Zheng, F.; Simpson, A.R.; Zecchin, A.C.

An efficient hybrid approach for multiobjective optimization of water distribution systems, *Water Resources Research*, 2014; 50(5):3650-3671.

DOI: 10.1002/2013WR014143

AGU PUBLICATIONS

Water Resources Research

RESEARCH ARTICLE
10.1002/2013WR014143

Key Points:
- Decomposition is effective in facilitating the multiobjective network design
- Mapping multiobjective problem by many single-objective problems is effective
- Coupled NLP and SAMODE significantly improves the efficiency

Correspondence to:
F. Zheng,
feifei.zheng@adelaide.edu.au

Citation:
Zheng, F., A. R. Simpson, and A. C. Zecchin (2014), An efficient hybrid approach for multiobjective optimization of water distribution systems, *Water Resour. Res.*, *50*, 3650–3671, doi:10.1002/2013WR014143.

Received 18 MAY 2013
Accepted 12 APR 2014
Accepted article online 17 APR 2014
Published online 5 MAY 2014

An efficient hybrid approach for multiobjective optimization of water distribution systems

Feifei Zheng[1], Angus R. Simpson[1], and Aaron C. Zecchin[1]

[1]School of Civil, Environmental and Mining Engineering, University of Adelaide, Adelaide, South Australia, Australia

Abstract An efficient hybrid approach for the design of water distribution systems (WDSs) with multiple objectives is described in this paper. The objectives are the minimization of the network cost and maximization of the network resilience. A self-adaptive multiobjective differential evolution (SAMODE) algorithm has been developed, in which control parameters are automatically adapted by means of evolution instead of the presetting of fine-tuned parameter values. In the proposed method, a graph algorithm is first used to decompose a looped WDS into a shortest-distance tree (T) or forest, and chords (Ω). The original two-objective optimization problem is then approximated by a series of single-objective optimization problems of the T to be solved by nonlinear programming (NLP), thereby providing an approximate Pareto optimal front for the original whole network. Finally, the solutions at the approximate front are used to seed the SAMODE algorithm to find an improved front for the original entire network. The proposed approach is compared with two other conventional full-search optimization methods (the SAMODE algorithm and the NSGA-II) that seed the initial population with purely random solutions based on three case studies: a benchmark network and two real-world networks with multiple demand loading cases. Results show that (i) the proposed NLP-SAMODE method consistently generates better-quality Pareto fronts than the full-search methods with significantly improved efficiency; and (ii) the proposed SAMODE algorithm (no parameter tuning) exhibits better performance than the NSGA-II with calibrated parameter values in efficiently offering optimal fronts.

1. Introduction

Multiobjective optimization methods are able to explore the trade-off between conflicting objectives within a water distribution system (WDS) design, including such as the construction cost of WDS, the network redundancy, and the water quality. A set of optimally competitive solutions is generated during multiobjective optimization. These optimal solutions form a front in the objective space, normally referred to as the *Pareto optimal front*, providing more design alternatives for practicing engineers.

Multiobjective optimization is not new, and various researchers have contributed to both the theory and the practice in the context of water systems. *Prasad and Park* [2004], for example, proposed a multiobjective genetic algorithm to optimize WDSs, with two objectives considered were the minimization of network costs and the maximization of the reliability index. *Khu and Keedwell* [2005] employed a multiobjective genetic algorithm to upgrade a WDS. The minimization of the upgrading cost and the minimization of the maximum pressure deficit were the objectives of their study.

Farmani et al. [2005] applied three multiobjective evolutionary algorithms (MOEAs) to optimize water network design, and concluded that the SPEA performed slightly better than MOGA and NSGA-II in terms of the quality of the Pareto front. *Reddy and Kumar* [2007] proposed a multiobjective differential evolution (MODE) algorithm to optimize reservoir system problems, and *Perelman et al*. [2008] developed a cross-entropy multiobjective optimization method for WDS design. *Fu and Kapelan* [2011] presented a fuzzy probabilistic approach for optimal design and rehabilitation of WDSs. Two objectives were considered in their work including the minimization of total design cost and the maximization of system performance measured by fuzzy random reliability.

More recently, *Ostfeld* [2012] proposed a new method to incorporate reliability into the design of WDSs by virtually increasing demand loadings (i.e., setting higher demands than the normal design). *Wu et al*. [2013] considered three objectives for the optimization of WDSs, including the traditional objectives of minimizing economic cost and maximizing hydraulic reliability, as well as the recently proposed objective of minimizing greenhouse gas (GHG) emissions.

One significant challenge in the use of multiobjective evolutionary algorithms for the design of real-world WDSs, however, is the enormous computational overhead [*Ostfeld*, 2012]. That is the result of (i) the need for a hydraulic simulation model to evaluate each solution, and (ii) the fact that a nondominant sorting algorithm is normally required by evolutionary algorithm-based multiobjective optimization techniques. It is difficult, if not impossible, for existing multiobjective evolutionary algorithms to find near-optimal Pareto fronts for large and complex real-world WDSs using a computational budget that is typically available in practice [*Fu et al*., 2012]. Improving computational efficiency is, therefore, highly desirable when applying multiobjective evolutionary algorithms to the optimization design of real-world WDSs.

Some attempts have been therefore made in order to reduce the computational effort. *Broad et al*. [2004], for example, proposed a metamodel approach to deal with WDS design taking into account water quality and reliability. In their work, an Artificial Neural Network (ANN) model was used as a surrogate of the hydraulic simulation model to reduce the computational overhead. *di Pierro et al*. [2009] developed two hybrid algorithms, i.e., PArEGO and LEMMO, to tackle multiobjective design problems of WDSs and reported greater efficiency in two WDS case studies in which they compared the performance of the hybrid algorithms with other standard multiobjective evolutionary algorithms.

Fu et al. [2012] recently used global sensitivity analysis to reduce the complexity of a multiobjective WDS design. In their method, a global sensitivity analysis was first performed to calculate the sensitivity indices for each decision variable. A user specified threshold of sensitivity was then used to determine the most sensitive decision variable, which was optimized using NSGA-II. Pareto optimal solutions were obtained for the simpler system in which only the most sensitive decision variables were considered. Finally, the original problem was solved using ε-NSGAII with the obtained Pareto solutions fed into the initial population.

The studies mentioned above have made significant contributions for research and knowledge building on improving the efficiency of the multiobjective optimization for WDSs. However, in some cases, their efficiency improvements were at the expense of the final solution quality, such as the metamodeling approach [*Broad et al*., 2004] and the hybrid algorithms [*di Pierro et al*., 2009]. Although *Fu et al*. [2012] stated that their sensitivity-informed optimization method showed great efficiency in identifying near-optimal fronts for WDSs, the water network case studies considered in their work were relatively small. Hence, the effectiveness of their sensitivity-analysis-based multiobjective optimization method needs to be further demonstrated by using real-world networks (such as water networks with greater than 100 decision variables). To this end, the aim of this study is to develop a new multiobjective optimization method for the design of real-world WDSs, in which the computational efficiency is dramatically improved without compromising the quality of the identified Pareto optimal fronts.

In addition to computational inefficiency, another issue with the use of MOEAs is that their search behaviors are heavily dependent on the selected values of control parameters [*Tolson et al*., 2009]. Typically, a trial-and-error approach is used to fine-tune parameter values for the MOEAs applied to given optimization problems. This results in a large computational overhead especially when dealing with real-world optimization problems. The tedious effort required for tuning parameter values has been frequently claimed by practitioners as one of the main reasons for their reluctance to embrace MOEAs in practice [*Geem and Sim*, 2010]. In order to solve this issue, we have developed a self-adaptive multiobjective differential evolution (SAMODE) algorithm in the current study. Instead of presetting fine-tuned parameter values, the control parameters of the SAMODE algorithm are automatically adjusted by means of evolution. Details of the SAMODE algorithm are presented in section 3.4.

In the current study, a hybrid nonlinear programming (NLP) and self-adaptive multiobjective differential evolution (NLP-SAMODE) approach has been developed for designing WDSs. The method is an extension of the nonlinear programming (NLP) and differential evolution (NLP-DE) method proposed by *Zheng et al*. [2011], which was developed to deal with the least cost single-objective optimization for WDS design. *Zheng et al*. [2011] demonstrated that the NLP-DE algorithm was able to find optimal solutions for WDSs with great efficiency. For the present study, this algorithm has been modified to reduce the computational issues associated with multiobjective WDS design. The four main new contributions of this paper relative to the work of *Zheng et al*. [2011] are given as follows:

1. The extension of the single-objective NLP-DE method [*Zheng et al*., 2011] to the case of multiobjective WDS design problems, where the objectives considered are the minimization of the cost and maximization of the network resilience. The utilization of network resilience as one of the objectives in the current study is able to offer greater insight for the WDS design problems compared to the use of the minimum pressure head as a constraint [*Zheng et al*., 2011].

2. The proposal of mapping a large-scale multiobjective WDS design problem to a series of single-objective subproblems. These subproblems are individually optimized by the NLP to generate an approximate Pareto front for the original entire problem in a computationally efficient manner. This approach is new and it is the main innovation within this research.

3. The development of the self-adaptive multiobjective differential evolution (SAMODE) algorithm. An important advantage of the SAMODE algorithm is that, unlike other MOEAs, the values of the control parameters do not need to be tuned. In the current study, the utility of the SAMODE was demonstrated by comparing it with the NSGA-II algorithm (one of the widely used MOEAs) in terms of efficiently finding Pareto fronts. This is entirely new, going beyond *Zheng et al*. [2011].

4. The proposal of dynamically assigning diameters for chords with the incorporation of domain knowledge and network resilience (for details, see section 3.3). In contrast, the minimum pipe diameters were assigned to the chords in *Zheng et al*. [2011].

Overall, the current study is the first known work in which the deterministic optimization method NLP has been combined with the multiobjective evolutionary algorithm (the SAMODE algorithm was used in this study) to optimize WDSs with multiple objectives. Details of the proposed NLP-SAMODE method are provided in section 3.

2. Problem Formulation

The objectives considered were the minimization of network cost and the maximization of the network reliability. The network resilience surrogate measure [*Prasad and Park*, 2004] was adopted in this study to represent network reliability as it has been demonstrated to be more effective than other surrogate measures, such as the resilience index and the flow entropy, in terms of avoiding pipe size discontinuities and the reliability under pipe failure conditions [*Raad et al*., 2010]. Given a WDS design problem involving the selection of pipe diameters $\mathbf{D}=[D_1 \ldots D_n]^T$ only, the two-objective optimization problem is given by:

$$\text{Minimize the cost}: F=a\sum_{i=1}^{n} D_i^b L_i \tag{1}$$

$$\text{Maximize the network resilience}: I_n=\min\left\{\frac{\sum_{j=1}^{m} U_j Q_{j,ml}(H_{j,ml}-H^*_{j,ml})}{\sum_{r=1}^{R} q_{r,ml}H^R_{r,ml}-\sum_{j=1}^{m} Q_{j,ml}(H^*_{j,ml}+z_j)}\right\} \quad ml=1,\ 2,\ldots,ML \tag{2}$$

where

$$\text{Indicator of diameter uniformity}: U_j=\frac{\sum_{p\in M_j} D_p}{|M_j|\times \max_{p\in M_j}\{D_p\}} \tag{3}$$

$$\text{Hydraulic constraints}: \mathbf{H}_{ml}=f(\mathbf{D},\mathbf{Q}_{ml}) \tag{4}$$

$$\text{Diameter choices}: D_i \in A \quad i=1,\ldots,n \tag{5}$$

where F = the total network cost (to be minimized), including pipe material cost and the construction cost; D_i = the diameter of pipe i = 1, . . ., n; L_i = length of pipe i; a, b = specified cost function coefficients; n = total number of pipes in the network; $\mathbf{H}_{ml}=[H_{1,ml}\cdots H_{m,ml}]^T$ is the vector of pressure heads at network nodes for the demand loading case ml = 1, 2, . . ., ML; $\mathbf{Q}_{ml}=[Q_{1,ml}\cdots Q_{m,ml}]^T$ is the vector of nodal demands for the demand loading case ml; m = the total number of demand nodes in the network; $H_{j,ml}$ and $Q_{j,ml}$ are, respectively, the pressure head and the nodal demands for node j = 1, . . ., m for demand loading case ml. z_j is the elevation of node j; $H^*_{j,ml}$ is the minimum allowable pressure head at node j for water demand loading case ml; $q_{r,ml}$ and $H^R_{r,ml}$ are, respectively, total demands and total heads (pressure head plus the elevation head) provided by the supply source (reservoirs or tanks) $r = 1, \ldots, R$ for demand loading case ml; M_j is the set of all pipes connected to node j and $|M_j|$ is the cardinality of M_j; $A =$ the set of commercially available pipe diameters. Equation (4) was handled by the hydraulic solver EPANET2.0 [*Rossman*, 2000] in this study.

U_j in equation (3) is an indicator of the diameter uniformity for pipes that immediately connect node j. For example, if D_1, D_2, and D_3 are diameters of three pipes connecting node j (i.e., $M_j = \{1, 2, 3\}$), $U_j = (D_1 + D_2 + D_3)/\{3 \times \max(D_1, D_2, D_3)\}$. It is noted that $U_j = 1$ when $D_1 = D_2 = D_3$, otherwise $U_j < 1$, with smaller U_j representing larger diameter variations for pipes connecting node j.

$Q_{j,ml}(H_{j,ml} - H^*_{j,ml})$ in equation (2) is the surplus power of node j for the demand loading case ml. A larger value of surplus power represents a greater reliability in handling water demand variations [*Todini*, 2000], while a large value of U_j in equation (2) indicates a higher reliability of network loops due to their similar diameters. This implies that the network resilience measure (equation (2)) simultaneously accounts for both the effects of surplus power and loop reliability, and hence is capable of offering highly robust design solutions. The terms $\sum_{r=1}^{R} q_{r,ml} H^R_{r,ml}$ and $\sum_{j=1}^{m} Q_{j,ml} H^*_{j,ml}$ in equation (2) are, respectively, the total power provided by the supply sources (reservoirs or tanks) and the minimum power used to deliver the required demands to users with pressure head constraints $H^*_{j,ml}$. The difference between these terms indicates the maximum power available to be dissipated internally for the feasible solutions (both demand and pressure head constraints are satisfied).

Typically, multiple demand loadings are considered for real-world WDSs. As such, the value of network resilience measure for the same design solution with given **D** may vary in response to demand loading cases due to variations of nodal demands and pressure heads. In the current study, the minimum value of the network resilience measure for all demand loading cases, I_n, is used to represent the network reliability as shown in equation (2). The value of I_n is within the range [0, 1]. A design solution with a larger value of I_n suggests a greater supply reliability for handling nodal demand variations and the pipe failures within loops.

3. Proposed NLP-SAMODE Method

In the proposed NLP-SAMODE method, four stages are involved for the multiobjective optimization of WDSs:

1. *Stage 1: Shortest-Distance Tree or Forest Identification*. A graph decomposition technique—the Dijkstra algorithm—is employed to partition a looped water network into a shortest-distance tree T or forest, and chords Ω.

2. *Stage 2: Nonlinear Programming Optimization for the Shortest-Distance Tree*. The original two-objective optimization problem is approximated by a series of single-objective optimization problems of the T to be solved by NLP.

3. *Stage 3: Assigning Diameters for the Pipes of the Chords*. The pipes in the chords Ω were not considered in the NLP optimization (Stage 2) as they have been removed to form the shortest-distance tree T in Stage 1. An algorithm is proposed here to assign the diameters for the pipes in Ω based on the NLP solutions obtained in Stage 2. The assigned diameters are combined with the NLP solutions for the T to form an approximate optimal front for the original whole network.

4. *Stage 4: Multiobjective Optimization for the Original Full Network*. A self-adaptive multiobjective differential evolution (SAMODE) algorithm is developed and employed to refine the approximate Pareto front from Stage 3 in order to achieve the best possible optimal front for the original WDS problem.

In the following sections, the details of the proposed method (NLP-SAMODE) are presented and illustrated by optimizing a small water network with the two objectives given in equations (1–5).

3.1. Shortest-Distance Tree or Forest Identification (Stage 1)

3.1.1. Shortest-Distance Tree Identification for Single-Source WDSs

A water distribution system (WDS) can be described as a graph $G(V, S)$, where G is the graph, V is the set of nodes, and S is the set of links [*Zheng et al.*, 2013a]. A demand node in a looped WDS $G(V, S)$ may have

 10.1002/2013WR014143

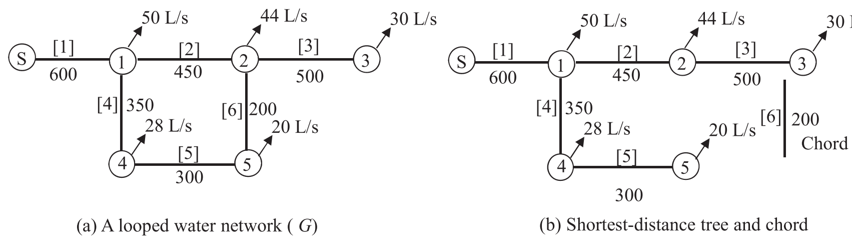


Figure 1. An example of decomposition results for a looped water network ((a) the original looped water network; (b) the shortest-distance tree and the chord). *S* is the supply or source node.

multiple possible paths by which to receive water from the source node. Of all possible paths, the one with the shortest total length is denoted the shortest path between the demand node and the source node. Each demand node has a shortest path to the source node and the union of these paths is designated the *shortest-distance tree T* [*Deo*, 1974]. The remaining edges that are not traversed by any shortest paths are termed *chords* Ω, where $G = T \cup \Omega$.

The shortest-distance tree *T* can be considered to be a meaningful spanning tree of the looped WDS as it represents the shortest paths from the sources to all demand nodes and the economic way to deliver water along its paths [*Kadu et al*., 2008; *Zheng et al*., 2011, 2013b]. The Dijkstra algorithm [*Deo*, 1974] is employed to find the shortest-distance tree for single-source WDSs. The details of the application of the Dijkstra algorithm to identify the *T* are given in *Zheng et al*. [2011, 2013b].

By applying the Dijkstra algorithm, the decomposition results for a small looped network with a single source shown in Figure 1a are presented in Figure 1b. For this demonstration network, the length for each pipe and the water demands for each node are shown in Figure 1.

3.1.2. Shortest-Distance Forest Identification for Multisource WDSs

The Dijkstra algorithm mentioned in section 3.1.1 is formulated for a single-source WDS, although in many cases there exist multiple sources (reservoirs or tanks) in a real-world and large-scale WDS. An approach is proposed in the current study to identify shortest-distance forest for WDSs with multiple sources, with details outlined as follows in two phases.

Phase 1: Identify the Subnetworks Based on the Number of Sources. In this phase, the decomposition method proposed by *Zheng et al*. [2013b] is employed to partition the original full WDSs with several sources (the number of sources is denoted as MS) into MS subnetworks, with each subnetwork $ms = 1, 2, \ldots, MS$ consisting of one and only one source node and a set of pipes and demand nodes. The available friction slope for the supply paths between each node and each supply source was used to identify the chords to enable the network decomposition [*Zheng et al*., 2013b]. This decomposition method was based on the heuristics that it is generally most cost effective for a demand node to receive flows from the source having relatively higher available heads and/or shorter distance to the node. In addition to the distance between supply source nodes and demand nodes, the topography of the network (elevations of the supply source nodes and demand nodes) are also considered in the decomposition method described in *Zheng et al*. [2013b].

Phase 2: Identify the Shortest-Distance Tree for Each Subnetwork. Since one and only one source node exists in each resultant subnetwork (*ms*) after the decomposition phase 1, the shortest-distance tree for each subnetwork T_{ms} can be determined by utilizing the Dijkstra algorithm. The shortest-distance forest (*SF*) is obtained as the union of the shortest-distance trees, i.e., $SF = \cup_{ms=1}^{MS} T_{ms}$. All the removed pipes for enabling the determination of the shortest-distance forest form the chords Ω.

Details of the decomposition algorithm for identifying subnetworks for WDSs with multiple sources are given in *Zheng et al*. [2013b] and case study 3 in section 4 is used to illustrate the decomposition method.

3.2. Nonlinear Programming Optimization for the Shortest-Distance Tree (Stage 2)

In Stage 2, the original two-objective optimization problem given in equations (1–5) is mapped to a series of single-objective optimization problems using the epsilon constraints method [*Cohon*, 1987]. In these single-objective optimization problems, the network cost is taken as the objective to be minimized while the network resilience I_n is considered to be constraints. Since it is difficult to explicitly incorporate I_n as

10.1002/2013WR014143

constraints, the minimum pressure head across the water network $H_{\min}$ is temporally used as a network reliability surrogate for single-objective optimization problems. This arrangement is possible because (i) $H_{\min}$ has previously been used as an indicator of network reliability, with a greater value indicating a reduced potential for the network to experience pressure head violations [*di Pierro et al.*, 2009]; and (ii) the use of $H_{\min}$ as a constraint greatly facilitates the formulation of single-objective optimization problems.

The process of mapping the two-objective problem to multiple single-objective problems involves assigning a set of minimum nodal head constraints $E=\{H_a^1, \ldots, H_a^K\}$ (where $H_a^k \geq H_{\min}$ is the assumed kth ($k = 1, \ldots, K$) head constraint, and K is the number of head constraints), and then minimizing the cost for a given head constraint. Consequently, for a given constraint $H_a^k \in E$:

$$\text{Minimize the cost of the shortest-distance tree}: F_1=a\sum_{i=1}^{n} D_i^b L_i, \forall i \in T \tag{6}$$

Subject to:

$$\text{Pressure head constraints}: H_{TS}-z_j-h_{s\to j}(ml) \geq H_a^k, \ \forall H_a^k \in E, j=1,\ldots,m, \ ml=1,\ldots,ML \tag{7}$$

$$\text{Head loss constraints}: h_{s\to j}(ml)=\sum_{i=1}^{n} \omega \frac{L_i}{C_i{}^{\alpha} D_i{}^{\beta}} Q_i(ml)^{\alpha}, \quad \forall i \in \Phi_{s\to j} \in T \tag{8}$$

$$\text{Pipe diameter range}: D_{\min} \leq D_i \leq D_{\max} \tag{9}$$

where F_1 = the total cost of the shortest-distance tree T; $h_{s\to j}$ (ml) = the total head loss in the shortest-distance path from the source s to the node $j = 1, \ldots, m$ for demand loading case $ml = 1, \ldots, ML$; H_{TS} = the available total head value at the source node s; $D_{\min}$ and $D_{\max}$ are the minimum and maximum allowable pipe diameters in set A; $\Phi_{s\to j}$ = the pipes in the shortest-distance path from the source s to the node j ($\Phi_{s\to j} \in T$). The Hazen-Williams (*H-W*) equation was used in the current study to calculate the head loss $h_{s\to j}$ (ml) as shown in equation (8) [*Simpson et al.,* 1994]. Thus, C_i = Hazen-Williams coefficient of pipe i and Q_i (ml) = pipe flow rate (m^3/s) of pipe i for the demand loading case ml. In this study, $\alpha = 1.852$ and $\beta = 4.871$.

When multiple trees are considered in the case of dealing with WDSs with multiple sources, equations (6–9) are formulated for each separate shortest-distance tree T and are optimized individually. In the proposed method, NLP is employed to solve these single-objective optimization problems. The decision variables used in equations (6–9) are continuous pipe diameters rather than the discrete values given in A as shown in equation (5) since NLP is only able to deal with a continuous search space.

Assigning an identical value H_a^k to each demand node, equations (6–9) can then be formulated to solve a single-objective optimization problem for the shortest-distance tree network T, in which the only objective is to minimize the tree network cost while ensuring the pressure head at each node $j = 1, \ldots, m$ is greater than the prespecified H_a^k value for all demand loading cases. Thus, a least cost design solution for the T is obtained by the NLP optimization for each particular H_a^k value. It should be highlighted that the minimum pressure head considered in the NLP is only used to provide initial guess to seed the SAMODE, and the network resilience is utilized when solving the original multiobjective optimization problem.

By assigning a sequence of different H_a^k values as the minimum pressure head constraints for the optimization model given in equations (6–9), a series of single-objective problems are established and individually solved by NLP optimization. The various H_a^k values are selected from a set E (i.e., $H_a^k \in E$) with a larger value of H_a^k representing greater network reliability. The set E is specified by the users. Normally, the lower bound H_l of E is the lower limit of what is required by WDS design criteria, which could differ for different cities. The upper bound H_u of E is limited by the available pressure head of the supply sources (reservoirs or pumps).

In the proposed method, H_a^k values are assigned to a series of integer values within E. For example, if the pressure head range for all nodes in a water network is between 20 and 30 m (i.e., H_l = 20 m and H_u = 30 m), then the integer values from 20 to 30 m with an increment of 1 m are used as H_a^k for the single-objective models given in equations (6–9), i.e., $E=\{20, \quad 21, \quad 22, \ldots, 30\}$. Therefore, 11 ($K = 11$) different NLP solutions are obtained.

Table 1. NLP Solutions for the Example Network

Assigned H_a^k Values (m) (Column 1)	Network Configurations (mm): From NLP Optimization in Stage 2: Pipe 1 (Column 2)	Pipe 2 (Column 3)	Pipe 3 (Column 4)	Pipe 4 (Column 5)	Pipe 5 (Column 6)	In Stage 3 Pipe 6[a] (Column 7)	Total Network Cost ($) (Column 8)	Minimum Pressure Head H_{min} (m) (Column 9)	Network Resilience I_n (Column 10)
20	396.7	291.9	224.8	230.5	178.9	178.9	135,273	19.71	0.19
21	411.8	303.1	233.4	239.3	185.7	185.7	143,235	20.76	0.29
22	431.1	317.3	244.3	250.5	194.4	194.4	153,621	21.81	0.39
23	457.3	336.6	259.2	265.7	206.3	206.3	168,129	22.85	0.48
24	497.1	365.8	281.7	288.8	224.2	224.2	190,933	23.90	0.58
25	573.1	421.7	324.8	333.0	258.4	258.4	237,308	24.95	0.67

[a]The chord of the example network given in Figure 1.

The flow distribution of the shortest-distance tree can be determined easily using a hydraulic solver (EPANET2.0 was used in this study) and satisfying the flow balance for each node (the flows in the chords Ω are assumed to be zero). This indicates that for the optimization model given in equations (6–9), the nodal mass balance is automatically satisfied before the NLP is run (i.e., Q_i (ml) is known in equation (8)). This leads to a significant reduction in the number of constraints within the NLP solver and hence optimal solutions can be found with great efficiency.

Assuming the minimum pressure head H_{min} for the example network given in Figure 1 is 20 m, it is very useful in practice to offer the decision maker a set of different solutions with different H_{min} values and a variety of network construction costs, rather than only providing the least cost solution for $H_{min} = 20$ m. For the example network under consideration, the total head at the source node is 26 m and $E=\{20, \quad 21, \ldots, 25\}$ was used to enable the NLP optimization for the shortest-distance tree T (Figure 1b).

The Hazen-Williams coefficient (C) is set to be 130 for all pipes in this network and the elevation is zero for all nodes. The coefficients of the unit pipe cost function are $a = 0.0104$ and $b = 1.5280$ (see equation (6)), and the units of cost and the diameter D in the cost function are $/meter and millimeter (mm), respectively. The total available discrete pipe diameters are $A = \{150, 200, 250, 300, 350, 400, 450, 500, 600, 700, 750, 800, 900, 1000\}$ mm and therefore $D_{min} = 150$ mm and $D_{max} = 1000$ mm for the NLP optimization. One demand loading case was considered for this example network. The 11 NLP solutions obtained for the T using different H_a^k values $\left(H_a^k \in E\right)$ are presented in Table 1. As can be seen from Table 1 (Columns 2–6), the pipe diameters for pipes 1–5 obtained from the NLP optimization in Stage 2 are overall larger when the assigned H_a^k increases as expected. It is noted that the pipe 6 is the chord of network and hence is not included during the NLP optimization. The pipe diameters for pipe 6 given in Column 7 are obtained by the next stage (Stage 3) of the proposed method.

3.3. Assigning Diameters for the Pipes in the Chords (Stage 3)

The pipes in the chords Ω were not included in the formulation of the NLP and hence were not optimized in Stage 2. In the proposed NLP-SAMODE method, the pipes in Ω are assigned to be the minimum diameters of the connecting pipes in the original full water network based on NLP solutions obtained in Stage 2. The rationale behind this includes: (i) it is reasonable to assign diameters to pipes that are similar to those of the surrounding pipes based on engineering judgment or experience of the water network design (i.e., maintain the high network resilience); (ii) assigning diameters to pipes in the set of chords Ω that resemble the surrounding pipes is useful to maintain a cost diversity in the initial solutions for the entire network; and (iii) the minimum diameters among the surrounding pipes assigned to the pipes in Ω are based on the consideration that these pipes are viewed to be less important than the pipes in the T in terms of demand delivery (see the decomposition in Stage 1).

The algorithm for dynamically assigning diameters to the pipes in the set of chords Ω can be given as:

$$\bar{D}_{k,j}=\min\left(\Delta_{k,j}\right) \qquad \forall j \in \Omega \tag{10}$$

where $\bar{D}_{k,j}$ = the diameter of pipe $j \in \Omega$ for the kth ($k = 1, 2, \ldots, K$) NLP solution, where K is the total number of NLP solutions, which equals to the total number of different assumed head values in the set of E;

10.1002/2013WR014143

$\Delta_{k,j}$ = the set of diameters comprised of the pipes connected to pipe j in the kth NLP solution. For the example, network given in Figure 1, pipe $j = 6$ is the chord in Ω, and consequently, $\Delta_{k,6} = \{D_{k,2}, D_{k,3}, D_{k,5}\}$, where $D_{k,2}$, $D_{k,3}$, and $D_{k,5}$ are pipe diameters for pipes 2, 3, and 5, respectively, in the kth NLP solution. The resulting diameters assigned to pipe $j = 6$ for different NLP solutions are given in Column 7 of Table 1. Values of $\Delta_{k,6}$ vary for different NLP solutions and hence the diameters assigned to the pipe in Ω are also different for different NLP solutions. It is observed that the assigned diameter for the chord pipe 6 is larger with the increase of the assigned H_a^k value.

For each assigned value of H_a^k, an optimal solution for the original full network is derived by combining the NLP solution and the assigned diameters for the pipes in the set of chords Ω. Subsequently, a final design $\mathbf{D} = [D_1, \ldots, D_n]^T$ is obtained, from which network cost, which is to be minimized, can be determined for each particular H_a^k value. The nodal pressure head can be obtained by hydraulic simulation (EPANET2.0 in this study) for each design solution with continuous pipe diameters, followed by the determination of the network resilience I_n (equation (2)), which is to be maximized. The resultant value of I_n for each approximate solution is presented in Column 10 of Table 1.

As shown in Table 1, a series of approximate optimal solutions for the original whole network is obtained and each approximate optimal solution is associated with a particular I_n value. The optimal solution with a relatively larger or smaller I_n value is associated with a relatively higher or lower network cost. The obtained approximate optimal solutions are therefore nondominated relative to each other (given the objectives are the minimization of cost and the maximization of the network resilience I_n); and therefore produce an approximate Pareto optimal front for the original whole network [*Deb et al.*, 2002].

It is observed that the actual H_{min} (Column 9) obtained by hydraulic simulation is slightly smaller than the assigned H_a^k (Column 1). For example, as shown in Table 1, H_{min} is 19.71 m for the first solution, which is slightly lower than its corresponding $H_a^k = 20$ m. This is because the return of the chords to the original network varies the flow distribution of the network slightly relative to the assumption made in Stage 1 that the flows in the chords Ω are zero. The variation of the flow distribution results in a slight change of the hydraulic properties (pressure head values) for the whole network. This was found for all case studies.

The approximate optimal front in terms of the network cost versus the network resilience I_n is given in Figure 2. As shown in Table 1 and Figure 2, the network design with a relatively higher or low I_n value is associated with a relatively higher or lower cost.

The approximate Pareto optimal front obtained after Stage 3 needs to be modified to generate an improved representation of the Pareto optimal front that applies to the original whole network. This is due to three facts: (i) the optimal solutions at the approximate Pareto front contain continuous pipe diameters since the diameters of the pipes are treated as continuous variables during the NLP optimization, and these continuous pipe diameters need to be converted to the available discrete pipe diameters; (ii) the pipe diameters of the chords are not actually optimized during Stage 3 of the proposed method as they are assigned the minimum diameters among the corresponding surrounding pipes; (iii) a very limited number of optimal solutions with specified integer H_a^k values are included in the approximate Pareto optimal front, and hence this front needs to be expanded to include the actual nondominated solutions. Stage 4 of the proposed NLP-SAMODE method described in the next section is used to drive the approximate Pareto optimal front toward an improved Pareto front for the entire original network.

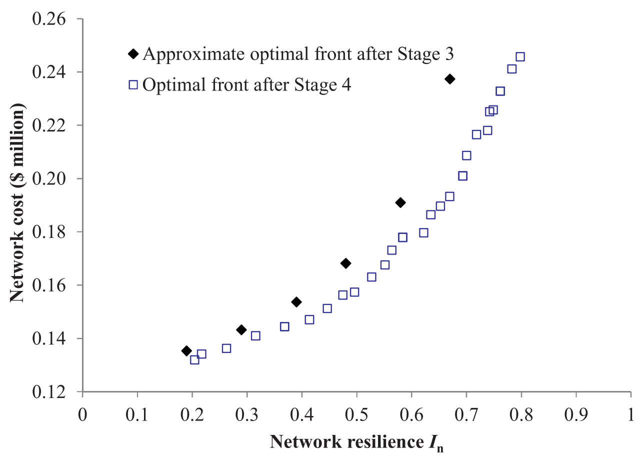


Figure 2. The approximate front after Stage 3 and the optimal front after Stage 4 for the example network.

It should be noted that the minimum pressure head across the whole network H_{min} was only used to replace the network resilience I_n in order to facilitate the mapping of the two-objective optimization problem to a set of single-objective problems to be solved by NLP. The I_n was taken back as a measure of the network reliability after the NLP optimization of the proposed method.

3.4. Multiobjective Optimization for the Original Full Network (Stage 4)

Previous studies have shown that the multiobjective differential evolution (MODE) algorithm is able to find comparable, if not better, fronts than the NSGA-II for multiobjective optimization problems [*Reddy and Kumar*, 2007]. As for other types of MOEAs, the performance of the MODE depends on the control parameter values, where the appropriate parameter values are normally optimization problem dependent [*Tolson et al*., 2009]. Typically, a few different parameter set values are tried for MOEAs applied to a new case study and the set exhibiting relatively better performance in terms of both the solution (front) quality and efficiency is selected. The necessity of fine tuning the parameters adds a large computational overhead, especially when dealing with large WDSs.

A self-adaptive multiobjective differential evolution (SAMODE) algorithm is proposed in the current study in order to remove the tedious effort required for tuning parameter values. In the proposed SAMODE algorithm, the two important control parameters: the mutation weighting factor (F, $0 < F \leq 1$) and the crossover rate (CR, $0 < CR \leq 1$) [*Storn and Price*, 1995], are encoded into the chromosome and are adapted by means of evolution. The essence of the SAMODE algorithm is given as follows: the F and CR values are initially randomly generated within given ranges for each individual in the initial population; the values of F and CR that produce new offspring dominating their corresponding individuals in the previous generation are directly passed onto the next generation, otherwise they are randomly generated again within the given ranges. An important feature of the proposed SAMODE is that the values of F and CR apply at the individual level rather than the generational level as for the standard MODE [*Reddy and Kumar*, 2007].

The range of F and CR is identical, set to be (0, 1], and does not need to be tuned when applied to different optimization problems. The details of the SAMODE algorithm are outlined below.

Step 1: Randomly generate N initial solutions $X_{i,G}=\{x^j_{i,G=0},\ldots,x^L_{i,G=0}\}, i=1,\ldots,N$ and values of mutation weighting factor (F) and the crossover rate (CR) within the range between (0, 1].

$$\begin{aligned} x^j_{i,G}&=rand[x^j_{\min},x^j_{\max}],\quad i=1,\ldots,N, j=1,\ldots,L \\ F_{i,G}&=rand\ (0,\ 1] \\ CR_{i,G}&=rand\ (0,\ 1] \end{aligned} \tag{11}$$

where $x^j_{\min}$ and $x^j_{\max}$ are the minimum and maximum bounds of the jth decision variable (minimum and maximum allowable pipe diameters in this study); L is the number of decision variables; *rand*[a, b] represents a uniformly distributed random variable taking values on the interval [a, b] or the semiopen interval (a, b] as if the case for F and CR; and $F_{i,G}$ and $CR_{i,G}$ are initial parameter values for the ith individual at generation $G=0$. As such, each initial solution is associated with a combination of random values of F and CR between (0, 1].

Step 2: Evaluate the two-objective values of the N initial solutions $F_1(X_{i,G})$ and $F_2(X_{i,G})$ as shown in equations (1) and (2). It should be noted that continuous pipe diameters are generated in the initialization process and these continuous values are altered to the nearest discrete diameters in A for hydraulic analysis.

Step 3: Perform the mutation operator to generate N mutant solutions $V_{i,G}=\{v^j_{i,G},\ldots,v^L_{i,G}\}$, $i=1,\ldots,N$.

$$V_{i,G}=X_{a,G}+F_{i,G}(X_{b,G}-X_{c,G}) \tag{12}$$

where $a \neq b \neq c$ and they are randomly generated for each $i=1,\ldots,N$.

Step 4: Perform the uniform crossover operator to generate N trial solutions $U_{i,G}=\{u^j_{i,G},\ldots,u^L_{i,G}\}$, $i=1,\ldots,N$.

$$u^j_{i,G}=\begin{cases} v^j_{i,G}, & \text{if } rand[0,1] \leq CR_{i,G} \\ x^j_{i,G}, & \text{otherwise} \end{cases} \tag{13}$$

where $rand\,[0,1]$ is generated independently for each $j=1,\ldots,L$, and $i=1,\ldots,N$.

Step 5: Round the continuous pipe diameters and evaluate the two objectives for the N trial solutions $U_{i,G}=\{u^j_{i,G},\ldots,u^L_{i,G}\}$, $i=1,\ldots,N$. This step is similar to Step 2 and $F_1(U_{i,G})$ and $F_2(U_{i,G})$ are obtained.

Step 6: Select the individuals and the F and CR values for the next generation $G=G+1$. A temporary solution set Ψ ($\Psi=\varnothing$ in the beginning) is created to store selected individuals during the solution comparisons, and constructed as follows:

$$\Psi=\Psi\cup\begin{cases} U_{i,G}, & \text{if } U_{i,G}\succ X_{i,G} \\ X_{i,G}, & \text{if } X_{i,G}\succ U_{i,G} \\ X_{i,G} \text{ and } U_{i,G} & \text{if } X_{i,G}\nprec U_{i,G} \text{ and } X_{i,G}\nsucc U_{i,G} \end{cases} \tag{14}$$

$$F_{i,G+1}=\begin{cases} F_{i,G}, & \text{if } U_{i,G}\succ X_{i,G} \\ rand(0,1], & \text{otherwise} \end{cases}$$
$$CR_{i,G+1}=\begin{cases} CR_{i,G}, & \text{if } U_{i,G}\succ X_{i,G} \\ rand(0,1], & \text{otherwise} \end{cases} \tag{15}$$

where $U\succ X$ means that solution U dominates solution X in a multiobjective sense, and $U\nsucc X$ means that solution U does not dominate solution X in a multiobjective sense [*Deb et al.*, 2002]. As shown in equation (14), each trial solution $U_{i,G}$ is compared with its corresponding $X_{i,G}$ with the same index i. If $U_{i,G}$ dominates $X_{i,G}$ ($U_{i,G}\succ X_{i,G}$, i.e., $F(U_{i,G})<F(X_{i,G})$ and $I_n(U_{i,G})>I_n(X_{i,G})$ in the context of the multiobjective problem considered here), $U_{i,G}$ is added to Ψ. In the meantime, the $F_{i,G}$ and $CR_{i,G}$ are given to $U_{i,G}$ to be used in the next generation ($G+1$) as shown in equation (15). In contrast, if $X_{i,G}\succ U_{i,G}$, $X_{i,G}$ is added to Ψ. When $U_{i,G}$ and $X_{i,G}$ are nondominated by each other ($X_{i,G}\nprec U_{i,G}$ and $X_{i,G}\nsucc U_{i,G}$), both of them are added to Ψ. Then, the nondominated ranking and the crowding distance algorithm are undertaken for the temporary vector Ψ to select N members for the next generation $X_{i,G+1}$ [*Deb et al.*, 2002]. For the selected members without associated F and CR values, new randomly generated F and CR values within (0, 1] are assigned to $X_{i,G+1}$. The temporary vector is emptied ($\Psi=\varnothing$) when the members for the next generation have been determined.

The final optimal front is obtained by repeating Steps 3–6 until the stopping criterion (such as the maximum allowable number of generations) is satisfied. In the current study, we restricted the range of the minimum pressure head values within [H_l, H_u] (i.e., the assigned minimum and maximum allowable pressure head for the network design). This is because a final solution with either too low or too high pressure head values is practically unacceptable [*Wu and Walski*, 2005]. If a solution violates the minimum pressure head constraint range, a positive and negative penalty cost are, respectively, added to the cost function in equation (1) and network reliability surrogate in equation (2) [details, see *Raad et al.*, 2010].

In the proposed method, the solutions at the approximate Pareto optimal front obtained by multiple NLP runs are used as the initial solutions for the SAMODE algorithm. However, the number of NLP solutions (N_a) is limited (N_a is normally smaller than the population size N). Thus, in addition to N_a approximate optimal solutions, the N-N_a randomly generated solutions within the whole search space are used to initialize the SAMODE algorithm.

For the example network given in Figure 1, the population size of SAMODE was set to be 30 ($N=30$), with $H_l=20$ m and $H_u=25$ m, respectively. Six optimal solutions ($N_a=6$) are included in the approximate optimal front as shown in Table 1. Thus, N-$N_a=24$ randomly generated solutions were combined with the six NLP solutions to form the total population $N=30$. The obtained optimal front after 100 generations is presented in Figure 2.

As shown in Figure 2, the solutions at the final optimal front after Stage 4 of the proposed method clearly dominate those at the approximate optimal front obtained in Stage 3 in terms of both the solution quality and front coverage. This indicates that the SAMODE used in Stage 4 is able to effectively drive the approximate optimal front toward an improved Pareto front for the original full network. Table 2 provides a few typical design solutions for the example network after Stage 4 of the proposed NLP-SAMODE method.

For the solution with the largest I_n (S_6 in Table 2), the diameters for pipes 2–6 are identical with a value of 350 mm, suggesting a high reliability for the loop within the water network. As mentioned in section 3.2,

10.1002/2013WR014143

Table 2. Typical Final Design Solutions for the Example Network

Solution Index	Network Configurations (mm) Pipe 1	Pipe 2	Pipe 3	Pipe 4	Pipe 5	Pipe 6	Total Network Cost ($)	Network Resilience I_n
S_1	400	300	250	200	150	150	131,967	0.20
S_2	400	350	250	200	150	200	140,982	0.32
S_3	450	350	250	200	200	200	156,287	0.47
S_4	450	400	250	250	150	250	167,511	0.55
S_5	500	350	250	300	250	300	186,365	0.64
S_6	600	350	350	350	350	350	245,705	0.80

the pipes in the chords Ω are not optimized by the NLP. Their diameters are assigned based on the heuristic method described in section 3.3 and then are used to seed the SAMODE optimization stage. These initially assigned diameters are further adjusted (optimized) during the SAMODE optimization process. As shown in Table 1, for the solution with $H_a^k = 25$ m, the pipe 6 (Ω) was given a pipe diameter of 258.4 mm (rounded to 250 mm) based on the proposed heuristic method, while this diameter was changed to 350 mm after the SAMODE optimization (S_6 in Table 2).

4. Case Studies

Three WDS case studies were used to demonstrate the effectiveness of the proposed NLP-SAMODE method. These include the Hanoi Problem (HP) [*Fujiwara and Khang*, 1990], Zhi Jiang network (ZJN) with three demand loading cases [*Zheng et al.*, 2011], and a completely new case study, the Lei Yang network (LYN) with 24 demand loading cases. The number of decision variables for the HP, ZJN, and LYN case studies are 34, 164, and 314, respectively. Extended period simulations were used for the case studies with multiple demand loading cases.

A SAMODE algorithm and a NSGA-II method seeded by purely random solutions were also applied to each case study to enable a performance comparison with the proposed NLP-SAMODE method. For the SAMODE algorithm, the ranges of F and CR were between (0, 1] for all the case studies and these two parameter values are self-adapted during the optimization process as described in section 3.4.

Previous studies have shown that a large value of crossover probability (P_c) is preferred when using the NSGA-II in order to obtain a good performance [*Deb et al.*, 2002], and hence P_c was fixed to be 0.9 for each case study. A number of different mutation probabilities (P_{mu}) were tried for each case study and the one for which the NSGA-II performed the best in terms of efficiently finding good quality fronts was selected.

The ranges of the minimum pressure head H_{min} across the whole network for the HP, ZJN, and LYN case studies were set to be in the range of [10, 40], [10, 30], and [10, 30] m, respectively. The pressure head range can be specified to any value by the user. For the three case studies discussed here, the H_{min} was restricted within the specified range for the three optimization methods (the proposed NLP-SAMODE, the SAMODE, and the NSGA-II) using the penalty approach described by *Raad et al.* [2010].

For each case study, each of the three methods was run 10 times with different starting random number seeds. It is observed that final optimal fronts generated by the proposed NLP-SAMODE method for each case study were similar overall, suggesting a high robustness and reliability of the search process. This is because, in the proposed method, the initial solutions in the multiple SAMODE runs shared the same NLP solutions and these preoptimized solutions guided the search toward to the promising regions. The results outlined below for each case study were taken from a typical run of the proposed method, while for the SAMODE and NSGA-II methods, the runs with the best performance in terms of the coverage and solution quality of the fronts were presented.

The commercial software Lingo12 [*LINDO Systems Inc.*, 2009] was employed to solve the NLP formulations. The nonlinear solver of the Lingo12 utilizes both the successive linear programming and generalized reduced gradient methods (for details, see *LINDO Systems Inc.* [2009]). For each case study, the NLP model of the equations (6–9) was solved with a number of different starting points. These include all variables with the minimum allowable pipe diameters, the maximum allowable pipe diameters, the middle value of the allowable diameter range, and 10 sets of initial values randomly generated within the diameter range

10.1002/2013WR014143

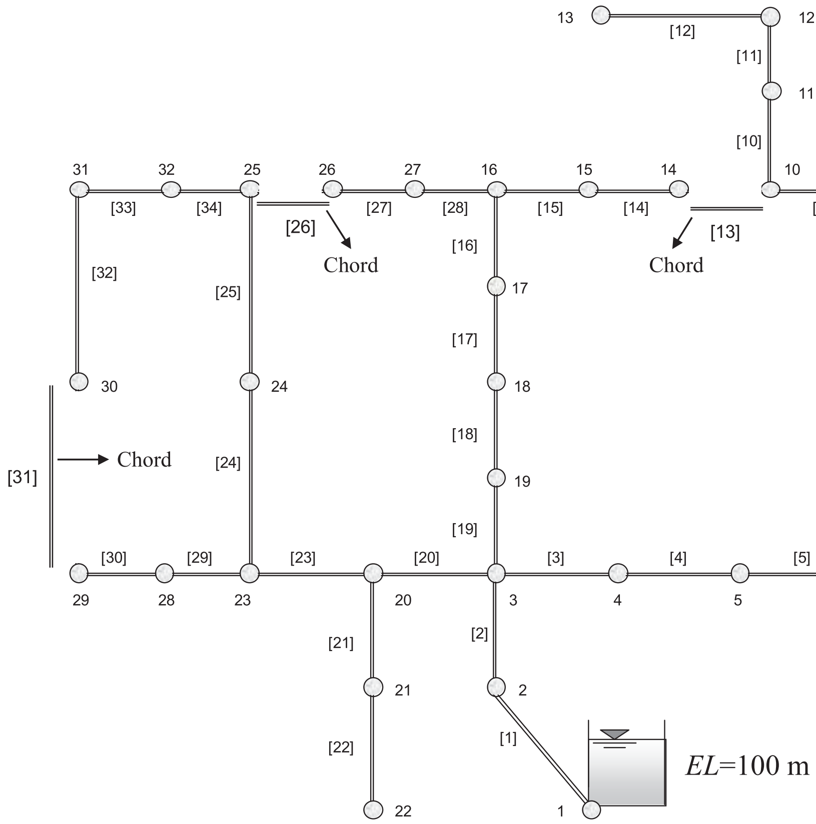


Figure 3. Shortest-distance tree and chords for the HP network.

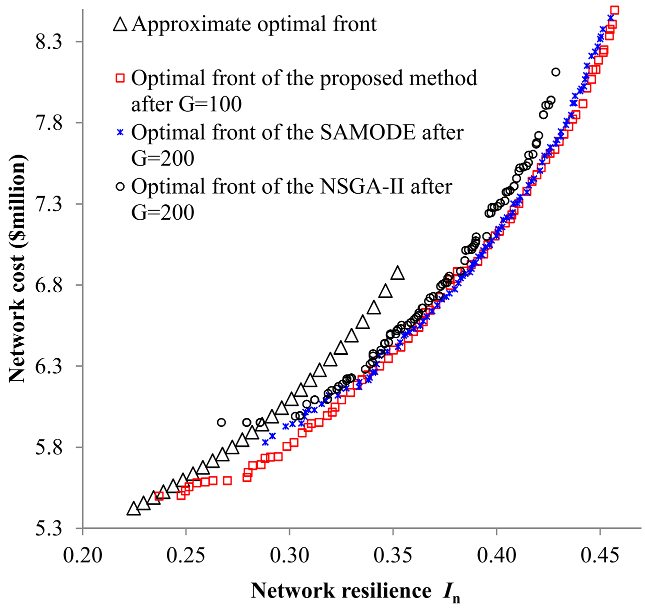
(different variables have different initial values). For these different initial guesses, the majority of the NLP optimization results were identical, while a few of them produced slightly different solutions in terms of the objective function values and the continuous pipe diameters. This shows that the solution space of the NLP model given in equations (6–9) is rather smooth, although it is not strictly convex. In the current study, the solution from the NLP model with starting points identically (all pipes) being the middle value of the allowable diameter range was used to seed the SAMODE for each case study.

4.1. Case Study 1: Hanoi Problem (34 Decision Variables)

The details of the Hanoi Problem (HP), including the pipe diameter choice table, network details, and the cost function, are given in *Fujiwara and Khang* [1990]. The shortest-distance tree and the chords of the Hanoi Problem (HP) after decomposition in Stage 1 are presented in Figure 3.

The assigned pressure head H_a^k for the HP case study was specified by the interval of [10, 40] m in the present study (i.e., $E=\{10,\ 11,\ 12,\ldots,40\}$). An NLP model was formulated and solved for each H_a^k value in Stage 2, generating 31 NLP solutions for the shortest-distance tree (T). For each NLP solution obtained in Stage 2, the algorithm proposed in Stage 3 was employed to assign diameters to the pipes in Ω. For the HP network, the set of chords $\Omega=\{13, 26, 31\}$ and their surrounding pipes are {9, 10, 14}, {25, 27, 34}, and {30, 32}, respectively. The diameters for pipes 13, 26, and 31 were set to be the minimum diameters from their corresponding surrounding pipes. Accordingly, 31 NLP approximate optimal solutions were obtained by combining the NLP solutions for the tree T and the assigned diameters for the pipes in the set of chords Ω. The results constituted the approximate optimal front for the HP case study, which is shown in Figure 4.

For the HP case study, a population size (N) of 100, was used for the proposed method, the SAMODE algorithm, and the NSGA-II. The 31 solutions at the approximate Pareto front plus 69 randomly generated solutions were taken as the initial solutions for the SAMODE used in the proposed method, while 100 randomly generated solutions within the whole search space were used for the SAMODE algorithm and NSGA-II. For



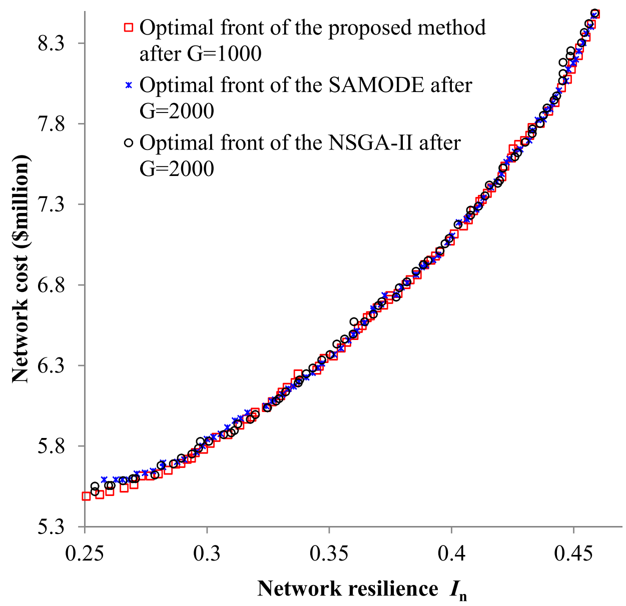


Figure 4. Optimal fronts obtained by the proposed NLP-SAMODE, SAMODE, and NSGA-II for the HP case study (34 decision variables).

the NSGA-II, different values of $P_{mu} = 0.01, 0.02, 0.03$, and 0.05 were tried for the HP case study and $P_{mu} = 0.02$ was selected as it performed the best in terms of the front quality. The optimal fronts produced by the three different methods are provided in Figure 4.

Figure 4 (left) shows that the optimal front generated by the proposed NLP-SAMODE method after 100 generations ($G = 100$) is significantly better than the fronts provided by the SAMODE algorithm and NSGA-II after 200 generations. This is especially the case when comparing the coverage of the fronts offered by the three methods. This implies that the proposed method is able to obtain a more resolved front than the SAMODE and NSGA-II methods with greater efficiency in the early generations of the HP case study.

The optimal fronts generated by the proposed method after $G = 1000$, and the SAMODE and NSGA-II after $G = 2000$ are presented in the Figure 4, right. In order to make a clear comparison between these three fronts, the nondominant sorting algorithm developed by *Deb et al*. [2002] was performed for the combined 300 solutions, with each of the three methods contributing 100 solutions. The algorithm found 192 nondominant solutions, of which 75, 68, and 49 were from the proposed method ($G = 1000$), the SAMODE algorithm ($G = 2000$), and the NSGA-II ($G = 2000$), respectively. In terms of solution spread and distribution, the optimal fronts produced by the three methods were comparable. These results showed that the proposed method was able to produce a front similar to the SAMODE and NSGA-II, but with a considerable reduction in computational overhead (50%). For the HP case study, the proposed SAMODE algorithm without parameter tuning outperformed the NSGA-II with calibrated parameter values in terms of the quality of the fronts as shown in Figure 4.

4.2. Investigation of Various Spanning Trees

We considered another two random spanning trees for the HP case study, in order to enable a comparison of performance with the proposed shortest-distance tree in terms of the quality of the optimal fronts. The chords of the two random spanning trees are $\Omega_1 = \{9, 27, 29\}$ and $\Omega_2 = \{5, 23, 30\}$ with the pipe index given in Figure 3. The NLP was employed to optimize the two random spanning trees separately. The approximate front, the fronts after 50 and 1000 SAMODE generations of the proposed method with the random spanning tree Ω_1 is given in Figure 5.

As evident in Figure 5, the approximate front of the random spanning tree Ω_1 is clearly dominated by that of the proposed shortest-distance tree. Given the same values of the network resilience I_n, the cost of the approximate solutions from the shortest-distance tree are dramatically lower than those from the random spanning tree Ω_1. Using the same SAMODE generations ($G = 50, 1000$), the optimal fronts achieved using

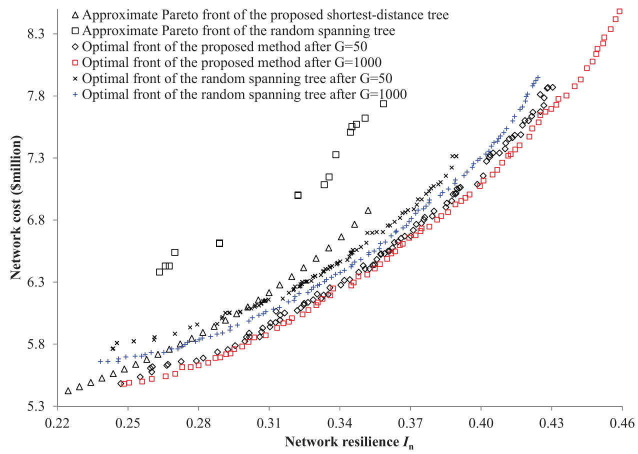


Figure 5. Optimal fronts obtained by the proposed NLP-SAMODE with the shortest-distance tree and the random spanning tree $\Omega_1 = \{9, 27, 29\}$ for the HP case study (34 decision variables).

the proposed shortest-distance tree are appreciably better that those obtained using the random spanning tree Ω_1 as shown in Figure 5. For the random spanning tree $\Omega_2 = \{5, 23, 30\}$, no feasible NLP solutions were found, showing that the solution regions determined by the Ω_2 is significantly far from that of the optimal solutions.

Similar observations were made when considering two random spanning trees for the case study 2 (ZJN case study). This implies that the performance of the proposed NLP-SAMODE method is sensitive to the decomposition approach used and the proposed shortest-distance tree is effective in providing good estimates for the Pareto front of the original entire WDS.

4.3. Case Study 2: ZJN Network (164 Decision Variables)

The ZJN network was first used by *Zheng et al*. [2011] as a single-objective design case study. In this study, this network was used as a multiobjective design problem. Only one demand loading case was involved in the original ZJN network, but now it has been changed to have three typical demand loading cases for each of the three supply regions as shown in Figure 6. The blue, red, and green nodes in Figure 6 represent the supply regions 1, 2, and 3, respectively, and their demand multiplier values based on the nodal base demands are given in Table 3 (i.e., for each node, the actual demands are the nodal base demands multiplied by the demand multiplier value). The Dijkstra algorithm was employed to identify the shortest-distance tree for this single-source water network (see section 3.1.1), with the tree presented in Figure 7.

The total number of available discrete pipe diameters and the cost function coefficients are the same as for the example network, i.e., $A = \{150, 200, 250, 300, 350, 400, 450, 500, 600, 700, 750, 800, 900, 1000\}$ mm, and a = 0.0104 and b = 1.5280. The pressure head range for each demand node was set to be the range of [10, 30] m. Each integer number between, including 10 and 30 was used as the minimum pressure head H_a^k for the NLP model $(E=\{10,\ 11,\ 12, \ldots, 30\})$. A total of 21 NLP solutions were obtained for the ZJN case study after Stages 2 and 3 of the proposed NLP-SAMODE method. Each solution was associated with a different value of network resilience I_n, resulting in 21 unique solutions for the approximate Pareto front (see Figure 8).

A population size of $N = 300$ was used for the proposed method, of which 21 was the NLP approximate solutions and 279 solutions were randomly generated within the whole search space. For the SAMODE and NSGA-II methods, $N = 300$ randomly generated solutions were used for the initial population. A value of $P_{mu} = 0.006$ was selected for the NSGA-II applied to the ZJN case study after trying a set of different values including $P_{mu} = 0.005$, 0.006, 0.007, 0.008, and 0.01. The optimal fronts produced by the proposed NLP-SAMODE method, the SAMODE, and the NSGA-II with $P_{mu} = 0.006$ for the ZJN case study are shown in Figure 8.

Table 3. Demand Multiplier Values for the Three Supply Regions of the ZJN Case Study

	Demand Patterns for Different Supply Regions of ZJN Network[a]		
	Region 1	Region 2	Region 3
Demand multiplier values	0.75	1.42	1.01
	1.49	0.88	0.93
	0.91	0.97	1.58

[a]The blue, red, and green dots in Figure 6 represent supply regions 1, 2, and 3, respectively.

Since the minimum pressure head for this case study is restricted between 10 and 30 m, the I_n values are accordingly within the range approximately between 0.2 and 0.5 as shown in Figure 8. The range of the pressure head H_a^k in the NLP optimization (Stage 2) needs to be expanded if one expects

10.1002/2013WR014143

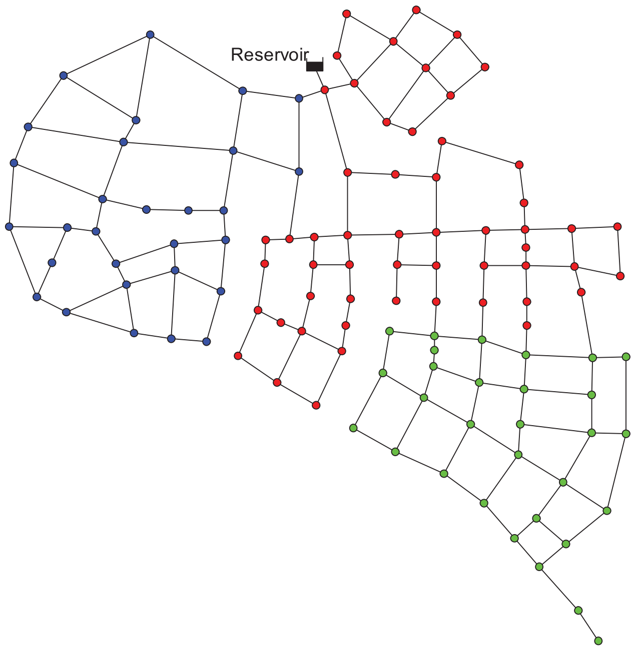


Figure 6. The network of the ZJN case study. The blue, red, and green nodes represent supply regions 1, 2, and 3, respectively, with different demand patterns (see Table 3).

solutions with further either larger or lower I_n values.

As shown in Figure 8, the proposed method after 300 generations achieved a front that substantially dominates those produced by the SAMODE algorithm and the NSGE-II after 500 generations in the region with relatively low I_n values ($I_n < 0.47$). Although the quality of the fronts was significantly improved for the SAMODE and NSGA-II methods when their computational budgets were extended to 4000 generations, the solutions with $I_n < 0.34$ were still dominated by those offered by the proposed method after only 300 generations.

The front provided by the proposed method after 1000 generations clearly dominates the fronts produced by the SAMODE and NSGA-II after 4000 generations. This was especially true when comparing the solutions in the low I_n region. The SAMODE and NSGA-II were unable to provide solutions with $I_n < 0.28$, while the proposed NLP-SAMODE was capable of finding such solutions with great efficiency. If one expects a network design with a relatively high value of I_n, such as $I_n \approx 0.45$, the proposed method was able to find such a solution with a cost of \$9.4 million using 1000 generations, while the SAMODE algorithm and the NSGA-II offered such solutions with approximately \$9.5 million using 4000 generations. This indicates that, for this relatively larger case study with multiple demand loading cases, the proposed method found a better quality front than the conventional full-search approaches (the SAMODE algorithm and the NSGA-II) with significantly improved efficiency.

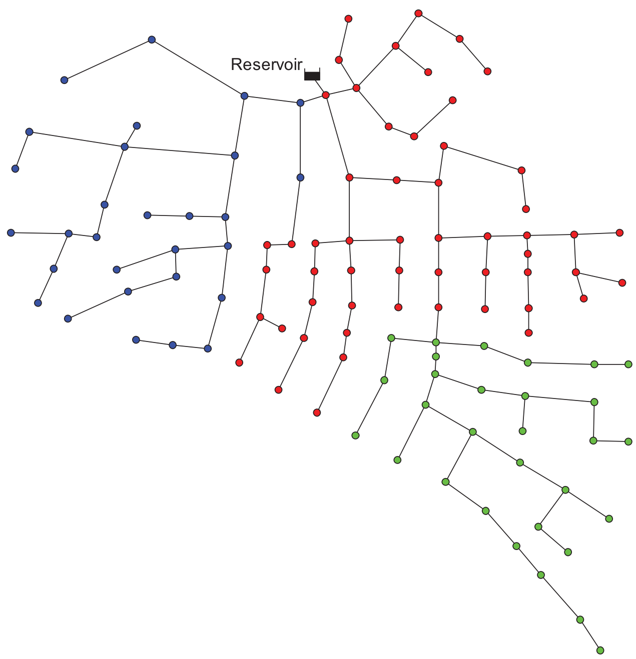


Figure 7. The shortest-distance tree of the ZJN network. Different colored demand nodes represent different supply regions.

Interestingly, the advantage of the proposed method over the conventional full-search approaches is more significant for the solutions with lower I_n values, while exhibiting similar performance in the region with very high I_n values. This is because the solutions at the approximate optimal front (that were used to seed the proposed method) are overall associated with low I_n values as shown in Figure 8. The

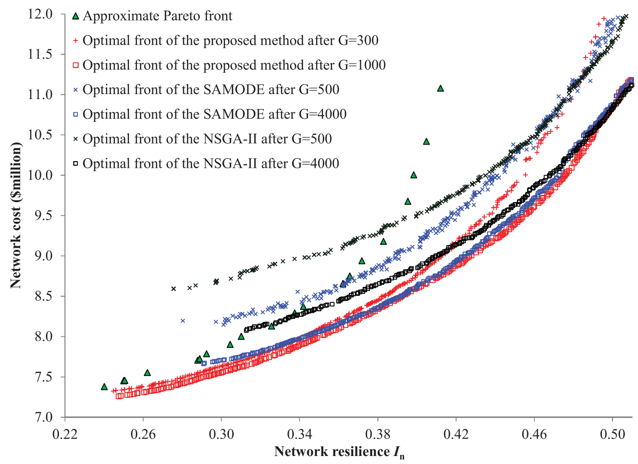


Figure 8. Optimal fronts obtained by the proposed NLP-SAMODE, SAMODE, and the NSGA-II methods for the ZJN case study (164 decision variables).

association of better solutions in regions of lower I_n values is a result of using the minimum pressure heads as constraints in the NLP optimization. Head excess was only considered to represent network reliability without the incorporation of diameter uniformity. Accordingly, I_n values of these NLP solutions are overall low, even if the minimum pressure head is very large (of say approximately 30 m).

Similar observations were made for case study 3 in section 4.4. Results in the case studies 3 and 4 indicated that a design solution with a high minimum pressure head excess across the water network does not necessarily possess high reliability when dealing with nodal demand variations and pipe failures (as indicated by the network resilience I_n). This observation agrees well with that found by *Prasad and Park* [2004].

Although the three methods being compared performed similarly in the region with very high I_n values, the majority portion the Pareto front provided by the proposed method clearly dominated those generated by the conventional full-search optimization methods. More importantly, the proposed method was able to offer a notably improved front with approximately three times faster computational speed than the SAMODE algorithm and the NSGA-II. For this case study 2, the SAMODE performed better than the NSGA-II with fine-tuned parameter in terms of both the solution quality and the coverage of the optimal front, as illustrated in Figure 8.

4.4. Case Study 3: LYN Case Study (315 Decision Variables)

The LYN case study is taken from a city in China. The layout of this network is given in Figure 9. There are 288 demand nodes, 315 links, and three reservoirs in the LYN water network. A total of 14 commercial piped diameters (from 150 to 1000 mm) are available as choices for designing this network and each pipe has a Hazen-Williams coefficient of 130. The 14 pipe diameters and the unit costs are from *Kadu et al*. [2008].

For the LYN network, there are three different supply regions based on the variation of the demand patterns, with blue, red, green demand nodes indicating supply regions 1, 2, and 3, respectively, as shown in Figure 9. For each supply region, 24 demand loading cases with extended period simulations are considered, with each representing the hourly demand multiplier based on the nodal base demands. Figure 10 shows the values of the hourly demand multiplier for each supply region of the LYN network.

The decomposition method proposed in section 3.1.2 was utilized to find the shortest-distance forest for this multiple-source water network. Three subnetworks were identified first using the partitioning method proposed by *Zheng et al*. [2013b], with each subnetwork consisting of only one source (reservoir) and a number of nodes and pipes (S_1, S_2, and S_3 in Figure 11). Then the Dijkstra algorithm was used to identify the shortest-distance tree for each of the subnetworks. The resultant shortest-distance forest for the LYN network is composed of three shortest-distance trees as shown in Figure 11.

The pressure head range for each node of the LYN case study was set between 10 and 30 m. A total of 21 H_a^k values ranging from 10 to 30 (including 10 and 30 m) with intervals of 1 m $(E=\{10,\ 11,\ 12,\ldots,30\})$ was used for the NLP models. For each given H_a^k value, the NLP given in equations (6–9) was formulated for each shortest-distance tree and was solved individually. The obtained NLP solutions were combined together to provide the NLP solutions for the original full network. The pipes in the chords of the shortest-distance tree were determined by using the proposed method described in Stage 3 (for details see

10.1002/2013WR014143

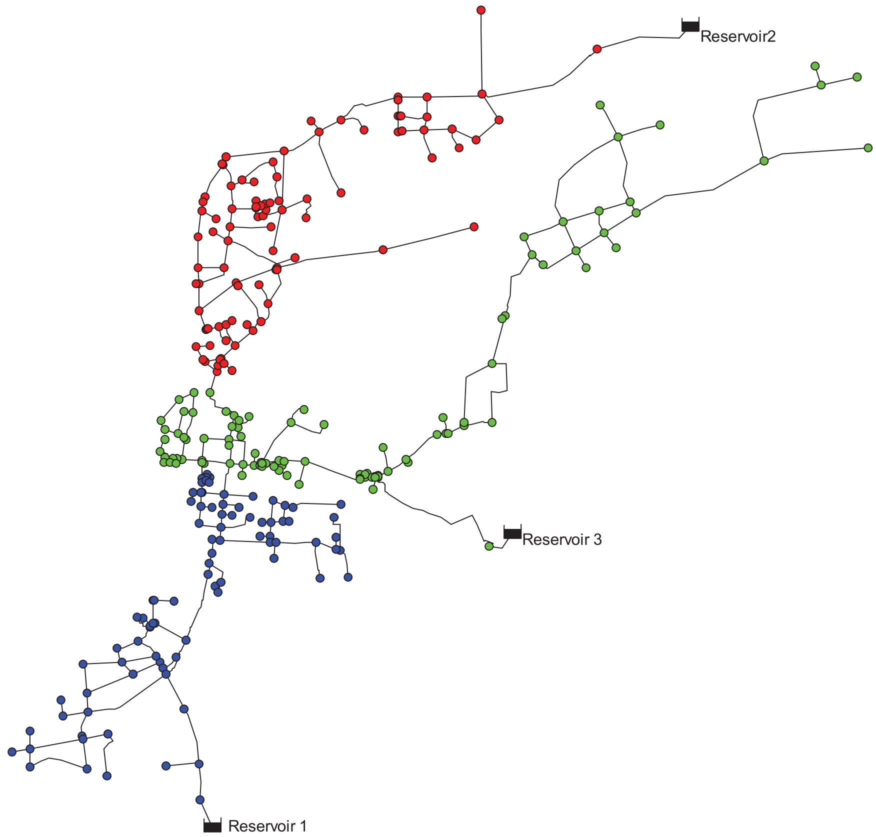


Figure 9. The network of the LYN case study. The blue, red, and green demand nodes, respectively, represent supply regions 1, 2, and 3 with different demand patterns.

section 3.3). The approximate front for the original LYN case study, formed from 21 NLP solutions after Stages 2 and 3, is presented in Figure 12.

A population size of $N = 300$ was used for the multiobjective stage of the proposed NLP-SAMODE method, as well as for the SAMODE and NSGA-II approaches. In addition to 21 solutions at the approximate Pareto

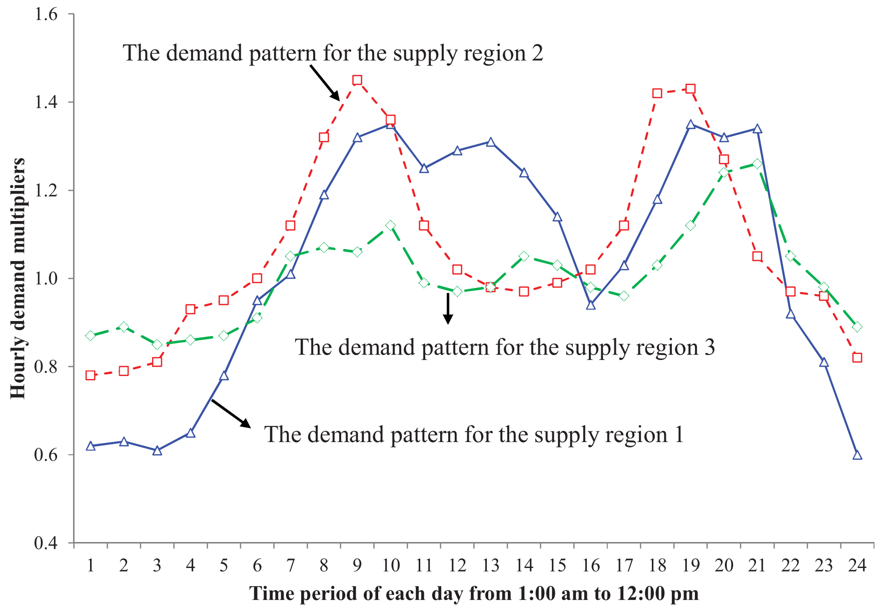


Figure 10. The hourly demand multiplier for supply regions 1, 2, and 3 of the LYN case study (see Figure 9).

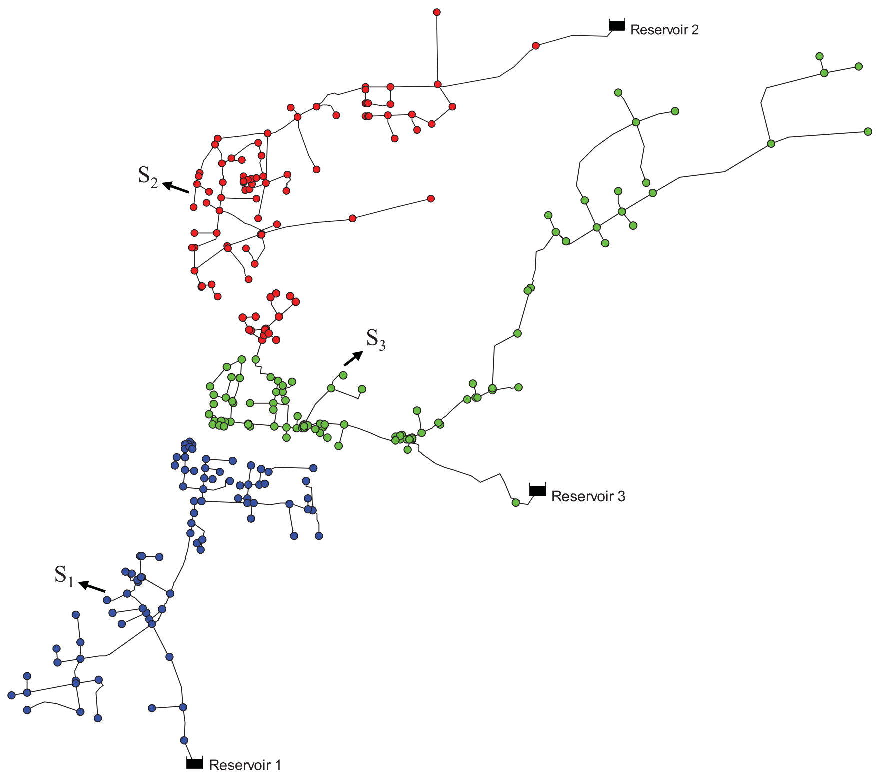


Figure 11. The shortest-distance forest of the LYN case study, with S_1, S_2, and S_3 representing three subnetworks. The blue, red, and green demand nodes, respectively, indicate different supply regions.

front, 279 solutions within the whole search space were randomly generated to provide a total of 300 solutions in the initial population for the SAMODE used in the proposed method. For the NSGA-II, $P_{\mathrm{mu}} = 0.004$ was ultimately selected as it outperformed other values such as $P_{\mathrm{mu}} = 0.001$, 0.002, 0.003, 0.005, and 0.01 in terms of the front quality. The fronts of the conventional full-search methods (the SAMODE and the NSGA-

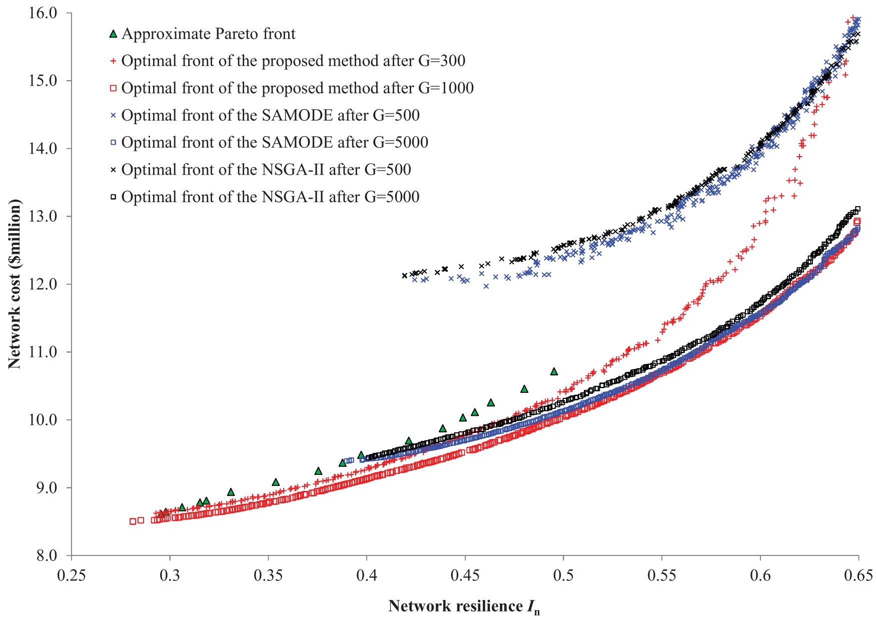


Figure 12. Optimal fronts obtained by the proposed NLP-SAMODE, the SAMODE algorithm, and the NSGA-II methods for the LYN case study (315 decision variables).

II) after 500 and 5000 generations, and the front of the proposed NLP-SAMODE method after 300 and 1000 generations are given in Figure 12.

As can been seen in Figure 12, the value of the network resilience for the LYN case study was limited approximately between 0.25 and 0.65, because the minimum pressure head value was set between 10 and 30 m in the current study. As clearly shown in Figure 12, the proposed method produced a front after only 300 generations that considerably dominates the fronts yielded by the SAMODE algorithm and the NSGA-II after 500 generations, and even 5000 generations for the region with relatively low network resilience values ($I_n < 0.42$).

On the other hand, when the NLP-SAMODE was run 1000 generations, the optimal front was improved further especially for the high I_n region. This front obtained using 1000 generations clearly dominate those found by the SAMODE algorithm and the NSGA-II after 5000 generations. Overall, with similar values of the network resilience, the proposed method was able to find approximately 2% lower cost solutions than the two conventional full-search approaches with approximately four times faster computational speed. This result demonstrates the superior performance of the proposed NLP-SAMODE method over the SAMODE algorithm and the NSGA-II in terms of efficiently providing good quality Pareto optimal front.

As for the previous case study 2, the advantage of the proposed method over the conventional full-search approaches is more noteworthy when the network resilience I_n decreases. In addition to the fact that the NLP solutions were overall located in the region with low I_n values as discussed in section 4.3, it is possible that in this complex case study 3, finding solutions in the Pareto front with relatively lower I_n values was more difficult. This can be reflected by that neither the SAMODE algorithm nor the NSGA-II were able to find any solutions with $I_n < 0.4$ within 500 generations. Furthermore, when the computational budget was significantly increased to 5000 generations, the coverage of the fronts was only slightly extended to approximately $I_n = 0.38$. In contrast, the proposed NLP-SAMODE method offered many solutions with $I_n < 0.35$ with only 300 generations and the minimum I_n reached a very low value of 0.27 as shown in Figure 12.

When comparing the performance of the two conventional full-search methods for the case study 3, the SAMODE clearly dominated the NSGA-II in terms of providing good quality fronts. This was especially the case when a relatively large computational budget with 5000 generations was assigned for both methods. Based on the results for the three case studies, it can be concluded that the proposed SAMODE in the current study (no parameter tuning) outperformed the NSGA-II with fine-tuned parameter values (a representative of the widely used MOEAs) in terms of efficiently offering Pareto fronts for WDS design problems.

5. Computational Analysis and Search Diversity Discussion

To enable an analysis of the computational efficiency of the Stages 1 and 2 of the proposed method, the total computational time required to find the shortest-distance tree or forest, and run multiple NLP models was first recorded for each case study. Then the computational run time of a single hydraulic simulation for each case study was obtained by taking the average of total time required to simulate 1000 times with randomly generated pipe diameters. Finally, the computational effort required by the Stages 1 and 2 of the proposed method for each case study was converted to the equivalent number of its full network simulations. Note that all these computational analysis used the same computer configuration (Pentium PC (Inter R) at 3.0 Hz). The results are given in Table 4.

As shown in Table 4, the total computational effort required to find the shortest-distance tree and run multiple NLPs for the HP, ZJN, and LYN case studies are 816, 4227, and 9022 equivalent numbers of their corresponding original whole network simulations, respectively. This shows that the computational effort required by the Stages 1 and 2 of the proposed method was very small compared to the total evaluations used by the SAMODE algorithm (200,000 for the HP, 1,200,000 for the ZJN, and 1,500,000 for the LYN). The computational run time for Stage 3 (assigning pipe diameters for the pipes in the chords) of the proposed method is negligible and hence is not included.

Potential issues when implementing the proposed method are a loss of search diversity and premature convergence since the original network is optimized by the SAMODE algorithm with the optimal solutions at the approximate Pareto front fed into the initial population. This kind of the preconditioning technique has

Table 4. Computational Effort Analysis for Finding Shortest-Distance Tree and Running the NLP Solver for Each Case Study[a]

Case Study	Number of Decision Variables	Computational Effort Required to Find the Shortest-Distance Tree (Stage 1)	Computational Effort Required to Solve the NLP for the Shortest-Distance Tree (Stage 2)	Number of NLP Runs	Total Equivalent Number of Original Network Evaluations	The Number of Evaluations Used by SAMODE[b]
HP	34	10	26	31	816	200,000
ZJN	164	6	201	21	4227	1,200,000
LYN	315	13	429	21	9022	1,500,000

[a]Note: The computational effort in finding shortest-distance tree and running NLP has been converted to an equivalent number of evaluations for its corresponding case study.
[b]The population sizes used for HP, ZJN, and LYN are, respectively, 100, 300, and 300.

been recognized to potentially reduce the diversity of the search and increase the risk of premature convergence for multiobjective applications [*Kollat and Reed*, 2006; *Kang and Lansey*, 2012].

In order to avoid the loss of search diversity, in addition to the solutions at the approximate Pareto front, other randomly generated solutions were used to initialize the SAMODE for the proposed method. In this way, the convergence to the best possible front was accelerated by exploiting the optimal solutions from the approximate Pareto front, while the search diversity was maintained by the addition of random solutions.

In the current study, a number of different population sizes N were used for each case study. For the HP case study seeded by 31 NLP solutions, $N = 100$ was sufficient to offer fronts with similar diversity (coverage) compared to fronted produced by the full-search SAMODE algorithm (initial solutions are randomly generated). For large case studies 2 and 3 (seeded by 21 NLP solutions), the value of N needs to increase to 300 in order to keep the front diversity.

Although it is difficult to derive an exact population size (N) for the proposed NLP-SAMODE method to preserve search diversity, it is empirically recommended that N should be no less than three times the number of NLP solutions at the approximate Pareto front for relatively small (number of decision variables $<$50) design problems. For large and complex WDSs, 15 times the number of NLP solutions at the approximate Pareto front are required.

6. Conclusion

This paper introduces an efficient hybrid method for multiobjective design of water distribution systems (WDSs). The objectives considered in this study were the minimization of the network cost and maximization of network reliability measured by network resilience. A self-adaptive multiobjective differential evolution (SAMODE) algorithm was developed, in which the two important parameters (F and CR) are automatically adjusted via evolution rather than presetting of fine-tuned parameter values.

In the proposed hybrid method, a shortest-distance tree/forest is first identified for a looped water network (Stage 1). Then the original two-objective optimization problem for the WDS design is approximated by a series of single-objective optimization problems based on the obtained shortest-distance tree. A nonlinear programming (NLP) is formulated for the shortest-distance tree and multiple NLP runs are conducted for different assigned minimum pressure head values (Stage 2). In Stage 3, an algorithm is proposed to assign diameters to the pipes in the chords based on the NLP solutions obtained in Stage 2. The NLP solutions for the tree network and the assigned diameters for the pipes in the chords constitute an approximate Pareto front for the original whole network. The solutions at the approximate Pareto front were finally used to seed a SAMODE algorithm to find the best possible front in Stage 4 with objectives being the minimization of network cost and maximization of the network resilience.

Three WDS case studies, including a benchmark problem (HP) with 34 decision variables, and two real networks (ZJN and LYN case studies) with 164 and 315 decision variables, respectively, were used to verify the effectiveness of the proposed method. Three and 24 demand loading cases with extended period simulations were, respectively, considered for the ZJN and LYN (three reservoirs) case studies. The results of the three case studies clearly show that the proposed method consistently generated optimal fronts superior to the conventional full-search methods (SAMODE and NSGA-II) seeded by purely random solutions with massively improved efficiency. Additionally, the advantage of the proposed method over the conventional full-search methods is more pronounced for the relatively larger and more complex case studies (ZJN and LYN case studies).

It is observed that the SAMODE algorithm developed in the current study consistently exhibited better performance than the NSGA-II (a representative of the widely used MOEAs) with tuned parameter values in terms of the quality of the Pareto fronts for the three case studies. This is especially the case for the large and more complex case studies 2 and 3, as the fronts offered by the proposed SAMODE are clearly better than those provided by the NSGA-II with calibrated parameter values. The removal of the parameter calibration process produces significant computational savings, and is one of the most important features of the SAMODE algorithm. Although the proposed SAMODE was demonstrated using the WDS design problems in this paper, it can be easily used to optimize other water resource problems.

The shortest-distance tree is used as a surrogate indicator of the main flow paths within the network (the network tree) in the proposed method. The determination of this tree is based on the distance between the supply nodes and the demand node, while does not take into account of the topography of the network. The quality of the initial NLP solutions may be deteriorated to some extent when dealing with a WDS where there are significant differences in nodal elevations. Incorporating the network topography into the decomposition is one focus of future study.

The NLP optimization in Stage 2 only considers the pressure head excess without the incorporation of the diameter uniformity. This results in relatively low network resilience (I_n) values for the NLP solutions, even if the pressure head is very large. The SAMODE seeded by these solutions is, therefore, more effective and efficient in exploring the region with relatively lower I_n values compared to its behavior in regions with very high I_n values as shown in Figures 4, 8, and 12. Accounting for the diameter uniformity in the NLP optimization to enable an effective search along the entire range of network resilience is a recommended future research direction.

Acknowledgment
For details of water distribution system case studies used in this paper, please contact feifei.zheng@adelaide.edu.au.

References

Broad, D. R., G. C. Dandy, and H. R. Maier (2004), A metamodeling approach to water distribution system optimization, paper presented at the World Water and Environmental Resources Congress, Salt Lake City, Utah, 27 June-1 July.

Cohon, J. L. (1987), *Multiobjective Programming and Planning*, Dover, Mineola, N. Y.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan (2002), A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, *6*(2), 182–197.

Deo, N. (1974), *Graph Theory With Applications to Engineering and Computer Science*, Prentice Hall, Englewood Cliffs, N. J.

di Pierro, F., S.-T. Khu, D. Savic, and L. Berardi (2009), Efficient multi-objective optimal design of water distribution networks on a budget of simulations using hybrid algorithms, *Environ. Modell. Software*, *24*(2), 202–213.

Farmani, R., D. A. Savic, and G. A. Walters (2005), Evolutionary multi-objective optimization in water distribution network design, *Eng. Optim.*, *37*(2), 167–183, doi:10.1080/03052150512331303436.

Fu, G., and Z. Kapelan (2011), Fuzzy probabilistic design of water distribution networks, *Water Resour. Res.*, *47*, W05538, doi:10.1029/2010WR009739.

Fu, G., Z. Kapelan, and P. Reed (2012), Reducing the complexity of multi-objective water distribution system optimization through global sensitivity analysis, *J. Water Resour. Plann. Manage.*, *138*(3), 196–207.

Fujiwara, O., and D. B. Khang (1990), A two-phase decomposition method for optimal design of looped water distribution networks, *Water Resour. Res.*, *26*(4), 539–549.

Geem, Z. W., and K.-B. Sim (2010), Parameter-setting-free harmony search algorithm, *Appl. Math. Comput.*, *217*(8), 3881–3889.

Kadu, M. S., R. Gupta, and P. R. Bhave (2008), Optimal design of water networks using a modified genetic algorithm with reduction in search space, *J. Water Resour. Plann. Manage.*, *134*(2), 147–160.

Kang, D., and K. Lansey (2012), Revisiting optimal water-distribution system design: Issues and a heuristic hierarchical approach, *J. Water Resour. Plann. Manage.*, *138*(3), 208–217.

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

Kollat, J. B., and P. M. Reed (2006), Comparing state-of-the-art evolutionary multi-objective algorithms for long-term groundwater monitoring design, *Adv. Water Resour.*, *29*(6), 792–807.

LINDO Systems Inc. (2009), *LINGO12 User's Guide*, LINDO Syst. Inc., Chicago, Ill.

Ostfeld, A. (2012), Optimal reliable design and operation of water distribution systems through decomposition, *Water Resour. Res.*, *48*, W10521, doi:10.1029/2011WR011651.

Perelman, L., A. Ostfeld, and E. Salomons (2008), Cross entropy multiobjective optimization for water distribution systems design, *Water Resour. Res.*, *44*, W09413, doi:10.1029/2007WR006248.

Prasad, T. D., and N.-S. Park (2004), Multiobjective genetic algorithms for design of water distribution networks, *J. Water Resour. Plann. Manage.*, *130*(1), 73–82, doi:10.1061/(ASCE)0733-9496(2004)130:1(73).

Raad, D. N., A. N. Sinske, and J. H. van Vuuren (2010), Comparison of four reliability surrogate measures for water distribution systems design, *Water Resour. Res.*, *46*, W05524, doi:10.1029/2009WR007785.

Reddy, M. J., and D. N. Kumar (2007), Multiobjective differential evolution with application to reservoir system optimization, *J. Comput. Civil Eng.*, *21*(2), 136–146.

Rossman, L. A. (2000), *EPANET 2-User Manual*, Natl. Risk Manage. Res. Lab., Off. of Res. and Dev., US Environ. Prot. Agency, Cincinnati, Ohio.
Simpson, A. R., G. C. Dandy, and L. J. Murphy (1994), Genetic algorithms compared to other techniques for pipe optimization, *J. Water Resour. Plann. Manage.*, *120*(4), 423–443.
Storn, R., and K. Price (1995), Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous space, technical report TR-95-012, Int. Comput. Sci. Inst., Berkeley, Calif.
Todini, E. (2000), Looped water distribution networks design using a resilience index based heuristic approach, *Urban Water*, *2*(2), 115–122.
Tolson, B. A., M. Asadzadeh, H. R. Maier, and A. Zecchin (2009), Hybrid discrete dynamically dimensioned search (HD-DDS) algorithm for water distribution system design optimization, *Water Resour. Res.*, *45*, W12416, doi:10.1029/2008WR007673.
Wu, Z. Y., and T. Walski (2005), Self-adaptive penalty approach compared with other constraint-handling techniques for pipeline optimization, *J. Water Resour. Plann. Manage.*, *131*(3), 181–192.
Wu, W., H. R. Maier, and A. R. Simpson (2013), Multiobjective optimization of water distribution systems accounting for economic cost, hydraulic reliability and greenhouse gas emissions, *Water Resour. Res.*, *49*, 1211–1225, doi:10.1002/wrcr.20120.
Zheng, F., A. R. Simpson, and A. C. Zecchin (2011), A combined NLP-differential evolution algorithm approach for the optimization of looped water distribution systems, *Water Resour. Res.*, *47*, W08531, doi:10.1029/2011WR010394.
Zheng, F., A. R. Simpson, A. C. Zecchin, and J. W. Deuerlein (2013a), A graph decomposition-based approach for water distribution network optimization, *Water Resour. Res.*, *49*, 2093–2109, doi:10.1002/wrcr.20175.
Zheng, F., A. R. Simpson, and A. C. Zecchin (2013b), A decomposition and multistage optimization approach applied to the optimization of water distribution systems with multiple supply sources, *Water Resour. Res.*, *49*, 380–399, doi:10.1029/2012wr013160.