
<td>≥0</td>
<td>&gt;&gt;0</td>
<td>No</td>
<td>Absolute value</td>
</tr>
<tr>
<td>Extent</td>
<td>EX</td>
<td>≥0%</td>
<td>&gt;&gt;0%</td>
<td>Yes (no)**</td>
<td>Percentage of the PFtrue extent</td>
</tr>
<tr>
<td>Spacing</td>
<td>SC</td>
<td>≥0</td>
<td>0</td>
<td>No</td>
<td>Absolute value</td>
</tr>
</tbody>
</table>

*Individual PFknown could be compared without using PFtrue (see Zitzler et al. 2003).
**If an absolute value was used i.e. EX = εouter, PFtrue would not be required.

Table 4 | Metric weighting factors and performance criteria

<table>
<thead>
<tr>
<th>Weighting factor</th>
<th>NN (%)</th>
<th>UN (%)</th>
<th>TN (%)</th>
<th>GD</th>
<th>IE</th>
<th>SM**</th>
<th>EX (%)</th>
<th>SC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual performance</td>
<td>Performance score</td>
<td>0.120</td>
<td>0.050</td>
<td>0.050</td>
<td>0.250</td>
<td>0.250</td>
<td>0.250</td>
<td>0.005</td>
</tr>
<tr>
<td>Very good</td>
<td>(5.0–4.0)</td>
<td>100–80</td>
<td>100–80</td>
<td>100–80</td>
<td>0–50</td>
<td>1–1.05</td>
<td>37.3°–34</td>
<td>313°–80</td>
</tr>
<tr>
<td>Good</td>
<td>(4.0–3.0)</td>
<td>80–60</td>
<td>80–60</td>
<td>80–60</td>
<td>50–100</td>
<td>1.05–1.1</td>
<td>34–31</td>
<td>80–60</td>
</tr>
<tr>
<td>Average</td>
<td>(3.0–2.0)</td>
<td>60–40</td>
<td>60–40</td>
<td>60–40</td>
<td>100–200</td>
<td>1.1–1.3</td>
<td>31–28</td>
<td>60–40</td>
</tr>
<tr>
<td>Poor</td>
<td>(2.0–1.0)</td>
<td>40–20</td>
<td>40–20</td>
<td>40–20</td>
<td>200–500</td>
<td>1.3–1.5</td>
<td>28–25</td>
<td>40–20</td>
</tr>
<tr>
<td>Very poor</td>
<td>(1.0–0.0)</td>
<td>20–0</td>
<td>20–0</td>
<td>20–0</td>
<td>500–12,816°</td>
<td>1.5–3.34°</td>
<td>25–21.3°</td>
<td>20–0</td>
</tr>
</tbody>
</table>

*Maximum or minimum values obtained.
**SM Metric performance criteria × 10°.
(Table 4) combine the individual performance scores of metrics into a combined performance score of all metrics (Table 5). A disadvantage of this approach is in the subjectivity of weighting factors. However, the clear advantage is the significantly more succinct set of results obtained which are more easily interpreted and presented.

The development of the metric performance criteria and weighting factors was undertaken by a trial-and-error approach. Initially, performance criteria for all metrics were distributed evenly with the same weighting factors assigned so all factors summed to 1.0. A process was then undertaken whereby the overall performance was assessed against a visual comparison where the proximity of the $P^*_\text{known}$ to the $P^*_\text{true}$ was considered of main importance. Basically, the visual comparison served as a means of ‘calibration’ of metric performance criteria and weighting factors, so the combined performance scores returned (see Table 5) were (more or less) in alignment with the location of the $P^*_\text{known}$ from the $P^*_\text{true}$. For that reason, metrics GD, IE and SM were assigned a higher weighting factor than the other metrics reflecting their higher importance. Similarly, metric TN was assigned a lower weighting factor because the bulk of the results (mainly SA1) for TN metric did not score well due to the lower population size applied. Fine tuning the performance criteria and weighting factors was an iterative task requiring careful considerations. Furthermore, these considerations will vary for different problems.

The following paragraphs illustrate a practical example of establishing the sensitivity of a particular parameter of interest using the performance metrics. Firstly, the metrics for the parameter setting combination are calculated and using linear interpolation, assigned a performance score from 0.0 to 5.0 (as per Table 4). For example, the results $NN = 100\%$, $UN = 94\%$, $TN = 5\%$, $GD = 117$, $IE = 1.06$, $SM = 37.3 \times 10^6$, $EX = 99\%$, $SC = 2,319$ are replaced with performance scores as $NN = 5.0$, $UN = 4.7$, $TN = 0.3$, $GD = 2.8$, $IE = 3.8$, $SM = 5.0$, $EX = 4.1$, $SC = 1.0$. Secondly, the combined performance score, which can adopt values from 0.0 to 5.0, is established as the sum of individual metric performance scores added together after each of them was multiplied by a weighting factor. Accordingly, the combined performance score is obtained as $5.0 \times 0.120 + 4.7 \times 0.050 + 0.3 \times 0.050 + 2.8 \times 0.250 + 3.8 \times 0.250 + 5.0 \times 0.250 + 4.1 \times 0.005 + 1.0 \times 0.025 = 3.8$, with the overall performance of the parameter setting combination ‘good’ (as per Table 5).

Finally, an additional step is undertaken to obtain the sensitivity of a particular parameter of interest. This involves grouping together parameter setting combinations, in which the parameter of interest is systematically changing (i.e. incrementally increasing) while the other parameters are held constant. These groups are further referred to as sensitivity groups (SEG). The sensitivity of the parameter of interest is then expressed in sensitivity points (SPs), which is the difference between the maximum and minimum combined performance scores within the SEG. For example, combined performance scores in the SEG ranging from 0.6 to 3.9 give 3.3 SPs (i.e. 3.9–0.6 = 3.3), indicating the high sensitivity of the parameter. Note that the maximum possible SPs $= 5.0$ (as per Table 5). Conversely, combined performance scores in the SEG ranging from 2.8 to 3.2 give 0.4 SPs, indicating low sensitivity of the parameter.

The methodology for performance and sensitivity analysis of algorithm parameters is summarised as a flowchart in Figure 3.

### RESULTS

All analyses were conducted using Intel Core i3 CPU M 380 processor with RAM 2.0 GB. Due to limited computer capacity for required analyses, about 50,000 function evaluations only were able to be run at once. Whereas the number of function evaluations required for one analysis ranged from 25,000 (i.e. population size of 50 for 500 generations) up to 800,000 (i.e. population size of 400 for 2,000 generations). Advantageously, the GANetXL’s capability to conduct one analysis in a sequence of interrupted
runs was used, where the previous run is resumed using the ‘resume’ function in the application’s toolbar (CWS 2011).

The results for all 152 analyses conducted are displayed in Figure 4. Note that only feasible solutions are shown in Figure 4 as only feasible solutions were obtained from the optimisation model (6) subject to constraints (3)–(5). SA1 (solid grey squares in Figure 4), which tested the sensitivity of mutation and crossovers for a constant value of population size of 100 and the number of generations of 2000, includes 80 Pareto fronts (i.e. PF\textsubscript{knowns}). SA2 (solid black triangles in Figure 4), which tested the sensitivity of the population size and number of generations for a constant value of mutation of 0.005 and crossover of 0.7 SXP and 1.0 S1P, includes 72 Pareto fronts (i.e. PF\textsubscript{knowns}). It can be seen that SA1 (solid grey squares) consist of two isolated clouds of Pareto fronts. This is caused by a particular value of a mutation, which is discussed further in the Mutation section.
As previously described, PF\textsubscript{true} is the union of the best solutions found in all analyses conducted. To be more exact, 152 Pareto fronts (i.e. PF\textsubscript{knowns}) from both SA1 and SA2 were joined together (solid grey dots in Figure 5) and the optimal solutions identified (solid black dots as PF\textsubscript{true} in Figure 5). Because some of these solutions were identical (i.e. had the same pump schedules), they were eliminated from the PF\textsubscript{true}. Those identical solutions were mainly gained from the same combinations of NSGA-II parameter settings, but different objective function scaling. The resulting PF\textsubscript{true}, which was used for subsequent calculations of some of the metrics, consists of a total of 333 non-dominated and unique solutions.

Some interesting observations were made regarding the PF\textsubscript{true} (Figure 6, Table 6), which relate to the NSGA-II parameter setting combinations and population sizes creating the PF\textsubscript{true}. The first observation was that the PF\textsubscript{true} was reached by several NSGA-II parameter setting combinations, more specifically 13 combinations. Interestingly, five of those combinations (i.e. 38\%) represented 91\% of all solutions in PF\textsubscript{true}. Conversely, eight combinations (i.e. 62\%) represented only 9\% of solutions in PF\textsubscript{true}. So, the minority of the parameter setting combinations, which reached the PF\textsubscript{true}, formed the vast majority of all PF\textsubscript{true} solutions and vice versa. The second observation was that as the solutions within the PF\textsubscript{true} progress from the top left to bottom right, they are derived from (more or less) dedicated sections to a particular and increasing population size. The last observation may indicate that population size performs inconsistently across a search space for this particular problem, which is discussed in the Discussion section.

**Objective function scaling**

The results for the sensitivity of objective function scaling (Figure 7) include two sets of 76 Pareto fronts (i.e. PF\textsubscript{knowns}) each gained from the same NSGA-II parameter setting combinations.
combinations, with the difference that only one set had its objective functions scaled. Figure 7 indicates that there may be a considerable number of identical solutions in the two Pareto front sets, because black dots representing scaled objective functions appear to lie on top of grey crosses representing non-scaled objective functions. Comparing individual solutions from the two Pareto front sets confirmed that about 70% of them are indeed identical, resulting in the majority of Pareto fronts (64 of 76) from the two sets being also identical. Accordingly, 64 NSGA-II parameter setting combinations (84%) delivered the same Pareto fronts regardless of the objective functions being or not being scaled. Whereas only 12 NSGA-II parameter setting combinations (16%) delivered varying Pareto fronts for scaled and non-scaled objective functions.

Differences in those two sets of 12 Pareto fronts were each examined using metrics. Results from the SC metric showed that uniformity of spacing between solutions in the Pareto fronts with scaled objective functions improved in five instances and deteriorated in seven instances. Consequently, it was not confirmed that adequate scaling of the objective functions assures more uniformly distributed solutions, for the optimisation model and network used. Similarly, the improvement in quality of Pareto fronts for scaled objective functions regarding other performance metrics was non-conclusive, as some of those metrics gave better results and some worse, with the overall performance being the same comparing scaled and non-scaled objective functions. An exception was the EX metric, which for scaled objective functions improved in nine instances and deteriorated in three instances only, so scaled objective functions generally delivered Pareto fronts with larger extents.

Considering the bias of the NSGA-II towards the water quality objective when objective functions are inadequately scaled, the results for non-scaled and scaled objective functions were compared in terms of concentrations of solutions along the Pareto front and the location of the Pareto front. It can be seen from Figure 7 that the concentration of solutions for minimum pump energy cost values improved (for example ellipses A and B in Figure 7) for scaled objective functions. Moreover, scaled objective functions tended to find better solutions for minimum water quality values (for example ellipse E in Figure 7), with an example provided in Table 7. It was thus confirmed that non-scaled (i.e. inadequately scaled) objective functions caused the NSGA-II bias towards the water quality objective for the optimisation model and network used. However, the discontinuity of Pareto fronts is still obvious (for example ellipses C and D in Figure 7), with the reasons discussed in the Discussion section.

**Table 6** | NSGA-II parameter setting combinations which reached the PFtrue

<table>
<thead>
<tr>
<th>NSGA-II parameter setting combination</th>
<th>Solutions in PFtrue</th>
</tr>
</thead>
<tbody>
<tr>
<td>POP</td>
<td>GEN</td>
</tr>
<tr>
<td>50</td>
<td>2,000</td>
</tr>
<tr>
<td>100</td>
<td>2,000</td>
</tr>
<tr>
<td>100</td>
<td>2,000</td>
</tr>
<tr>
<td>100</td>
<td>2,000</td>
</tr>
<tr>
<td>150</td>
<td>2,000</td>
</tr>
<tr>
<td>150</td>
<td>1,000</td>
</tr>
<tr>
<td>200</td>
<td>2,000</td>
</tr>
<tr>
<td>300</td>
<td>2,000</td>
</tr>
<tr>
<td>300</td>
<td>2,000</td>
</tr>
<tr>
<td>300</td>
<td>1,000</td>
</tr>
<tr>
<td>300</td>
<td>1,000</td>
</tr>
<tr>
<td>400</td>
<td>2,000</td>
</tr>
<tr>
<td>400</td>
<td>1,000</td>
</tr>
<tr>
<td>Total</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 7** | Pareto fronts for sensitivity of objective function scaling.
Because the two sets of results for non-scaled and scaled objective functions were very similar, with scaled objective functions proved to deliver slightly better results, the subsequent sensitivity analysis of NSGA-II parameters was undertaken using results with scaled objective functions only.

The graphical results for exploring the sensitivity of mutation, crossover (both types S1P and SXP), population size and the number of generations are displayed in Figures 8–12. In those figures, each set of results representing a single parameter value (mutation value of 0.002 as white triangles in Figure 8, or crossover value of 0.6 as solid black squares in Figure 9, and so on) exist as a ‘cloud’ rather than a single front. The reason for this is that a single parameter value was always combined with other parameters, some as constants, but some as a range.

### NSGA-II parameters

Table 7 | Example of minimum solution for economic and water quality objective

<table>
<thead>
<tr>
<th>Analysis ID</th>
<th>POP</th>
<th>GEN</th>
<th>MUT</th>
<th>CRS</th>
<th>Objective functions scaled</th>
<th>Economic objective</th>
<th>Water quality objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>104</td>
<td>100</td>
<td>2,000</td>
<td>0.005</td>
<td>1.0 S1P</td>
<td>No</td>
<td>148.7</td>
<td>98,149.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>204.8</td>
<td>1,815.9</td>
</tr>
<tr>
<td>140</td>
<td>100</td>
<td>2,000</td>
<td>0.005</td>
<td>1.0 S1P</td>
<td>Yes</td>
<td>148.7</td>
<td>98,149.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>204.5</td>
<td>1,521.8*</td>
</tr>
</tbody>
</table>

POP = population size, GEN = number of generations, MUT = mutation, CRS = crossover.

*Note that the water quality objective improved when scaled.

---

**Table 7 | Example of minimum solution for economic and water quality objective**

**Figure 8 | Pareto fronts for sensitivity of mutation (POP 100, GEN 2,000, CRS 0.6-1.0 both S1P and SXP).**

**Figure 9 | Pareto fronts for sensitivity of S1P crossover (POP 100, GEN 2,000, MUT 0.002-0.1).**

**Figure 10 | Pareto fronts for sensitivity of SXP crossover (POP 100, GEN 2,000, MUT 0.002-0.1).**
as noted in the figure titles. For example, the data set for the aforementioned mutation value of 0.002 (white triangles in Figure 8) includes results for population size of 100, the number of generations of 2,000 and five crossover values from 0.6 to 1.0 both S1P and SXP, so this data set contains 10 Pareto fronts.

Mutation

The observation gleaned from Figure 8 is that the best and most competitive results were obtained from the mutation values of 0.005 and 0.01. The mutation value of 0.002 gave poorer results and the value of 0.1 provided Pareto fronts very distant from the optimal solutions, creating an isolated cloud. This indicates that the mutation value of 0.1 is too high for this particular problem, which caused the search to degenerate into a random process as previously described by Savic & Walters (1997). In contrast, the mutation value of 0.002 is too low for this particular problem, which caused the premature convergence of the algorithm to a local optimum as previously described by Srinivas & Patnaik (1994).

The sensitivity of mutation was investigated using metrics. Table 8 lists two sample SEGs from 10 available SEGs for the mutation. The performance of mutation varies from ‘very poor’ to ‘good’ for the both SEGs, with the corresponding 3.3 and 2.9 SPs, respectively. Because of the wide range of scores across all SEGs with the average of 2.9 SPs, the sensitivity of mutation is high for the optimisation model and network used. This means that the selection of a suitable mutation value to achieve good results requires careful attention.

Crossover

It can be observed from Figures 9 and 10 that particular crossover values are very widely spread and, unlike mutation, are not formed into designated performance areas. This may not make the identification of the best performing crossover value as straightforward as for the mutation. Indeed, the metrics with the following sensitivity analysis confirmed the low sensitivity to different crossover values, meaning that the change of crossover setting does not affect the final solution significantly.

Table 9 presents two sample SEGs from eight available SEGs for the crossover. The performance of crossover varies from ‘average’ to ‘good’ for the first SEG and is ‘very poor’ across the second SEG, with the corresponding 0.9 SPs and 0.4 SPs, respectively. Because the range of scores across all SEGs is narrow with the average of 0.7 SPs, the sensitivity of crossover is low for the optimisation model and network used. The selection of a suitable crossover value for this particular problem is still important to achieve good results, yet it does not seem as crucial as the identification of a suitable mutation value.
Population size

The performance of different population sizes (Figures 6 and 11) was discussed in relation to the PFtrue in previous sections, so only sensitivity is examined here. Table 10 lists two sample SEGs from six available SEGs for the population size. The performance of population size varies from ‘average’ to ‘very good’ for the first SEG and is ‘good’ across the second SEG, with the corresponding 1.4 SPs and 0.7 SPs, respectively. Because of the intermediate range of scores across all SEGs with the average of 1.2 SPs, the sensitivity of population size is moderate for the optimisation model and network used. The selection of a suitable population size to achieve good results needs careful attention, which is emphasised by the finding that a particular population size appears to perform better at some locations within the search space and worse in others as demonstrated by the PFtrue (Figure 6). This inconsistent performance is discussed in the Discussion section.

Number of generations

The first observation gleaned from Figure 12 is that generally, the results show steady improvement with the increasing...
number of generations. Hence, the best results were obtained from the number of generations of 2,000, which was confirmed by the PF\text{true}, where 319 (i.e. 96\%) of total 333 solutions in the PF\text{true} were gained from this number of generations.

The sensitivity of the number of generations was assessed using metrics. Table 11 lists two sample SEGs from 12 available SEGs for the number of generations. The performance of the number of generations varies from ‘average’ to ‘good’ for the first SEG and is ‘good’ across the second SEG, with the corresponding 0.4 SPs and 0.8 SPs, respectively. Because the range of scores across all SEGs is narrow with the average of 0.7 SPs, the sensitivity of number of generations is low for the optimisation model and network used. In spite of the steady improvements in results with increased number of generations, there are additional associated computational costs which may be considered to negate the perceived benefits.

### DISCUSSION

#### Performance metrics and their application

The UN metric (9) reports the percentage of unique non-dominated solutions in a PF\text{known} (i.e. solutions with

---

**Table 10** | Sensitivity of population size (two sample SEGs)

<table>
<thead>
<tr>
<th>ID</th>
<th>Population size</th>
<th>Constant parameters</th>
<th>NN (%)</th>
<th>UN (%)</th>
<th>TN (%)</th>
<th>GD</th>
<th>IE</th>
<th>SM (&lt;10^6)</th>
<th>EX (%)</th>
<th>SC</th>
<th>Combined performance score</th>
<th>Overall performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>118</td>
<td>50</td>
<td>GEN: 1,000 MUT: 0.005 CRS: 0.7 SXP</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>202</td>
<td>1.13</td>
<td>32.1</td>
<td>83</td>
<td>4,119</td>
<td>2.8</td>
<td>Average</td>
</tr>
<tr>
<td>121</td>
<td>100</td>
<td>GEN: 2,000 MUT: 0.005 CRS: 1.0 S1P</td>
<td>100</td>
<td>98</td>
<td>2</td>
<td>67</td>
<td>1.10</td>
<td>31.8</td>
<td>55</td>
<td>2,474</td>
<td>3.2</td>
<td>Good</td>
</tr>
<tr>
<td>124</td>
<td>150</td>
<td>GEN: 1,000 MUT: 0.005 CRS: 0.7 SXP</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>23</td>
<td>1.08</td>
<td>30.6</td>
<td>22</td>
<td>927</td>
<td>3.2</td>
<td>Good</td>
</tr>
<tr>
<td>127</td>
<td>200</td>
<td>GEN: 2,000 MUT: 0.005 CRS: 1.0 S1P</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>28</td>
<td>1.07</td>
<td>30.8</td>
<td>22</td>
<td>526</td>
<td>3.2</td>
<td>Good</td>
</tr>
<tr>
<td>130</td>
<td>300</td>
<td>GEN: 1,000 MUT: 0.005 CRS: 0.7 SXP</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>29</td>
<td>1.11</td>
<td>31.3</td>
<td>48</td>
<td>988</td>
<td>3.5</td>
<td>Good</td>
</tr>
<tr>
<td>133</td>
<td>400</td>
<td>GEN: 2,000 MUT: 0.005 CRS: 1.0 S1P</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>24</td>
<td>1.13</td>
<td>33.7</td>
<td>38</td>
<td>1,139</td>
<td>3.3</td>
<td>Good</td>
</tr>
</tbody>
</table>

**Table 11** | Sensitivity of the number of generations (two sample SEGs)

<table>
<thead>
<tr>
<th>ID</th>
<th>Number of generations</th>
<th>Constant parameters</th>
<th>NN (%)</th>
<th>UN (%)</th>
<th>TN (%)</th>
<th>GD</th>
<th>IE</th>
<th>SM (&lt;10^6)</th>
<th>EX (%)</th>
<th>SC</th>
<th>Combined performance score</th>
<th>Overall performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>120</td>
<td>500</td>
<td>POP: 100 MUT: 0.005 CRS: 0.7 SXP</td>
<td>100</td>
<td>71</td>
<td>0</td>
<td>72</td>
<td>1.14</td>
<td>29.5</td>
<td>52</td>
<td>2,764</td>
<td>2.8</td>
<td>Average</td>
</tr>
<tr>
<td>121</td>
<td>1,000</td>
<td>POP: 300 MUT: 0.005 CRS: 1.0 S1P</td>
<td>100</td>
<td>33</td>
<td>0</td>
<td>43</td>
<td>1.14</td>
<td>32.5</td>
<td>40</td>
<td>675</td>
<td>3.1</td>
<td>Good</td>
</tr>
<tr>
<td>122</td>
<td>2,000</td>
<td>POP: 100 MUT: 0.005 CRS: 0.7 SXP</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>28</td>
<td>1.07</td>
<td>30.8</td>
<td>22</td>
<td>526</td>
<td>3.2</td>
<td>Good</td>
</tr>
<tr>
<td>147</td>
<td>500</td>
<td>POP: 300 MUT: 0.005 CRS: 1.0 S1P</td>
<td>100</td>
<td>33</td>
<td>0</td>
<td>43</td>
<td>1.14</td>
<td>32.5</td>
<td>40</td>
<td>675</td>
<td>3.1</td>
<td>Good</td>
</tr>
<tr>
<td>148</td>
<td>1,000</td>
<td>POP: 100 MUT: 0.005 CRS: 0.7 SXP</td>
<td>100</td>
<td>71</td>
<td>0</td>
<td>72</td>
<td>1.14</td>
<td>29.5</td>
<td>52</td>
<td>2,764</td>
<td>2.8</td>
<td>Average</td>
</tr>
<tr>
<td>149</td>
<td>2,000</td>
<td>POP: 300 MUT: 0.005 CRS: 1.0 S1P</td>
<td>100</td>
<td>33</td>
<td>0</td>
<td>43</td>
<td>1.14</td>
<td>32.5</td>
<td>40</td>
<td>675</td>
<td>3.1</td>
<td>Good</td>
</tr>
</tbody>
</table>

**GEN** – number of generations, MUT – mutation, CRS – crossover.

**POP** – population size, MUT – mutation, CRS – crossover.
unique pump schedules). There were some solutions found which had different pump schedules, yet objective function values for both objectives were nearly the same, for example, only differing in the third decimal place for one objective. Note that solutions with the same objective function values, but different decision variables (i.e. pump schedules) are called alternate solutions (Bhattacharjya 2003). In this work, the formerly described solutions were considered ‘unique’ for the following reason. If a solution has desirable objective function values but is impractical, the decision maker would probably be interested in a different solution with similar objective function values.

There is a limitation to the GD metric (12), which measures the distance between a PFknown and the PFtrue. This metric may give misleading results if the length (i.e. extent) of the PFknown is significantly greater than the extent of the PFtrue as demonstrated in Figure 13. Despite the proximity of PFknownA and PFknownB to the PFtrue being the same, the GD metric returns larger value for the PFknownA due to inclusion of the points A1–A4 in the equation, calculating their distance from the outer PFtrue point T1. This limitation was recognised but was considered adequate for the purpose of the sensitivity analysis. This limitation is overcome by using the ε-indicator metric.

In addition, both the GD metric and ε-indicator metric require knowledge about the PFtrue, which is not normally available. The metric with a similar function, which measures the closeness of a PFknown to the PFtrue, yet does not require the PFtrue information, is S-metric for 2-dimensional objective space (Zitzler & Thiele 1998) or hypervolume for n-dimensional objective space (Fleischer 2003; Zitzler et al. 2003). The disadvantage of this metric is, however, that the choice of the reference point can influence the evaluation of the PFknowns (Knowles & Corne 2002).

The methodology for performance and sensitivity analysis using metrics may be applied for the calibration of multi-objective algorithm parameters for any WDS optimisation problem. It needs to be noted, nevertheless, that the metric performance criteria, performance scores and weighting factors (see Table 4) were specifically developed for the purpose of this work and, that these parameters need to be developed and fine-tuned to suit the particular problem. Furthermore, the selection of the metrics and their performance evaluation also depends on the purpose for their application. In the case of calibration, the metrics GD, IE, SM and TN would likely adopt the higher weighting factors than the other metrics. For applications comprising more than two objectives, metrics can easily be adjusted to the required n-dimensional format.

The disadvantage of the methodology used is that a large computational effort is required. This effort may be partially reduced by using only metrics, which do not require knowledge about the PFtrue (refer to Table 3). Alternatively, algorithms with adaptive parameter settings, such as Borg multi-objective evolutionary algorithm (Hadka & Reed 2015) or algorithms with few parameters, such as Pareo archived dynamically dimensioned search (PA-DDS) with only one parameter (Asadzadeh & Tolson 2011) could be considered. It is expected that these algorithms may well require less computational effort to calibrate their parameters.

**Objective function scaling**

In spite of the adequate scaling of objective functions, the Pareto fronts obtained were discontinuous, which is possibly due to the following:

1. Discrete decision variables.
2. Model simplifications, for example, water quality analysis (the length of time step, i.e. 1 hour rather than 1–10 minutes, the length of simulation period, i.e. 1 day rather than 1 week or more).
3. Only a single run was performed from one initial population of random solutions.

![Figure 13 | GD metric limitation.](https://iwaponline.com/jh/article-pdf/17/6/891/388672/jh0170891.pdf)
NSGA-II parameters

Parameter sensitivity

Performance metrics enabled to evaluate the sensitivity of the NSGA-II parameters on a quantitative and more objective basis than the use of visual comparison only. In particular, the sensitivity of NSGA-II parameters was expressed as SPs of maximum possible SPs, which assisted in understanding of sensitivity hierarchy (Table 12). The most sensitive parameter was identified mutation and the least sensitive crossover, with the limitation to the optimisation model and network used. Population size was more sensitive than the number of generations for this problem. Note that the more sensitive parameter is, its change has greater influence on results.

A comment needs to be made though that the identification of reasonable range of settings for each parameter was important in this exercise. Authors acknowledge that the value for the mutation of 0.1 may be a little too high and if this value was not used, the sensitivity of mutation would not be evaluated as high. However, quite a wide range of settings for each parameter was identified purposely so the impact of those settings on the final results could be clearly demonstrated.

Parameter interdependency

The performance of a mutation in relation to crossover and vice versa is now examined by exploring charts detailing concrete mutation and crossover values, and crossover types. Figure 14 indicates that an increase in mutation from 0.005 to 0.01 using the crossover of 0.6 improved the solution (arrow 1a in Figure 14); conversely, the solution deteriorated for the same increase in mutation, but crossover of 1.0 (arrow 1b in Figure 14). Vice versa, when crossover increased from 0.6 to 1.0 using mutation of 0.005, the solution improved (arrow 2a in Figure 14); conversely, the solution deteriorated for the same increase in crossover, but mutation of 0.01 (arrow 2b in Figure 14). Figure 15 displays Pareto fronts for varying mutation and crossover values, and crossover types. It demonstrates that the change of crossover type from S1P to SXP for crossover value of 0.6 and mutation value of 0.005 resulted in better results (arrow 3a in Figure 15); and vice versa, change of crossover type from S1P to SXP for crossover value of 1.0 and mutation value of 0.01 resulted in worse results (arrow 3b in Figure 15). The likely conclusion drawn from those observations is that the performance of a particular mutation depends on the crossover value and crossover

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Average</th>
<th>Median</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Parameter sensitivity</th>
<th>Number of sensitivity groups (SEGs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mutation</td>
<td>2.9</td>
<td>2.9</td>
<td>2.7</td>
<td>3.3</td>
<td>High</td>
<td>10</td>
</tr>
<tr>
<td>Population size</td>
<td>1.2</td>
<td>1.3</td>
<td>0.6</td>
<td>1.8</td>
<td>Moderate</td>
<td>6</td>
</tr>
<tr>
<td>Number of generations</td>
<td>0.7</td>
<td>0.8</td>
<td>0.0</td>
<td>2.1</td>
<td>Low</td>
<td>12</td>
</tr>
<tr>
<td>Crossover</td>
<td>0.7</td>
<td>0.7</td>
<td>0.2</td>
<td>1.4</td>
<td>Low</td>
<td>8</td>
</tr>
</tbody>
</table>

*Difference between the maximum and minimum combined performance scores within SEGs.

Figure 14 | Pareto fronts for varying mutation and crossover values (POP 100, GEN 2,000).
type, and the performance of a particular crossover value and its type is probably dependent upon the mutation value. In other words, it appears that mutation and crossover are somehow interdependent. If this is the case, they ought to be evaluated as an interdependent pair rather than in isolation. More research is needed in this area. Plotting the values of computed metrics for different combinations of crossover and mutation operators on a surface or contour chart is suggested as one way of visualizing the combined effects of these two parameters.

Performance inconsistency

Due to Pareto fronts for different parameter setting combinations being non parallel often intersecting (Figures 14–16), it indicates that some parameter setting combinations may perform better at some locations of the search space and worse in others. At Figures 14 and 15, for example, mutation value of 0.005 and crossover value of 0.6 S1P (solid grey dots) perform well towards minimum water quality values, but quite irregularly diverge from minimum pump energy costs values. Another example is in Figure 16, where all population sizes were able to capture (more or less) minimum water quality values, but their level of performance (or accuracy) over the large search space differs greatly. Hence, (certain) parameter setting combinations seem to perform with various degrees of inconsistency across the search space. This inconsistent performance was also demonstrated by the PFtrue (Figure 6), which was not reached by one but several parameter setting combinations. This is thought to reflect the complexity of the search space, and requires further research.

Unlike single-objective approaches, where the calibration process results in the ‘best’ algorithm parameter values to find the optimal solution as demonstrated in Younis & Dandy (2012), the identification of those ‘best’ values in multi-objective approach does not seem as straightforward. On the contrary, it appears that the multi-objective approach requires a range of parameter values (i.e. a range of parameter setting combinations) to reach or identify all optimal solutions within the (true) Pareto front. This may be caused by the fact that the single-objective approach seeks one particular solution, whereas the multi-objective approach searches for an entire front of optimal solutions.

Limitations

There are three limitations to this work. The first limitation is that, due to the large number of parameter setting combinations being evaluated, only a single run of the NSGA-II was performed for each parameter setting combination. This approach is contrary to the common application of genetic algorithms (GAs) where multiple runs are used because of
the stochastic nature of GAs. The reader, therefore, needs to keep in mind that the analyses of the generated results and conclusions are limited to a single run only (i.e. are not based on statistical assessment of multiple runs). The second limitation is that the number of parameter setting combinations to be evaluated needs to be limited (i.e. based on literature values and/or preliminary testing), otherwise time required to run analyses may become unacceptable. The third limitation is that one example network was used only. These limitations need to be addressed in future research.

CONCLUSION

In this work, performance metrics were applied to comprehensively analyse the sensitivity of NSGA-II parameters and their settings and objective function scaling for multi-objective optimisation problem in WDSs. Algorithm parameters considered included population size, number of generations, crossover and mutation. A two-objective optimisation model involved the competing operational objectives of pump energy costs and water quality, and was applied to a test network adopted from the literature representing a regional non-drinking WDS.

Performance metrics enabled the evaluation of the sensitivity of the NSGA-II parameters on a quantitative and more objective basis than the use of visual comparison only. Moreover, they assisted in determining the sensitivity hierarchy of NSGA-II parameters for the particular optimisation model and network used. Performance metrics were thus demonstrated suitable for sensitivity analysis of NSGA-II parameters for multi-objective optimisation problems in WDSs. Additionally, the methodology for performance and sensitivity analysis may be used by researchers and practitioners, after converting metrics to the adequate n-dimensional objective space, for the calibration of multi-objective algorithm parameters for any WDS optimisation problem. In order to repeat a process, it is summarised in the form of a flowchart in Figure 3.

Adequate scaling of the objective functions to cover similar ranges of numerical values was confirmed important for the optimisation model and network used, because inadequately scaled objective functions caused the NSGA-II to bias towards the water quality objective. However, adequately scaled objective functions did not assure continuity of Pareto fronts. In fact, the discontinuity of Pareto fronts is likely attributed to other factors, for example discrete nature of the problem.

NSGA-II parameters were found sensitive for the optimisation model and the network used. Performance of the algorithm and the quality of the solution was thus dependent on parameter settings. Furthermore, some parameters (i.e. mutation and crossover) appeared interdependent and some parameter setting combinations performing with various degrees of inconsistency across the search space. It appears that unlike a single-objective approach, a multi-objective approach requires a range of parameter values (i.e. a range of parameter setting combinations) to cover the entire search space and to reach or identify all optimal solutions. However, additional future research is required in this area to confirm the results and conclusions of this work by the inclusion of different seed values into performance and sensitivity analysis of algorithm parameters, and using different applications in multi-objective optimisation problems of WDSs.

ACKNOWLEDGEMENT

This research was supported by the Australian Research Council as Project LP0990908.

REFERENCES


Babayan, A. V., Savic, D. A. & Walters, G. A. 2007 Multiobjective optimisation of water distribution system design under uncertain demand and pipe roughness. In: Topics on System


First received 13 May 2014; accepted in revised form 15 June 2015. Available online 20 July 2015