Hybrid Pareto archived dynamically dimensioned search for multi-objective combinatorial optimization: application to water distribution network design

Masoud Asadzadeh and Bryan Tolson

ABSTRACT

Pareto archived dynamically dimensioned search (PA-DDS) has been modified to solve combinatorial multi-objective optimization problems. This new PA-DDS algorithm uses discrete-DDS as a search engine and archives all non-dominated solutions during the search. PA-DDS is also hybridized by a general discrete local search strategy to improve its performance near the end of the search. PA-DDS inherits the simplicity and parsimonious characteristics of DDS, so it has only one algorithm parameter and adjusts the search strategy to the user-defined computational budget. Hybrid PA-DDS was applied to five benchmark water distribution network design problems and its performance was assessed in comparison with NSGAII and SPEA2. This comparison was based on a revised hypervolume metric introduced in this study. The revised metric measures the algorithm performance relative to the observed performance variation across all algorithms in the comparison. The revised metric is improved in terms of detecting clear differences between approximations of the Pareto optimal front. Despite its simplicity, Hybrid PA-DDS shows high potential for approximating the Pareto optimal front, especially with limited computational budget. Independent of the PA-DDS results, the new local search strategy is also shown to substantially improve the final NSGAII and SPEA2 Pareto fronts with minimal additional computational expense.

Key words | combinatorial problems, heuristic optimization, hypervolume performance metric, local search, multi-objective optimization, water distribution network design

Masoud Asadzadeh (corresponding author)
Bryan Tolson
Department of Civil and Environmental
    Engineering,
University of Waterloo,
200 University Avenue West,
Waterloo, ON N2L 3G1,
Canada
E-mail: *masadzad@engmail.uwaterloo.ca*

INTRODUCTION

Developing multi-objective optimization (MOO) algorithms has been an active area of research for many years. Schaffer (1984) introduced the vector evaluated genetic algorithm (GA) based on the GA to solve multi-objective problems (MOPs). Since then, several other multi-objective evolutionary algorithms (MOEAs) have been introduced based on single objective EAs. Example algorithms include PAES by Knowles & Corne (2000), SPEA2 by Zitzler *et al.* (2001), NSGAII by Deb *et al.* (2002), MOPSO by Parsopoulos & Vrahatis (2002), MO shuffled complex evolution by Vrugt *et al.* (2003) and MO cross entropy by Perelman *et al.* (2008). MOEAs generate a population of solutions and evolve it generation by generation until a termination criterion occurs. They archive some or all non-dominated solutions and use them as elite solutions for new generations.

Ideally, solving a combinatorial MOP should identify the complete set of Pareto optimal solutions. These solutions are not dominated by any other feasible solution. Usually, finding all Pareto optimal solutions is very time consuming, if not impossible. Therefore, the goal of solving a MOP is to find a set of solutions that represents an approximate front as close as possible to the Pareto optimal front (proximity) and as broadly spread out as possible (diversity).

Asadzadeh & Tolson (2009) introduced the Pareto archived dynamically dimensioned search (PA-DDS) MOO

algorithm for solving problems with continuous decision variables and showed good comparative performance on multi-objective test problems with two objectives. PA-DDS uses dynamically dimensioned search (DDS) (Tolson & Shoemaker 2007) as a search engine and archives all non-dominated solutions during the search as in $(1 + 1)$-PAES (Knowles & Corne 2000). DDS is a simple single-solution-based, single objective optimization algorithm that has only one algorithm parameter with a robust default setting such that tuning it for each problem is not recommended (Tolson & Shoemaker 2007). DDS also adjusts the search strategy to the user input computational budget. In this study, the DDS component of PA-DDS is substituted with the discrete DDS (D-DDS) component of hybrid discrete DDS (HD-DDS) (Tolson *et al.* 2009) to solve combinatorial MOPs.

Hybridizing MOO algorithms

Local search techniques can be used to improve proximity (Ishibuchi & Murata 1996; Jaszkiewicz 2002) and/or diversity (Talbi *et al.* 2001; Bosman & De Jong 2006) of the MOP solution. Heuristic neighbourhood search strategies such as hill-climbing, simulated annealing, and Tabu search are probably the most common approaches to hybridize MOO algorithms (example applications are Ishibuchi & Murata 1996; Knowles & Corne 2000; Deb & Goel 2001; Talbi *et al.* 2001; Jaszkiewicz 2002; Kleeman *et al.* 2007). However, Brown & Smith (2003) combined the steepest-descent MOO theory and evolutionary computation to guide the search towards the dominating search direction in each generation. Also, Bosman & De Jong (2006) successfully combined three different gradient techniques (local search strategies) with typical genetic operators. Moreover, Jourdan *et al.* (2005) introduced LEMMO that uses the learnable evolution model (LEM) to characterize some rules during the search of a MOO algorithm such as NSGAII to improve its convergence speed. However, modeller time is required to develop these problem specific rules which can highly affect the quality of the coupled NSGAII and LEM solutions.

Deb & Goel (2001) introduced some hybridization terminology to distinguish between applying local search at the end of the genetic search (*posteriori*) and during the genetic search (online). Example posterior hybridization include Talbi *et al.* (2001) and Deb & Goel (2001) and example

online hybridization include Ishibuchi & Murata (1996), Jaszkiewicz (2002), Brown & Smith (2003) and Bosman & De Jong (2006). Goel & Deb (2002) compared *posteriori* and online approaches for hybridizing NSGAII and concluded that the *posteriori* approach is more efficient since the online approach places too much emphasis on the local search. Also, Ishibuchi *et al.* (2003) pointed out the need for a balance between global and local searches and applied the local search to only a few offspring in each generation. Therefore, a new parameter is often required to divide the computational budget between local and global searches. Similar to the other algorithm parameters, finding a proper value for this parameter might be problem specific and time-consuming.

In this study, a simple *posteriori* neighbourhood search strategy replaces the D-DDS sampler in PA-DDS at the point in the search when the expected value of number of perturbed decision variables per iteration becomes one. This approach is based on the hybridization approach in Tolson *et al.* (2009) and removes the need for a new algorithm parameter to change the search strategy.

Comparing MOO algorithm performance

Performance metrics aim to measure the quality of an approximate front by a single number (see Coello *et al.* 2007 for a detailed list of MO performance metrics). Zitzler *et al.* (2003) defined compatibility and completeness for performance metrics based on the dominance relation. Based on the definition, comparing two approximate fronts A and B, a performance metric is complete in terms of the weak dominance relation, if it prefers A that weakly dominates B. However, if a metric is not complete in terms of weak dominance, it may fail to prefer A when A weakly dominates B. A performance metric is compatible in terms of the weak dominance relation, if its preference in A over B indicates that A weakly dominates B. However, if a metric is not compatible in terms of weak dominance, it may prefer A while A does not weakly dominate B. Only if a performance metric is complete and compatible in terms of weak dominance relation, can its result show whether one approximate front outperforms the other one (Zitzler *et al.* 2003).

Zitzler *et al.* (2003) studied various performance metrics and showed that neither a single performance metric nor

a combination of finite number of performance metrics can represent a complete and compatible metric with respect to weak dominance relation. Therefore, they concluded that based on a single performance metric or a combination of them it would not be possible to indicate if an approximate front outperforms the other. However, a performance metric can at best indicate if an approximate front is not worse than (weakly dominated by) another one. As Zitzler & Thiele (1998) showed hypervolume, HV, is a complete metric with respect to weak dominance relation, that is, if it prefers solution A to solution B it means that A is not weakly dominated by B.

In this study, HV was revised and used to assess the results. The revised HV evaluates the algorithm performance relative to the best and worst observed performance across all algorithms in the comparison. Similar to the original HV, the revised HV is complete. Moreover, it is more interpretable than HV when the difference between the worst and the best solutions are practically meaningful.

METHODOLOGY

Hybrid PA-DDS for solving combinatorial MOPs

The proposed MOO algorithm utilizes an implementation of DDS for solving combinatorial problems referred to as D-DDS (Tolson et al. 2009) to first approximate the Pareto optimal front followed by a general local search strategy to improve the quality of the approximation. The pseudo code in Figure 1 represents the algorithm in detail.

In addition to having discrete decision variables instead of continuous decision variables, the only other difference between the PA-DDS in Figure 1 and the PA-DDS in Asadzadeh & Tolson (2009) is that the PA-DDS in Asadzadeh & Tolson (2009) was specifically adapted for test problems with many local fronts. Unfortunately, this adaptation introduces an algorithm parameter associated with how much of the budget should be used initially to search for individual minima. In order to make the algorithm as simple and parsimonious as possible, we revised the PA-DDS in this paper (Figure 1) so that no computational budget needs to be allocated to initially search for individual minima. Initial testing with PA-DDS on one of

STEP 0. Inputs and Initialization

- maximum number of solution evaluations, M
- number of decision variables D
- number of objectives to be minimized, $\#obj$
- network inputs (layout, available pipe diameters, costs, ...)
- empty set of non-dominated solutions, **ND_set** = [ ]
- initial solutions, that respect decision variable bounds
- archive non-dominated solutions in **ND_set**

STEP 1. Global Search: Discrete DDS + Archiving ND solutions

- run the discrete DDS with computational budget M and archive non-dominated solutions during the search
- update $M = M\text{-}m$, m = # solutions evaluated in this step

STEP 2. Local Search for Current Extreme Points of ND_set

(Best Solutions for Individual Objectives)

- FOR $k = 1, ..., \#obj$
  - find extreme point corresponding to objective k
  - While $M > 0$ and L has not yet converged for selected extreme point
    - perform L for selected extreme point
    - update $M = M\text{-}m$, m = #solutions evaluated in this step
    - IF $M = 0$, Go to STEP4

STEP 3. Local Search for Other Solutions in ND_set

- divide first objective range to $n = M / 2D$ equal intervals
- IF $n <$ (# solutions in **ND_set** that are not polished by L)
  - FOR $i = 1, ... n$
    - If there is any **ND_set** solution in interval i that is not polished by L, choose a solution at random, x^{cur} and perform local search L starting at x^{cur}
- ELSE
  - FOR $i = 1, ...,$ number of solutions in **ND_set**
    - IF solution i in **ND_set** has not been polished by L
      - choose solution i as x^{cur}
      - perform local search L starting at x^{cur}
- update $M = M\text{-}m$, m = # solutions evaluated in this step
- **STEP 4. Check Termination Criteria** IF $M > 0$ and there is any solution in **ND_set** for which L is not converged
  - Go to STEP 2
- ELSE,
  - Return **ND_set** as the approximate tradeoff, STOP

Figure 1 │ Pseudo code of the hybrid PA-DDS.

our case studies in this paper indicated that primarily searching for individual minima was unnecessary.

Discrete DDS (D-DDS)

D-DDS (Tolson et al. 2009) is the search engine of PA-DDS for solving combinatorial MOPs. Single objective D-DDS

always uses the overall best solution as the centre of the search neighbourhood. However, in MOPs instead of a single current best solution, a set of current non-dominated solutions exists. Therefore, as noted in Step 1 of Figure 1, the D-DDS is modified to archive all the non-dominated solutions during the search. Also, whenever the current selected solution generates a dominated solution as determined in Step 4 of Figure 1, another current archived non-dominated solution is selected based on the crowding distance measure (as in Deb 2001) as the centre of neighbourhood. Selection is based on the parameterless process roulette-wheel to guide the search towards less crowded areas in the objective space. The pseudo code in Figure 2 describes this search engine in detail.

Hybridizing the algorithm

Based on the decision to hybridize the D-DDS in Tolson et al. (2009), PA-DDS is hybridized when the expected number of perturbed decision variables per iteration becomes one ($P(i) \leq 1/D$ in Step 2 of Figure 2). In other words, at this time, the D-DDS is replaced by the local search referred to as L which is defined in Figure 3. L is designed to polish a current non-dominated solution by cycling through all possible ways for decreasing or increasing one decision variable at a time by one discrete option.

In hybrid PA-DDS, L is called iteratively in Steps 2 and 3 (Figure 1). L is first invoked to polish solutions corresponding to the extreme points of the current approximate front. For polishing extreme points, L restarts the search at decision variable 1 to ensure convergence to a solution that can no longer be improved by L. In Step 3 of Figure 1, L will polish a portion of the other non-dominated solutions depending on the remaining computational budget. Unlike for extreme solutions, L does not restart the search to converge for other non-dominated solutions. Instead, L spends at most $2D$ (twice the number of decision variables) iterations to polish each of these solutions and this limit helps to ensure that the local search occurs along the entire front. If the remaining computational budget in hybrid PA-DDS is not enough for polishing all archived non-dominated solutions, the range of the first objective function is divided into $n = (the\ remaining\ budget)/2D$

STEP 0. Inputs and Initialization

- maximum number of objective function evaluations, M
- neighbourhood perturbation size parameter, r (default: 0.2)
- vector of maximum options for all D decision variables, x^{max}. Note that $x^{min} = [1, 1, ..., 1]$
- set of non-dominated solutions, **ND_set**
- initialize the solution evaluation counter to 0, $i = 0$
- calculate crowding distance for solutions in **ND_set**

STEP 1. Selection

- select current solution x^{cur} from **ND_set** by Roulette-Wheel based on crowding distance measure

STEP 2. Neighbourhood Definition

- Randomly select J of the D decision variables for inclusion in neighbourhood, $\{N\}$:
- set $i = i + 1$
- calculate probability each decision variable is included in $\{N\}$ as a function of i: $P(i) = 1 - \ln(i) / \ln(M)$
- IF ($P(i) \leq 1/D$) OR IF ($i > M$), return **ND_set**, STOP
- ELSE
  - FOR $d = 1, ..., D$ decision variables, add d to $\{N\}$ with probability $P(i)$
  - IF $\{N\}$ is empty, select one random d for $\{N\}$

STEP 3. Perturbation from Discrete Probability Distributions

- FOR $j = 1, ..., J$ decision variables in $\{N\}$
  - Sample a standard normal random variable, $N(0,1)$
  - $x_j^{new} = x_j^{cur} + \sigma_j N(0,1)$, where $\sigma_j = r \times (x_j^{max} - x_j^{min})$
    - IF $x_j^{new} < (x_j^{min} - 0.5)$, with 50% chance: $x_j^{new} = x_j^{min}$, with 50% chance: $x_j^{new} = 2 x_j^{min} - x_j^{new} - 1$
    - IF $x_j^{new} > (x_j^{max} + 0.5)$, set $x_j^{new} = x_j^{min}$
    - IF $x_j^{new} > (x_j^{max} + 0.5)$, with 50% chance: $x_j^{new} = x_j^{max}$, with 50% chance: $x_j^{new} = 2 x_j^{max} - x_j^{new} + 1$
    - IF $x_j^{new} < (x_j^{min} - 0.5)$, set $x_j^{new} = x_j^{max}$
  - Round x_j^{new} to nearest integer, discrete option number
  - IF $x_j^{new} = x_j^{cur}$, sample x_j^{new} from a discrete uniform distribution, $U(x_j^{max}, x_j^{min})$, until $x_j^{new} \neq x_j^{cur}$

STEP 4. Evaluation

- Evaluate objective function values of x^{new}, $F(x^{new})$
- IF x^{new} is not dominated by x^{cur}
  - Check dominance of x^{new} against solutions in **ND_set**
  - IF x^{new} is a new non-dominated solution
    - $x^{cur} = x^{new}$
    - Go to STEP 3
  - ELSE, Go to STEP 1
- ELSE, Go to STEP 1

Figure 2 | Pseudo code of the global search engine of PA-DDS for solving combinatorial MOPs.

equal intervals and at least one randomly selected non-dominated solution from each nonempty interval is selected to be polished by L.

STEP 0. Inputs and Initialization

- initial solution x^{cur}, and corresponding objective function values $F^{cur} = F(x^{cur})$
- maximum number of objective function evaluations, M
- vector of maximum option for all D decision variables, x^{max}. Note that $x^{min} = [1, 1, ..., 1]$
- set of non-dominated solutions, **ND_set**
- initialize the network simulation counter to 0, $i = 0$

STEP 1. Reduce decision variables by one option

- $x^{ini} = x^{cur}$
- FOR $d = 1, ... D$
  - $x^{new} = x^{cur}$
  - If $x^{new}(d) > x^{min}(d)$
    - set $i = i + 1$
    - if $i > M$, STOP L and return **ND_set**
    - $x^{new}(d) = x^{new}(d) - 1$
    - Evaluate $F^{new} = F(x^{new})$
    - IF x^{new} is not dominated by x^{cur}
      - Check dominance of x^{new} against **ND_set** solutions
      - If x^{new} dominates x^{cur} then $x^{cur} = x^{new}$, $F^{cur} = F^{new}$

STEP 2. Increase decision variables by one option

- $x^{cur} = x^{ini}$
- FOR $d = 1, ... D$
  - $x^{new} = x^{cur}$
  - If $x^{new}(d) < x^{max}(d)$
    - set $i = i + 1$
    - if $i > M$, STOP L and return **ND_set**
    - $x^{new}(d) = x^{new}(d) + 1$
    - Evaluate $F^{new} = F(x^{new})$
    - IF x^{new} is not dominated by x^{cur}
      - Check dominance of x^{new} against **ND_set** solutions
      - If x^{new} dominates x^{cur} then $x^{cur} = x^{new}$, $F^{cur} = F^{new}$

Figure 3 │ Local search L to polish one solution on the approximate front.

Optimization model formulation

In this study, the bi-objective WDN design problem is solved to minimize cost and minimize the highest pressure deficit throughout the network (the same problem formulation is utilized in Atiquzzaman *et al.* 2006; Farmani *et al.* 2005; Perelman *et al.* 2008; Di Pierro *et al.* 2009). The two objectives are conflicting since pipes of larger diameter cost more and usually reduce the pressure deficit. Decision variables of the problem are pipe diameters that can be selected from a finite set of available pipe sizes and all other network characteristics are known. The mathematical form of the objectives is presented as follows

$$\min_x f_1 = \left(\sum_{t=1}^{D} L_i C(x_i) \right)$$

$$\min_x f_2 = \max \left(0, \max_j \{ H_j - h_j(x) \} \right)$$

$$Subject\ to: \ x_i \in \{1, 2, ..., x_i^{max}\}, \quad \forall i = 1, ..., D \qquad (1)$$

where i and j are the pipe and demand node indices respectively, L is the length of each pipe, C is the cost per unit length of each pipe as a function of the decision variable x_i that is an integer-valued pipe diameter option number for pipe i and is between option 1 (the smallest diameter) and the maximum diameter option, x_i^{max}, for all D pipes in the network to be sized, H is the minimum required pressure head for each demand node in the network, and h is the pressure at each demand node as a function of $x = [x_1, ..., x_D]$ and is determined by the network hydraulic simulator which was EPANET2 in this study.

Benchmark WDN design problems

The following five WDN design problems are selected from the literature, modelled in EPANET2, and solved in the bi-objective optimization problem formulation (1). The EPANET2 input files and all necessary information for replicating these WDNs are available online (http://www.civil.uwaterloo.ca/btolson/links.htm).

New York tunnels problem (NYTP)

NYTP (Schaake & Lai 1969; Zecchin *et al.* 2005) involves the rehabilitation of an existing WDN with 21 pipes and 16 design options per pipe (parallelization with one of 15 tunnel sizes or a do-nothing option). This defines a search space size of 16^{21} ($\sim 1.93 \times 10^{25}$). The best known least-cost design of NYTP costs \$38.638 million (Maier *et al.* 2003). Therefore (0 m, \$38.638 million) is one of the extreme points of the Pareto optimal front in problem formulation (1) for NYTP. Obviously, the other extreme point is (47.6 m, \$0) corresponding to no additional pipe to the current

network and accepting the maximum of 47.6 m pressure deficit.

Doubled New York tunnels problem (NYTP2)

NYTP2 (Zecchin et al. 2005) is twice as big as NYTP with 42 pipes to be sized from 16 options. This defines a search space size of 16^{42} ($\sim 3.74 \times 10^{50}$). The best known least-cost design of NYTP2 costs $77.276 million (Zecchin et al. 2005). Therefore, the best known Pareto front for NYTP2 in problem formulation (1) has the following extreme points: (0 m, $77.276 million) and (47.6 m, $0).

Hanoi problem (HP)

HP (Fujiwara & Khang 1990) has 32 pipes with six options resulting in a search space of 6^{32} combinations (2.87×10^{26} solutions). The single objective version of this problem is reportedly difficult to simply find a feasible solution for (Eusuff & Lansey 2003; Zecchin et al. 2005, 2007). Moreover, Farmani et al. (2005) noted that in the bi-objective optimization problem formulation (1), finding a fully feasible solution remains difficult for HP. The best known least-cost design of HP costs $6.081 million (Perelman & Ostfeld 2007) that suggests (0 m, $6.081 million) as an extreme point of the best known Pareto front in problem formulation (1). The other true extreme point that corresponds to the smallest pipe size for all pipes is (17,678.5 m, $1.802 million). Readers are referred to Zecchin et al. (2005) for detailed information about these first three networks.

GoYang problem (GYP)

GYP (Kim et al. 1994) is a WDN in South Korea with 30 pipes that should be sized from eight diameter options. This defines a search space size of 8^{30} ($\sim 1.24 \times 10^{27}$). The best known least-cost design of the network costs 177.01 million Won (Tolson et al. 2009); therefore, one extreme point of the best known Pareto front is (0 m, 177.01 million Won) and the other one is (125.1 m, 174.673 million Won).

Balerma irrigation network problem (BP)

BP is a large and complex WDN with 443 demand nodes, 454 pipes, eight loops, and four reservoirs (Reca & Martínez 2006). Each of the 454 pipes must be sized from 10 possible diameters that define a search space size of 10^{454}. The best known least-cost-design of BP costs €1.9409 million (Tolson et al. 2009). Therefore, the best known Pareto front of this network has the following extreme points: (0 m, €1.9409 million) and (5213.7 m, €0.724 million); the latter corresponds to the smallest size of all pipes.

Selected performance metrics

Normalized hypervolume (NHV)

The hypervolume (HV) metric (Zitzler & Thiele 1998) measures the volume bounded by the approximate front and a reference point (e.g. the shaded area in Figure 4). Therefore, the bigger HV value is preferred.

Deb (2001) suggested the calculation of HV in the normalized objective space which is called the normalized hypervolume (NHV) and Van Veldhuizen (1999) proposed the hypervolume ratio (HVR) that is the ratio of HV for
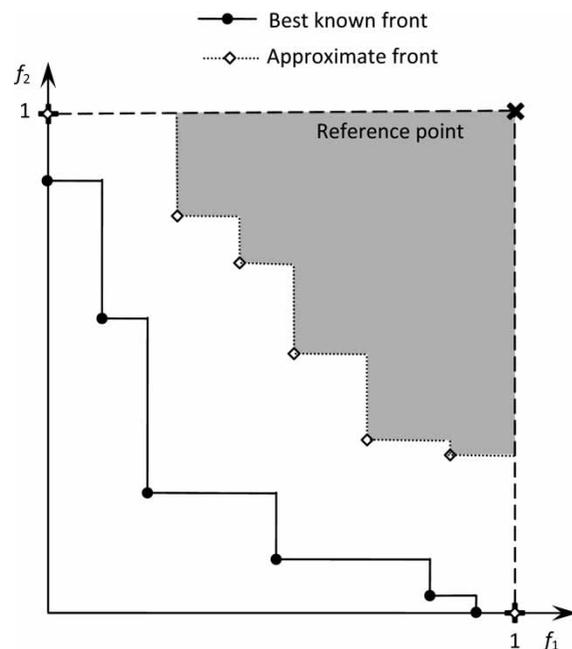

Figure 4 │ Normalized hypervolume performance metric for minimization of two objectives.

the approximate front to the HV for the Pareto optimal front. Hence, the HVR shows the quality of an approximate front in comparison with the Pareto optimal front. Although theoretically the HVR can take any value between 0 and 1, it is not necessarily close to 0 for all poor approximate fronts. In fact, as our results show, very different approximate fronts can have an HVR with a very small numerical difference. As a result, the interpretation of differences in the HVR values between algorithms is not always straightforward. Therefore, a modified version of the HV is proposed that is much easier to interpret than the HV and HVR.

Comparative normalized hypervolume (CNHV)

The proposed MOO algorithm performance metric is referred to as the comparative NHV (CNHV) and is a modified version of the HVR calculated in the normalized objective space for comparing multiple optimization trials of multiple algorithms. With reference to areas A and B in Figure 5, the CNHV for an approximate front is equal to $B/(A + B)$. The main difference between CNHV and its precedent performance metrics HVR and HV can be summarized as follows.

- The single reference point is replaced by a set of reference points corresponding to the worst attained front that can be constructed from all algorithm results in the comparison.
- Both the best and the worst attained fronts used in the CNHV are extracted from the results of all MOO algorithms that are included in the comparison while in HV and the HVR, the reference point and the best known front are fixed.

To calculate the CNHV, the best and the worst attained fronts (the solid and dashed lines in Figure 5, respectively) must be identified. To do this, final approximate fronts of all the trials of MOO algorithms in the comparison are collected in a set. The best attained front is then identified as the subset of solutions from this set that are non-dominated. The worst attained front contains all solutions in this set that are weakly dominated by at least one solution from each optimization trial in the comparison.
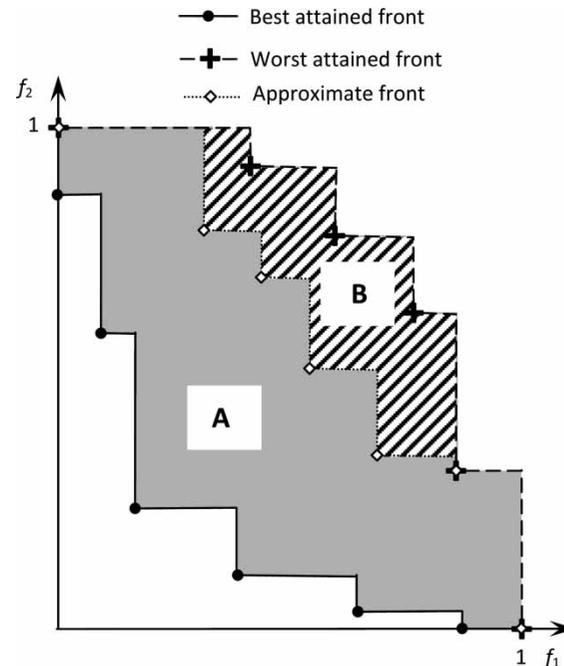

Figure 5 | CNHV performance metric for minimization of two objectives.

Similar to the original HV, the CNHV is complete with respect to the weak dominance relation, that is, it always prefers an approximate front that weakly dominates the other one. In other words, comparing two approximate fronts, the better value of the CNHV indicates that the corresponding approximate front is not weakly dominated by the other one.

The value of the CNHV is more directly interpretable than the value of HV or the HVR since it determines how much of all attained results are dominated by each approximate front. As such, CNHV values close to 0 are relatively poor and values close to 1 are relatively good. However, CNHV is not recommended if the best and the worst results are not practically different. In that situation, all results are practically the same quality and it might be misleading to assign a value close to 0 to one or more of the algorithms in the comparison.

Benchmark optimization algorithms

In order to assess the performance of hybrid PA-DDS, NSGAII and SPEA2 are implemented and applied to the same bi-objective WDN design problems. The search engine of NSGAII and SPEA2 is an integer coded GA

with a 90% chance of uniform crossover and average mutation rate of 1/(number of decision variables). The other parts of NSGAII and SPEA2 are implemented as in Deb *et al.* (2002) and Zitzler *et al.* (2001), respectively. The population size in both NSGAII and SPEA2 is set to 100 except for solving GYP with the limited budget of 2000 model evaluations where a smaller population size of 50 is used. The population size of 100 and the probability of crossover and mutation were selected based on the parameter values specified in Deb *et al.* (2002) and we did not spend any further effort to fine tune them.

RESULTS

The results are presented in two subsections. First, a comparison is made between hybrid PA-DDS, NSGAII and SPEA2 for solving problem formulation (1) for all five bi-objective WDN case studies. Second, the effectiveness of the local search is evaluated by applying it to both NSGAII and SPEA2.

Hybrid PA-DDS versus NSGAII and SPEA2

Each of the five WDN case studies is solved by hybrid PA-DDS, NSGAII, and SPEA2, with a rather large computational budget (determined from typical budgets utilized for the case studies in previous publications) and with a more limited computational budget (one order of magnitude less). Hybrid PA-DDS has two termination criteria, computational budget and the local search convergence. For NYTP, NYTP2 and GYP solved with the higher computational budget, the local search usually converges before spending the whole computational budget. This is why the computational effort for these three cases is less than the budget. Table 1 summarizes some statistics of the two performance metrics called NHV and CNHV proposed in this study. Bold numbers in Table 1 represent the best result for each case study. With the limited computational budget, Hybrid PA-DDS achieves the best NHV and CNHV values for all case studies. However, with the higher computational budget, NSGAII performs better than Hybrid PA-DDS in NYTP2 and BP.

Figure 6 compares the best attained front and the worst CNHV fronts (corresponding to the italic numbers in Table 1) for all three algorithms solving each of the five case studies with the higher computational budget. It should be noted here that the best attained front is obtained from results of all trials of all three algorithms and is a discrete front; however, its points are piecewise linearly connected for illustrative purposes. For NYTP, NYTP2, and GYP, the best known endpoints (see Benchmark WDN Design Problems section) are captured in the best front. However, the best obtained endpoint corresponding to least-cost design of HP is (0 m, \$6.096 M) instead of (0 m, \$6.081 M) and for BP (0 m, €2.115 M) instead of (0 m, €1.9409 M). The other best known endpoints of these two cases are identified in the best attained front.

Comparing the NHV and CNHV values in each row of Table 1, CNHV more clearly detects the difference between the approximate fronts and therefore between algorithms. For example, in the HP case study with a computational budget equal to 10,000, the best trial of hybrid PA-DDS has a NHV equal to 0.96 while it is 0.94 for the best trial of NSGAII. This value means that the best trial of Hybrid PA-DDS covered (dominated) 96% of the area between the best attained front and the reference point while the best trial of NSGAII covered 94% of this area. Although this difference seems negligible, Figure 7(a) shows that there is a considerable difference between the corresponding approximate fronts. For the same trials, the CNHV values are 0.98 and 0.84 denoting that the best trial of hybrid PA-DDS dominated 98% of the area between the best and worst attained fronts considering all 150 approximate fronts from 50 trials of each algorithm compared to only 84% for the best trial of NSGAII. Also comparing the best trial of these two algorithms for solving BP with a computational budget equal to 1,000,000, NHV is equal to 0.96 and 0.97 for Hybrid PA-DDS and NSGAII respectively, while CNHV magnifies the difference and results in 0.82 and 0.92, respectively. Figure 7(b) demonstrates that the difference detected by CNHV is really considerable and NSGAII performed better than Hybrid PA-DDS.

Local search performance assessment

To evaluate the effectiveness of the proposed local search strategy, it was also applied to the results of NSGAII and

Table 1 | Algorithm comparison based on NHV and CNHV

| Case Study	Budget[a]	Statistic	NHV Hybrid PA-DDS	NSGAII	SPEA2	CNHV Hybrid PA-DDS	NSGAII	SPEA2
NYTP 50 trials	50,000	Avg.	**0.85**	0.85	0.85	**1.00**	0.96	0.95
		Best	**0.85**	0.85	0.85	**1.00**	0.98	0.99
		Worst	**0.85**	0.85	0.85	*0.99*	*0.92*	*0.80*
	5,000	Avg.	**0.82**	0.74	0.76	**0.90**	0.53	0.62
		Best	**0.84**	0.79	0.81	**0.96**	0.75	0.83
		Worst	**0.79**	0.67	0.70	**0.78**	0.27	0.38
NYTP2 50 trials	300,000	Avg.	0.85	**0.85**	0.85	0.87	**0.91**	0.84
		Best	0.85	**0.85**	0.85	0.95	**0.95**	0.92
		Worst	0.84	**0.85**	0.84	*0.77*	**0.83**	*0.69*
	30,000	Avg.	**0.82**	0.79	0.79	**0.82**	0.60	0.62
		Best	**0.83**	0.82	0.83	**0.91**	0.81	0.84
		Worst	**0.80**	0.74	0.73	**0.69**	0.29	0.21
HP 50 trials	100,000	Avg.	**1.00**	0.98	0.80	**0.99**	0.96	0.65
		Best	**1.00**	0.98	0.93	**1.00**	0.97	0.88
		Worst	**0.99**	0.95	0.67	*0.99*	*0.91*	*0.43*
	10,000	Avg.	**0.92**	0.90	0.89	**0.96**	0.54	0.51
		Best	**0.96**	0.94	0.93	**0.98**	0.84	0.75
		Worst	**0.88**	0.86	0.82	**0.93**	0.30	0.04
GYP 50 trials	20,000	Avg.	**0.93**	**0.93**	0.93	**1.00**	1.00	0.99
		Best	**0.93**	**0.93**	**0.93**	**1.00**	1.00	1.00
		Worst	**0.93**	0.93	0.91	*1.00*	*0.99*	*0.58*
	2,000	Avg.	**0.89**	0.62	0.65	**0.94**	0.43	0.49
		Best	**0.92**	0.79	0.77	**0.99**	0.74	0.71
		Worst	**0.83**	0.39	0.42	**0.83**	0.02	0.07
BP 10 trials	1,000,000	Avg.	0.96	**0.97**	0.93	0.77	**0.88**	0.35
		Best	0.96	**0.97**	0.94	0.82	**0.92**	0.45
		Worst	0.96	**0.96**	0.93	*0.72*	**0.83**	*0.30*
	100,000	Avg.	**0.91**	0.83	0.81	**0.85**	0.35	0.28
		Best	**0.92**	0.84	0.83	**0.87**	0.42	0.38
		Worst	**0.90**	0.81	0.78	**0.78**	0.24	0.10

[a]Hybrid PA-DDS converged before spending the entire computational budget in NYTP, NYTP2 and GYP for the higher computational budget. Therefore, its computational effort for these cases is less than the computational budget. 35,000 for NYTP, 263,400 for NYTP2 and 17,330 for GYP.

SPEA2. However, as these two algorithms spent the whole computational budget for the global search, the local search is only applied to the extreme points of the resultant fronts with the computational budget equal to the average budget of local search in Hybrid PA-DDS. This computational budget and the average improvement in results based on the CNHV are summarized in Table 2.

Based on Table 2, it can be concluded that although the local search is only applied to the extreme points of the front with very limited computational budget, it highly improved the results of NSGAII and SPEA2 especially when the total computational budget is limited.

DISCUSSION

It should be noted that the local search L is implemented such that it always evaluates one option change at a time relative to the current solution. Therefore, the local search order (starting for example at decision variable D instead of decision variable 1 as in our implementation) can change the results. However, finding the best order is not the purpose of this study. Moreover, it is not claimed that the proposed neighbourhood search strategy is the most efficient local search, but perhaps the simplest one that adequately improves the algorithm efficiency.
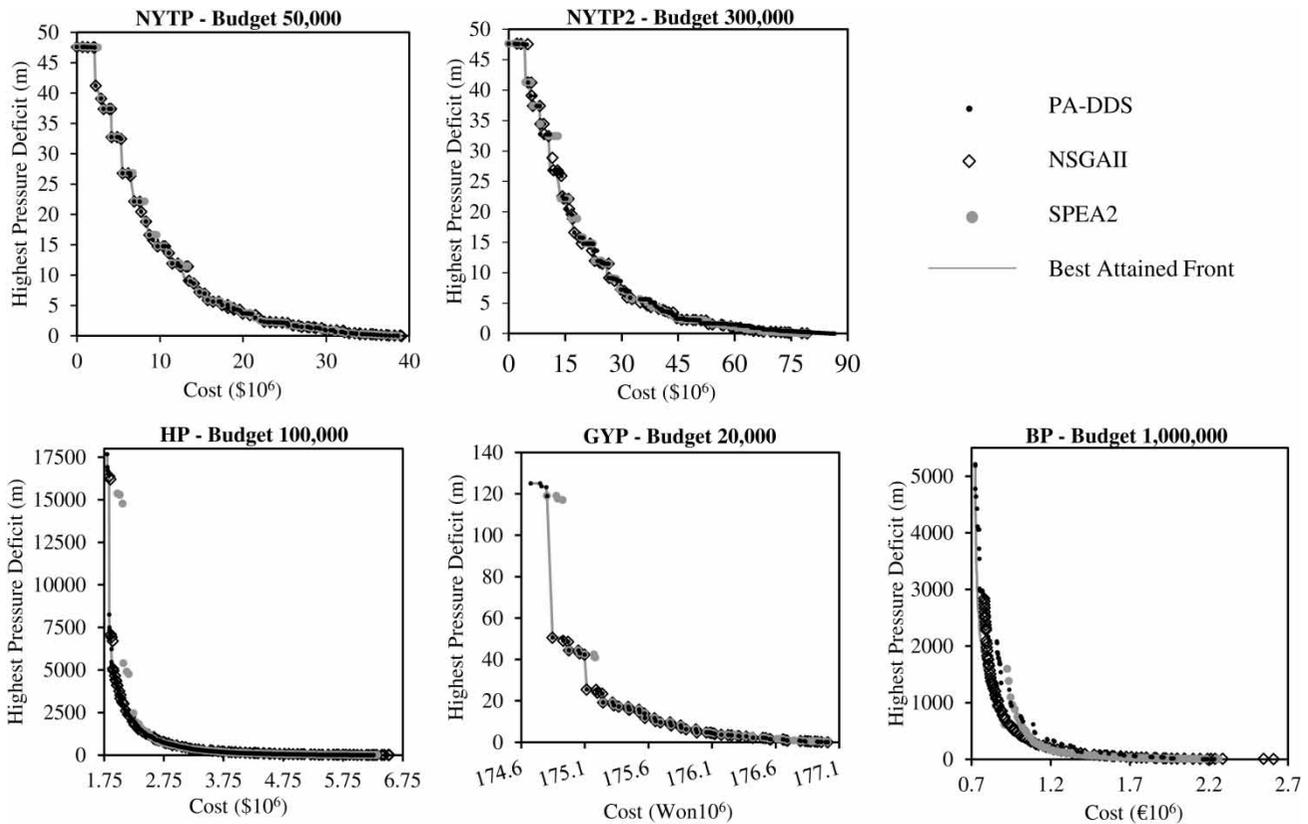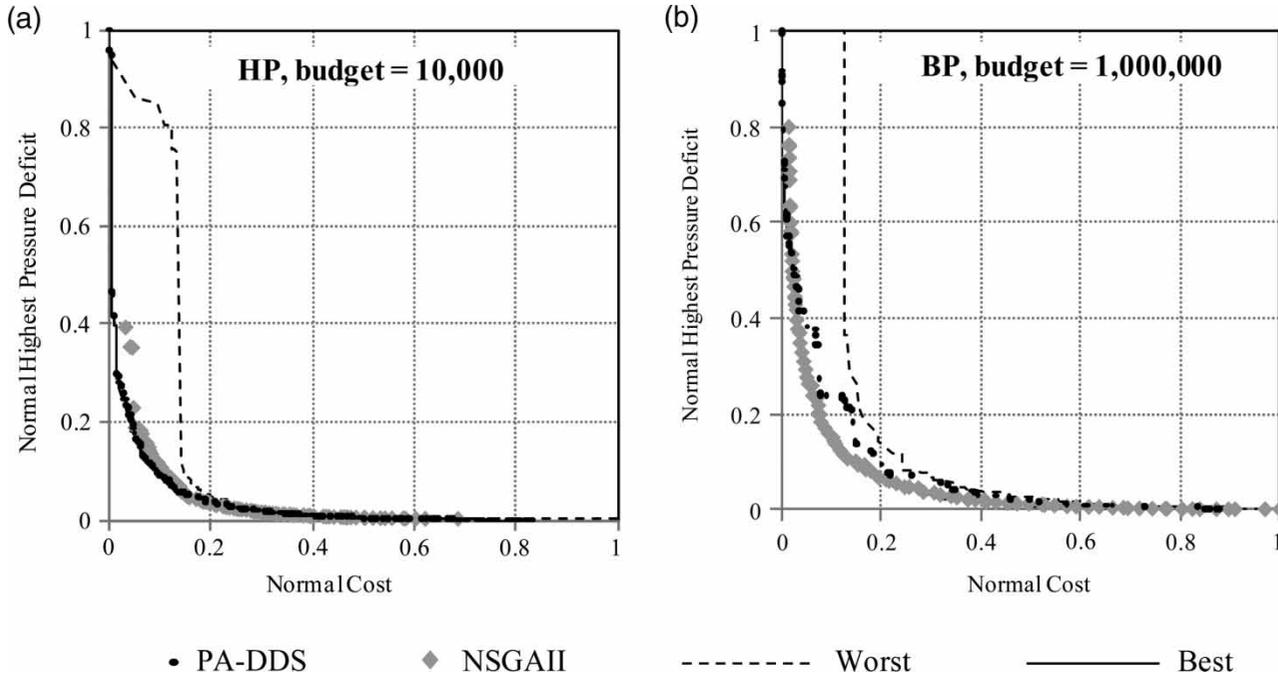
Figure 6 | Comparison between the best attained front and the worst CNHV trials of all three algorithms solving each case study.

If the purpose of MOO algorithm comparison is to assess the performance of various algorithms with various computational budgets, we recommend using the best and worst attained fronts corresponding to each computational budget. This recommendation makes the comparison hard to replicate. Nonetheless, calculating computational budget specific CNHV (and thus the best and the worst attained fronts) is more appropriate since the attainable objective space obviously varies as a function of computational budget.

The comparison of algorithms was based on default configurations/parameter settings that WDN modellers would most likely utilize when trying to solve their own MO design problems with these algorithms. Comparative results might change if each algorithm was fine tuned to optimally solve each problem but that would generally require substantial computational experiments. Instead, parameters of NSGAII and SPEA2 are set to the recommended values from literature while the design

decisions and single parameter value of the Hybrid PA-DDS algorithm were based on previous decisions for DDS (Tolson & Shoemaker 2007) and HD-DDS (Tolson et al. 2009).

NSGAII and SPEA2 have a fixed-size archive and if it becomes full of non-dominated solutions, they ignore some current non-dominated solutions. Although this strategy controls and limits the complexity of the algorithm, it may have a disadvantage. Laumanns et al. (2002) showed that the standard archiving strategy of NSGAII or SPEA2 allows the algorithm eliminate some high quality solutions that in the long run might even dominate some of the non-dominated solutions in the final archive. To deal with this issue some new archiving strategies have been proposed (see for example Laumanns et al. 2002; Beume et al. 2007). The current version of PA-DDS archives all non-dominated solutions during the search. Therefore, it does not lose any non-dominated solution; however, this leads to two related challenges for PA-DDS.

Figure 7 | Comparison between the best CNHV trial of Hybrid PA-DDS and NSGAII for solving: (a) HP, Extreme solutions (1,7678.5 m, $1.803 M) and (0 m, $7.470 M) are used to normalize the fronts; and (b) BP, Extreme solutions (5,213.7.5 m, €0.724 M) and (0 m, €2.3620 M) are used to normalize the fronts.

The first challenge corresponds to the algorithm efficiency (runtime) for solving large scale problems. In PA-DDS, any new non-dominated solution is checked against all current non-dominated solutions. Hence, the higher number of archived solutions the more time required for the dominance check. This may affect the efficiency of PA-DDS for solving problems with many objective functions and a huge computational budget since either of these factors can generate excessive non-dominated solutions. However, in real-world engineering problems, where simulation is very time consuming and hence the total computational budget is not so huge to allow numerous solutions in the archive, time of dominance check may not have a considerable impact on the algorithm runtime. For these five WDN bi-objective problems, PA-DDS always had a shorter serial runtime than SPEA2 while NSGA2

Table 2 | Evaluating the effectiveness of the local search for improving the extreme points of the approximate fronts based on the average per cent improvement in CNHV

| Case study | Global search budget | NSGAII | | SPEA2 | |
		Local search budget	Average improvement in CNHV (%)	Local search budget	Average improvement in CNHV (%)
NYTP	50,000	49	1.0	48	1.0
	5,000	183	34.0	160	17.7
NYTP2	300,000	111	0.0	127	0.0
	30,000	413	20.3	376	19.4
HP	100,000	166	1.0	251	36.9
	10,000	808	72.2	788	84.3
GYP	20,000	56	0.0	67	1.0
	2,000	171	107.0	191	83.7
BP	1,000,000	27,600	6.8	24,200	114.3
	100,000	2,500	193.8	2,500	256.0

was always the quickest among all three algorithms. Over all of the five case studies with both computational budgets, PA-DDS runtimes were on average only 13% (with extremes of 7 and 36%) longer than NSGAII. Therefore, the first challenge was not an important issue for solving these five problems.

The second possible challenge is related to the selection process. The current selection scheme of PA-DDS is designed to sample more from less crowded parts of the approximate front. However, we believe that the performance of PA-DDS can be improved by modifying the selection process to guide the search towards the most interesting parts of the tradeoff, which may not coincide with the less crowded part of the front. This improvement can be significant especially for solving computationally intensive problems where a limited computational budget is available. Currently, we are investigating new selection schemes in PA-DDS.

Although we did not investigate relative PA-DDS performance on large distribution networks (i.e. thousands of pipes/decision variables), we did apply the algorithm with the same parameters and configuration to problems with 21–454 decision variables and a computational budget ranging from 2,000 to 1,000,000 hydraulic model evaluations. As such, PA-DDS could be applied to even larger distribution networks and in such a case we would suggest applying PA-DDS without any algorithmic and/or parameter modifications. For larger distribution networks, the efficiency of local search L to refine the extreme solutions will degrade substantially relative to efficiencies reported in Table 2. Future algorithm comparison studies focused only on very large distribution networks are necessary to properly assess relative PA-DDS performance in this context.

It should be noted here that, problem formulation (1) is an artificial WDN bi-objective problem. Therefore, its result cannot be used for designing real WDN problems. For example, all Pareto optimal solutions returned in this study have a pressure deficit (some even have negative pressures). Even the extreme point corresponding to the least-cost design of the network is impractical since it tends to reduce pipe sizes or completely eliminate some pipes. This may lead to an insufficient capacity to handle system failures (Walski 2001). We solved problem formulation (1) as the benchmark

WDN problem type to assess the relative performance of our proposed MOO algorithm just as many previous studies have done (e.g. Farmani et al. 2005; Atiquzzaman et al. 2006; Perelman et al. 2008; Di Pierro et al. 2009). Although not demonstrated here, Hybrid PA-DDS can be applied to more realistic WDN design studies that have more than two objectives. Further work should be conducted to compare Hybrid PA-DDS performance relative to other algorithms on problems with more than two objectives.

CONCLUSION

A simple and parsimonious optimization algorithm for solving combinatorial MOPs was introduced. The algorithm is hybridized with a straightforward neighbourhood search and is successfully applied to five benchmark bi-objective WDN problems. The hypervolume performance metric is modified to define the new comparative normalized hypervolume metric which makes the hypervolume metric more interpretable for comparing multiple trials of multiple algorithms. Results show the comparable performance of the proposed algorithm to two of the most well known MOO algorithms, NSGAII and SPEA2. In the future, we will try to improve the algorithm performance mainly by modifying the selection criteria to guide the search towards the most interesting parts of the front rather than the less crowded parts of it.

Moreover, we will assess the algorithm performance in solving real world WDN combinatorial problems with more than two objectives such as selection and placement of best management practice for pesticide control (e.g. Maringanti et al. 2008) and multi-objective long-term groundwater monitoring design (e.g. Kollat & Reed 2006).

comments which definitely improved the presentation of our findings.

REFERENCES

Asadzadeh, M. & Tolson, B. A. 2009 A new multi-objective algorithm, Pareto Archived DDS. In *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers (GECCO'09), July 8–12, 2009, Montreal, QC, Canada*. ACM, New York, NY, USA, pp. 1963–1966.

Atiquzzaman, M., Liong, S. & Yu, X. 2006 Alternative decision making in water distribution network with NSGAII. *J. Water Res. Plan. Manage.* **132** (2), 122–126.

Beume, N., Naujoks, B. & Emmerich, M. 2007 SMS-EMOA: Multiobjective selection based on dominated hypervolume. *Eur. J. Oper. Res.* **181** (3), 1653–1669.

Bosman, P. A. N. & De Jong, E. D. 2006 Combining gradient techniques for numerical multi-objective evolutionary optimization. In *Proceedings of the 8th Annual Conference Companion on Genetic and Evolutionary Computation Conference (GECCO'06), July 8–12, 2006, Seattle, WA, USA*. ACM, New York, NY, USA, pp. 627–634.

Brown, M. & Smith, R. 2003 Effective use of directional information in multi-objective evolutionary computation. In *Proceedings of the 2003 international conference on Genetic and Evolutionary Computation: PartI (GECCO'03), July 12–16, 2003, Chicago, Illinois, USA*. Springer-Verlag, Berlin, Heidelberg, pp. 778–789.

Coello Coello, C. A., Lamont, G. B. & Van Veldhuizen, D. A. 2007 *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, USA.

Deb, K. 2001 *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons Inc., NY, USA.

Deb, K. & Goel, T. 2001 A hybrid multi-objective evolutionary approach to engineering shape design. In *Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)* (E. Zitzler, L. Thiele, K. Deb, C. A. Coello Coello & D. Corne, eds.), *Vol. 2632 of Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 519–533.

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

Di Pierro, F., Khu, S. T., Savic, D. A. & Berardi, L. 2009 Efficient multi-objective optimal design of water distribution networks on a budget of simulations using hybrid algorithms. *Env. Model. Softw.* **24** (2), 202–213.

Eusuff, M. M. & Lansey, K. E. 2003 Optimization of water distribution network design using the shuffled frog leaping algorithm. *J. Wat. Res. Plann. Manage.* **129** (3), 210–225.

Farmani, R., Savic, D. A. & Walters, G. A. 2005 Evolutionary multi-objective optimization in water distribution network design. *Eng. Optimization* **37** (2), 167–183.

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

Goel, T. & Deb, K. 2002 Hybrid methods for multi-objective evolutionary algorithms. In *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning Computational Intelligence for the e-age (SEAL'02)*, November 18–22, 2002, Nanyang Technical University, Singapore, Vol. 1, pp. 188–192.

Ishibuchi, H. & Murata, T. 1996 Multi-objective genetic local search algorithm. In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation (ICEC'96), May 20–22, 1996, Nagoya, Japan*. IEEE Press, Piscataway, NJ, USA, pp. 88–93.

Ishibuchi, H., Yoshida, T. & Murata, T. 2003 Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. *IEEE Trans. Evol. Comput.* **7** (2), 204–223.

Jaszkiewicz, A. 2002 Genetic local search for multi-objective combinatorial optimization. *Eur. J. Oper. Res.* **137** (1), 50–71.

Jourdan, L., Corne, D., Savic, D. & Walters, G. 2005 Preliminary investigation of the learnable evolution model' for faster/ better multiobjective water systems design. In *Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)* (C. A. Coello Coello, A. H. Aguirre & E. Zitzler, eds.), *Vol. 3410 of Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 841–855.

Kim, J. H., Kim, T. G., Kim, J. H. & Yoon, Y. N. 1994 A study on the pipe network system design using non-linear programming. *J. Korean Water Resour. Assoc.* **27** (4), 59–67.

Kleeman, M. P., Lamont, G. B., Cooney, A. & Nelson, T. R. 2007 A multi-tiered memetic multiobjective evolutionary algorithm for the design of quantum cascade lasers. In *Conference on Evolutionary Multi-Criterion Optimization (EMO 2007)* (S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu & T. Murata, eds.), *Vol. 4403 of Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 186–200.

Knowles, J. D. & Corne, D. W. 2000 Approximating the nondominated front using the Pareto archived evolution strategy. *Evol. Comput.* **8** (2), 149–172.

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

Laumanns, M., Thiele, L., Zitzler, E. & Deb, K. 2002 Archiving with guaranteed convergence and diversity in multi-objective optimization. In *Proceedings of the 2002 Annual Conference Companion on Genetic and Evolutionary Computation Conference, (GECCO'02), July 9–13, 2002, New York, USA*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 439–447.

Maier, H. R., Simpson, A. R., Zecchin, A. C., Foong, W. K., Phang, K. Y., Seah, H. Y. & Tan, C. L. 2003 Ant colony optimization for design of water distribution systems. *J. Wat. Res. Plann. Manage.* **129** (3), 200–209.

Maringanti, C., Chaubey, I., Arabi, M. & Engel, B. 2008 A multi-objective optimization tool for the selection and placement of BMPs for pesticide control. *Hydrol. Earth Syst. Sci. Discuss.* **5** (4), 1821–1862.

Parsopoulos, K. E. & Vrahatis, M. N. 2002 Particle swarm optimization method in multiobjective problems. In *Proceedings of the ACM Symposium on Applied Computing (SAC2002), March 10–14, 2002, Madrid, Spain.* ACM, New York, NY, USA, pp. 603–607.

Perelman, L. & Ostfeld, A. 2007 An adaptive heuristic cross-entropy algorithm for optimal design of water distribution systems. *Eng. Optimization* **39** (4), 413–428.

Perelman, L., Ostfeld, A. & Salomons, E. 2008 Cross entropy multiobjective optimization for water distribution systems design. *Water Resour. Res.* **44**, W09413.

Reca, J. & Martínez, J. 2006 Genetic algorithms for the design of looped irrigation water distribution networks. *Water Resour. Res.* **42** (5), W05416.

Schaake, J. C. & Lai, D. 1969 *Linear Programming and Dynamic Programming Applications to Water Distribution Network Design.* Rep. No. 116, Hydrodynamics Laboratory. MIT, Cambridge, MA.

Schaffer, J. D. 1984 *Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms.* PhD Dissertation, Vanderbilt University, Nashville, TN, USA.

Talbi, E., Rahoual, M., Mabed, M. & Dhaenens, C. 2001 A hybrid evolutionary approach for multicriteria optimization problems: Application to the Flow Shop. In *Proceedings of the 1st International Conference on Evolutionary Multi-Criterion Optimization (EMO'01)* (E. Zitzler, L. Thiele, K. Deb, C. A. Coello Coello & D. Corne, eds.), *Vol. 1993 of Lecture Notes in Computer Science.* Springer, Berlin, Heidelberg, pp. 416–428.

Tolson, B. A. & Shoemaker, C. A. 2007 Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resour. Res.* **43** (1), 01413.

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 Veldhuizen, D. A. 1999 *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations.* PhD Thesis, Air Force Institute of Technology, Wright Patterson AFB, OH, USA.

Vrugt, J. A., Gupta, H. V., Bastidas, L. A., Bouten, W. & Sorooshian, S. 2003 Effective and efficient algorithm for multiobjective optimization of hydrologic models. *Water Resour. Res.* **39** (8), 1214.

Walski, T. M. 2001 The wrong paradigm: Why water distribution optimization doesn't work. *J. Water Res. Plann. Manage.* **127** (4), 203–205.

Zecchin, A. C., Simpson, A. R., Maier, H. R. & Nixon, J. B. 2005 Parametric study for an ant algorithm applied to water distribution system optimization. *IEEE Trans. Evol. Comput.* **9** (2), 175–191.

Zecchin, A. C., Maier, H. R., Simpson, A. R., Leonard, M. & Nixon, J. B. 2007 Ant colony optimization applied to water distribution system design: Comparative study of five algorithms. *J. Water Res. Plann. Manage.* **133** (1), 87–92.

Zitzler, E. & Thiele, L. 1998 Multiobjective optimization using evolutionary algorithms-A comparative case study. In *Proceedings of Parallel Problem Solving from Nature – PPSN V* (A. E. Eiben, T. Back, M. Schoenauer & H. P. Schwefel, eds.), *Vol. 1498 of Lecture Notes in Computer Science.* Springer, Berlin, Heidelberg, pp. 292–301.

Zitzler, E., Laumanns, M. & Thiele, L. 2001 SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In *Proceedings of the EUROGEN'2001 Conference, Evolutionary Methods for Design, Optimisation, and Control with Applications to Industrial Problems,* September 19–21, 2001, Athens, Greece (K. C. Giannakoglou, ed.). International Center for Numerical Methods in Engineering, The Universitat Politecnica de Catalunya, BarcelonaTech (UPC), Barcelona, Spain, pp. 95–100.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M. & Da Fonseca, V. G. 2003 Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans. Evol. Comput.* **7** (2), 117–132.