

Hybrid metaheuristics for multi-objective design of water distribution systems

Qi Wang, Dragan A. Savić and Zoran Kapelan

ABSTRACT

Multi-objective design of Water Distribution Systems (WDSs) has received considerable attention in the past. Multi-objective evolutionary algorithms (MOEAs) are popular in tackling this problem due to their ability to approach the true Pareto-optimal front (PF) in a single run. Recently, several hybrid metaheuristics based on MOEAs have been proposed and validated on test problems. Among these algorithms, AMALGAM and MOHO are two noteworthy representatives which mix their constituent algorithms in contrasting fashion. In this paper, they are employed to solve a wide range of benchmark design problems against another state-of-the-art algorithm, namely NSGA-II. The design task is formulated as a bi-objective optimisation problem taking cost and network resilience into account. The performance of three algorithms is assessed via normalised hypervolume indicator. The results demonstrate that AMALGAM is superior to MOHO and NSGA-II in terms of convergence and diversity on the networks of small-to-medium size; however, for larger networks, the performance of hybrid algorithms deteriorates as they lose their adaptive capabilities. Future improvement and/or redesign on hybrid algorithms should not only adopt the strategies of adaptive portfolios of sub-algorithms and global information sharing, but also prevent the deterioration mainly caused by imbalance of constituent algorithms.

Key words | hybrid metaheuristics, hypervolume, multi-objective design, resilience, water distribution system

Qi Wang (corresponding author)
Dragan A. Savić
Zoran Kapelan
Centre for Water Systems,
College of Engineering, Mathematics and Physical
Sciences,
University of Exeter,
North Park Road,
Exeter EX4 4QF,
United Kingdom
E-mail: qw212@exeter.ac.uk;
zero3315263@gmail.com

INTRODUCTION

The design of Water Distribution Systems (WDSs) by multi-objective evolutionary algorithms (MOEAs) has attracted considerable attention during recent years (Keedwell & Khu 2003; Prasad & Park 2004; Khu & Keedwell 2005; Farmani *et al.* 2006; Prasad & Tanyimboh 2008; Fu *et al.* 2012a). The primary goal of the MOEA is to generate a trade-off between the total cost and system benefits, while meeting consumer demands and other system constraints (e.g. pressure, velocity, etc.). As combinatorial optimisation problems with Non-deterministic Polynomial-time hard (NP-hard) feature (Papadimitriou & Steiglitz 1998), it is challenging to tackle the design of a real-world WDS as it often incurs expensive computational efforts, especially when extended period simulations are required for objective evaluations (Keedwell & Khu 2003). MOEAs are suitable and

popular for this task due to their ability to approach the true Pareto-optimal front (PF) in a single run (Zitzler & Thiele 1999; Farmani *et al.* 2005a).

Farmani *et al.* (2005a) compared the performance of three commonly used MOEAs, i.e. Non-dominated Sorting Genetic Algorithm II (NSGA-II), Strength Pareto Evolutionary Algorithm 2 (SPEA2) as well as Multi-Objective Genetic Algorithm (MOGA), on multi-objective design of a WDS applying them to two benchmark networks, as well as a large real-life network. They concluded that SPEA2 (Zitzler *et al.* 2002) outperformed other techniques in satisfying both goals of multi-objective optimisation, i.e. closeness to the true PF and diversity among the non-dominated solutions, especially on a large network. Subsequently, Farmani *et al.* (2005b) used NSGA-II to solve an expanded rehabilitation

problem of the Anytown network (Walski *et al.* 1987) as a realistic benchmark taking cost and resilience index (Todini 2000) into account.

In order to yield acceptable near optimal solutions and reduce the overall number of hydraulic evaluations, Keedwell & Khu (2003) investigated the possibility of combining NSGA-II with a neighbour search to solve the multi-objective design of the New York tunnels network. Results showed an encouraging improvement of the hybrid algorithm given a budget of model simulations. Later on, they tried to combine a novel cellular automaton-based initialisation technique with a Genetic Algorithm (GA) to solve the least cost design of a WDS (Keedwell & Khu 2006). The applications to two large networks from industry highlighted the benefits of using this approach to discover better results in a fixed time span.

Besides integrating a local search strategy with current MOEAs, Raad *et al.* (2009) applied a hybrid metaheuristic algorithm, called a multi-algorithm, genetically adaptive multi-objective method (AMALGAM) proposed by Vrugt & Robinson (2007), for the first time to address the optimal design of a WDS considering the total cost and network resilience (Prasad & Park 2004). Instead of using the original sub-algorithms, they employed a greedy design heuristic, two variants of NSGA-II and discrete particle swarm optimisation (PSO) because of their tendency to succeed in a discrete multi-objective optimisation setting. The results obtained from three benchmark models as well as a real WDS in South Africa proved the strength of the AMALGAM-type algorithm as a faster, more reliable tool for multi-objective design of a WDS.

Wolpert & Macready (1997) presented a number of 'no free lunch' theorems and demonstrated the danger of analysing algorithms by their performance on a small set of cases. Most of the previous work tests several MOEAs (often built on different concepts) on quite a few benchmark and/or real-world WDS design problems, therefore, the conclusions might be biased since it is impossible for a specific optimisation algorithm to be effective on a wide range of problems. Hybrid algorithms arise with an attempt to overcome this difficulty by combining the power of different methods. However, many such schemes proposed for WDS design often require the parameters to be fine-tuned, hence the lack of adaptability, robustness and popularity.

In this paper, we applied two recently-proposed hybrid algorithms, i.e. AMALGAM and Multi-Objective Hybrid Optimisation (MOHO), to solve the multi-objective design of a WDS. More specifically, we tested the strength of two different hybrid schemes (Talbi 2002), namely high-level teamwork hybrid (HTH) and high-level relay hybrid (HRH), by conducting the bi-objective optimal design on a wide range of benchmark models collected from the literature, including the Anytown network which is regarded as one of the challenging benchmarks receiving less attention in the past (Prasad & Tanyimboh 2008). The problem was formulated to minimise the total cost and to maximise the network resilience, as defined by Prasad & Park (2004). In order to compare the performance of hybrid algorithms with state-of-the-art MOEAs in the domain, we used NSGA-II to solve the aforementioned problem as well. In addition, with an attempt to clearly evaluate the performance of each algorithm, we employed a well-established indicator, i.e. hypervolume (Deb 2001), to assess the quality of final solutions. Multiple independent optimisation runs were carried out on each problem, which served to generate unbiased evaluation based on statistics. The main contributions of this paper are the investigation of the capability of hybrid metaheuristics to perform multi-objective design of a WDS and comparison of their performance with that of modern MOEAs by extensive testing. Therefore, this work aims to uncover the reasons for success and/or failure of the two algorithms, and in turn, to establish how the hybrid algorithms could benefit from further improvements.

The remainder of this paper is organised as follows: first, multi-objective design of a WDS is briefly introduced followed by the mechanisms of AMALGAM and MOHO in more detail. Then, the benchmark problems used in this paper are summarised and the performance metric is given. After comparing the results obtained from each algorithm, conclusions are drawn at the end.

MULTI-OBJECTIVE DESIGN OF WDSs

The design of a WDS always involves optimising multiple and usually conflicting objectives at the same time, such as, total cost, system reliability and water quality. The goal of multi-objective design of a WDS is to get as close as

possible to the true trade-off between cost and benefit, which offers a range of alternatives for the decision making process. A typical WDS design problem consists of providing cost-effective specification of various components, i.e. pipes, pumps, valves tanks, etc., within the network given the system layout. In a more narrow sense, various investigators considered the design task to be the specification of the best combination of pipe sizes from within a discrete range of commercial diameters that meets the water demand and other system requirements. Herein, we focus on this narrow definition of the problem using bi-objective optimisation to minimise the total capital cost and maximise the performance benefits of the network. The value of the latter objective is calculated based on hydraulic simulation through the EPANET2.0 package (Rossman 2000).

A series of indicators (Todini 2000; Prasad & Park 2004; Prasad & Tanyimboh 2008) have been proposed in the literature as a surrogate of performance benefit giving preference to a 'looped network'. Recently, the resilience index (Todini 2000) has gained more attention due to its ability to account for failure conditions in a risk type measure. It is defined based on the concept that the total input power into a network consists of the power dissipated in the network and the power delivered at demand nodes. In response to Todini's measure, Prasad & Park (2004) developed the network resilience metric by taking the uniformity of pipes connected to a certain node into account. The advantage of the latter is that it explicitly rewards redundancy of similarly sized pipes as improving the reliability of network under pipe failure scenarios (Raad *et al.* 2009). A new approach was recently proposed to provide flexibility to the design of water supply (Zhang & Babovic 2012) by considering innovative Real Options technology. However, this approach deals with the design of water systems under uncertainty which is not considered here.

Given the above and the fact that this paper focuses on the comparison of hybrid metaheuristics for the WDS design, the optimisation methodology presented here is based on the conventional WDS design driven by the trade-off between the WDS design cost and performance, the latter being evaluated by using the network resilience metric.

HYBRID METAHEURISTICS

Unlike the self-contained algorithms, hybrid metaheuristics combine two or more different mechanisms (usually built on population-based evolutionary algorithms) to facilitate the efficiency of the search towards the global optima. In an attempt to classify hybrid algorithms using common terminology, Talbi (2002) presented a taxonomy mechanism for current hybrid metaheuristics in a qualitative way considering both design and implementation issues. The taxonomy combined a hierarchical classification scheme with a flat classification scheme to provide a clear and structural framework for comparative purposes. Here, we mainly focus on the design issues of hybrid algorithms.

At the first level of the hierarchical classification, low-level and high-level hybridisations can be distinguished. This is done by ascertaining whether the component metaheuristics are embedded or self-contained. In the low-level hybrid class, a certain functional part of an algorithm is substituted with another algorithm. While in the high-level hybrid class, each algorithm works on its own without depending on other metaheuristics. At the second level of the hierarchical classification, each class (low-level or high-level hybrid) is further divided into relay and teamwork classes according to the working fashion, i.e. optimising a problem in turn or cooperatively. Therefore, four general types of algorithms are derived from the hierarchical taxonomy, i.e. Low-level Relay Hybrid, Low-level Teamwork Hybrid (LTH), HRH and HTH. According to the flat classification, all abovementioned hybridisation classes can be categorised into homogeneous/heterogeneous, global/partial and specialist/general schemes. In homogeneous hybrids, all the constituent algorithms use the same metaheuristic. While in heterogeneous hybrids, different metaheuristics are employed. The hybrid schemes can also be viewed as global or partial hybrids depending on whether the whole search space will be the same for all the sub-algorithms or decomposed into sub-areas (one for each sub-algorithm). From the perspective of function of metaheuristics, specialist hybrids can be distinguished from general hybrids as they combine sub-algorithms which aim to solve different problems from the others.

Van Zyl *et al.* (2004) proposed an LTH algorithm, which incorporated a hill-climber strategy with a GA method, to solve operational optimisation of a WDS. They concluded that the hybrid algorithm outperformed pure GA by finding good solutions quickly. They also showed that a local search method complemented GA by efficiently finding local optima.

Cisty (2010) combined a GA with Linear Programming (LP) as an LTH to solve three least-cost design problems of a WDS. This method employed a GA to decompose looped network configurations into a group of branched networks. LP was then applied to optimise the branched networks as it was more reliable than heuristic methods in finding the global optimum. The results demonstrated the hybrid's superiority in consistently generating better solutions when compared to GA and Harmony Search.

Tolson *et al.* (2009) extended dynamically dimensioned search (DDS), which is a continuous global optimization algorithm (Tolson & Shoemaker 2007) and developed an LTH (called hybrid discrete DDS, HD-DDS) by introducing two local search strategies. These local search heuristics involved one-pipe change and two-pipe change local moves in the process of solving a discrete, single-objective, constrained WDS design problem. The main advantages of the algorithm were that it does not require fine-tuning of a number of parameters and that it is computationally efficient when compared to GA or PSO. The results obtained (especially on a large network) revealed that it outperformed the state-of-the-art existing algorithms in terms of searching ability and computational efficiency.

As most low-level hybrid schemes commonly combine various local search strategies or a mechanism different from population-based techniques into the structure of evolutionary algorithms, they turn out to be tailored to cope with specific problems. This is most often done by experimenting with a rule that determines when to switch from one algorithm to another. However, this makes such a hybrid less flexible as it would generally fail to adapt to other applications. On the other hand, few of these low-level hybrid algorithms are designed for multi-objective optimisation except Creaco & Franchini (2012). Since the main concern of this paper is about multi-objective design of a WDS, herein, we focus on the comparison of two different high-level hybrid schemes, i.e. HTH and HRH, which employ

the population-based evolutionary algorithms in contrasting fashions. In particular, two instances of high-level hybrid scheme are analysed by solving the bi-objective design problems using 12 WDS benchmark networks collected from the literature.

Instance of HTH: AMALGAM

Vrugt & Robinson (2007) proposed a multi-algorithm, genetically adaptive multi-objective method, known as AMALGAM. This can be classified as an HTH, heterogeneous, global, general framework. It simultaneously employs four sub-algorithms within the framework, including NSGA-II, PSO, adaptive metropolis search (AMS) and differential evolution (DE). The main aim of the developed algorithm was to overcome the drawbacks, as well as possible failure of an individual algorithm on a specific problem. The new concepts of multi-method search and genetically adaptive offspring creation are developed to ensure a fast, reliable and computationally efficient algorithm for multi-objective optimisation. Results on a set of well-known multi-objective test functions suggest that this hybrid method achieved a tenfold improvement in convergence metric (Deb *et al.* 2002) over NSGA-II for the more complex, higher dimensional problems. Besides its extraordinary performance, AMALGAM provides a general template which is flexible and extensible, and could easily accommodate any other population-based algorithms. A sequential version of AMALGAM code was requested from Vrugt for this work. The pseudocode of AMALGAM is illustrated in Figure 1.

The parameter settings of three population-based sub-algorithms within AMALGAM are summarised in Table 1. Besides using GA, PSO and DE, AMALGAM also includes AMS as a Markov Chain Monte Carlo (MCMC) sampler that proactively avoids the search being trapped in local optima. The algorithm works by substituting the parents with offspring of lower fitness (Haario *et al.* 2001). This sampler also shows superior efficiency in exploring the search space of high-dimensionality. Therefore, AMS is capable of rapidly travelling across the entire Pareto distribution when the optimisation process progresses towards the PF. Readers are referred to the supporting information of (Vrugt & Robinson 2007) for more details.


```

Initialise population  $P_0$  of size  $N$  by using Latin hypercube sampling (LHS)
Sort  $P_0$  using the fast non-dominated sorting (FNS) algorithm (Deb et al. 2002)
Generate offspring  $Q_0$  of size  $N$  using four sub-algorithms (GA, PSO, AMS, DE) at equal number
Set  $t=0$ , and  $T$ =maximum number of generations
While ( $t < T$ )
    Combine  $P_t$  with  $Q_t$  as an intermediate population  $R_t$  of size  $2N$ 
    Apply FNS to  $R_t$ 
    Select members of size  $N$  from  $R_t$  to form  $P_{t+1}$  based on the rank and crowding distance
    Calculate the contribution of offspring obtained by each sub-algorithm in  $P_{t+1}$ 
    Update the number of offspring creation by each sub-algorithm based on reproductive success
    Generate offspring  $Q_{t+1}$  of size  $N$  using four sub-algorithms at specified number
     $t=t+1$ 
End

```

Figure 1 | Pseudocode of AMALGAM.

Table 1 | Setting of parameters in AMALGAM

GA		PSO		DE	
Crossover rate	0.9	Inertia factor	$0.5 + 0.5u(0,1)$	Scaling factor 1	$u(0.6,1.0)$
Mutation rate	$1/L$	Cognitive weight	1.5	Scaling factor 2	$u(0.2,0.6)$
Distribution index for crossover	20	Social weight	1.5	–	–
Distribution index for mutation	20	Turbulence factor	$u(-1,1)$	–	–

Note: L is the number of decision variables; $u(a,b)$ is a uniform random number between a and b .

Instance of HRH: MOHO

Moral & Dulikravich (2008) focused on another hybrid scheme following the concept of Pareto-dominance. They presented an MOHO algorithm as a HRH, heterogeneous, global, general metaheuristic which implements three sub-algorithms in a sequential manner. The MOHO hybrid coordinates SPEA2, Multi-Objective Particle Swarm Optimisation (MOPSO) and Non-dominated Sorting Differential Evolution (NSDE) and decides which one of them will generate offspring using the automatic switching procedure. More specifically, MOHO proceeds by choosing one of them for producing the next generation based on the performance of the currently employed algorithm. Five different indicators for measuring improvements on finding non-dominated solutions, including the quality of approximation and distribution, are used to decide whether to continue with a particular algorithm or change to another one. In this paper, we are not able to implement the original MOHO software; instead, we tried to recreate it following

the primary idea in a Matlab environment. Figure 2 shows the pseudocode of MOHO.

MOHO evaluates the performance of its sub-algorithms on five distinct improvements: (1) changes in the size of non-dominated set; (2) whether there exists a solution from the new generation which dominates any members in the last generation; (3) changes in the hypervolume indicator; (4) changes in average Euclidian distance; (5) increase in the spread indicator. The innovative part of this evaluation strategy is that MOHO considers not only the quality of the non-dominated set in the next generation (i.e. in terms of convergence and diversity), but also takes into account the perturbation introduced by the potential solutions which may bring substantial improvement in later iterations. The main differences between the original MOHO and the one reported here are twofold. First, the initial population is generated using uniformly distributed random sampling rather than Sobol's quasi-random sequence generator (Bratley & Fox 1988) as the advantage of this method vanishes for higher dimensional problems (Rahnamayan *et al.* 2007).

```

Initialise the first population  $P_0$  of size  $N$  by uniformly distributed random sampling
Sort  $P_0$  using the FNS algorithm (Deb et al. 2002)
Select one of sub-algorithms (i.e. SPEA2, MOPSO or NSDE) randomly to generate offspring  $Q_0$ 
Set  $t=0$ , and  $T$ =maximum number of generations
While ( $t < T$ )
    Combine  $P_t$  with  $Q_t$  as an intermediate population  $R_t$  of size  $2N$ 
    Apply FNS to  $R_t$ 
    Select members of size  $N$  from  $R_t$  to form  $P_{t+1}$  based on the rank and crowding distance
    Evaluate the performance of currently employed sub-algorithm in  $P_{t+1}$  on five aspects
    Decide whether to continue this sub-algorithm or switch to another one
    Generate offspring  $Q_{t+1}$  of size  $N$  using the selected sub-algorithm in previous step
     $t=t+1$ 
End

```

Figure 2 | Pseudocode of MOHO.

Secondly, since the parameter settings of each sub-algorithm were not clearly stated in the original MOHO, we configure these values by trial-and-error method on some difficult test functions and choose the best combination based on the experimental results. Additionally, the maximum number of consecutive iterations of a certain sub-algorithm is set to 1/50 of total generations.

Greater details about two hybrid algorithms and their performance can be found in the original authors' papers (Vrugt & Robinson 2007; Moral & Dulikravich 2008). Apart from using hybrid algorithms with distinct schemes, we also applied NSGA-II to solve the benchmark problems for the purpose of comparison of the quality of final solutions. For more details about NSGA-II, the readers are referred to Deb *et al.* (2002). The latest version of NSGA-II (revision 1.1.6) was downloaded from the website of Kanpur Genetic Algorithms laboratory (<http://www.iitk.ac.in/kangal/codes.shtml>).

CASE STUDIES

Benchmark problems

To well compare the performance of AMALGAM and MOHO against NSGA-II, 12 WDS networks were collected from the literature and served as benchmarks for optimisation tests. The number of pipes in these models ranges from eight to 454, which, together with various design criteria, provide a wide range of problems and search spaces

with the number of candidate solutions ranging between 10^7 and 10^{454} . The name of benchmark models, number of pipes, diameter options and relevant design criteria are summarised in Table 2.

It is worth mentioning that four benchmark models, BLA, FOS, PES and MOD, adopted from Bragalli *et al.* (2008) are more realistic compared with others (except ANT) as they all take a reasonable range of pressure head (not only minimum pressure requirement) as well as the upper bound on flow velocity in the network. Although ANT was introduced as a hypothetical network, it contains most common features (multiple loading conditions, pipe duplication or reconditioning (i.e. cleaning and re-lining), new pipe installation, tank location and operation as well as pump scheduling) found in many real systems. For a detailed description of design criteria on each model, interested readers are referred to Dong *et al.* (2012), Raad (2011) as well as via <http://centres.exeter.ac.uk/cws>.

Performance indicator

It should be emphasised here that there is no ideal indicator which can give consistent and definite evaluation of both convergence and diversity of multi-objective optimisation. Among the various metrics which are designed to measure the achievement of MOEAs, it is established that hypervolume (HV) is a single metric which can assess the performance of both aspects in a combined sense (Deb 2001). In order to remove the bias caused by the magnitude

Table 2 | Benchmark models used for comparison of each algorithm

No.	Model	Pipe Count	Option Count	Design Criteria			
				Min Head	Max Head	Max Velocity	Multiple Loading Condition
1	Two-loop Network (TLN)	8	14	Yes	No	No	No
2	BakRyan Network (BAK)	9	11	Yes	No	No	No
3	New York Tunnel Network (NYT)	21	16	Yes	No	No	No
4	Blacksburg Network (BLA)	23	15	Yes	Yes	Yes	No
5	GoYang Network (GOY)	30	8	Yes	No	No	No
6	Hanoi Network (HAN)	34	6	Yes	No	No	No
7	Fossolo Network (FOS)	58	22	Yes	Yes	Yes	No
8	Pescara Network (PES)	99	13	Yes	Yes	Yes	No
9	Modena Network (MOD)	317	13	Yes	Yes	Yes	No
10	Balerma Irrigation Network (BIN)	454	10	Yes	No	No	No
11	Two Reservoir Network (TRN)	8	8	Yes	No	No	Yes
12	Anytown Network (ANT)	43	10	Yes	No	No	Yes

Note: For TRN network, three of eight pipes are existing pipes which have three options including 'do nothing', cleaning or duplication; for ANT network, although there are only 43 pipes to be considered, its formulation contains up to 112 decision variables, which makes it the most challenging problem in the list.

of different objective functions, we take the normalised version of HV, called the ratio of the HV of approximation set and of true Pareto-optimal front (HVR) (Deb 2001), to evaluate the quality of final solutions obtained from each algorithm. The expression of HV and HVR are shown as Equations (1) and (2), respectively:

$$HV = \text{volume} \left(\bigcup_{i=1}^{|Q|} v_i \right); \quad (1)$$

$$HVR = \frac{HV(Q)}{HV(P^*)} \quad (2)$$

where v_i is the hypercube constructed with a reference point (normally a vector of worst objective values) and the solution i as the diagonal corners; Q is the non-dominated solutions obtained by an algorithm and P^* is the solutions in the true PF.

Since we do not have a theoretical true PF for each benchmark problem, to assist the evaluation of performance, a quasi-true Pareto-Optimal front (quasi-PF) was generated for each problem. This was achieved by applying a non-dominated sorting procedure to the aggregated Pareto fronts obtained by all three algorithms through multiple runs.

RESULTS AND DISCUSSION

The benchmark networks adopted in this paper encompass a wide range of network sizes, with up to several hundreds of pipes. Hence, various computational budgets (Table 3) were tested to make sure each algorithm converged well before their performance could be compared. It is worth noting that these budgets (i.e. population size and number of generations for each benchmark problem) are kept the same for all three algorithms. As such, the number of function evaluations via EPANET2.0 (Rossman 2000) varied from 25,000 to 500,000. Because each algorithm produces a first generation in a different way, multiple runs are implemented to eliminate the

Table 3 | Configuration of computational budget

Population Size	Generation	Pipe No.	Problems
100	250	≤ 50	TLN, BAK, NYT, BLA, GOY, HAN, TRN
	500	≤ 100	FOS, PES
	1000	≤ 500	MOD, BIN
	5000	N/A	ANT

Note: The numbers of population size and generation (except ANT) are decided based on trial runs in order to ensure the convergence of NSGA-II given the specified computational budgets. The number of generation on the ANT problem follows the same setting chosen by Farmani et al. (2005b).

influence of the initial population, and the statistical results of HVR are used to assess their performance. Thirty independent runs were carried out for all cases except ANT, which was run for 10 times as it requires many more generations to ensure convergence and thus is extremely time-consuming.

Figure 3 shows the box plot of statistical performances of three algorithms on 12 benchmark problems. The top and bottom edges of the grey bar in each plot represent the maximum and minimum values of HVR for each algorithm, respectively. The intermediate short lines in dark colour denote the average values of HVR for each algorithm.

Reference points are also provided in terms of cost (in million units) and network resilience values. For example, for TLN the reference point is (5.0, 0.1). The results clearly demonstrate that AMALGAM consistently outperforms MOHO and NSGA-II on the networks of small-to-medium size (Wang *et al.* 2012), i.e. TLN, BAK, NYT, GOY, FOS, PES, MOD and TRN. The performance of MOHO was comparable to that of AMALGAM and NSGA-II on smaller networks, i.e. TLN, BAK, NYT, BLA, GOY and TRN; however, it became less efficient on larger networks, i.e. HAN, FOS, PES, MOD, BIN and ANT, as the complexity of the

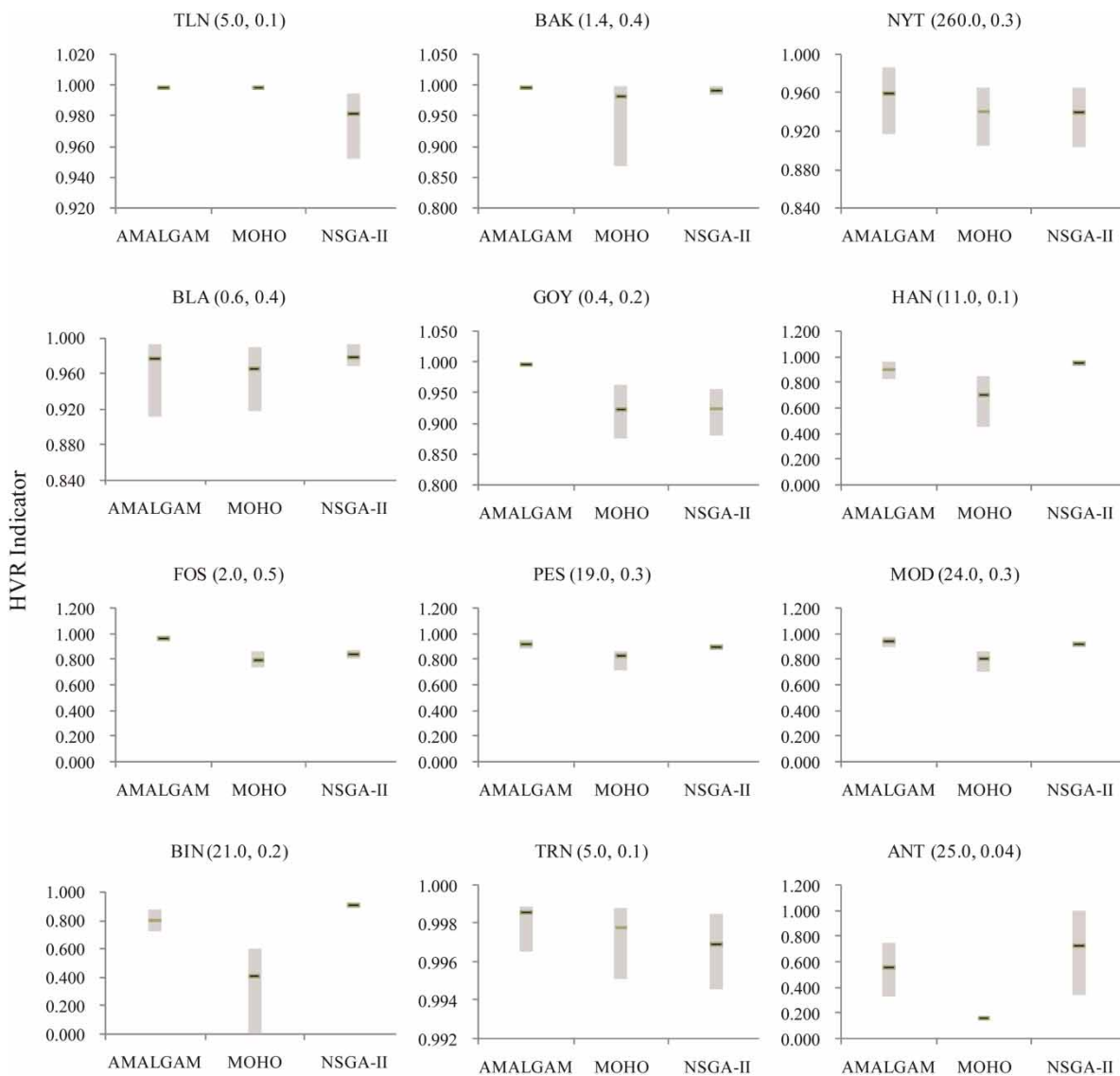


Figure 3 | Statistical performances of each algorithm on each problem using HVR indicator.

problem increased. Even worse, it was able to find only one feasible solution on the ANT problem in 10 runs. On smaller networks (less than 400 pipes), NSGA-II performed worse than hybrid algorithms except on the BLA and HAN problems; on the contrary, it dominated hybrid algorithms on larger networks, i.e. BIN and ANT. Admittedly, none of the algorithms converged on the ANT problem, which also implies that it was the most complex problem in the selected cases by considering many aspects simultaneously. Furthermore, it is important to emphasise that the convergence of MOHO and NSGA-II were highly dependent on initial random seeds. For instance, twice out of 10 runs, improper seeds resulted in complete failure of NSGA-II as there were no feasible solutions found in the final population. By contrast, AMALGAM, which uses Latin hypercube sampling

for initialisation, successfully discovered non-dominated solutions all the time.

Another way to compare the performance of three algorithms is to illustrate their contributions to the Pareto front obtained via multiple runs on each case (see Figure 4). Herein, only four cases, namely NYT, HAN, PES, and BIN, are chosen as they exhibit different levels of complexity within the problems considered in the paper. Each figure is produced in the following manner. Firstly, the objective function values of the non-dominated solutions obtained by each algorithm (via 30 runs) are rounded to four-digit precision and the duplicate solutions are removed. Next, the quasi-PF for each case is generated using the non-dominated sorting procedure (Deb *et al.* 2002). Seven data sets are then obtained by counting the common contribution of all three algorithms (denoted as

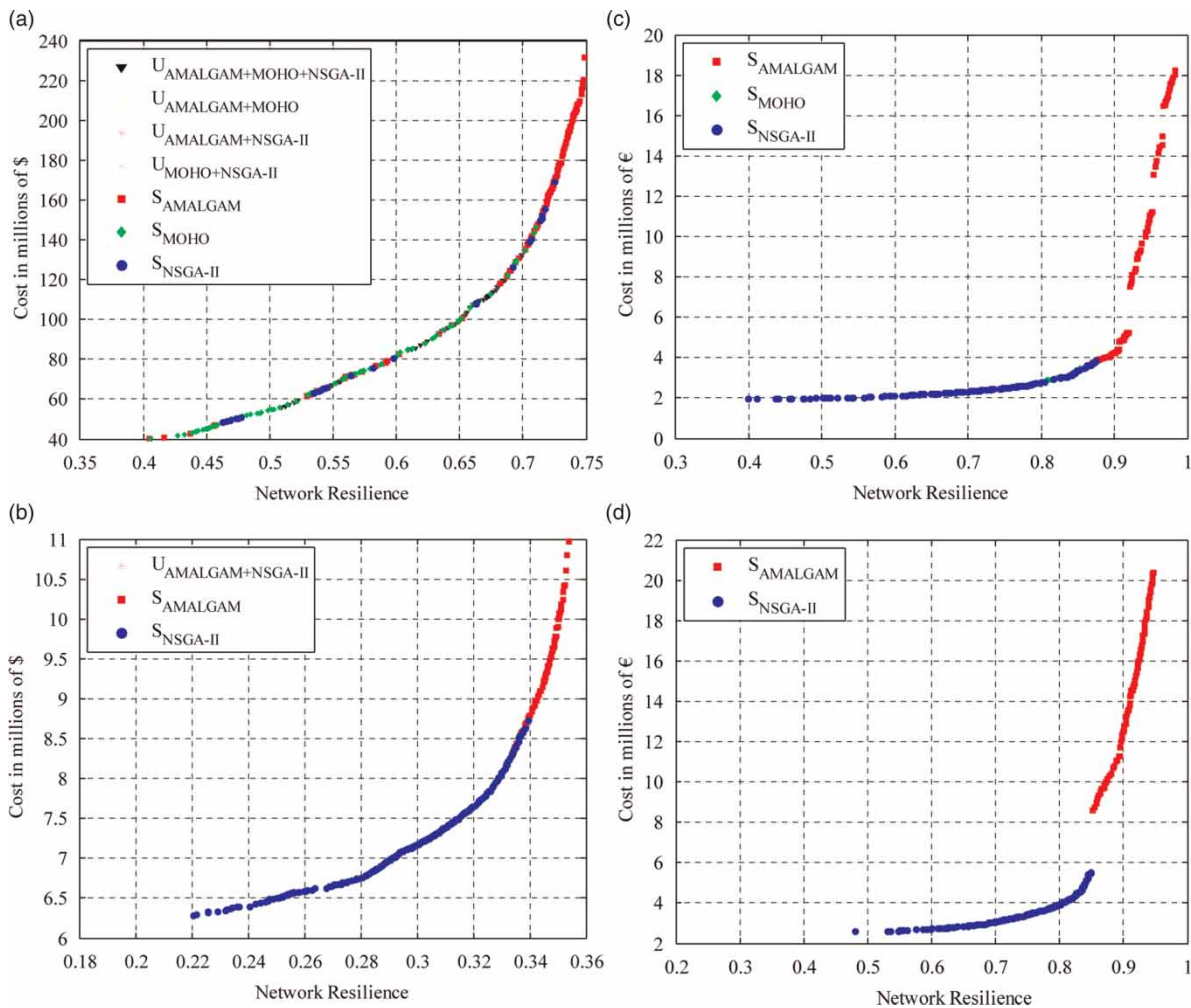


Figure 4 | Pareto fronts obtained via multiple runs by AMALGAM, MOHO, and NSGA-II. (a) Case NYT, (b) Case HAN, (c) Case PES, (d) Case BIN.

$U_{AMALGAM+MOHO+NSGA-II}$), common contribution of every two algorithms (denoted as $U_{AMALGAM+MOHO}$, $U_{AMALGAM+NSGA-II}$, $U_{MOHO+NSGA-II}$), and individual contributions of each algorithm (denoted as $S_{AMALGAM}$, S_{MOHO} , $S_{NSGA-II}$), which has already excluded the common ones. Finally, these data sets are plotted in Figure 4. It should be noted that these sets can be empty and therefore are not necessarily shown on the figure. Similar algorithm performance trends can be observed as discussed previously. AMALGAM was consistently superior to the others in terms of diversity by identifying the solutions in the region of high network resilience. NSGA-II outperformed hybrid algorithms in terms of convergence towards the region of low cost, especially on larger networks (i.e. HAN, PES, and BIN) while MOHO was able to find solutions in the quasi-PFs of NYT and PES, albeit completely failing on HAN and BIN problems.

To compare quantitatively the contributions of each algorithm, Table 4 summarises the percentage of solutions

obtained by each algorithm in the quasi-PF set for each test case. On seven of these benchmark problems (mainly on larger networks), NSGA-II found a significant number of solutions in the quasi-PF sets. For smaller test cases, like TLN and TRN, its contribution was similar to that of AMALGAM and MOHO. It was worth noting that on the ANT problem the quasi-PF was comprised solely of the solutions obtained by NSGA-II. This highlighted the superiority of NSGA-II in terms of convergence given a fixed computational budget. Conversely, AMALGAM successfully produced more solutions on five small-to-medium size networks when compared to NSGA-II. Such performance was due to its better achievement in terms of diversity and convergence. It can also be observed that AMALGAM always found extreme points of the quasi-PF sets in the region of high network resilience, which were often neglected by NSGA-II and MOHO. Interestingly, MOHO failed to generate any members in the quasi-PF sets on HAN, MOD, BIN and ANT. Furthermore, it only found a feasible solution set once out of 10 runs on the ANT problem.

In order to investigate the reasons why hybrid algorithms failed on some cases, the evolutionary processes of each sub-algorithm within AMALGAM and MOHO on all design problems were recorded and analysed. Four cases, i.e. HAN, PES, BIN and ANT, were selected and discussed here as they represented the most difficult ones under limited computational budget levels, i.e. 250, 500, 1000 and 5000 generations, respectively. Since the bottom line of creating offspring points in AMALGAM was maintained at 5, the number of individuals provided by a specific sub-algorithm was expected to vary between 5 and 85. As shown in Figure 5, for the HAN problem, AMS outperformed

Table 4 | Percentage of contribution from each algorithm for each design problem

Contribution in percentage (%)						
Problem	TLN	BAK	NYT	BLA	GOY	HAN
AMALGAM	88	100	67	52	62	25
MOHO	91	98	60	21	32	0
NSGA-II	98	96	39	31	42	78
Contribution in percentage (%)						
Problem	FOS	PES	MOD	BIN	TRN	ANT
AMALGAM	31	38	56	38	98	0
MOHO	31	12	0	0	95	0
NSGA-II	38	50	44	62	99	100

Note: The maximum contribution to each problem is shown in boldface.

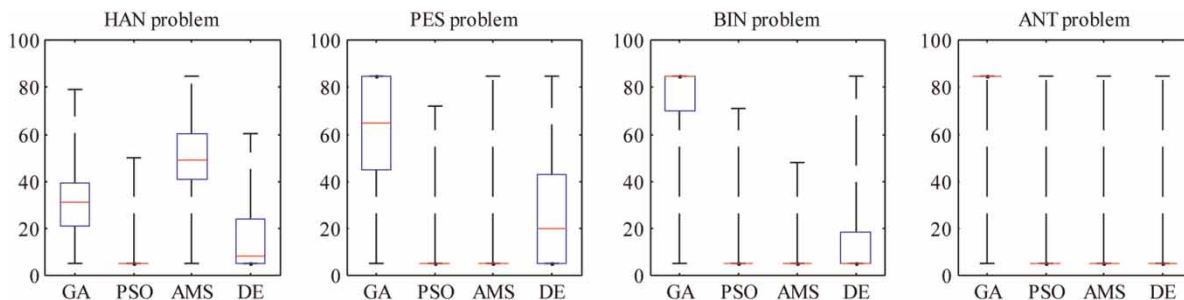


Figure 5 | Statistical performances of sub-algorithms within AMALGAM on four selected cases.

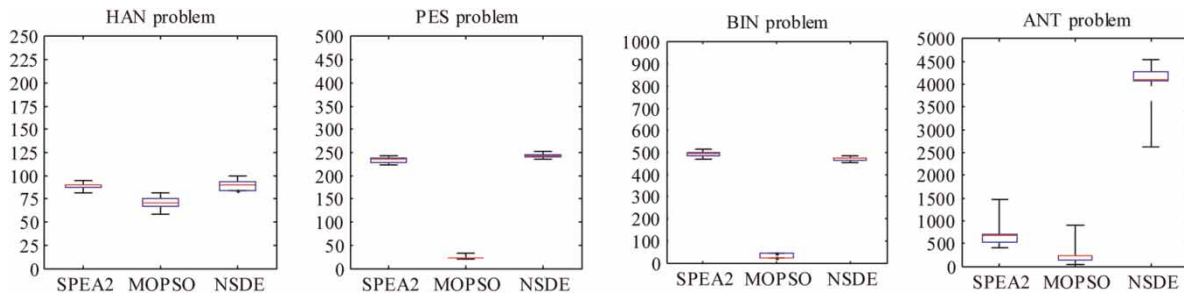


Figure 6 | Statistical performances of sub-algorithms within MOHO on four selected cases.

the other three sub-algorithms by generating a median value of 50 points within the 250 generations. GA worked better than DE followed by PSO which always stayed around the bottom line. However, this behaviour changes steadily from less complex (i.e. PES) to more complex (ANT) problems as GA consistently dominated other sub-algorithms. Only DE was comparable to GA on the PES and BIN problems, while PSO and AMS seldom made a contribution to the population and stayed at the minimum level most of the time. For the ANT problem, GA steadily produced most offspring. In other words, AMALGAM behaved like NSGA-II. Therefore, the failure of AMALGAM on the ANT problem could be attributed to the fact that PSO, AMS and DE were not effective and consequently wasted search resources. In three of the four selected cases and the MOD problem, MOHO completely failed to contribute any solutions in sets of the quasi-PF. In Figure 6, it can be observed that MOPSO was inefficient especially on large networks as on average it ran less than 1/20 of total iterations. Although SPEA2 was comparable with NSDE on the first three cases, it was not selected to produce a next generation on the ANT problem. This resulted in MOHO working similarly to NSGA-II while wasting nearly 20% of iterations to explore the search space. Another explanation for MOHO's inefficiency is that the adaptive feature may be significantly weakened as the inefficiency of a certain constituent algorithm produces poor solutions.

CONCLUSIONS

Two hybrid algorithms, namely AMALGAM and MOHO, as well as NSGA-II were applied to a wide range

of multi-objective design of WDS benchmark networks. AMALGAM employs four sub-algorithms simultaneously and adapts offspring creation genetically based on the success rate of each algorithm in producing the next population. MOHO, on the other hand, selects in sequence when to switch from one of its sub-algorithms to another by monitoring performance on five separate aspects. NSGA-II was used as a representative of state-of-the-art MOEAs for the purpose of comparison. Multiple independent runs were carried out on each test cases and the HVR metric was adopted to assess their performance in terms of convergence and diversity.

The results clearly reveal that AMALGAM (HTH scheme) is superior to NSGA-II on the networks of small-to-medium size, which indicates that this achievement benefits from the strategies of adaptive multi-method search and global information sharing. On the other hand, the HTH scheme has potential to achieve better performance compared to the HRH scheme through taking full advantage of each sub-algorithm more efficiently. However, on larger networks, the behaviour of hybrid algorithms gradually deteriorated or completely failed. The underlying reason why hybrid metaheuristics perform worse on larger networks was also investigated by monitoring the evolutionary process of its sub-algorithms in detail. The failure is attributed to the loss of effectiveness in terms of proactive adaptation. Actually, it is observed that, on the ANT problem, AMALGAM performed nearly the same as NSGA-II because GA dominated other sub-algorithms completely most of the time.

Admittedly, there is still a lack of theoretical analysis in the literature about the impact of problem characteristics on the performance of metaheuristics, which makes them and

associated hybrid methods (like AMALGAM and MOHO in this paper) as black-box approaches and thus results in them receiving criticism. Future work on this aspect is needed to change this situation substantially. There is also a gap between the design and application stages of hybrid schemes, which verifies the effectiveness and efficiency of a specific combination of different sub-algorithms from a mathematical point of view. Without this step, it can be misleading when creating a new hybrid scheme. Moreover, the parameterisation issue of hybrid algorithms should be carefully investigated giving consideration to different problem characteristics.

In addition, with the development of both hardware and software in computer technology, the computational capacity of modern PCs has been significantly improved; hence, we suggest that any newly-developed hybrid frameworks or MOEAs should be tested on a wide range of benchmark networks as shown in this work. Furthermore, considerable attention should be focused on the networks of medium-to-large size which give sufficient consideration to the requirements of real-world cases. On the other hand, there are additional concerns other than cost and reliability (e.g. water quality issues) in real cases. The multi-objective design of a WDS may need to adapt to a many-objective (more than three objectives) design process (Fu *et al.* 2012b). Thus, the future development of hybrid metaheuristics should cope with the expansion of dimensionality in both objective function space and decision variable space.

REFERENCES

- Bragalli, C., D'Ambrosio, C., Lee, J., Lodi, A. & Toth, P. 2008 Water Network Design by MINLP. IBM Research Report.
- Bratley, P. & Fox, B. L. 1988 Algorithm 659: implementing Sobol's quasirandom sequence generator. *ACM Trans. Math. Softw.* **14** (1), 88–100.
- Cisty, M. 2010 Hybrid genetic algorithm and linear programming method for least-cost design of water distribution systems. *Water Resour. Manage.* **24**, 1–24.
- Creaco, E. & Franchini, M. 2012 Fast network multi-objective design algorithm combined with an a posteriori procedure for reliability evaluation under various operational scenarios. *Urban Water J.* **9** (6), 385–399.
- Deb, K. 2001 *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Chichester, UK.
- Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. 2002 A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **6** (2), 182–197.
- Dong, X., Liu, S., Tao, T., Li, S. & Xin, K. 2012 A comparative study of differential evolution and genetic algorithms for optimizing the design of water distribution systems. *J. Zhejiang Univ-Sci A (Appl. Phys. Eng.)* **13** (9), 674–686.
- Farmani, R., Savic, D. A. & Walters, G. A. 2005a Evolutionary multi-objective optimization in water distribution network design. *Eng. Optim.* **37** (2), 167–183.
- Farmani, R., Walters, G. A. & Savic, D. A. 2005b Trade-off between total cost and reliability for Anytown water distribution network. *J. Water Res. Plan. Manage.* **131** (3), 161–171.
- Farmani, R., Walters, G. & Savic, D. 2006 Evolutionary multi-objective optimization of the design and operation of water distribution network: Total cost vs. reliability vs. water quality. *J. Hydroinf.* **8** (3), 165–179.
- Fu, G., Kapelan, Z. & Reed, P. 2012a Reducing the complexity of multiobjective water distribution system optimization through global sensitivity analysis. *J. Water Res. Plan. Manage.* **138** (3), 196–207.
- Fu, G., Kapelan, Z., Kasprzyk, J. & Reed, P. 2012b Optimal design of water distribution systems using many-objective visual analytics. *J. Water Res. Plan. Manage.* 10.1061/(ASCE)WR.1943-5452.0000311.
- Haario, H., Saksman, E. & Tamminen, J. 2001 An adaptive metropolis algorithm. *Bernoulli* **7** (2), 223–242.
- Keedwell, E. & Khu, S. 2006 Novel cellular automata approach to optimal water distribution network design. *J. Comput. Civil Eng.* **20** (1), 49–56.
- Keedwell, E. C. & Khu, S. T. 2003 *More Choices in Water System Design Through Hybrid Optimisation*. Computing and Control for the Water Industry 2003, London, UK, pp. 257–264.
- Khu, S.-T. & Keedwell, E. 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.
- Moral, R. J. & Dulikravich, G. S. 2008 Multi-objective hybrid evolutionary optimization with automatic switching among constituent algorithms. *AIAA J.* **46** (3), 673–681.
- Papadimitriou, C. H. & Steiglitz, K. 1998 *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, New York.
- Prasad, T. D. & Park, N.-S. 2004 Multiobjective genetic algorithms for design of water distribution networks. *J. Water Res. Plan. Manage.* **130** (1), 73–82.
- Prasad, T. D. & Tanyimboh, T. T. 2008 Entropy based design of “Anytown” water distribution network. In: *Water Distribution Systems Analysis 2008* (J. E. Van Zyl, A. A. Illembade & H. E. Jacobs, eds). ASCE, Kruger National Park, South Africa, pp. 450–461.
- Raad, D. N. 2011 *Multi-objective Optimisation of Water Distribution Systems Design Using Metaheuristics*. University of Stellenbosch, Stellenbosch.
- Raad, D., Sinske, A. & Van Vuuren, J. 2009 Robust multi-objective optimization for water distribution system design using a meta-metaheuristic. *Int. Trans. Oper. Res.* **16** (5), 595–626.

- Rahnamayan, S., Tizhoosh, H. R. & Salama, M. M. A. 2007 [A novel population initialization method for accelerating evolutionary algorithms](#). *Comput. Math. Appl.* **53** (10), 1605–1614.
- Rossman, L. A. 2000 *EPANET 2 Users Manual*. U.S. Environment Protection Agency, Cincinnati, Ohio, USA.
- Talbi, E. G. 2002 [A taxonomy of hybrid metaheuristics](#). *J. Heuristics* **8**, 541–564.
- Todini, E. 2000 [Looped water distribution networks design using a resilience index based heuristic approach](#). *Urban Water* **2** (2), 115–122.
- Tolson, B. A. & Shoemaker, C. A. 2007 [Dynamically dimensioned search algorithm for computationally efficient watershed model calibration](#). *Water Resour. Res.* **43**, W01413.
- Tolson, B. A., Asadzadeh, M., Maier, H. R. & Zecchin, A. 2009 [Hybrid discrete dynamically dimensioned search \(HD-DDS\) algorithm for water distribution system design optimization](#). *Water Resour. Res.* **45**, W12416.
- Van Zyl, J. E., Savić, D. A. & Walters, G. A. 2004 [Operational optimization of water distribution systems using a hybrid genetic algorithm](#). *J. Water Resour. Plan. Manage.* **130** (2), 160–170.
- Vrugt, J. A. & Robinson, B. A. 2007 [Improved evolutionary optimization from genetically adaptive multimethod search](#). *Proc. Natl. Acad. Sci. USA* **104** (3), 708–711.
- Walski, T. M., Brill, J. E. D., Gessler, J., Goulter, I. C., Jeppson, R. M., Lansey, K., Lee, H.-L., Liebman, J. C., Mays, L., Morgan, D. R. & Ormsbee, L. 1987 [Battle of the network models: Epilogue](#). *J. Water Res. Plan. Manage.* **113** (2), 191–205.
- Wang, Q., Savić, D. & Kapelan, Z. 2012 [Hybrid optimisation algorithms for multi-objective design of water distribution systems](#). *10th International Conference on Hydroinformatics*, Hamburg, Germany.
- Wolpert, D. H. & Macready, W. G. 1997 [No free lunch theorems for optimization](#). *IEEE Trans. Evol. Comput.* **1** (1), 67–82.
- Zhang, S. X. & Babovic, V. 2012 [A real options approach to the design and architecture of water supply systems using innovative water technologies under uncertainty](#). *J. Hydroinf.* **14** (1), 13–29.
- Zitzler, E. & Thiele, L. 1999 [Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach](#). *IEEE Trans. Evol. Comput.* **3** (4), 257–271.
- Zitzler, E., Laumanns, M. & Thiele, L. 2002 [SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization](#). In: *Evolutionary Methods for Design, Optimisation and Control* (K. Giannakoglou, D. Tsahalis, J. Periaux, K. Papailiou & T. Fogarty, eds). International Center for Numerical Methods in Engineering (CIMNE), Barcelona, Spain, pp. 95–100.

First received 17 January 2013; accepted in revised form 24 June 2013. Available online 24 July 2013