Dynamic Scenario Selection in Optimal Design Problems and Evolutionary Optimization with Uncertain System Knowledge

Nathan Sankary[1], Avi Ostfeld[2]
[1,2] Technion – Israel Institute of Technology, Faculty of Civil and Environmental Engineering, Haifa, Israel, 32000
[1]*nesankar@mtu.edu*

ABSTRACT

The design of water resource management and control systems have provided a promising space for evolutionary algorithms. In many cases a system for managing a water resource requires a large degree of planning and design before implementation and many stake holders perceive different objectives with different importance. Multiobjective evolutionary algorithms inherently provide a tool that can best satisfy the desires of many stakeholders (many objectives) through computation of a non-dominated solution set. However, the performance of an optimal solution provided by a multiobjective evolutionary algorithm is likely to deteriorate during real-world implementation if design conditions of the optimization framework are not identical to those imposed on the system in practice. This paper focuses on evaluating a scenario based multiobjective evolutionary algorithm for real-world design problems in which the environment where a system will operate is dynamic, and uncertain. A previously developed genetic algorithm termed the "RNSGA-II" used for water distribution system design is augmented to incorporate robust objectives and simple Monte Carlo sampling to solve the classic water quality sensor placement problem. This study aims to further develop an understanding of scenario based optimization methods for optimizing solutions to perform well in the face of uncertainty.

Keywords: Multiobjective optimization, robust optimization, min-max optimization, evolutionary algorithms

1 BACKGROUND

When designing a system for real-world implementation, a common difficulty arises in attempting to characterize the unknown conditions a system may operate within. In many cases, the conditions that a system will operate under cannot be explicitly known during a system's design phase. This issue is especially characteristic to water resource optimization problems. Typically, it is impossible to truly know the future state of water resource systems, such as future reservoir levels, future rainfall and stream flow levels, or future demand levels. To cope with the incomplete future knowledge inherent to the design of a water resource system, scenario planning techniques have been applied to evolutionary optimization schemes used in various water resource problems. [1]–[4]

Monte Carlo (MC), Latin Hypercube (LH), and other sampling techniques have been previously employed to generate large ensembles of possible scenarios that may be realized, and optimization typically attempts to minimize or maximize some objective measure across all sampled scenarios. However, it is often infeasible to generate all possible realizations of an operational scenario, and the computational demand to evaluate an ensemble of scenarios limits the size of an ensemble suite that can be used for optimization. In an attempt to optimize better solutions for uncertain operational scenarios, this work explores a dynamic scenario selection scheme, previously used to reduce overfitting in genetic programming [5] and employed for water distribution system (WDS) design [3]. For evaluating potential solution designs the evaluation scenario suite is dynamically re-sampled during each generation of a multiobjective genetic algorithm. Dynamically re-sampling the scenarios present in the evaluation suite is expected to increase computational efficiency by using smaller evaluation suites for evaluation, leading to robust solutions whose non-dominance is robust with respect to the objective measures, and other solutions along the Pareto front.

2 LITERATURE REVIEW

To cope with real-world design problems, scenario based optimization methods have been studied to incorporate real-world uncertainty in to optimization problems. Stochastic optimization methods [6] [7] have considered scenario based optimization as a method for robust optimization. However, less work has considered explicit scenario based optimization in an evolutionary algorithm (EA) framework. Deb et al. [8] applied a EA to a three-bar truss and cantilever beam design problem evaluated in multiple operational scenarios. To optimize a single system's design for multiple distinct operational scenarios the authors aggregated objective function values computed for each operational scenario.

In most real-world design problems, it is infeasible to enumerate all possible operational environments. Thus, system are designed to operate against a sample of scenarios which appropriately approximates the characteristics of all possible operational environments. As such, an optimization algorithm will strive to develop solutions that will perform well against "unknown" operational environments based on a fraction of feasable operational environments. Within this framework the optimization problem is similar those found in Genetic Programming (GP). Stochastically re-sampling the evaluation suite has previously been employed in GP studies to reduce solution bloat, and simultaneously improve the solutions ability to perform well in unknown scenarios [5] [9].

Over the last two decades, EAs have been extensively applied to a wide variety of water resource problems [10], [11]. Evolution algorithms have been advantageous in the water resource field because they do not require function landscape information, easily handle objective function non-linarites, and can simultaneously optimize along multiple competing objectives (multiobjective evolutionary algorithms (MOEAs)). Evolutionary algorithms have been employed to solve: WDS design and rehabilitation problems [2], [3], [12]–[15]; groundwater monitoring system and treatment design [16]–[18], water quality sensor placement [1], and more recently evolutionary algorithms were employed to solve large scale rainfall runoff model calibration [19], and watershed portfolio allocation [20].

A GA previously developed in [3], termed the RNSGA-II, inspired the exploration of the authors proposed "*R*" operator in the NSGA-II. The RNSGA-II was employed to evaluate robustness in WDS solution designs by sampling (via Latin Hypercube sampling [21]) a small number of scenario realizations (20) that the WDS may operate within. The percentage to scenarios where the WDS did not provide adequate service was used to calculate the robustness of each solution. Throughout the RNSGA-II run, the solution performance was averaged across the previous 10-30 RSNGA-II generations to describe the performance against a larger number of evaluations scenarios, without the intensive computation demand required to evaluate a large number of scenario realizations during each solutions evaluation. Further description and discussion of the RNSGA-II can be found in [3].

The methods used within the RNSGA-II were augmented to operate in a robust optimization framework using min-max objectives. The goal of min-max optimization is to find a solution which leads to the best worst case performance (optimizing solution to perform best in the worst operational case). This formulation is appealing in that it incorporates uncertainty in the operational scenarios by "insuring" that some optimal solution will provide some performance guarantee, even in the worst case. However, the min-max formulation has issues that need to be addressed.

In cases where system uncertainty is incorporated in to optimization by sampling potential operational scenarios, the quality of a solution is dependent on the condition that the worst-case scenario has been sampled for evaluation, otherwise a performance guarantee is not valid. In many cases, even if many thousands of scenarios are sampled it is unlikely that the single worst case scenario is sampled due to large scenario sampling spaces. Sampling and evaluating a large number of scenarios leads to a dramatically increased computational demand in the optimization algorithm. Aslo, min-max solutions are highly susceptible to "overfitting" the solution to the single worst case scenario that evaluation presents. In the case that all possible operational scenarios have been evaluated "overfitting" does not exist, however, in all practical cases a small sample of scenarios is sampled due to a limited computational budget. In optimizing a solution to perform well in the "expected case" (min-mean optimization) a small number of scenarios is often sufficient to describe the expected performance level of a solution, however, it is unlikely that this condition will hold for min-max optimization.

The previously developed RNSGA-II has been modified to exploit its computational savings and search efficiency for min-max optimization objectives, by removing inter-generational performance averaging.

3 METHODS

The GA employed in this work is described in the pseudo code below. In short, prior initialization of the GA an initial scenario evaluation suite is generated termed *EvalSuiteInitial* via Monte Carlo sampling from all possible evaluation scenarios. During each generation of the GA, a new, smaller evaluation suite (termed *EvalSuite*) is sampled from *EvalSuiteInitial* prior evaluation of the current population.

Algorithm: Framework for a scenario based EA using a stochastic resampling operator of the evaluation suite

Stochastic Scenario Ga (*GAoptions*, *EvalSuiteTotal*, f_1, f_2, $objective$)
Sample EvalSuiteInitial $\in EvalSuiteTotal$
Initialize a population of candidate solutions
while *termination criteria not met* **do**
    Sample EvalSuite $\in EvalSuiteInitial$
    Evaluate $objective(i, EvalSuite)\ \forall\ i \in Population$
    Select candidate solution for placement in the next population
    Crossover and *Mutate* the candidate solution
    Place candidate solutions in the next population
    Check termination criteria
End **while**

For investigation of generational sampling within the GA, a large number of optimization runs were performed using various sizes of *EvalSuiteInitial*, and *EvalSuite*, outlined in the table below. Each entry in the table represents the size of *EvalSuite* used within the GA. All GA runs were limited to 50,000,000 function evaluations.

Table 1. Evaluation suite sizing for Monte Carlo sampling within the GA.

Size of *EvalSuiteInitial*	**100**	**250**	**500**	**750**	**1000**
(*EvalSuite* / *EvalSuiteInitial*) =0.25	25	62	125	187	250
(*EvalSuite* / *EvalSuiteInitial*) =0.5	50	125	250	375	500
(*EvalSuite* / *EvalSuiteInitial*) =0.75	75	187	375	562	750
(*EvalSuite* / *EvalSuiteInitial*) =0.9	90	225	450	675	900
(*EvalSuite* / *EvalSuiteInitial*) =1	100	250	500	750	1000

4 CASE STUDY

The sensor placement problem for contamination event detection previously presented in [1] is used for evaluation in this study. This problem has been extensively explored within the literature; the task to place a set of water quality monitoring stations (sensors) at junctions throughout a WDS to provide the best (fastest, most reliable, etc.) detection of contamination. Limited by cost, sensor networks typically only provide sparse data and the performance of a set of sensors is highly sensitive to the location of each sensor. As an increasing number of sensors are placed within a distribution system, the network provides better event detection performance, however, at increased cost.

In this work, a modified RNSGA-II incorporating simple Monte Carlo random sampling and min-max optimization objectives is employed to determine the best locations within a WDS to place a set of water quality monitoring stations (sensors). Instead of defining a number of sensor a priori, the number of sensors is defined as an objective of the GA. The multiobjective nature of the employed RNSGA-II accordingly seeks to minimize the number of sensors placed in the network, while maximizing the performance of a sensor network. Messy genetic algorithm cut and splice operators [22] are used to “cross” variable length strings, which define the sensor networks.

To evaluate the performance of the sensor networks, the population affected metric is used [1], and each solution is exposed to a suite of contamination events. For each contamination event the population affected prior contamination detection is computed and reported. The minimum maximum population affected across all contamination events within the evaluation suite is reported as a respective solution’s performance. For further discussion of the population affected metric and its computation the reader is directed to [1].

5 RESULTS

The plots below present the results of the optimization scenarios conducted according to Table 1, for clarity, only selected solutions are presented here. Full Pareto fronts are presented in Figure 1, while the aggregated Pareto front from all optimization scenarios is presented in Figure 2.

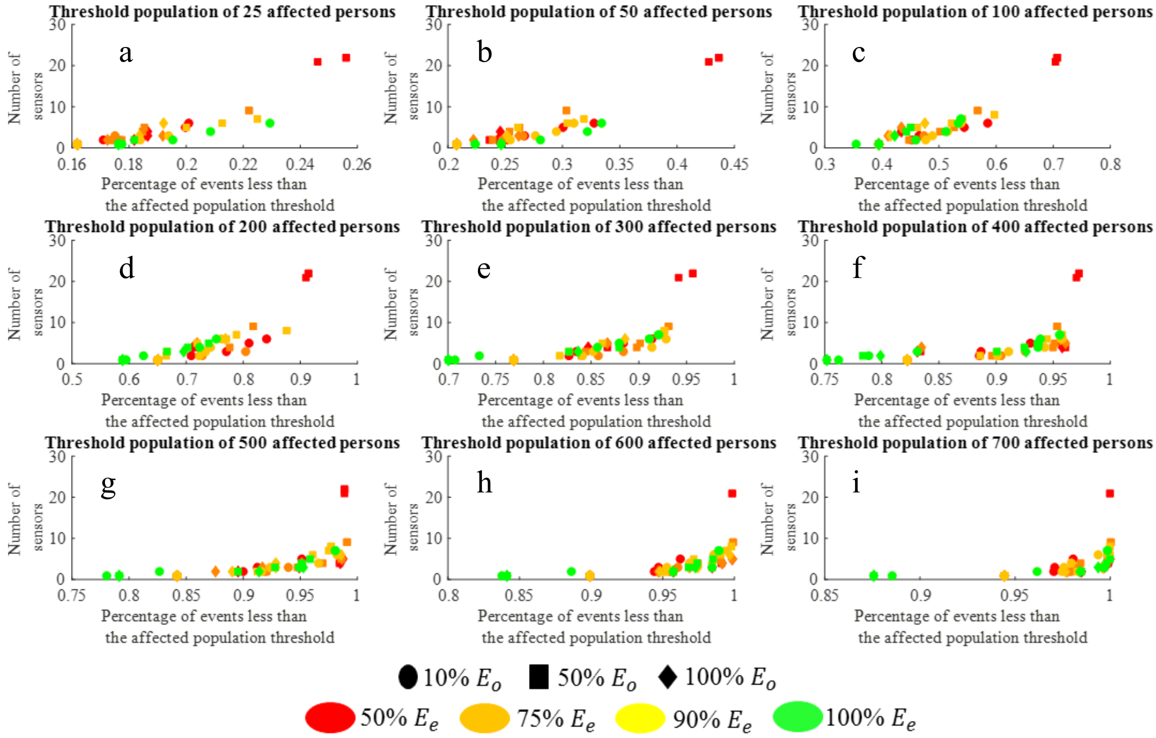

Figure 1: Performance of "optimal" sensor networks found using the dynamic evaluation suite mechanism. Color and shape of the points correspond to the respective GA optimization scenario.

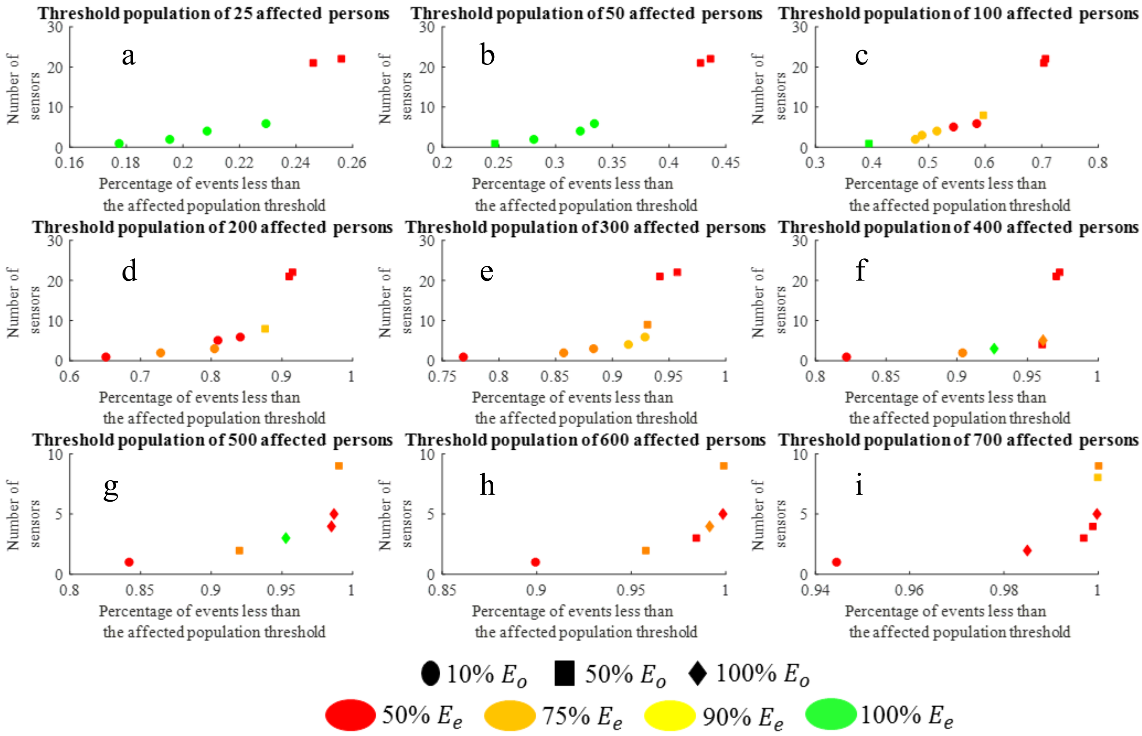

Figure 2: Aggregated Pareto fronts of "optimal" sensor networks found using the dynamic evaluation suite mechanism. Color and shape of the points correspond to the respective GA optimization scenario.

Observing the Pareto fronts presented in Figures 1 and 2 demonstrates the advantages gained by using a dynamic generational evaluation framework. For a case which demands extremely stringent performance, ie. a threshold allowable population affected of less than 100 people the best solutions are generally those generated by using a static evaluation suite (all green points are those generated using a static evaluation suite). However, these solutions provide dramatically low robustness; at best allowing 40% of all contamination events to affect populations greater than the prescribed affected population threshold (Figure 2c). Of the solutions that show more practical robustness, as seen in the plots of threshold allowable populations of 100 people or more, almost all were generated using dynamic evaluation suites (Figure 2d-i).

For comparison to a previous study, one of the highest performing solutions from the original Battle of the Water Sensor Networks [1], [23] challenge was evaluated in the same framework as the solutions generated using the augmented RNSGA-II from this study, Figure 3. The previous solution (black points) shows to be out-performed by the solutions generated using an augmented RNSGA-II.

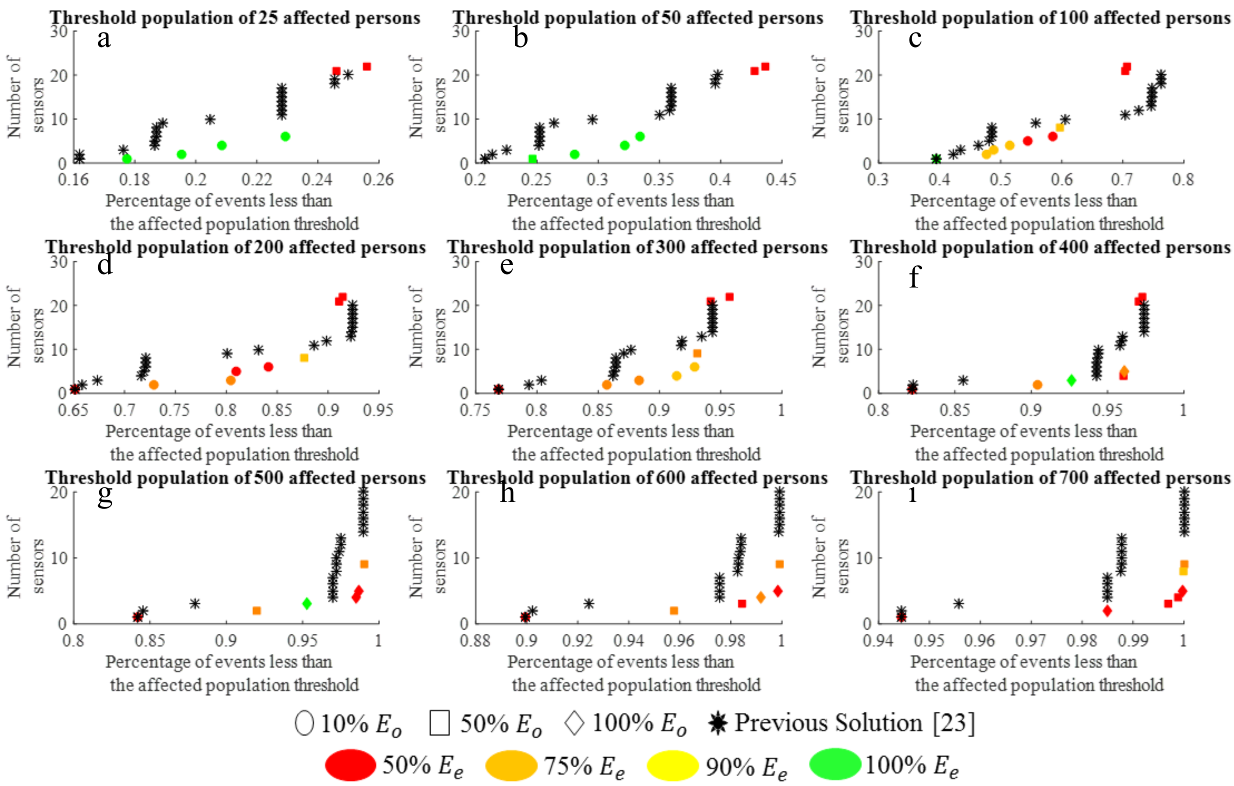

Figure 3: Aggregated Pareto fronts of "optimal" sensor networks found using the dynamic evaluation suite mechanism. Color and shape of the points correspond to the respective GA optimization scenarios.

6 CONCLUSIONS

This paper investigates a genetic algorithm operator previously proposed and studied for water distribution system design. With the intent of better understanding how to optimize difficult water resource problems under uncertain conditions, this study briefly explores the use of a dynamic scenario evaluation suite re-sampled using Monte Carlo sampling during each generation of the genetic algorithm. Opposed to sampling a large number of evaluation scenarios (in this case contamination events) and evaluating each solution against each scenario during each genetic algorithm generation, resampling the evaluation suite during each genetic algorithm generation allows for small evaluation suites. This can increase the efficiency of a genetic algorithm and allow for longer optimization runs, which would likely lead to higher performing solutions for equal computational burden. In comparison to a historically high performing solution for the sensor placement problem on the case study network, the solutions developed herein show increased robustness with respect to the population affected by an unknown contamination event. This shows the importance in considering robust performance in wireless sensor networks for contamination event detection, something that was not considered in the design of the historical comparison solution.

7 AKNOWLEGMENTS

This study was supported by the United States - Binational Science Foundation (BSF) and by the Technion Funds for Security research.

References

[1] A. Ostfeld, J. G. Uber, E. Salomons, J. W. Berry, W. E. Hart, C. A. Phillips, J.-P. Watson, G. Dorini, P. Jonkergouw, Z. Kapelan, F. di Pierro, S.-T. Khu, D. Savic, D. Eliades, M. Polycarpou, S. R. Ghimire, B. D. Barkdoll, R. Gueli, J. J. Huang, E. A. McBean, W. James, A. Krause, J. Leskovec, S. Isovitsch, J. Xu, C. Guestrin, J. VanBriesen, M. Small, P. Fischbeck, A. Preis, M. Propato, O. Piller, G. B. Trachtman, Z. Y. Wu, and T. Walski, “The Battle of the Water Sensor Networks (BWSN): A Design Challenge for Engineers and Algorithms,” *J. Water Resour. Plan. Manag.*, vol. 134, no. 6, pp. 556–568, Nov. 2008.

[2] A. Babayan, Z. Kapelan, D. Savic, and G. Walters, “Least-Cost Design of Water Distribution Networks Under Demand Uncertainty,” *J. Water Resour. Plann. Manag.*, vol. 131, no. October, pp. 375–382, 2005.

[3] Z. S. Kapelan, D. a. Savic, and G. a. Walters, “Multiobjective design of water distribution systems under uncertainty,” *Water Resour. Res.*, vol. 41, no. 11, pp. 1–15, 2005.

[4] J. Xu, M. P. Johnson, P. S. Fischbeck, M. J. Small, and J. M. VanBriesen, “Robust placement of sensors in dynamic water distribution systems,” *Eur. J. Oper. Res.*, vol. 202, no. 3, pp. 707–716, 2010.

[5] I. Goncalves, S. Silva, J. B. Melo, and J. M. B. Carreiras, “Random Sampling Technique for Overfitting Control in Genetic Programming,” *EuroGP*, pp. 218–229, 2012.

[6] M. C. Campi and G. Calafiore, “Decision making in an uncertain environment : the scenario-based optimization approach,” *Mult. Particip. Decis. Mak.*, no. 1, pp. 99–111, 2004.

[7] H. G. Beyer and B. Sendhoff, “Evolutionary algorithms in the presence of noise: To sample or not to sample,” *Proc. 2007 IEEE Symp. Found. Comput. Intell. FOCI 2007*, no. Foci, pp. 17–24, 2007.

[8] K. Deb, L. Zhu, and S. Kulkarni, “Multi-Scenario, Multi-Objective Optimization Using Evolutionary Algorithms: Initial Results,” *2015 IEEE Congr. Evol. Comput.*, pp. 1877–1884, 2015.

[9] J. Fitzgerald, R. M. A. Azad, and C. Ryan, “A bootstrapping approach to reduce over-fitting in genetic programming,” *Proc. 15th Annu. Conf. companion Genet. Evol. Comput.*, pp. 1113–1120, 2013.

[10] J. Nicklow, P. Reed, D. Savic, T. Dessalegne, L. Harrell, A. Chan-Hilton, M. Karamouz, B. Minsker, A. Ostfeld, A. Singh, and Others, “State of the art for genetic algorithms and beyond in water resources planning and management,” *J. Water Resour. Plan. Manag.*, vol.

136, no. 4, pp. 412–432, 2009.
[11] P. F. Maier H.R., Kapelan Z., Kasprzyk J., Kollat J., Matott L.S., Cunha M.C., Dandy G.C., Gibbs M.S., Keedwell E., Marchi A., Ostfeld A., Savic D., Solomatine D.P., Vrugt J.A., Zecchin A.C., Minsker B.S., Barbour E.J., Kuczera G., "Evolutionary Algorithms and Other Metaheuristics in Water Resources: Current Status, Research Challenges and Future Directions," *Environ. Model. Softw.*, vol. Under revi, 2014.
[12] A. R. Simpson, G. C. Dandy, and L. J. Murphy, "Genetic Algorithms Compared to Other Techniques for Pipe Optimization," *J. Water Resour. Plan. Manag.*, vol. 120, no. 4, pp. 423–443, 1994.
[13] D. A. Savic and G. A. Walters, "Genetic algorithms for least-cost design of water distribution networks," *J. Water Resour. Plan. Manag.*, vol. 123, no. 2, pp. 67–77, 1997.
[14] D. Halhal, G. A. Walters, D. Ouazar, and D. A. Savic, "Water Network Rehabilitation with Structured Messy Genetic Algorithm," *J. Water Resour. Plan. Manag.*, vol. 123, no. 3, pp. 137–146, 1997.
[15] T. D. Prasad and N.-S. Park, "Multiobjective Genetic Algorithms for Design of Water Distribution Networks," *J. Water Resour. Plan. Manag.*, vol. 130, no. 1, pp. 73–82, Jan. 2004.
[16] A. Singh and B. S. Minsker, "Uncertainty-based multiobjective optimization of groundwater remediation design," *Water Resour. Res.*, vol. 44, no. 2, pp. 1–20, 2008.
[17] J. B. Kollat and P. M. Reed, "Comparing state-of-the-art evolutionary multi-objective algorithms for long-term groundwater monitoring design," *Adv. Water Resour.*, vol. 29, no. 6, pp. 792–807, 2006.
[18] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A Niched Pareto Genetic Algorithm for Multiobjective Optimization," *Proc. First IEEE Conf. Evol. Comput.*, pp. 82–87, 1994.
[19] P. M. Reed, D. Hadka, J. D. Herman, J. R. Kasprzyk, and J. B. Kollat, "Evolutionary multiobjective optimization in water resources: The past, present, and future," *Adv. Water Resour.*, vol. 51, pp. 438–456, 2013.
[20] J. R. Kasprzyk, P. M. Reed, B. R. Kirsch, and G. W. Characklis, "Managing population and drought risks using many-objective water portfolio planning under uncertainty," *Water Resour. Res.*, vol. 45, no. 12, pp. 1–18, 2009.
[21] M. D. Mckay, R. J. Beckman, and W. J. Conover, "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, vol. 42, no. 1, p. 55, 2000.
[22] D. Goldberg, B. Korb, and K. Deb, "Messy genetic algorithms: Motivation, analysis, and first results," *Complex Syst.*, vol. 3, pp. 493–530, 1989.
[23] J. Krause, Andreas Leskovec and P. Isovitsch, Shannon Xu, Jianhua Guestrin, Carlos VanBriese, Jeanne Small, Mitchel Fishbech, "Optimizing Sensor Placements in Water Distribution Systems Using Submodular function maximization," pp. 1–17, 2006.