APPLICATIONS OF MULTI-OBJECTIVE, MIXED-INTEGER AND HYBRID GLOBAL OPTIMIZATION ALGORITHMS FOR COMPUTATIONALLY EXPENSIVE GROUNDWATER PROBLEMS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Ying Wan

January 2015

APPLICATIONS OF MULTI-OBJECTIVE, MIXED-INTEGER AND HYBRID GLOBAL OPTIMIZATION ALGORITHMS FOR COMPUTATIONALLY EXPENSIVE GROUNDWATER PROBLEMS

Ying Wan, Ph.D.

Cornell University 2015

This research focuses on the development and implementation of efficient optimization algorithms that can solve a range of computationally expensive groundwater simulation-optimization problems. Because groundwater model evaluations are expensive, it is important to find accurate solutions with relatively few function evaluations. As a result, all the algorithms tested in this research are evaluated on a limited computation budget.

The first contribution to the thesis is a comparative evaluation of a novel multi-objective optimization algorithm, GOMORS, to three other popular multi-objective optimization methods on applications to groundwater management problems within a limited number of objective function evaluations. GOMORS involves surrogate modeling via Radial Basis Function approximation and evolutionary strategies. The primary aim of the analysis is to assess the effectiveness of multi-objective algorithms in groundwater remediation management through multi-objective optimization within a limited evaluation budget. Three sets of dual objectives are evaluated. The objectives include minimization of cost, pollution mass remaining/pollution concentration, and cleanup time. Our results indicate that the overall performance of GOMORS is better than three other algorithms, AMALGAM, BORG and NSGA-II, in identifying good trade-off solutions. Furthermore, GOMORS incorporates modest parallelization to make it even more efficient.

The next contribution is application of SO-MI, a surrogate model-based algorithm designed for computationally expensive nonlinear and multimodal mixed-integer black-box optimization problems, to solve groundwater remediation design problems (NL-MIP). SO-MI

utilizes surrogate models to guide the search thus save the expensive function evaluation budget, and is able to find accurate solutions with relatively few function evaluations. We present numerical results to show the effectiveness and efficiency of SO-MI in comparison to Genetic Algorithm and NOMAD, which are two popular mixed-integer optimization algorithms. The results indicate that SO-MI is statistically better than GA and NOMAD in both study cases.

Chapter 4 describes DYCORS-PEST, a novel method developed for high dimensional, computationally expensive, multimodal calibration problems when the computation budget is limited. This method integrates a local optimizer PEST into a global optimization framework DYCORS. The novelty of DYCORS-PEST is that it uses a memetic approach to improve the accuracy of the solution in which DYCORS selects the point at which the search switches to use of the local method PEST and when it switches back to the global phase. Since PEST is a very efficient and widely used local search algorithm for groundwater model calibration, incorporating PEST into DYCORS-PEST is a good enhancement for PEST and easy for PEST users to learn. DYCORS-PEST achieves the goal of solving the computationally expensive black-box problem by forming a response surface of the expensive function, thus reducing the number of required expensive function evaluations for finding accurate solutions. The key feature of the global search method in DYCORS-PEST is that the number of decision variables being perturbed is dynamically adjusted in each iteration in order to be more effective for higher dimensional problems. Application of DYCORS-PEST to two 28-parameter groundwater calibration problems indicate this new method outperforms PEST by a large margin for high dimensional, computationally expensive, groundwater calibration problems.

BIOGRAPHICAL SKETCH

Ying Wan was born in November 1987 in Hunan, China. After graduating from high school in 2005 in Hunan, China, she joined the Wuhan University, one of the top universities in China. She obtained a Bachelor of Science in Hydrology and Water Resources System at Wuhan University in June 2009. In 2009, she started her studies in Civil & Environmental Engineering MS/PhD program at Cornell University with Prof. Christine Shoemaker as an advisor. Her research concerns the development and implementation of efficient stochastic optimization algorithms for simulation based real world problems that are computationally expensive to evaluate.

This thesis is dedicated to my family and friends.

ACKNOWLEDGEMENTS

Thanks to my advisor Prof. Christine Shoemaker for her guidance, her time and financial support. Through her passion for using Mathematics to solve real world environmental problems, she has been a key to my development in optimization linked to groundwater flow and transport problems.

Thanks to the CEE department, for years of funding as a teaching assistant.

Thanks to my committee members, who provided occasional and exceptionally helpful advice.

Thanks to Amandeep Singh, Suenee Tan and Min Pang from our EWRS (Environmental Water Resources and Systems) group for their kindly help and support.

Finally, thanks to my family for their constant and unwavering moral support, as well as the core values of hard work and resilience they have transmitted to me and without which this work could not have been done.

Table of Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

Computationally expensive simulation models exist in high numbers. Some examples of such models are watershed simulations, groundwater model simulations, etc. Computationally expensive models are often like black-box functions, where the mathematical characteristics, including derivatives, are not known. These models aim to mimic natural phenomena, where the relationship between the varying inputs and outputs is highly complex. As a result, obtaining the output of each simulation can be time-consuming. Models which take considerable amount of computer time in evaluation of one scenario are commonly referred to computationally expensive simulations, or computationally expensive black box functions, since the relationship between inputs and outputs are hard to define.

These expensive simulation models provide a basis to forecast and simulate different alternatives. The decision makers or engineers can then choose the best possible option or optimal solution. The need for optimization algorithms to choose the optimal solution arises when the range of parameters and number of parameter combinations is too large to enumerate and test all the possible alternatives. So optimization algorithms are associated with the simulation model to guide the search for good solutions. The typical framework used for these kind of real world optimization problems is 'Simulation-Optimization', where the simulation model attempts to mimic reality using numerical approximations of partial differential equations and the optimization model then tries to find the best set of input parameters for the simulation model. Most of the earlier work done for addressing these kinds of simulation-optimization problems are based on the standard linear programming and global optimization tools like Genetic Algorithms (GA) etc, which need simulation model to be run many many times. For a computationally expensive models, it is practically impossible to perform a large number of simulations e.g. millions or sometimes even thousands. Therefore, efficient optimization algorithms that requires relatively fewer expensive simulation are desirable. Since the optimization problems that we focus on in this thesis belong to class of 'Global Optimization', the ability of efficiently explore the whole search space within a certain time frame for a global optimization problem of an algorithm is another important area to examine. The focus of this thesis is to address this issue of efficient optimization of computationally expensive real world environmental system problems.

Simulation optimization problems can be divided into two categories: (1) calibration optimization and (2) decision optimization. Chapter 2 and chapter 3 will focus on decision optimization of a groundwater contamination problem based on a groundwater EPA superfund site. Decision optimization involves the identification of optimal decisions (e.g. pumping rates for groundwater contamination cleanup). The total cost for cleaning up contaminated groundwater can exceed many millions of dollars and the cleanup process can take many years. Groundwater remediation at a contaminated site involves multiple conflicting objectives. Consequently, by providing a new dimension to the optimization problem, multi-objective optimization is introduced to groundwater management problems in chapter 2. We compare performance of four multi-objective optimization algorithms for application to groundwater contamination cleanup problems. Chapter 3 addresses the issue of computational expense of some mixed-integer nonlinear problems. These problems arise in the groundwater sites where installation costs must be accounted in addition to Operation and Maintenance cost to determine optimal policy. A novel mixed-integer optimization algorithm SO-MI is introduced and applied to these problems. Calibration optimization involves the deduction of values for simulation model parameters (e.g. hydraulic conductivity in groundwater models) that ensure that model outputs are as close to observed data as possible. Given that both of the problems are computationally expensive, efficient and robust optimization algorithms are necessary and required. Using optimization for groundwater calibration problems was studied in my Master thesis. There I discussed the performance of three optimization algorithms and proposed a new hybrid optimization method. The algorithms were applied to calibration of a groundwater model for part of Beijing water supply. Chapter 4 proposes a new method to enhance a widely used local optimizer PEST by integrating PEST into a novel surrogate based algorithm DYCORS. PEST is the most popular tool for groundwater model calibration in water resources field given it is very efficient in finding the local optimal for the convex objective function. But the limitation of local optimization algorithm is that it may trap into local minimum when the problem is multimodal. DYCORS-PEST provides a way to integrate a widely used local search algorithm into a global optimization algorithm. This algorithm is applied to two study cases of calibration of Beijing groundwater model, and compared against Stochastic RBF and PEST on the same problems.

Chapter 2

Comparison of Algorithms for Multi-objective Optimization on Computationally Expensive Groundwater Contaminant Transport Models

2.1 Introduction

Groundwater flow and transport models have been widely used by researchers and decision makers to understand aquifer remediation movements and processes. Numerous simulation-optimization models have been developed for obtaining optimal groundwater remediation strategies. Major optimization approaches include linear programming, dynamic programming, simulated annealing and genetic algorithms; robust optimization, and evolution strategies. These papers consider a single objective, such as minimization of remediation cost, maximization of cleanup, and minimization of uncertainty, and there exists many studies dealing with single-objective optimization algorithms for optimal groundwater remediation design and management problems. However, solving real-world groundwater remediation and management problems based on a single aggregated performance metric could lead to a significant loss of information within the optimization process. The problems are generally multidisciplinary, which require inputs from different parties and thus involve multiple and often conflicting objectives. The use of multi-objective optimization has added a new dimension to groundwater remediation cleanup and management problems, providing added insight into decision making. In multi-objective optimization a set of "non-dominated" solutions (the Pareto front) are computed. Non-dominated means none of the solutions are better or worse than each other. As a result the Pareto Front is also called the trade-off curve.

There are a few studies dealing with groundwater remediation problems using multi-objective optimization technique. McKinney and Lin ([55]) used a nonlinear programming algorithm to solve multi-objective groundwater remediation problems. However, the trade-offs were obtained by single-objective optimization technique. Deb has shown that this approach can miss part of the Pareto Front if the feasible set (solutions satisfying the constraints) is not convex [20]. In general, groundwater models involving transport are not convex so this is a shortcoming of the nonlinear programming method. Erickson et al. ([30]) used niched Pareto genetic algorithm (NPGA) to solve the pump-and-treat groundwater remediation problem. Ren and Minsker ([75]) applied non-dominated sorting genetic algorithm (NSGA-II) [21] to two multi-objective remediation problems. Singh and Minsker ([78]) developed a robust multi-objective optimization method and applied it to a field-scale pump-and-treat design problem at the Umatilla Chemical Depot at Hermiston, Oregon. Singh and Chakrabarty ([79]) used a new technique of coupling NSGA-II (non-dominated sorting genetic algorithm) with both MODFLOW 2000 and MT3DMS to find a trade-off between remediation cost and extraction rate.

Multiple algorithms have been proposed algorithms for multi-objective optimization, within the simulation-optimization framework ([17, 20]). Various algorithmic contributions, within the water resources community have also been made, focusing primarily on evolutionary strategies. Evolutionary strategies are frequently referred as multi-objective evolutionary algorithms (MOEA) in contemporary literature. Tang et. al provide a comparative analysis of various evolutionary algorithms, in order to deduce their effectiveness in hydrological calibration.[48]compare the performances of four state-of-the-art evolutionary multi-objective optimization algorithms on a four-objective long-term groundwater monitoring (LTM) design problem. [68]provide a comprehensive diagnostic assessment of ten benchmark MOEAs including NSGA-II [21],, BORG[36] and AMALGAM [86] , for water resources applications addressing rainfall–runoff calibration, long-term groundwater monitoring (LTM), and risk-based water supply portfolio planning. However, there are no journal papers outside this one that applies BORG and AMALGAM to groundwater remediation problem.

While many multi-objective optimization techniques have been studied and applied to groundwater remediation problems, the computational complexity of the numerical groundwater models pose a huge challenge to the use of multi-criteria optimization algorithms. It should be noted that the groundwater remediation and cleanup design problem is a computationally expensive simulation optimization problem, since the objective functions are evaluated via simulation and a simulation evaluation for groundwater flow and transport model take a very long time. Hence, algorithms that require fewer model evaluations to produce good trade-off solutions are desirable.

In this study, we compare performance of four multi-objective optimization algorithms for application to groundwater remediation management problems. The algorithm compared in this study includes the (NSGA-II) [21], AMALGAM [86] , BORG [36], which are popular evolutionary algorithms. Performance of these algorithms is compared against a response surface assisted optimization method GOMORS[7]. The focus of our analysis is on comparing algorithm performance, within a limited evaluation budget. Our results indicate that

GOMORS outperforms all other algorithms within a limited model evaluation budget, when applied to groundwater remediation cleanup case studies.

2.2 Optimization Model Formulations

2.2.1 Overview of Multi-Objective Optimization Technique

The framework of the multi-objective optimization problem can be stated as:

$$\min_{x \in \Omega} F(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T \tag{2.1}$$
$$\text{subject to } x_i^{min} \leq x_i \leq x_i^{max}, i = 1, 2, \ldots n$$

where $x = [x_1, x_2, \ldots, x_n]$ is the vector of decision variables, bounded by x^{min} and x^{max}. $F(x)$ is the vector of user-defined objective functions and k is the number of objectives to be optimized. Some important concepts are defined as follows:

Domination: One candidate solution x_1 dominates another solution x_2 if and only if $f_i(x_1) \leq f_i(x_2)$ for all $1 \leq i \leq k$, and $f_i(x_1) < f_i(x_2)$ for some $1 \leq i \leq k$.

Non-Domination: Given a set of solution S, a candidate solution $x^* \in S$ is non-dominated in S if there does not exist anther solution which dominates x^*.

Pareto Optimality: If S is the entire feasible domain space of the defined problem, a non-dominated solution $x^* \in S$ is called Pareto-optimal solution.

Trade-off Curve/Pareto Front: The set of Pareto Optimal solutions.

The purpose of the multi-objective optimization problem is to find a set of pareto-optimal solutions, such that x^* is a non-dominated solution for the set of objective functions in the entire domain. In practice, the true pareto-front cannot be found exactly especially when the objective functions are computationally expensive. Therefore, an additional challenge for the multi-objective optimization problems is that good solutions need to be found within a limited number of function evaluations. The aim of a multi-objective optimization algorithm with application to groundwater problems is to find a solution set which is adequately close to the optimal solution set within a limited calculation budget. In later section, we discuss how to measure the goodness of the solutions found by multi-objective optimization algorithms.

2.2.2 Case Study Application

The three groundwater applications used in our analysis are derived from the Umatilla Chemical Depot which are adapted from NAVFAC (Naval Facilities Engineering Command) technical report TR-2237-ENV, NAVFAC 2004. Umatilla Chemical Depot is located at Hermiston, Oregon. It is a 19728 acre military reservation established in 1941 as an ordnance depot for storage and handling of munitions. From the 1950s until 1960s the depot was operated as an onsite explosives washout plant which processed munitions to remove and recover explosives with a pressurized hot water system. During this time, about 85 million gallons of wash water was discharged into two unlined lagoons, from where the wash water infiltrated into the soil system.

The two major chemicals, RDX (Hexahydro-1,3,5-trinitro-1,3,4-triazine, and commonly known as Royal Demolition Explosive) and TNT (2,4,6-Trinitrotoluene) are the focus of the remediation and it is assumed that in the process of removing them, other contaminants will also be removed. A pump-and-treat system was designed by the U.S.Army Corps of Engineers (USACE, 1996 and 2000) to remediate the RDX and TNT plumes(Figure 2.1). The designed pump-and-treat system consists of 3 pumping wells and 4 extraction wells (one of which is not operating currently so it is marked as inactive in the figure 2.1) which play role in the cost definitions. According to USACE design, the cleanup levels for RDX and TNT are set at 2.1 μg/l and 2.8 μg/l, respectively. In the pump-and-treat system, the contaminated water is pumped from the extracting wells. After being treated by GAC (granulated activated carbon) units, water is discharged into the infiltration basins.

The proposed multi-objective optimization framework consists of three components that are (1) groundwater simulation models, (2) optimization formulations and (3) multi-objective optimization algorithms. In our study cases, groundwater flow and transport systems are simulated by MODFLOW 2000 and MT3DMS[97] to predict the contaminant concentrations at every model grid point. The study model has 125 rows, 132 columns and 5 layers, with variable grid spacing of 24.8ft - 647.9ft along the rows and 21.6ft - 660.7ft along the columns. The formulation only focuses on contaminants transport in the first layer of the model and the boundary conditions for all four sides of the model domain were simulated as constant head. The model inputs are Hydro-geological data, Domain-discretization data and the pumping data, where the pumping/extracting data includes the well locations and the corresponding pumping/extracting rate. However, well locations are fixed in this study and only pumping/extracting rates are considered as the decision variables. After taking the input information, MODFLOW and MT3DMS start to simulate the model and concentrations of RDX and TNT are calculated as the model outputs which are used to formulate the objective functions. The multi-objective optimization algorithms links with both simulation models and the optimization formulations. In this framework, the optimization algorithm prepares inputs or decision variables of the optimization formulations. Then simulation models use the inputs to simulate model outputs, such as hydraulic heads from MODFLOW and contaminant concentrations from MT3DMS, which along with the decision variables contribute to calculate the objective functions. The optimization routine generates a new set of inputs based on the objective function values calculated earlier and passes them to the simulation models again. The whole process runs through iteratively until the maximum number of function evaluations is exceeded.

2.2.3 Formulation for Optimization

Three formulations are developed from the Umatilla Chemical Depot groundwater system which try to do the cleanup by finding the optimal pumping rates for fixed well locations and the lowest variable cost. Figure 2.1 shows the fixed wells locations.

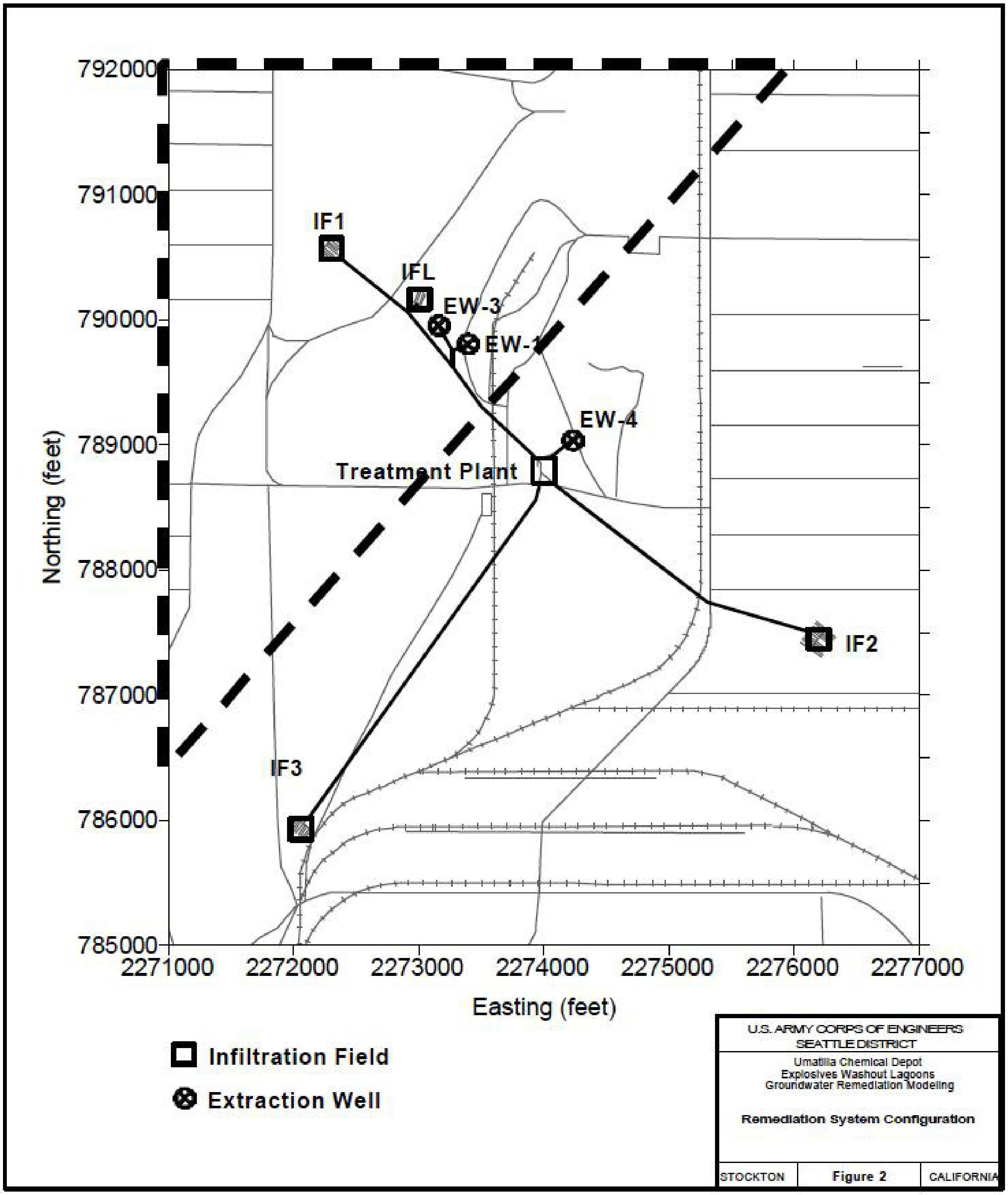


Figure 2.1: Site Map: NAVFAC (Naval Facilities Engineering Command) technical report TR-2237-ENV, 2004 showing location of extraction wells and infiltration fields (recharge basins)

2.2.3.1 Formulation 1

This formulation aims at highlighting the trade-off between the total operation costs and the mass remaining above the cleanup level for the entire project duration. The first objective is to minimize the total cost for the pump-and-treat groundwater remediation system and the other objective of the problem is the minimization of the mass of the contaminants remaining above cleanup level after the remediation period. In all formulations, the box constraints in 2.1 are imposed.

The model formulation 1 is as follows:

$$\min \text{Cost} = VCE(Q) + VCG(Q) + PenaltyCost(Q) \tag{2.2}$$

$$\min \text{Mass} = \sum_c \sum_i \max(0, Conc_{c,i} - CleanupLevel_c) * Volume_i \tag{2.3}$$

where, $VCE(Q)$ is the variable electric cost of operation wells, $VCG(Q)$ is the variable cost of GAC units, $PenaltyCost(Q)$ is the cost for violating the total pumping constraint, as the decision variable, $Q = [Q_1, Q_2, \ldots, Q_{10}]$ is a vector of pumping rates for 10 wells, among which well 1-8 are pumping wells and well 9-10 are recharge basins with the last recharge basin getting a recharge as constraint 3, all the costs are computed in net present value (NPV) with a discount rate of $r = 5\%$, c states the chemical type (RDX or TNT in this study), $Conc_{c,i}$ is the concentration of chemical c from cell i of the groundwater numerical model, $CleanupLevel_c$ is the fixed maximum concentration level that chemical c can get up to which is discussed earlier and the clean up levels for RDX and TNT are 2.1 μg/l and 2.8 μg/l, respectively, $Volume_i$ is the volume of cell i of the groundwater model, and the function $\max(0, Conc_{c,i} - CleanupLevel_c)$ demonstrates that the mass of chemical c in cell i only gets accounted when its concentration is above the cleanup level. The second objective formulation is also discussed by [78].

The formulation includes the following constraints:

1. This model consists of 1 management period of 4 years, within which the pumping rates of all the wells are kept throughout this period.

2. the total pumping rate, after adjustment for the average amount of system uptime, cannot exceed 1300 gpm. Therefore, the current maximum capacity of the treatment plant is $Q_{total} \leq 1300\alpha$ gpm, where α is the coefficient of the average amount of system time ($\alpha = 0.9$ in this study). The total pumping constraint is implemented by using the penalty function thus the solutions not satisfying this constraint are penalized to force the algorithm to look for solution points satisfying the constraint.

3. The pumping capacity of individual wells must not exceed 400 gpm in the less permeable portion of the aquifer and 1000 gpm in permeable portion. This constraint is implemented as the lower and upper bounds of the decision variables.

4. The total amount of pumping must equal the total amount of injection through the infiltration basins within an error tolerance. Thus, the pumping rate of the third recharge basin is set to be (Total pumping)-(Total recharge).

2.2.3.2 Formulation 2

This formulation aims at finding the trade-off between the total operation costs and the total contaminant concentration above the cleanup level for the entire project duration. Similar to formulation 1, the first objective is to minimize the total cost for the pump-and-treat groundwater remediation system. The second objective is the minimization of the concentration of RDX and TNT of the contaminants remaining above cleanup level after the remediation period.

The model formulation 2 is as follows:

$$\min \text{Cost} = VCE(Q) + VCG(Q) + PenaltyCost(Q) \tag{2.4}$$

$$\min TotalConcentration = \sum_{c} \max(0, MaxConc_c - CleanupLevel_c) \tag{2.5}$$

where, $VCE(Q)$ is the variable electric cost of operation wells, $VCG(Q)$ is the variable cost of GAC units, $PenaltyCost(Q)$ is the cost for violating the total pumping constraint, as the decision variable, $Q = [Q_1, Q_2, \ldots, Q_{10}]$ is a vector of pumping rates for 10 wells, among which wells 1-8 are pumping wells and wells 9-10 are recharge basins with the last recharge basin getting a recharge as constraint 3. All the costs are computed in net present value (NPV) with a discount rate of $r = 5\%$. The variable$c$ states the chemical type (RDX or TNT in this study). $MaxConc_c$ is the maximum concentration of chemical c of the groundwater numerical model among all the nodes. $CleanupLevel_c$ is the fixed maximum concentration level that chemical c can get up to which is discussed earlier and the clean up levels for RDX and TNT are 2.1 μg/l and 2.8 μg/l, respectively. The function $\max(0, MaxConc_c - CleanupLevel_c)$ demonstrates that the concentration of chemical c only gets accounted when its maximum concentration is above the cleanup level. This formulation is very similar to formulation 1, but it provides some insights focusing on contaminant concentration instead of mass. The constraints are identical to formulation 1 and are given in 2.1.

2.2.3.3 Formulation 3

In the first and second formulation, the remediation period is fixed to be 4 years. In the groundwater remediation problems, reducing the remediation time is as important as reduce the cost. Therefore, incorporating remediation time as a decision variable in the optimization problem may give better results in minimizing the remediation cost. Thus similar to Formulation 1, the first objective is the minimization of the total operation cost. The second objective is the minimization of the remediation time.

The formulation 3 is as follows:

$$\min \mathrm{Cost} = VCE(Q,t) + VCG(Q,t) + PenaltyCost(Q,t) \tag{2.6}$$

$$\min \text{Time} = t \tag{2.7}$$

where, Q represents the vector of pumping rates for the 10 wells, t stands for the remediation time, $VCE(Q, t)$ is the variable electric cost of operation wells, $VCG(Q, t)$ is the variable cost of GAC units, and $PenaltyCost(Q, t)$ is the cost for violating the constraints. Different from the Formulation 1 and 3, the penalty function consists of two components: the penalty cost for violating concentration constraints and the penalty cost for violating the total pumping constraint. The total pumping constraints remains the same as the Formulation 1. If the predicted concentration is above the cleanup level, the penalty cost will be added to the total cost. The constraints of Formulation 3 are listed as follows:

1. The total pumping rate, after adjustment for the average amount of system uptime, cannot exceed 1300 gpm. Therefore, the current maximum capacity of the treatment plant $Q_{total} \leq 1300\alpha$ gpm, where α is the coefficient of the average amount of system time ($\alpha = 0.9$ in this study). Same as Formulation 1, the total pumping constraint is implemented by using the penalty function thus the solutions not satisfying this constraint are penalized to force the algorithm to look for solution points satisfying the constraint.

2. RDX and TNT concentrations cannot exceed their respective cleanup levels at the end of the remediation time. Similar to the first constraint, it is built into the penalty function.

3. The pumping capacity of individual wells must not exceed 400 gpm in the less permeable portion of the aquifer and 1000 gpm in permeable portion. This constraint is implemented as the lower and upper bounds of the decision variables.

4. The lower and upper bounds of the remediation time is 0 and 8, respectively

5. The total amount of pumping must equal to the total amount of injection through

the infiltration basins within an error tolerance. Thus, the pumping rate of the third recharge basin is set to be the balance of (Total pumping)-(Total recharge).

One purpose of constructing three different 2 objective problems is to have three test cases on which to test the relative performance of multiple algorithms.

2.3 Optimization Algorithms

2.3.1 Non-Dominated Sorting Genetic Algorithm - II (Parallel version)

The Elitist Non-Dominated Sorting Genetic Algorithm II (NSGA-II), proposed by [21], is a widely known evolutionary algorithm for solving multi-objective optimization problems. NSGA-II is a second generation MOEA which made significant improvement to NSGA. NSGA-II has been used as a classic benchmark algorithm in many studies and has been applied to various real engineering problems across numerous fields, especially where the objectives are highly nonlinear [17]. Some of recent applications of NSGA-II to water resources problems include multi-objective automatic calibration of SWAT model[13], Long-term groundwater monitoring design [69], watershed water quality management [26], and decision making in water distribution network [10]. In the evolutionary search optimization process of NSGA-II, it ranks and archives parent and child populations based on the non-dominating sorting and crowding distance on a particular front. The non-dominating sorting approach makes use of fitness value to rank the solutions and assign them to different fronts. Crowing distance is a measure of diversity of a solution. NSGA-II moves the non-dominated front towards convergence by using Pareto-dominance during the whole search process.

2.3.2 AMALGAM - A MultiAlgorithm Genetically Adaptive Multi-Objective Optimization Method

AMALGAM, proposed by [86], is a multi-method evolutionary multi-objective optimization algorithm, which adaptively incorporates multiple MOEAs, NSGA-II, PSO, DE, and adaptive metropolis (AM). The key features of AMALGAM is the simultaneous multi-method search and self-adaptive offspring creation. The recent application of AMALGAM includes a complex hydrologic model calibration problem in [95]with comparison against SPEA2 and NSGA-II.

2.3.3 BORG

BORG is a multi-objective evolutionary algorithm designed for many-objective, multi-modal optimization problems, which is proposed by Hadka et al.[36]. The key features of BORG are 1) ϵ-dominance archive and ϵ-progress to insure a well-spread Pareto front; 2) a restart mechanism triggered to avoid local minima; 3) adaptive multiple search operator selection. Similar to AMALGAM, BORG incorporates a class of operators, Simulated Binary Crossover (SBX), Differential Evolution (DE), Parent-Centric Crossover (PCX), Unimodal Normal Distribution Crossover (UNDX), Simplex Crossover (SPX) and Uniform Mutation (UM). These operators can be adaptively selected by BORG based on the optimization problem. Applications of BORG to water resources engineering problems and comparison against other multi-objective optimization algorithms are extended by[68]. Across these applications, BORG is the top overall algorithm. A commercial package for BORG [36] that is highly parallelized is also available, but has not been applied to groundwater problems. This paper uses the serial version of BORG.

2.3.4 GOMORS - Gap Optimized Multi-Objective Optimization with Response Surfaces

GOMORS [7] is a novel surrogate-based multi-objective optimization strategy, which is designed for multi-modal and computationally expensive optimization problems. GOMORS consists of 4 major components: 1) sampling initial points, 2) constructing a Response Surface Model, 3) solving the multi-objective problem from the surrogate model by MOEA, 4) picking additional points from the surrogate based Pareto front. The unique feature of GOMORS is that GOMORS is an iterative scheme which makes use of Radial Basis surrogate response surface model to approximate the expensive objective functions such that multi-objective search is guided towards the optimal solution with limited computation budget. During the search in the surrogate model approximated via Radial Basis Functions, MOEAs (NSGA-II and MOEA/D) are embedded and the solutions obtained from the cheap function evaluations are selected for expensive function evaluations. The other advantage of GOMORS is that it can be easily modified to incorporate parallelization during the expensive function evaluations, and hence can be more efficient. In this study, GOMORS is parallelized up to 8 processors and NSGA-II is also modified into a parallel version with 8 processors.

2.4 Algorithm Comparison Methodology

The effectiveness of a multi-objective optimization algorithm can be assessed via analysis of algorithm "efficiency". Efficiency corresponds to the effectiveness of an algorithm in deducing a set of good "quality" trade-off solutions, quickly and within a limited simulation evaluation budget. Due to the stochastic nature of all algorithms compared in this analysis, it is also important to understand and compare algorithm "reliability", i.e the ability of an algorithm to produce good "quality" solutions, consistently over multiple trial runs. Defining "quality" of trade-off solutions is an ongoing debate within the research community today. [20] defines

”quality” of a multi-objective solution (also called approximate front and non-dominated front) to be a combination of two (potentially conflicting) properties: 1) ”convergence” and 2) ”diversity”. Convergence can be described as the proximity of an algorithm’s solutions, to the Pareto front, and ”diversity” is the extent of the true trade-off represented by an algorithm’s solutions.

A visual analysis and comparison of trade-offs obtained via different algorithms is the most common comparison methodology employed in prior literature [82, 20]. Efforts have also been employed to quantify ”quality” of trade-off solutions, within a single performance measure. In our analysis, we employ a combination of visual trade-off analysis, and performance metric based analysis, in order to understand and compare performance of algorithms on the test suite. Three performance metrics have been used in this analysis, the hypervolume metric [99], the Inverted Generational Distance (IGD) [85]metric and the Generational Distance metric [84].

2.4.1 Performance Metrics

Performance metrics are used to evaluate the approximation sets produced by running an MOEA, allowing the comparison of approximation sets using numeric values. It provides a measure of “convergence” and “diversity” by calculating the volume of the feasible objective space not dominated by solutions obtained by an algorithm. The feasible objective space is the hypercube bounded by the “Reference Set” and the “worst point”. The “Reference Set” of a multi-objective optimization problem is the vector depicting minimum attainable values of all objectives. In our case studies, we would like to reduce all the objectives to zeros. Hence, the “Reference Set” for all the groundwater management problems is the zero vector. It should be noted that the “Reference Set” is usually an unattainable solution, and the Pareto front of a groundwater problem is dominated by the “Reference Set”. Since the true Pareto front is not known for all the groundwater management problems discussed in this analysis, we generated our best approximations of the problems’ Pareto fronts across all runs of all the algorithms as our 'Reference Sets', which served as references for assessing convergence through the performance metrics. The "worst point" is the vector with the worst attainable values of all objectives. Since the worst attainable values are not known in the problems discussed in this study, the "worst point" vector is estimated by the worst values of all objectives obtained in our computer experiments.

Our results and analysis will focus on three performance measures: (1) hypervolume , (2) inverted generational distance (IGD), and (3) generational distance (GD) . Each of these measures provides unique insights into the performance of MOEAs.

The hypervolume [99] is the most challenging of the three measures to satisfy. Hypervolume measures the volume of objective space dominated by an approximation solution set. The Approximate Solution Set is the final approximation of the Pareto front deduced by a single run of an algorithm. The hypervolume indicator is calculated as the difference in hypervolume between the Approximate Solution Set and the "Reference Set". The hypervolume calculations is performed across the normalized objective function values, hence the volume integral is calculated between the reference points 0 to 1. In our study, we focus on the uncovered hypervolume, which is the volume between the "Reference Set" and the Approximate Solution Set. The sum of hypervolume and uncovered hypervolume is 1. The hypervolume provides a comprehensive quantification of algorithms' convergence and diversification abilities. Figure 2.2(a) provides a visual illustration of the hypervolume metric, i.e., the blue shaded region is the hypervolume and the pink shaded region is the uncovered hypervolume.

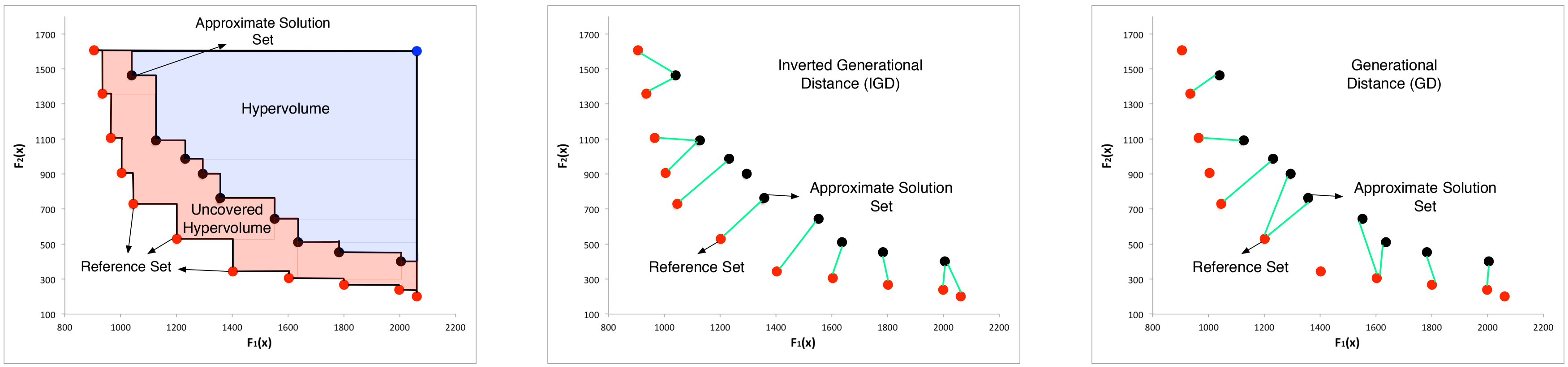
The Inverse Generational Distance (IGD) metric [85] is illustrated in Figure 2.2(b). The IGD metric is the minimum distance of the Approximate Solution Set from the "Reference Set" and is depicted by the green line in Figure 2.2(b). For the algorithms discussed in this analysis, the Approximate Solution Set is the set of non-dominated solutions obtained from a single algorithm run. The IGD metric is a measure of convergence of an algorithm to the "Reference Set".

The Generational Distance (GD) metric [84] is illustrated in Figure 2.2(c) GD metric is the average distance between the Approximate Solution Set and the "Reference Set" as depicted by green lines. Similar to IGD, GD is a measure of convergence towards the "Reference Set".

2.4.2 Experimental Setup

We initiated the algorithm comparison methodology by deducing suitable values for parameters of all algorithms under discussion. A small trial-and-error exercise was performed to tune population sized for all multi-objective evolutionary algorithms (MOEAS). Since performance of MOEAs is highly dependent on population sizes, we ran multiple trials of NSGA-II, AMALGAM on the Umatilla groundwater test case studies, with population sizes of 20, 50, 100 and 200, with an evaluation limit of 500. The initial trial-and-error analysis showed that within the limited evaluation budget of 500, a population size of 20 was desirable for all MOEAs, for all three test cases. BORG's configuration recommended by Hadka et. al.[36] was employed.

Due to the stochastic nature of all algorithms, multiple trial runs were performed for all groundwater case studies and each trial was using the same starting points for all algorithms. Our analysis was focused at comparing algorithms' performances, in terms of efficiency, effectiveness, and reliability, within a simulation evaluation budget of 200. Efficiency of all algorithms was compared via plotting performance metric values (averaged over multiple trial runs) against number of function evaluations. These plots are called progress graphs in our discussion. Since it is hard to quantify quality of solutions obtained via an algorithm, through a single metric, we used three metrics in this analysis, namely, hypervolume, IGD and GD, all of which have been discussed before. We performed 10 trials for each algorithm with 200 function evaluations for the three groundwater study cases.



(a) (b) (c)

Figure 2.2: Illustration of performance metrics used in this study: a) The Hypervolume provides a comprehensive quantification of algorithms' convergence and diversification abilities, b) The Inverse Generational Distance (IGD) is a good measure of convergence. c) The Generational Distance (GD) also predominantly captures convergence

Effectiveness and reliability of all algorithms was compared via a visual comparison of worst-trade-offs obtained through each algorithm after a fixed number of function evaluations, along with a box plot comparison of the hypervolume metric. The worst trade-off obtained by an algorithm refers to the worst approximation of the Pareto front obtained by an algorithm in multiple trials, according to the hypervolume metric value.

2.5 Results

2.5.1 Time Analysis - Performance Metrics

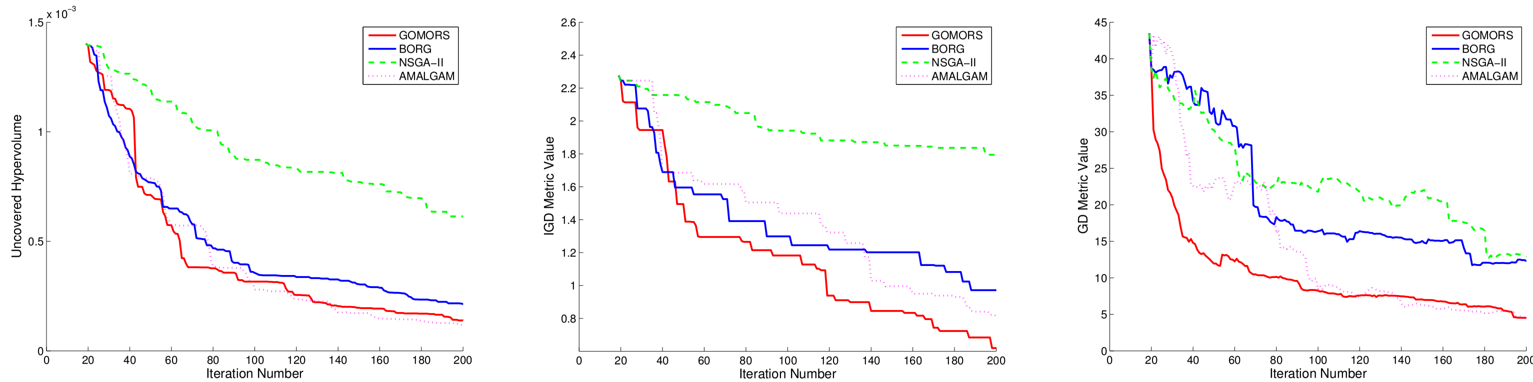
The relative efficiency of the algorithms discussed in this analysis can be summarized by progress graphs. Progress graphs plot values of performance metrics against number of function evaluations, to visualize algorithm progress with time. Before moving towards a detailed discussion of comparative performance of the algorithms, we summarize their relative performances on all groundwater test problems in Figure 2.3-2.5 through progress graphs. Each figure corresponds to one groundwater test problem and provides visualization of algorithm progress with number of function evaluations according to the three performance metrics discussed earlier. Each figure contains 3 plots, depicting progress of algorithms according to a) hypervolume metric, b) IGD metric and c) GD metric. As was mentioned earlier, the hypervolume metric tends to highlight both convergence and diversification, while the IGD and GD metrics capture convergence capabilities of algorithms. Please note that lower values of all metrics are desirable.

Two deductions are evident from the analysis in 2.3-2.5: 1) Overall average performance of GOMORS is better than all other algorithms for all test problems within a limited evaluation budget of 200 and 2) NSGA-II is the least efficient algorithm for all groundwater test problems within a limited evaluation budget of 200. The progress graph also shows that results can vary significantly according to the choice of performance metric. Hence, we employed three different metrics to ensure a fair comparison and to identify if any algorithm performs consistently better for all metrics. The performances of BORG and AMALGAM are also consistently good. 2.3 and 2.5 indicate a better performance of AMALGAM than BORG, but the difference between them are very small. Overall, the progress graphs indicate relative superiority of GOMORS, AMALGAM and BORG over NSGA-II for a limited evaluation budget of 200.

2.5.2 Box Plot Analysis

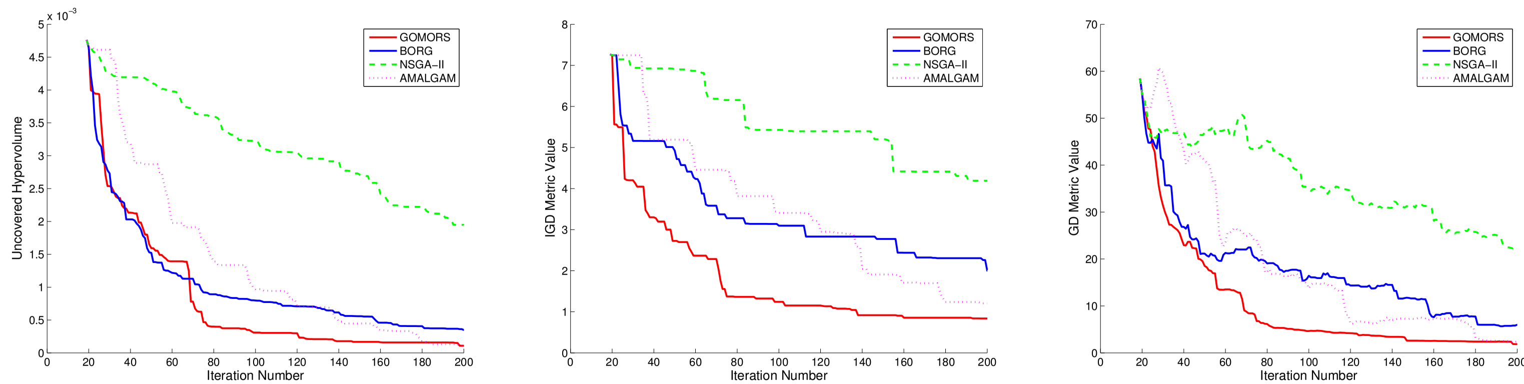
While the analysis of 2.5.1indicates relative superiority of GOMORS over all other algorithms according to a performance metric based analysis based on average across trials, a box plot comparison is employed to show the effectiveness and consistency in producing a better trade-off (with better convergence and diversification), and robust over multiple trial runs. The box plot comparisons, based on the hypervolume metric, depict superior performance of GOMORS, AMALGAM, and BORG, within a limited evaluation budget of 200, for all case studies.

In Figure 2.6 GOMORS obtains the smallest median in all three case studies and smallest spread in Case 2 and 3 for both 100 and 200 function evaluations among all four algorithms. In Case 1, AMALGAM has the smallest spread and median and followed by BORG and GOMORS. It is also noticed that GOMORS, BORG and AMALGAM converge very fast before 100 function evaluations except NSGA-II in Case 3. AMALGAM performs slightly better than BORG on average. The overall trend of this analysis shows that performance of GOMORS is consistently good across all test case studies for 100 and 200 function evaluations.



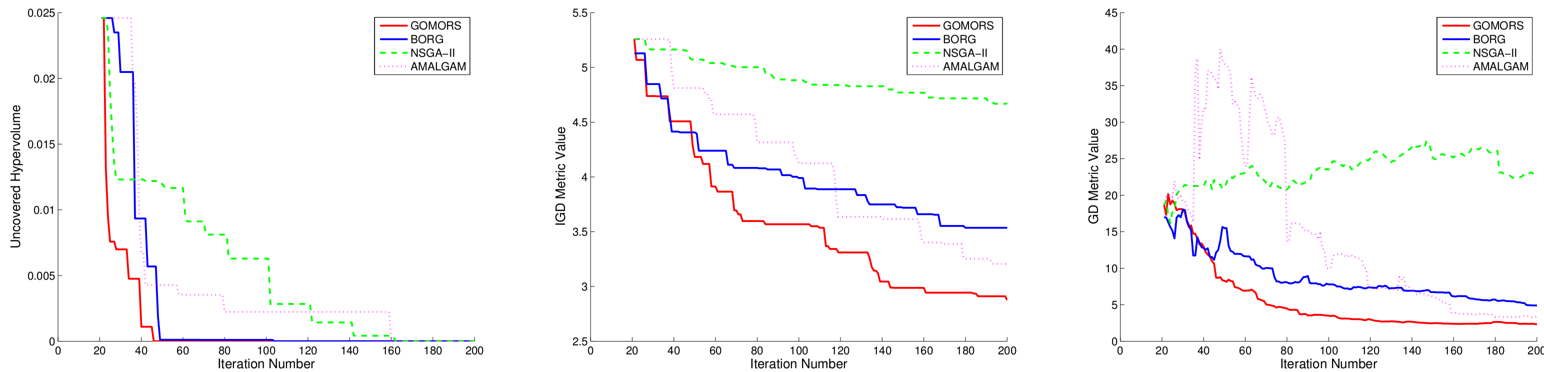
(a) (b) (c)

Figure 2.3: Problem UMA-1: Progress graphs with plots of average values of : a) Hypervolume, b) IGD metric, c) GD metric against number of function evaluations, averaged over 10 trials for four algorithms.



(a) (b) (c)

Figure 2.4: Problem UMA-2: Progress graphs with plots of average values of : a) Hypervolume, b) IGD metric, c) GD metric against number of function evaluations, averaged over 10 trialsfor four algorithms.



(a) (b) (c)

Figure 2.5: Problem UMA-3: Progress graphs with plots of average values of : a) Hypervolume, b) IGD metric, c) GD metric against number of function evaluations, averaged over 10 trialsfor four algorithms.

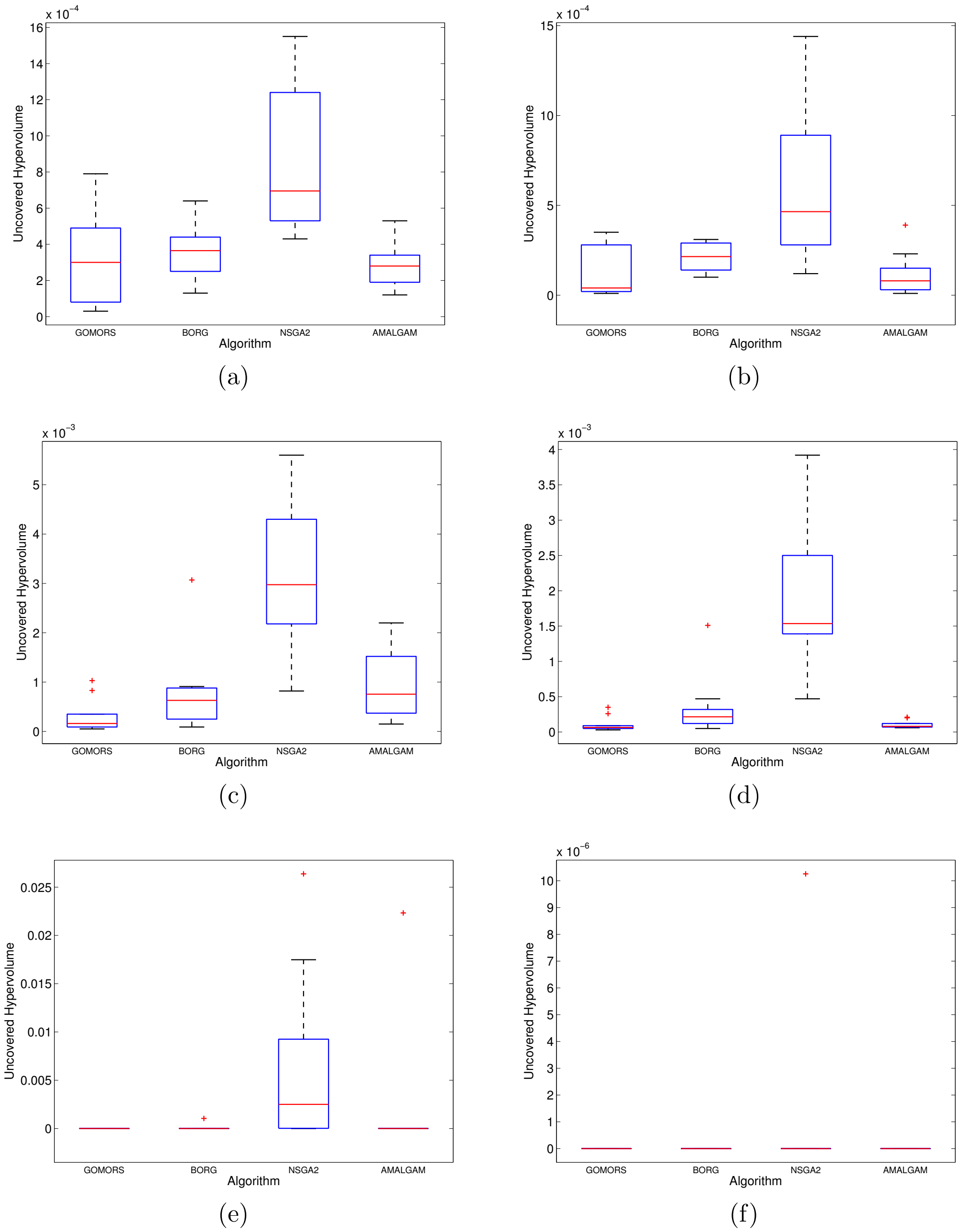


Figure 2.6: Box plot algorithm comparison of the uncovered hypervolume metric after 100 and 200 function evaluations (10 trials): (a-b) Problem UMA-1 after 100 and 200 function evaluations; (c-d) Problem UMA-2 after 100 and 200 function evaluations; (e-f) Problem UMA-3 after 100 and 200 function evaluations;

We assess the difference in performance between GOMORS, AMALGAM and BORG through the Mann-Whitney Rank Sum test performed over the hypervolume metric values obtained for each algorithm in multiple trials. The Rank Sum test is a non-parametric statistical hypothesis test for deducing whether results obtained from one algorithm in multiple trials runs are significantly different from results obtained from another algorithm in multiple trials. The algorithms are compared in pairs, the hypervolume metric value is used as the performance quality measure and the Rank Sum Test is performed for all three formulations (three cases) after 50, 100 and 200 evaluations (except for Case 3 because of the fast convergence of all the algorithms). Hence, there are 18 Rank Sum tests, 6 for each test problem.

A summary of the Mann-Whitney Rank Sum Test is provided in Table 2.1. The indicated results in the table cells are the p-values of the Rank Sum Test. * after the p-value denotes that GOMORS is significantly different from the algorithm in the row with significant level 5%. The results indicate that performance of GOMORS is better than AMALGAM, in 5 out of 9 comparisons. AMALGAM is never statistically better than GOMORS. Performance of GOMORS is better than BORG in 4 out of 9 comparison. However, BORG is never statistically better than GOMORS.

Table 2.1: Summary of p-values from a statistical comparison via Mann-Whitney Rank Sum Test applied to GOMORS, AMALGAM, BORG, according to hypervolume metric after 50, 100, and 200 function evaluations. * after the p-value denotes that GOMORS is better than the algorithm in the row with significance level at $\alpha = 5\%$.

Problem	Algorithm	50 evals (30 evals for Case 3)	100 evals (40 evals for Case 3)	200 evals (50 evals for Case 3)
Case 1	AMALGAM	0.7959	0.9560	0.9557
	BORG	0.9118	0.5920	0.1964
Case 2	AMALGAM	0.0433*	0.0094*	0.2520
	BORG	0.7394	0.0409*	0.0194*
Case 3	AMALGAM	0.0054*	0.0188*	0.0205*
	BORG	0.0437*	0.0478*	0.4270

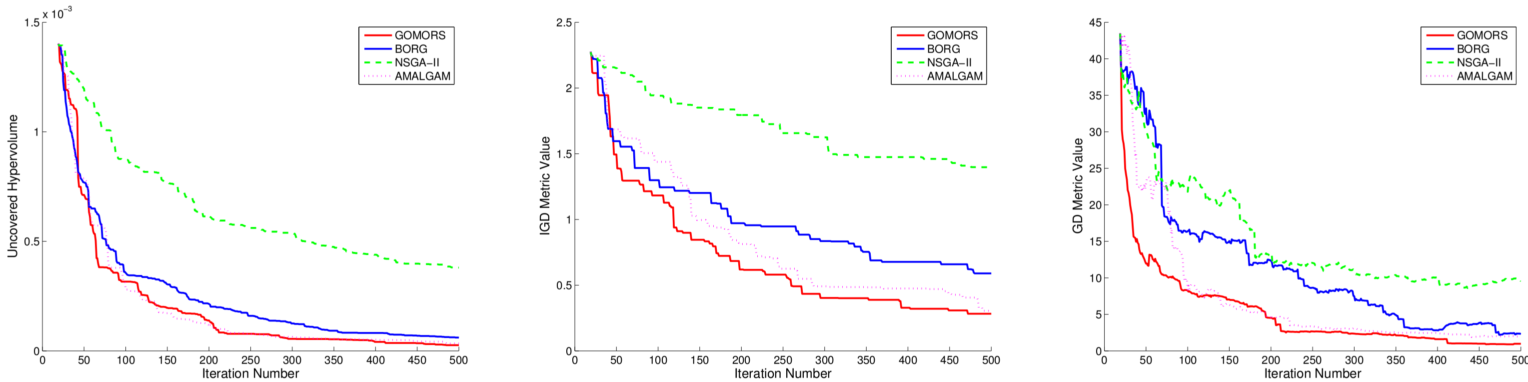
2.5.3 Long-term Analysis

While GOMORS, AMALGAM and BORG have good performances within a function evaluation of 200, it is also important to know, how consistent the performances will be after considerably more function evaluations. Figure 2.7-2.9 provides a progress graph of GOMORS, AMALGAM, BORG and NSGA-II with a function evaluation budget of 500. The same as the time analysis in section 2.5.1, each figure corresponds to one groundwater test problem and provides visualization of algorithm progress with number of function evaluations according to the three performance metrics a) hypervolume metric, b) IGD metric and c) GD metric.

The results indicate the same overall trend as the time analysis in section 2.5.1 that the overall performance of GOMORS is superior to all other algorithms and NSGA-II is least efficient algorithm among the four algorithms. When the iteration number approaches 500, the differences between GOMORS and AMALGAM become very small and AMALGAM has slightly lower value than GOMORS in 2 out of 9 cases. However, we can still observe that GOMORS has the fastest drop at the beginning for almost all the cases in all three performance metrics indicating its superior efficiency with small number of function evaluations. It is evident from Figure 2.9 as well, that GOMORS tends to converge quickly within a limited evaluation budget. BORG has better performance than NSGA-II in all the test cases, but it's not as efficient and consistent as GOMORS and AMALGAM.

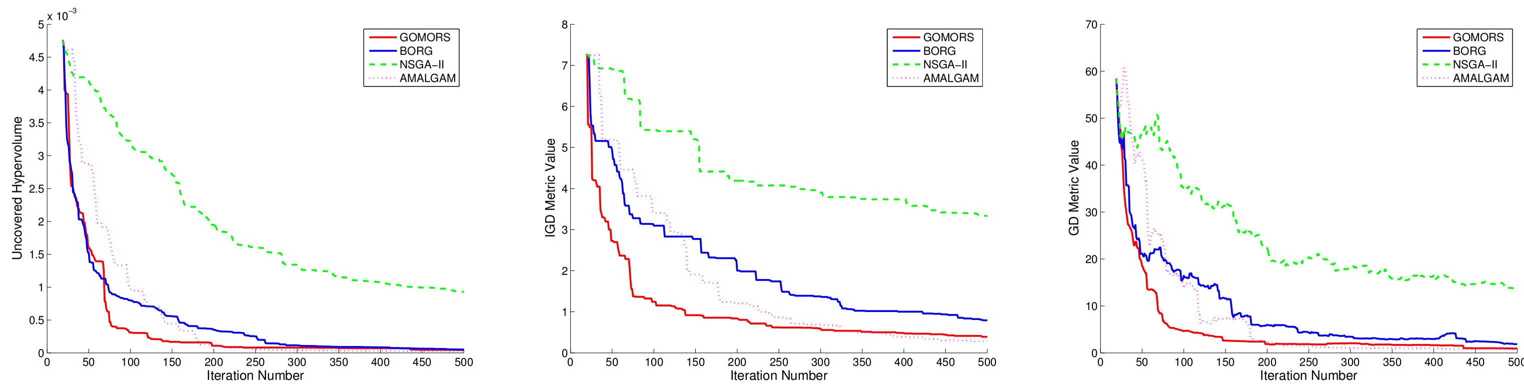
2.5.4 Meaningful Trade-offs Analysis

While it is important to know that the level of convergence and diversity obtained within a limited simulation evaluation budget is acceptable, an equally important ability of an



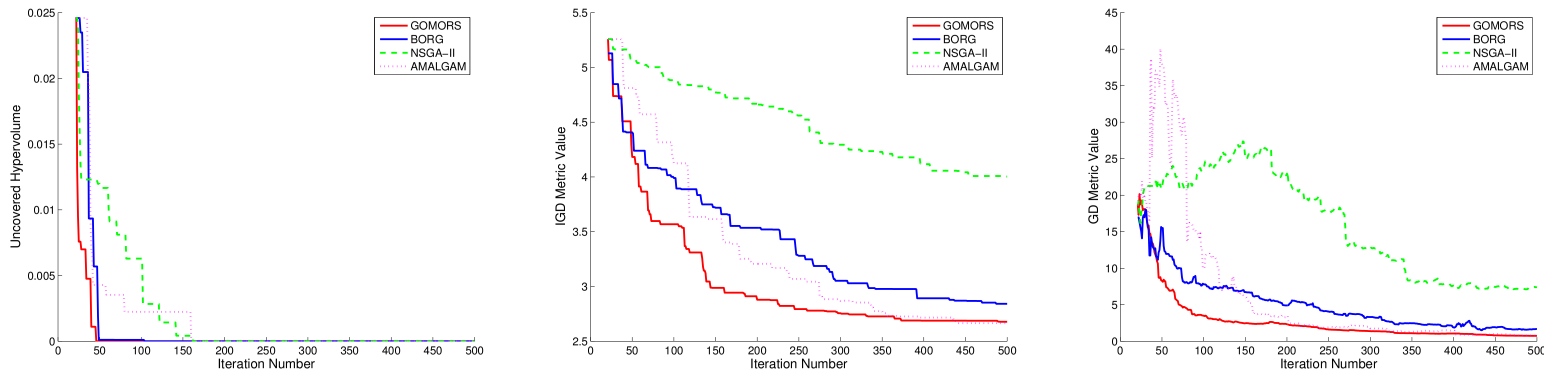
(a) (b) (c)

Figure 2.7: Long-term Analysis. Problem UMA-1: Progress graphs with plots of average values of : a) Hypervolume, b) IGD metric, c) GD metric against number of function evaluations, averaged over 10 trials with 500 evaluations



(a) (b) (c)

Figure 2.8: Long-term Analysis. Problem UMA-2: Progress graphs with plots of average values of : a) Hypervolume, b) IGD metric, c) GD metric against number of function evaluations, averaged over 10 trials with 500 evaluations.



(a) (b) (c)

Figure 2.9: Long-term Analysis. Problem UMA-3: Progress graphs with plots of average values of : a) Hypervolume, b) IGD metric, c) GD metric against number of function evaluations, averaged over 10 trials with 500 evaluations.

algorithm is to produce "meaningful trade offs" effectively. The term "meaningful trade-offs" has been defined and used by various authors in the water resources literature (e.g.[29, 49]). It essentially refers to trade-off (non-dominated) solution obtained by multi-objective optimization, which are typically within upper or lower bounds on the range of permissible values of each of the objective functions. As a result, the definition of a meaningful trade-off is subjective which depends on the preference of the stakeholders or the features of the optimization problem. We define box constraints on objective function values to identify meaningful solutions for trade-offs obtained from different algorithms. Figure 2.10 provides an illustration of the implementation of meaningful trade-offs. While many trade-off solutions might exist between competing objectives for groundwater flow and transport problems, the number of meaningful trade-off solutions might be very limited. This is illustrated in the trade-offs obtained in our analysis as well. For instance, clean up cost values for Case 1 for trade-off solutions obtained via GOMORS range up to hundreds of million dollars. One can argue that any cost greater than $10million is not acceptable. Consequently, many non-dominated solutions will not be considered acceptable or meaningful if the cost is greater than $10 million.

We investigate the ability of GOMORS, AMALGAM, BORG and NSGA-II, in producing acceptable or meaningful solutions, by using constraints to remove trade-off solutions that do not achieve acceptability thresholds according to each objective. The meaningful trade-off analysis is performed for all three groundwater test cases. The first step in this process is to define acceptability thresholds for every objective in three test cases. According to previous studies of Umatilla chemical depot [12, 78], the acceptability thresholds for Case 1 are no greater than $10 million as the cleanup cost and 8kg as the maximum contaminant mass remaining. In the case of the second test problem, all trade-off solutions with cleanup cost less than $10 million and total concentration less than 10 are considered as meaningful or acceptable trade-offs. For Case 3, trade-off solutions with cleanup cost less than $20 million and cleanup time less than 5 years are deemed meaningful or acceptable.

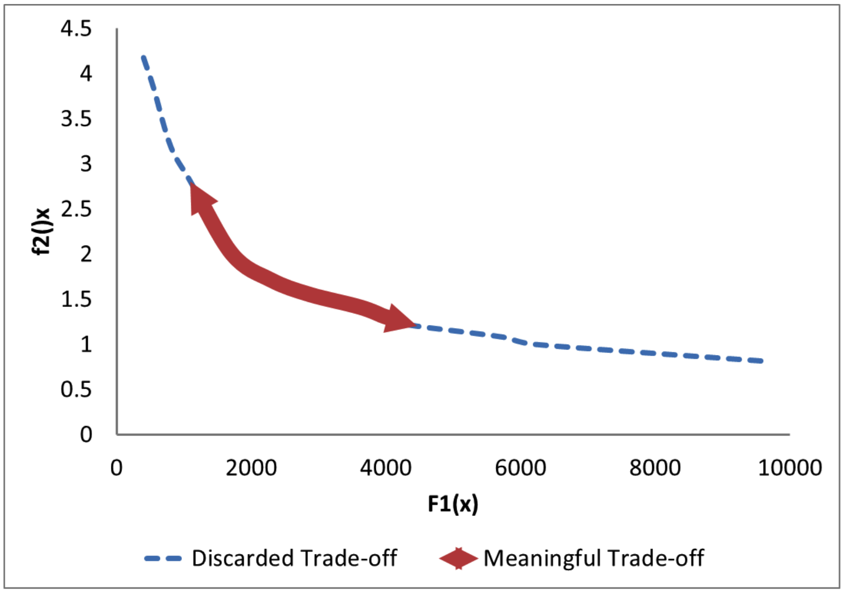


Figure 2.10: The heavy line is an illustration of the region on the Pareto Front that has meaningful tradeoff. In this case, there are two objectives, relative bias and relative variability. Acceptability thresholds are defined according to user preference and solutions in the trade-off set which are within acceptability thresholds, are called meaningful.

Figure 2.11 provides the uncovered hypervolume progress graph in terms of producing meaningful trade-off solutions of the four algorithms on a limited evaluation budget of 200 and 500, respectively. The progress graph shows that GOMORS outperforms all other algorithms with 200 function evaluations and then followed by AMALGAM and BORG. The advantage of GOMORS is more obvious than the time analysis in section 2.5.1 evident by GOMORS's stochastic dominance of all the other algorithms for all number of function evaluations. After 200 iterations, the differences between among GOMORS, AMALGAM and BORG become smaller as indicated by Figure2.11d) and f).

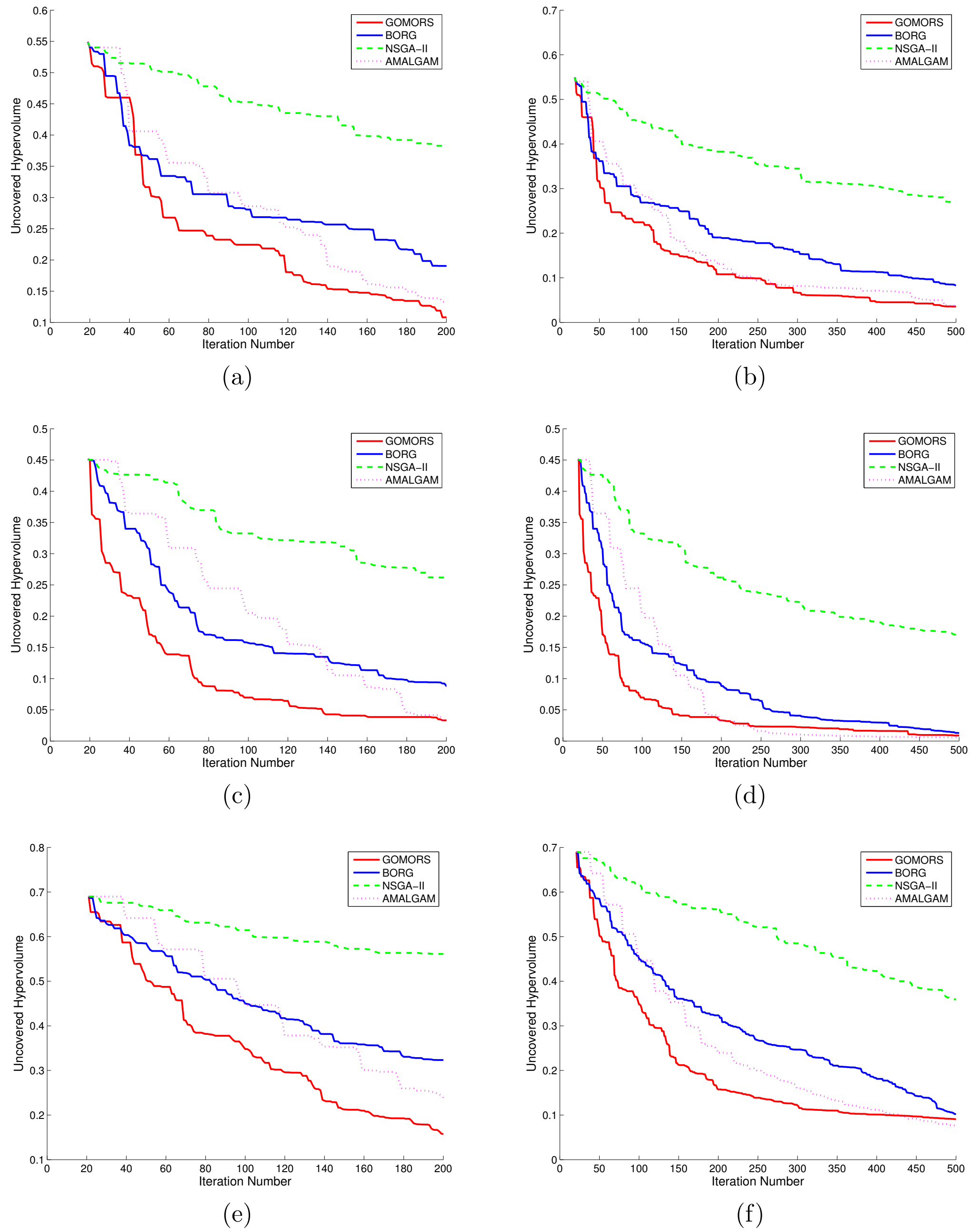


Figure 2.11: Meaningful Trade-offs analysis: Progress graphs with plots of average values of uncovered Hypervolume metric against number of function evaluations, averaged over 10 trials with meaningful constraints: (a-b) Problem UMA-1 after 200 and 500 function evaluations; (c-d) Problem UMA-2 after 200 and 500 function evaluations; (e-f) Problem UMA-3 after 200 and 500 function evaluations;

The Rank Sum Test result of meaningful trade-off analysis is provided in Table 2.2. Similar to the results from the earlier hypothesis testing, GOMORS frequently outperforms

AMALGAM and BORG, whereas none of the other algorithms outperforms GOMORS on any case study.

Table 2.2: Summary of statistical comparison of meaningful trade-offs analysis via Mann-Whitney Rank Sum Test applied to GOMORS, AMALGAM, BORG, according to hypervolume metric after 50, 100, and 200 function evaluations. * after the p-value denotes that GOMORS is better than the algorithm in the row with significance level at $\alpha = 5\%$.

Problem	Algorithm	50 evals	100 evals	200 evals
Case 1	AMALGAM	0.1859	0.1620	0.3847
	BORG	0.7337	0.1859	0.0376*
Case 2	AMALGAM	< 0.001*	0.0036*	0.4274
	BORG	0.0376*	0.0211*	0.0173*
Case 3	AMALGAM	0.1405	0.0487*	0.0307*
	BORG	0.4727	0.0472*	0.0046*

2.6 Conclusions

This study provides a brief introduction to various popular and state-of-art multi-objective algorithms, including MOEAs and surrogate assisted search methods, and investigates their relative effectiveness in groundwater contaminant cleanup management problem, within a limited budget of simulation evaluations. The effectiveness of all algorithms is evaluated in terms of their ability to reliably produce trade-off solutions with good convergence and diversification capabilities.

We employed three difference performance metrics, hypervolume, IGD and GD, to assess the effectiveness of four multi-objective optimization algorithms. Our initial time analysis clearly shows that GOMORS, AMALGAM and BORG are significantly more efficient than NSGA-II. Furthermore, statistical testing analysis is conducted, which shows that GOMORS is not outperformed by any algorithm, within an evaluation budget of 200, after application to 3 groundwater management test cases.

A box plot comparison is also conducted to show the effectiveness and consistency in producing good solutions over multiple trial runs. The result indicates the superiority of GOMORS, AMALGAM and BORG in all the cases with 100 and 200 function evaluations. A further analysis, with more function evaluation, supports that findings of the initial time analysis and indicates the advantages of GOMORS with fewer function evaluations. The meaningful trade-off solution analysis shows that GOMORS outperforms AMALGAM, BORG and NSGA-II with a computational budget of 200 and has more advantages with fewer function evaluations.

Given the computational burden associated with large distributed groundwater numerical models, computationally efficient algorithms can be employed to produce good and meaningful solutions within a very limited simulation evaluation budget. GOMORS , AMALGAM and BORG are a very promising algorithms, which can be used more frequently to efficiently analyze groundwater flow and transport numerical models. While all three algorithms are very promising, GOMORS is more efficient with even more limited computation budget. Moreover, GOMORS can be easily modified to incorporate modest parallelization, and hence can be more efficient in producing good trade-off solutions.

Chapter 3

Groundwater Remediation Long Term Optimal Policy Design by a Surrogate Model Based Mixed-integer Global Optimization Algorithm

3.1 Introduction

This study applies a new surrogate model based nonlinear mixed-integer optimization algorithm to solve fixed cost problems i.e. a problem with dual discrete and continuous value decision variables. These problems generally arise when the objective is to minimize the sum of installation cost and the operation-maintenance cost. The installation cost is a fixed cost that can be represented in the objective function by $c_j I_j$, where c_j is the fixed cost at well j and I_j is a binary integer variable that is 1 if and only if a facility is installed at location j. Thus the first objective is to choose the best possible location (i.e.pick the optimal values of the I_j) out of all feasible options and then to choose the respective rates such that the operation-maintenance cost is minimized over time. These problems can be categorized under Nonlinear Mixed Integer problems (NL-MIP). Nonlinear mixed-integer optimization problems are in general difficult to solve due to their large search spaces. For example in fixed cost problems, both integer and continuous decision variables need to be determined. When decision variable has high dimension and multiple local minima are involved, solving the NL-MIP problems becomes more challenging.

The design of groundwater optimal remediation system is an example of complicated NL-MIP problem. There are generally two types of decision variables that need to be considered in a typical practical groundwater remediation design problem. First type is the integer variable that is which of the available locations for a groundwater are to be used. Second type of decision variable is the continuous variable i.e. (pumping or injection rates for the selected wells). The first type of variable decides the installation cost whereas the second type decides the operation and maintenance cost. Once the integer variable configuration is fixed, the installation cost is no longer a variable, then the objective function is to minimize the variable cost for that configuration. Such types of multimodal NL-MIP problems in water resources management are very difficult to optimize not only due to the large search domain, but also because of the computational complexity of the numerical groundwater models, which poses a huge challenge to the multimodal NL-MIP problem. Groundwater optimal remediation design is a computationally expensive simulation optimization problem, since the objective functions are evaluated via simulation and a simulation evaluation for groundwater flow and transport model may take long time. As a result, algorithms that require fewer model evaluations to produce good solutions are desirable.

Most of the earlier research [35, 6, 19, 55, 76]was more focused on integrating mathematical optimization techniques with simulation for important environmental management issues without addressing the issue of decisions both about optimization of pumping well locations and pumping rates. Zheng and Wang [98]developed an integrated approach to solve the mixed-integer remediation design problem using Tabu Search. Yan and Minsker [92] used an adaptive Neural Network Genetic Algorithm to solve a groundwater remediation design problem. Babbar and Minsker [11] proposed multiscale strategies for GAs that evaluate designs on different spatial grids at different stages of the algorithm to save some computation effort. Such approach becomes computationally expensive for large-scale models where one model simulation takes significant time. Therefore, investigating more effective and robust nonlinear mixed-integer optimization methods becomes the goal and is necessary if the models are highly computationally expensive.

A global mixed-integer optimization algorithm SO-MI [59] is introduced in this study. SO-MI is a surrogate model based algorithm for computationally expensive mixed-integer black-box global optimization problems with both binary and non-binary integer variables. It can be applied to both nonlinear and even multimodal problems. One of the main attractive features of surrogate model based methods is that the methods are "gradient free" i.e. the methods do not need actual gradient information for optimization run. This feature makes this kind of method very suitable for environmental problems as they generally have black box formulation. None of the response surface methods (including the method used) needs any additional information other than function values for minimization.

The SO-MI algorithm is tested for an optimal pumping strategy for groundwater remediation on a hypothetical case and a real groundwater case derived from the Umatilla Chemical Depot, which is an EPA groundwater site. The objective functions for both problems are to minimize the sum of fixed and variable costs subject to some constraints. The fixed cost depends on the integer variable configuration, whereas the variable cost depends on the continuous variables. The integer variables are essentially binary variables and the continuous variables are bounded with upper and lower limit. The results from the SO-MI algorithm are compared to other two widely used mixed-integer optimization algorithms, GA (Genetic Algorithm) and NOMAD (Nonsmooth Optimization by Mesh Adaptive Direct Search).

This paper is organized as follows. Section 3.2 introduces the application problems and optimization formulations. Section 3.3 gives an overview of the algorithms discussed in this study. Experiment setup and results are explained in Section 3.4 and the last Section 3.5

addresses the conclusions.

3.2 Mixed-integer Optimization Problems and Study Cases

3.2.1 Overview of Mixed-integer Optimization Problem

The optimization problems considered in this study are global or nonlinear mixed integer optimizations applied in groundwater management. This type of optimization problems can be formulated as

$$\underset{\boldsymbol{x},\boldsymbol{i}}{\text{Minimize}} \qquad F(\boldsymbol{x},\boldsymbol{i}) = \boldsymbol{c}^T\boldsymbol{i} + f(\boldsymbol{x}) \tag{3.1}$$

subject to

$$\boldsymbol{h}(\boldsymbol{x}) = 0$$

$$\boldsymbol{g}(\boldsymbol{x}) \leq 0$$

$$\boldsymbol{x} \in \mathbf{X} = \left\{\boldsymbol{x} : \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{x}_l \leq \boldsymbol{x} \leq \boldsymbol{x}_u, \boldsymbol{x} \leq \boldsymbol{x}^T \cdot \boldsymbol{i}\right\}$$

where $\boldsymbol{x}$ is the vector with n dimension of continuous decision variables representing the pumping rates and $\boldsymbol{i}$ denotes the m dimension vector of 0-1 binary variables, representing the potential existence of wells at various locations in an aquifer or a treatment system. The objective function is usually the cost function, which consists of both fixed costs $\boldsymbol{c}^T\boldsymbol{i}$ and variable cost $f(\boldsymbol{x})$, where $\boldsymbol{c}$ is a vector of constants associated with fixed costs. Equality constraints are represented by the vector equation $\boldsymbol{h}(\boldsymbol{x}) = 0$ whereas inequality constraints, conditions such as capacity constraints, are represented by the vector inequality $\boldsymbol{g}(\boldsymbol{x}) \leq 0$. The continuous decision variables $\boldsymbol{x}$ is bounded by a lower bound vector $\boldsymbol{x}_l$ and a upper bound vector $\boldsymbol{x}_u$. In addition, continuous decision vector $\boldsymbol{x}$ is subject to the constraint that

$\boldsymbol{x} \leq \boldsymbol{x}^T \cdot \boldsymbol{i}$ to ensure no variable cost associated with locations where wells are not installed.

3.2.2 Model Description

The new algorithm SO-MI is tested and compared with other popular mixed-integer algorithms on two groundwater management problems: one hypothetical groundwater problem and one real demonstration site. The demonstration site, Umatilla Chemical Depot, is adapted from NAVFAC (Naval Facilities Engineering Command) technical report TR-2237-ENV. The objective function both of the two groundwater problems is the sum of fixed and variable cost for the contaminant cleanup as described in Section 3.2.1. The fixed costs are associated with the integer variable configuration i.e. which locations will have wells installed, and the variable costs depend on the continuous decision variables, i.e. the pumping or rejection rates of the chosen facilities. A 'fixed cost constraint' ensures that a continuous variable exists only if the well is chosen to be constructed. More details of the problem formulation for the two test groundwater problems are discussed in this section.

3.2.2.1 Hypothetical Case

The objective function for the hypothetical groundwater problem is to minimize the sum of fixed and variable cost such that the contaminant concentration constraint is satisfied at the end of management period. This hypothetical problem has 32 binary decision variables representing the potential 32 locations for facility construction and 32 continuous decision variables representing variable costs associated with the corresponding facilities in the 32 locations. This hypothetical problem is formulated as

$$\underset{\boldsymbol{x},\boldsymbol{I}}{\text{Minimize}} \qquad \sum_{j=1}^{p} I_j F_j + VC(\boldsymbol{x}) + penalty(C) \tag{3.2}$$

subject to

$$\boldsymbol{x}_{min} \leq \boldsymbol{x} \leq \boldsymbol{x}_{max}$$

$$\boldsymbol{x} \leq \boldsymbol{I} \cdot \boldsymbol{x}_{max}$$

$$C \leq C_{max}$$

where, p is the number of integer variables; $\boldsymbol{x}$ is the continuous decision vector; I is the vector with elements I_j, the binary variable associated with j^{th} well; F_j is the fixed cost associated with j^{th}well; $VC(\boldsymbol{x})$ is the variable cost; C is the maximum concentration at the end of management period measured at observation wells; $penalty(C)$ represents a penalty cost is incorporated to the objective function if the concentration constraint is violated.

3.2.2.2 Umatilla Chemical Depot Study Case

Umatilla Chemical Depot is a 19728 acre military reservation established in 1941 for storage and handling of munitions. From the 1950s until 1960s the depot was operated as an onsite explosives washout plant to remove and recover explosives with a pressurized hot water system. During this time, about 85 million gallons of wash water was discharged into two unlined lagoons, from where the wash water infiltrated into the soil system.

The two major chemicals, RDX and TNT are treated as contamination indicator parameters. A pump-and-treat system was designed by the U.S.Army Corps of Engineers (USACE, 1996 and 2000) to remediate the RDX and TNT plumes(Figure 3.1). The designed pump-and-treat system consists of 3 pumping wells and 2 extraction wells which play role in the cost definitions. The cost of activating the inactive well is considerably less than the cost of installing a new well whereas is no installation cost associated with any other existing wells. This study uses binary integer variables associated with all of the pumping well locations, i.e. '1' if active and '0' if inactive. According to USACE design, the cleanup levels for RDX and TNT are set at 2.1 μg/l and 2.8 μg/l, respectively. In the pump-and-treat system, the contaminated water is pumped from the extracting wells. After being treated by GAC (granulated activated carbon) units, water is discharged into the infiltration basins.

In both hypothetical and Umatilla cases, MODFLOW 2000 and MT3DMS[97] are utilized to simulate groundwater flow and transport systems. The models are used to predict the contaminant concentrations at every model grid point. The formulation only focuses on contaminant transport in the first layer of the model and the boundary conditions for all four sides of the model domain were simulated as constant head. The model inputs are hydrogeological data, domain discretization data and the pumping data, where the pumping/extracting data includes the well locations and the corresponding pumping/extracting rate. After taking the input information, MODFLOW and MT3DMS start to simulate the model, and concentrations of RDX and TNT are calculated as the model outputs which are used to calculate the objective functions.

The optimization algorithm links with both simulation models and the optimization formulations. In this framework, the optimization algorithm prepares inputs or decision variables of the optimization formulations. Then simulation models use the inputs to simulate model outputs, such as hydraulic heads from MODFLOW and contaminant concentrations from MT3DMS, which along with the decision variables contribute to calculate the objective functions. The optimization routine generates a new set of inputs based on the objective function values calculated earlier and passes them to the simulation models again. The whole process runs through iteratively until the maximum number of function evaluations is exceeded.

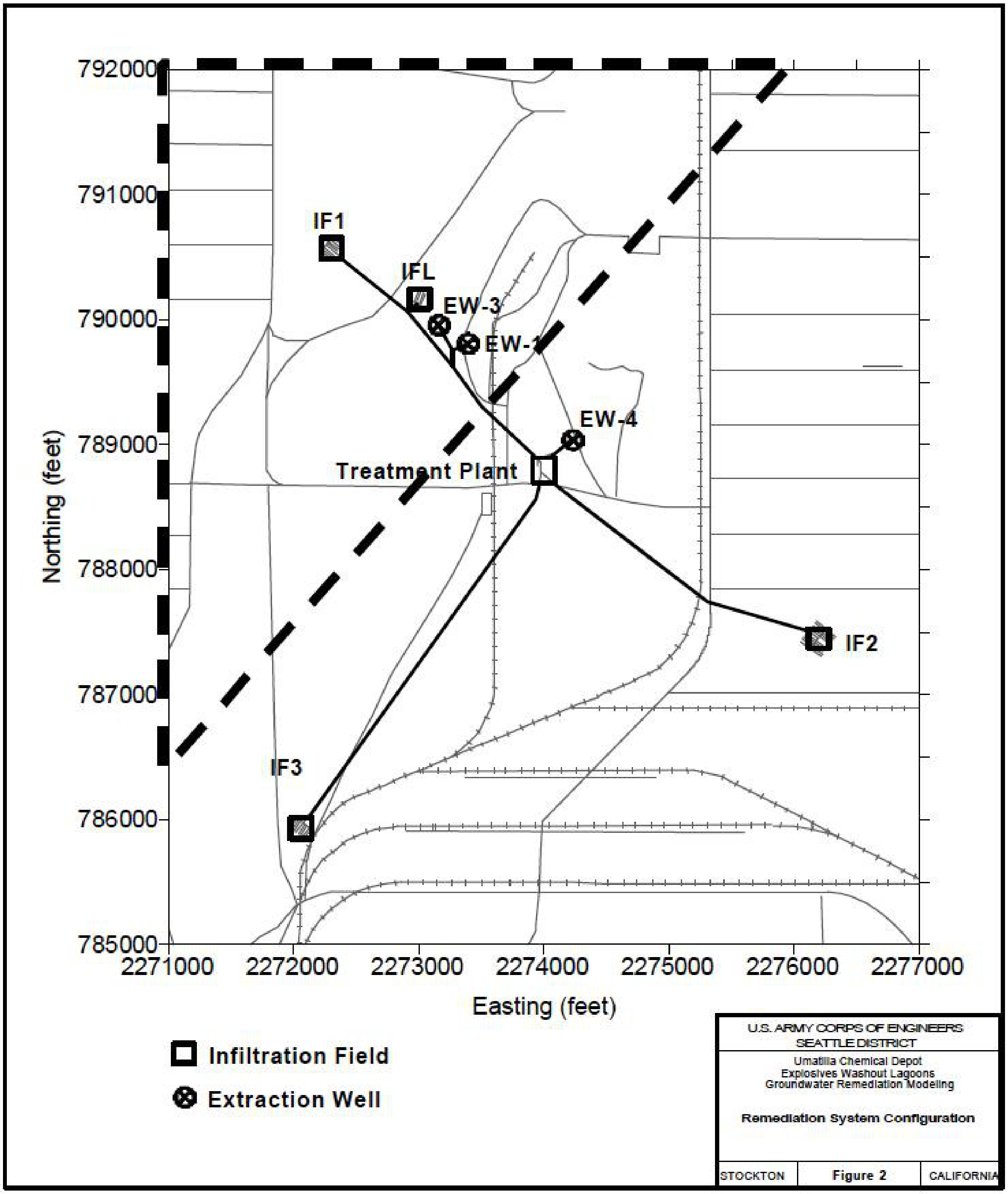


Figure 3.1: Site Map: NAVFAC (Naval Facilities Engineering Command) technical report TR-2237-ENV, 2004 showing location of extraction wells and infiltration fields (recharge basins)

The overall objective for this problem is to obtain the optimal well locations and the corresponding pumping/injection rates of the active wells. Each pumping well has a binary variable (location) with value '1' if it's active well or '0' if it's inactive and respective pumping rates (continuous variables). For this case, since there are two infiltration basins with fixed locations, we have in total 8 binary variables and 10 continuous variables. Similar to the hypothetical case, the objective of this formulation is to minimize the total operations and maintenance costs for the project duration and the objective function is provided as follows:

$$\underset{\boldsymbol{Q},\boldsymbol{I}}{\text{Minimize}} \qquad \sum_{j=1}^{p} I_j F_j + VCE(\boldsymbol{Q}) + VCG(\boldsymbol{Q}) + penalty(\boldsymbol{Q}, C) \tag{3.3}$$

where, p is the total number of wells and I_j is the binary variable associated with j^{th} well; F_j is the fixed cost associated with j^{th}well with \$75,000 for installing a new well and \$25,000 for converting an existing inactive well to services. The decision variable, $\boldsymbol{Q} = [Q_1, Q_2, \ldots, Q_{10}]$ is a vector of pumping rates for 10 wells, among which wells 1-8 are pumping wells and wells 9-10 are recharge basins with the last recharge basin getting a recharge as constraint 3. C is the maximum concentration at the end of management period measured at observation wells. The $penalty(\boldsymbol{Q}, C)$ represents a penalty cost incorporated to the objective function if the concentration or pumping constraints are violated.$VCE(\boldsymbol{Q})$ is the variable electric cost of operation wells, and $VCG(\boldsymbol{Q})$ is the variable cost of GAC units. Note that all the costs are computed in net present value (NPV) with the following discount function $NPV = cost_y/(1 + r)^{y-1}$, where NPV is the net present value of a cost incurred in year y with a discount rate of $r = 5\%$.

The formulation includes the following constraints that must be satisfied while the objective function is minimized:

1. This model consists of 1 management period of 4 years, within which the pumping rates of all the wells are kept throughout this period.

2. Cleanup must be achieved at the end of 4 years, meaning the maximum concentration

of RDX and TNT in model layer 1 must be less than their respective cleanup targets by the end of 4 years: $C_{RDX}^{max} \leq 2.1ppb$ and $C_{TNT}^{max} \leq 2.8ppb$.

3. The total pumping rate, after adjustment for the average amount of system uptime, cannot exceed 1300 gpm. Therefore, the current maximum capacity of the treatment plant $Q_{total} \leq 1300\alpha$ gpm, where α is the coefficient of the average amount of system time ($\alpha = 0.9$ in this study). The total pumping constraint is implemented by using the penalty function thus the solutions not satisfying this constraint are penalized to force the algorithm to look for solution points satisfying the constraint.

4. The pumping capacity of individual wells must not exceed 400 gpm in the less permeable portion of the aquifer and 1000 gpm in permeable portion. This constraint is implemented as the lower and upper bounds of the decision variables.

5. The total amount of pumping must equal to the total amount of injection through the infiltration basins within an error tolerance. Thus, the pumping rate of the third recharge basin is set to be the balance of (Total pumping)-(Total recharge).

6. Binary constraint: I_j is 0 or 1.

7. Well pumping constraint: $\boldsymbol{Q} \leq \boldsymbol{I} \cdot \boldsymbol{Q}_{max}$.

3.3 Optimization Algorithms

3.3.1 SO-MI

The goal of this study is to introduce a novel mixed integer global optimization algorithm SO-MI with applications to the two groundwater study cases and access the performance of this new algorithm as well as two other widely used algorithms. SO-MI, a surrogate model based algorithm designed for computationally expensive nonlinear mixed-integer black-box optimization problems, is developed by Müller et al. [59]. The key feature of this algorithm is that it utilizes surrogate models to guide the search and thus saves the expensive function evaluation budget, and is able to find accurate solutions with relatively few function evaluations. A full description of the inner working of SO-MI can be found in Müller et al.[59], where SO-MI is applied to some computationally expensive mixed integer optimization test problems and is approved to be more efficient than other algorithms. SO-MI has never been used in any water resources management problems, including expensive groundwater model analysis.

The specific steps of the surrogate model based algorithm SO-MI for computationally expensive mixed-integer problems described in [59] are summarized below:

Step 1: Use Latin hypercube design to initially generate $2k+1$ points, where k is the dimension of decision variables. Round the discrete variables to the closest integers, and add one known feasible point to the design such that the total initial points are $n_0 = 2(k+1)$.

Step 2: Do the costly function evaluations of the initial points to obtain $y_t = f(z_t)$, and constraints $c_j(z_t)$, where z_t denotes the initial points, $t = 1, \ldots, n_0, j = 1, \ldots, m$. Find the best feasible point with lowest function value and the worst feasible point with the highest function value.

Step 3: Compute the adjusted objective function values according to

$$f_p(z) = \begin{cases} f_{max} + c_p v(z) & \text{if z is not feasible} \\ f_z & \text{otherwise} \end{cases}$$

where, c_p denotes the penalty factor, f_{max} is the worst feasible function value found so far, and $v(z)$ is the constraint violation function.

Step 4: Calculate the surrogate Radial Basis Function (RBF) model parameters by using of the data points.

Step 5:

(a) Create four groups of candidate points by randomly (i) perturbing only continuous variable values of z_{min}, (ii) perturbing only the discrete variable values of z_{min}, (iii) perturbing both continuous and discrete variable values of z_{min}, (iv) uniformly sampling points from decision space.

(b) Calculate the scoring criteria for every candidate point. Note that this step is inexpensive.

(c) Choose the points with the best score and do the expensive function evaluations at these points (in parallel)

(d) Update the adjusted objective function values and RBF model parameters. Iterate until the maximum iteration number is achieved.

Step 6: Return the best feasible solution.

The features of SO-MI that are different from Stochastic RBF, a method developed by Regis and Shoemaker [71, 72], are summarized as follows: (1) A radial basis function surrogate model is used to select candidate points for both integer and continuous decision variable points at which the computationally expensive objective and constraint functions are to be evaluated, whereas Stochastic RBF can only be used for continuous decision variables. (2) In every iteration multiple new points are selected based on different methods to increase the likelihood of finding good new solutions, and the function evaluations are done in parallel. (3) The adjusted objective function is used to calculate the surrogate model to make the algorithm model efficient.

3.3.2 Genetic Algorithm

Genetic Algorithm (GA), one of the most widely used evolutionary optimization algorithms, is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution [34]. The algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm randomly selects individuals from the current population and uses them as parents to produce the children for the next generation. Over successive generations, the population converge toward an optimal solution. GA involves several operators i.e. crossover, mutation, elitism. Reed et.

al. [67] discusses the importance of setting appropriate GA operators and initial population size to achieve a better performance in water resources problems. GA can be applied to solve problems that are not well suited for standard optimization algorithms, including problems in which the objective function is multimodal, discontinuous, non-differentiable, stochastic, or nonlinear. GAs have been used for water resources problems due to its capabilities of solving global optimization problems without knowing the derivatives. Many previous studies in groundwater remediation design have used GAs to find the optimal design scheme[8, 42, 54, 57, 94]. In this study, Genetic Algorithm solver in Global Optimization Toolbox from MATLAB R2014a is employed to solve the two aforementioned groundwater remediation problems and the results are compared with the other two global mixed-integer optimization algorithms.

3.3.3 NOMAD

Another derivative-free algorithm compared with SO-MI in this study is NOMAD (Nonsmooth Optimization by Mesh Adaptive Direct Search) [4, 2, 5]. NOMAD implements MADS (mesh adaptive direct search) algorithm, which is based on Generalized Pattern Search algorithms. MADS is an extension of generalized pattern search algorithms with superior convergence properties as shown in [3]. These algorithms are iterative methods with each iteration consisting of two phases: a search and a local Poll phase. In the search phase, the objective function is evaluated over a finite number of mesh points to find a new point with better objective function value. It calls the Poll procedure if the algorithm fails to find an improved mesh point, where a barrier objective function (at neighboring mesh points) is evaluated to find a better function value. If Poll also fails to find an improved point, then the mesh is refined and the procedure is repeated.

NOMAD is applicable to mixed-integer global optimization problems. It is an open source algorithm and is encapsulated into the OPTimization Interface (OPTI) toolbox, which is a free MATLAB toolbox for constructing and solving linear, nonlinear, continuous and discrete optimization problems [51]. A range of open source and academic solvers are implemented into this MATLAB optimization solver.

3.4 Numerical Results

3.4.1 Experiment Setup

The goal of this study is to implement the new mixed-integer optimization algorithm SO-MI to identify good solutions for computationally expensive groundwater remediation design problems when computational budget is limited. Thus the experimental runs for the study were designed to test SO-MI against two open source mixed-integer optimization algorithms (Genetic Algorithm and NOMAD) with a fixed number of function evaluations. Branch and Bound method was not used because [59] showed branch and bound did not work at all well on global optimization problems.

Due to the stochastic nature of all algorithms, 20 trial runs were performed for all groundwater case studies and each numbered trial was using the same starting points for that number for all algorithms. SO-MI uses a symmetric Latin hypercube design which consists of $2(k+1)$ points, where the integer constraints are satisfied by rounding the corresponding variable values to the closest integers. To ensure a fair comparison, GA and NOMAD are also using the same Latin hypercube design initial points. Since Genetic Algorithm was obtained from the Global Optimization Toolbox from MATLAB, the algorithm parameters are defined by MATLAB's default, which sets population size of 100. The initial 100 population of GA is obtained from SO-MI's first 100 solution points. The starting point for NOMAD is obtained from the best point of Latin hypercube initial design since NOMAD requires one initial starting point.

This analysis focuses on comparing algorithms' performances, in terms of efficiency and reliability, within a simulation evaluation budget of 1000. Efficiency of all algorithms was compared via plotting progress graphs (average best objective function values among 20 trial runs vs. number of function evaluations)in our discussion. Effectiveness and reliability of all algorithms was compared via box plot comparisons of the objective function values. The next section presents the results for the algorithm comparison with discussion about the algorithm.

3.4.2 Algorithm Comparison

This study compares a surrogate response surface based mixed-integer optimization algorithm SO-MI to two earlier discussed mixed-integer problems. The results for the two groundwater remediation design problem are summarized by the progress graph, which plot values of average objective function values (cleanup costs) against number of function evaluations, to visualize algorithm progress with time. Figure 3.2 and Figure 3.3 depict the progress graphs of the hypothetical groundwater test case and Umatilla groundwater study case. For each figure, the function value (y-axis) is plotted against the specific i^{th}function evaluation (x-axis). This function value is the average over 20 trial runs for the best solution found on or before the i^{th} function evaluation, respectively. In this study, the objective functions for both study cases are the groundwater remediation cleanup costs.

For both hypothetical case (Figure 3.2) and Umatilla case (Figure 3.3), with a fixed number of total allowed function evaluations, the SO-MI algorithm significantly outperforms the other two methods (GA and NOMAD). Since the goal is to minimize the cleanup cost, the superior algorithm should be a curve with lowest average objective function value and fastest drop. As shown in the progress graphs, SO-MI (red solid line) drops the fastest (after 110 function evaluations) and remains the lowest function value for all the number of function evaluations.

In other words, SO-MI has the best efficiency with the whole optimization procedure among the three algorithms. The performances of GA and NOMAD in the hypothetical study case are very close to each other towards the end of 1000 function evaluations and NOMAD is better than GA in the range of 110 to 400 evaluations. Notably even after

1000 evaluations, the GA and NOMAD solutions (which have flattened out) are much worse than the SO-MI solution. For the Umatilla study case, GA performs relatively better than NOMAD as compared to the hypothetical study case. It can also be noticed that the difference between SO-MI and other two algorithm are more significant in Figure 3.3, which is the Umatilla groundwater study case. The Umatilla groundwater case, as discussed in earlier sections, involves two types of chemicals and the model scale itself is larger than the hypothetical groundwater problem. Thus Umatilla case is a more difficult optimization problem than the hypothetical one in terms of its nonlinearity and multi-modal nature. SO-MI demonstrates its advantage of being used in a harder problem than other algorithms as shown in Figure 3.3, whereas GA and NOMAD need significant number of additional function evaluations to locate a relatively good solution.

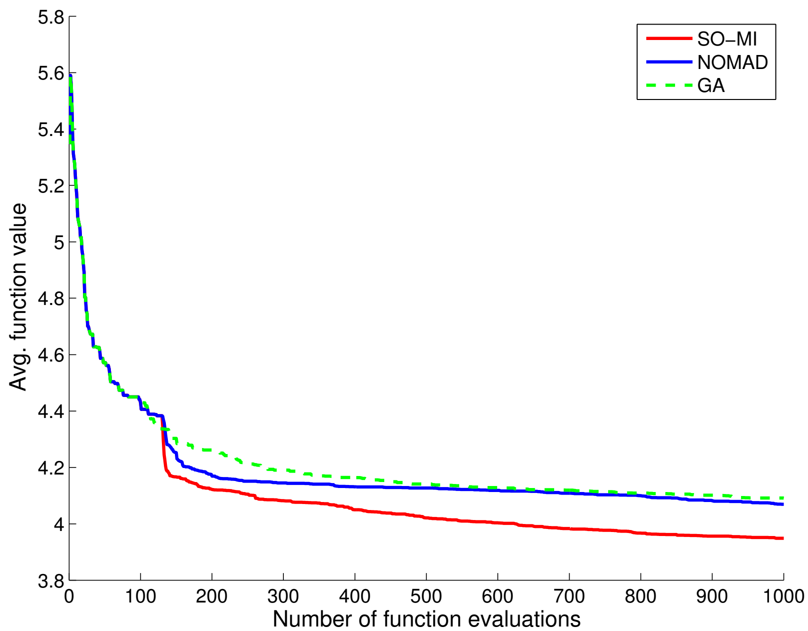


Figure 3.2: Objective function value for feasible points averaged over 20 trials vs. number of function evaluations; for groundwater hypothetical problem

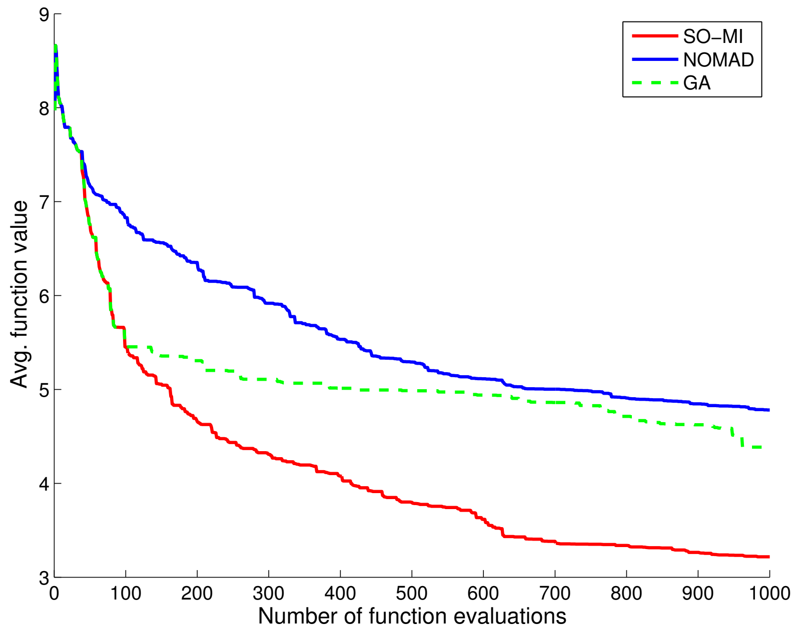


Figure 3.3: Objective function value for feasible points averaged over 20 trials vs. number of function evaluations; for Umatilla groundwater problem

Table 3.1 shows the mean and standard deviation of objective function value for the best solution at the end of 200, 500 and 1000 function evaluations of all three algorithms for both Hypothetical and Umatilla Cases, where the best results of the mean values are marked by bold. The results indicate that SO-MI has the lowest mean among all three algorithms with different number of function evaluations in both cases (marked bold). The standard deviation results show that SO-MI gets the smallest standard deviation as the number of function evaluation increasing. To analyze the reliability of the algorithms, box plot comparisons are employed to show the effectiveness and consistency in producing a good solution, and robust over multiple trial runs. The box plot comparisons are based on the average best function evaluation at the end of 1000 function evaluations. In both study cases, SO-MI has the smallest median and spread among as the comparison to GA and NOMAD. NOMAD has the largest spread in the hypothetical groundwater study case depicting its unstable performance with multiple trials, though it has lower median than GA. The spread of NOMAD and GA are not much difference in the Umatilla study case, but GA has slightly lower median than NOMAD. The overall trend of this analysis shows that performance of SO-MI is consistently good in both groundwater study cases for 1000 function evaluations.

Table 3.1: Mean objective function values over 20 trials with 200, 500 and 1000 function evaluations for Hypothetical Case and Umatilla Case. Best result of the mean values among all algorithms is marked by bold.

Problem	Algorithm	Statistics	200 eval.	500 eval.	1000 eval.
Hypothetical Case	SO-MI	Mean	**4.123**	**4.021**	**3.948**
		SD.	0.052	0.035	0.022
	NOMAD	Mean	4.169	4.127	4.069
		SD.	0.072	0.072	0.085
	GA	Mean	4.262	4.141	4.091
		SD.	0.116	0.030	0.030
Umatilla Case	SO-MI	Mean	**4.657**	**3.800**	**3.218**
		SD.	0.916	0.864	0.862
	NOMAD	Mean	6.351	5.292	4.780
		SD.	0.686	0.972	1.127
	GA	Mean	5.306	4.985	4.385
		SD.	0.757	0.691	0.989

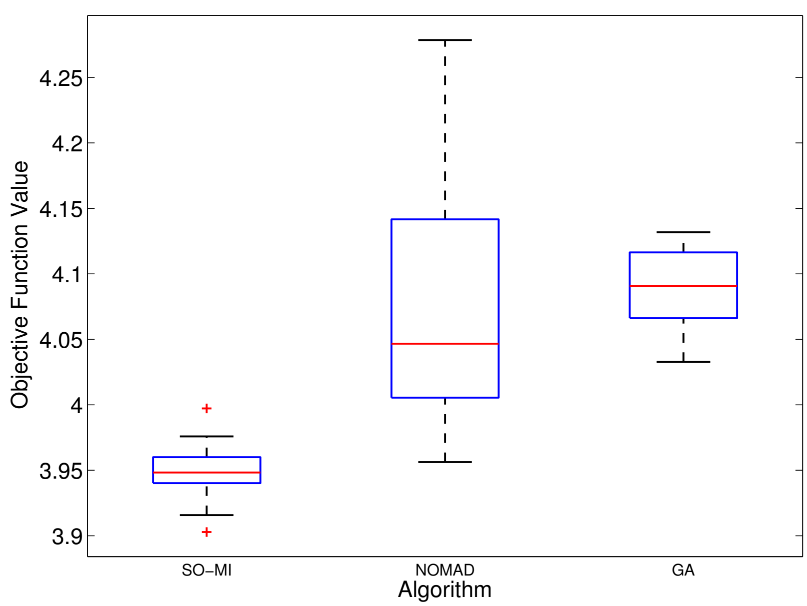


Figure 3.4: Box plot algorithm comparison after 1000 function evaluations (20 trials); for groundwater hypothetical problem

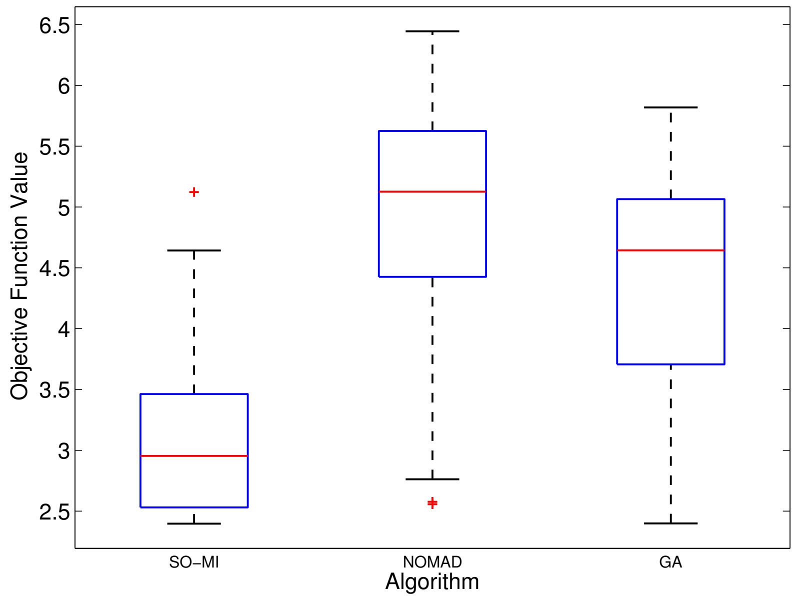


Figure 3.5: Box plot algorithm comparison after 1000 function evaluations (20 trials); for Umatilla groundwater problem

The hypothesis testing for differences in means between the different algorithms in Table 3.2 indicates that for both Hypothetical and Umatilla Cases SO-MI produces significantly lower mean at significance level $\alpha = 5\%$ or even $\alpha = 1\%$ from all the other tested algorithms. Thus, this result further supports the evidence of SO-MI's superior algorithm performance. Statistical test results for comparison of NOMAD and GA fails to indicate any significant difference at 5% significance level between the results for NOMAD and GA.

The comparisons in the previous section indicate clearly that SO-MI performs well with a limited computational budget and is much better suited to handle global mixed-integer blackboard optimization problems with multiple local minima.

Table 3.2: Hypothesis testing for differences in means at the final function evaluation of the Hypothetical Case and Umatilla Case (1000th simulation) with significant at the $\alpha = 5\%$. * after the p-value denotes that the algorithm in the row is better than the one in the row with significance level at $\alpha = 5\%$.

Problem	Algorithm	SOMI	NOMAD	GA
	SO-MI	1.000	< 0.001*	< 0.001*
Hypothetical Case	NOMAD		1.000	0.2457
	GA			1.000
	SO-MI	1.000	< 0.001*	< 0.001*
Umatilla Case	NOMAD		1.000	0.2838
	GA			1.000

To test the quality of the results obtained from SO-MI for Umatilla groundwater site, the pumping rates and well configuration results are compared to the results presented by Becker et.al [12]. The results presented by Becker et.al for this formulation the optimal solution point after 1000-8000 simulations uses 2 new wells, 2 existing wells and 2 existing basins. Out of 20 trials done using SO-MI 20 obtained the same configuration with slightly different pumping rates as optimal solution point.

3.5 Conclusions

The results of optimal policy design for pump and treat system, which is a problem involving both integer and continuous decision variables, illustrate the success of the new SO-MI algorithm. The study compared the suggested SO-MI method with NOMAD and Genetic Algorithm (GA) for two mixed integer value optimization problems. The algorithm, SO-MI, iteratively evaluates the computationally expensive simulation model and updates a radial basis function surrogate model to reduce the amount of function evaluations to be done to find optimal solution.

The results presented indicate that under limited computational budget, SO-MI method is able to find significantly better solutions than the other algorithms for both the hypothetical groundwater study case and the real groundwater case. The genetic algorithm has the worst performance for both study cases, whereas NOMAD is able to find competitive solutions in some trial runs. These results are based on groundwater pump and treat system problems do not prove that SO-MI will always be better than other algorithms in mixed-integer optimization problems. However SO-MI was tested in many other problems in [59] and was proved to be the best algorithm among many widely used mixed-integer algorithms. As a result, this study suggests that SO-MI should be considered as alternatives to widely used methods such as evolutionary algorithm, especially with applications to groundwater remediation designs.

In conclusion, the introduced algorithm SO-MI extends the research area of using surrogate models for solving mixed-integer optimization problems. The computational results indicate that SO-MI is a promising algorithm and performs significantly better than commonly used algorithms for groundwater remediation design problems (mixed-integer problem).

Chapter 4

Implementation of DYCORS-PEST Algorithm for Computationally Expensive Groundwater Calibration Problems

4.1 Introduction

Groundwater numerical models have been widely used as effective tools to analyze and manage water resources. Numerous mathematical models have been developed to solve groundwater problems [74, 89, 93, 37]. Generally speaking, physically based mathematical models are solved by finite-difference or finite-element methods and most of these models are distributed parameter models. The parameters used to characterize the groundwater numerical models are not directly measurable and have to be determined through parameter estimation processes or model calibration. Thus, parameter calibrations in groundwater models are essential for successful modeling and the inaccuracy of parameter estimation may cause unreliable model output for future predictions or management purposes. The problem of parameter identification has been studied extensively during the past decades, and numerous approaches have been developed for solving this problem. Groundwater inverse methods have been reviewed by Yeh [93], Kuiper [50], Ginn and Cushman [33], Sun [80], Sun et al. [81], McLaughlin and Townley [56], Hyun and Lee [43], Carrera et al. [15], Hill and Tiedeman [40] and Hill et al. [41].

To solve the parameter estimation problem, various techniques have been developed. Manual calibration, which is also named trial-and-error method, has been popular and is a frequently used approach for model calibration in the old days. The whole processes require a large amount of human time as well as perception of the model. Thus, the manual calibration is very tedious and time-consuming. To reduce computation time and human effort, automatic calibration that involves the use of an optimization method to search for the parameter set subject to a specified goodness-of-fit function were developed. Compared to manual calibration, automatic calibration requires much less human time and has higher possibility finding a better parameter set since more model simulations could be performed without worrying about saving human effort. To solve the automatic calibration problem, Gradient-based optimization methods, such as Gauss-Newton, gradient steepest descent, conjugate gradient, quasi-Newton, truncated-Newton, and Levenberg-Marquardt methods, have been widely used in groundwater model calibration. Previous researches have demonstrated the performance of applying gradient-based algorithms to groundwater calibration problems [52, 9, 93, 16, 61, 39, 41, 45]. The variations of the Gauss-Newton optimization approach were written into solution codes for applying into groundwater inverse problems, such as UCODE [**?**, 40], iTough2 [31] , PEST [23]. The model independent Levenberg-Marquardt (LM) method based parameter estimation software PEST, which quantifies model-to-measurement misfit in the weighted least squares sense, has been widely used for environmental numerical model calibration. This software is efficient in terms of its model run requirements. The major advantage of using these gradient-based algorithms is that these local optimization methods are computationally efficient at searching for local minima of the non-linear objective functions. However, groundwater inverse problems are typically highly nonlinear with multiple local minima. Thus, these gradient-based methods can be easily trapped into local optimum and cannot necessarily find the global optimum solution.

To solve this problem, many classic heuristic optimization methods are developed and applied to water resources problems, i.e. Genetic Algorithm [34, 32], Artificial Neural Networks [46, 47], tabu search [96], simulated annealing [96], and ant colony optimization (ACO) [1] etc. In addition, there are some other heuristic methods were developed and implemented to inverse problems. For example, shuffled complex evolution (SCE) algorithm [27], dynamically dimensioned search (DDS) algorithm [28], Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [38]. This algorithm has been encapsulated into the PEST software package [24] as the global optimizer with name CMAES_P. The difficulty with heuristic methods is that they tend to require hundreds of thousands of function evaluations to obtain adequately good solutions. Since simulations in engineering applications can be computationally extremely expensive and time consuming, several surrogate based methods have been proposed to solve such global optimization problems. Examples include Efficient Global Optimization (EGO) method, which is the first surrogate global optimization method [44], UOBYQA [63], which has been applied to watershed model calibration by Shoemaker et al. [77], NEWUOA [62, 64], DFO [18], ORBIT [91], FA-RS [70], which was applied to bioremediation models [58], and Stochastic RBF [72], which uses a weighted score computed based on the objective function value prediction by the response surface and the distance to previously evaluated points to iteratively select sample points. The sampling method has also been employed as a framework in DYCORS (DYnamic COordinate search using Response Surfaces), which is developed by Regis and Shoemaker [73]. DYCORS is a surrogate-based optimization algorithm for high dimensional expensive black-box functions that incorporates an idea from the DDS algorithm. This algorithm is applied to a watershed calibration problem and produces very promising results[73].

In this study, we introduce the new method DYCORS-PEST, which uses the DYCOR, a surrogate global optimization method, in conjunction with PEST, a widely used local search algorithm. This method is developed for high dimensional, computationally expensive, multimodal calibration problems when computation budget is limited. DYCORS-PEST achieves the goal of solving the computationally expensive black-box problem by forming a response surface of the expensive function. This response surface function is used as a surrogate for the expensive function during the search, thus the number of required expensive function evaluations for finding accurate solutions can be significantly reduced. The global search method in DYCORS-PEST perturbs the best point found so far in order to find a new sample point. The number of decision variables being perturbed is dynamically adjusted in each iteration in order to be more effective for higher dimensional problems. The procedure for dynamically changing the dimensions perturbed is drawn from earlier work on the DYCORS algorithm. The novelty in DYCORS-PEST is a memetic approach to improve the accuracy of the solution in which we use a local optimization search around a subset of the previously evaluated points. Since PEST is a very efficient and widely used local search algorithm, incorporating PEST into DYCORS-PEST is a good enhancement for PEST and easy for PEST users to learn.

The new integration methodology is applied to two real groundwater calibration problems, and the results are then compared to Stochastic RBF and PEST, which are studied in [87]. This study is organized as follows. Section 4.2 introduces the details of the new method and the other two algorithms compared to. Section 4.3 introduces and explains the model calibration and application problems with details of objective function and constraints. Experiment setup and results are then discussed in section 4.4. Section 4.5 highlights the conclusions.

4.2 Optimization Algorithm Description

The goal of this study is to develop and demonstrate a global optimization algorithm for relatively high dimensional calibration problems that improves the accuracy of the solution while maintaining high efficiency for computationally expensive problems with limited computation budget. This method integrates a widely used local optimization method (PEST) with a response surface based global optimization method (DYCORS). In the following subsections, the general Stochastic Response Surface (SRS) Framework, Stochastic Radial Basis Function (Stochastic RBF), DYCORS, PEST and details of DYCORS-PEST framework will be discussed. Stochastic RBF [72] is another global optimization algorithm designed for computationally expensive Black-box functions. In the earlier study of the performance of Stochastic RBF with application to groundwater calibration problems, Stochastic RBF shows its superiority with comparison to PEST and other two global optimizers [87]. In this study, we compare the performance of the new method with Stochastic RBF in the same groundwater calibration problems.

4.2.1 Stochastic Response Surface (SRS) Framework

Regis and Shoemaker [72] introduced a class of stochastic response surface algorithms, called SRS (Stochastic Response Surface) which is a framework for expensive global optimization. Examples of methods based on SRS are LMSRBF [72] and DYCORS [73]. The SRS framework is given in Algorithm 4.1. The iteration of any surrogate based optimization consists of performing the function evaluations of the points in the initial experimental design and building an RBF model to approximate an expensive objective function. The algorithm then makes use of this computationally inexpensive response surface to determine the location of the next function evaluation point. Finally, the algorithm updates the surrogate after the new evaluated point is added in.

Algorithm 4.1 Stochastic Response Surface (SRS)

Input: initial experimental design

1: build initial response surface
2: **repeat**
3: generate candidate points around x_{best};
4: select a point for function evaluation;
5: update x_{best}, the best point found so far;
6: update response surface;
7: **until** termination condition is met

Output: the approximate global minimum

4.2.2 Stochastic Radial Basis Function (Stochastic RBF)

Regis and Shoemaker [72] introduced the Stochastic Radial Basis Function (Stochastic RBF) method based on the SRS Framework. Two new global optimization methods were developed: Global Metric Stochastic Radial Basis Function (Global MSRBF) and Multistart Local Stochastic Radial Basis Function (Multistart Local MSRBF). In this paper, we used Multistart Local MSRBF. The key idea of these two global optimization methods is to utilize radial basis functions (RBF) [65, 66, 14] as the response surface model to approximate the expensive objective function and thus reduce the number of function evaluations needed. Global MSRBF and Multistart Local MSRBF are both designed for continuous, multimodel and computationally expensive functions, especially if no derivative information is inexpensively available.

With the idea from SRS Framework, Stochastic RBF uses radial basis functions (RBF) to approximate the objective functions that are expensive to compute. The RBF model can then be represented as

$$s(x) = \sum_{i=1}^{n} \lambda_i \phi(\|x - x_i\|) + p(x), \ x \in \mathbb{R}^d \tag{4.1}$$

where $s(x)$ is the surrogate model prediction at the point x, $\|\cdot\|$ is the Euclidean norm, and $p(x)$ is a polynomial tail. The order of the polynomial tail depends on the chosen RBF type. for a cubic RBF $\phi(r) = r^3$ with a linear polynomial $p(x) = \boldsymbol{\alpha}^T x + \alpha_0$, where $\boldsymbol{\alpha} = (\alpha_{1,\ldots,}\alpha_d)^T \in \mathbb{R}^d$, Eq. 4.1 can be simplified to the following form:

$$s(x) = \sum_{i=1}^{n} \lambda_i \phi(\|x - x_i\|) + \boldsymbol{\alpha}^T x + \alpha_0,\ x \in \mathbb{R}^d \tag{4.2}$$

4.2.3 DYCORS

Dynamic Coordinate Search using Response Surface Models (DYCORS) is one of the algorithms from the family of surrogate-based optimization algorithm, and it is developed by Regis and Shoemaker [73]. While most of the previous surrogate-based optimization methods are suitable and have only been tested on problems with low dimension, DYCORS, on the other hand, was developed especially for problems in a class of HEB (High dimensional Expensive Black-box) problems. DYCORS incorporates an idea from the DDS [83], where it does not perturb all variables of the best point found so far in order to create candidate points, but rather each variable is perturbed with probability

$$P(n) = p_0 \left[1 - \frac{\log(n - m + 1)}{\log(N_{max} - m)} \right], \tag{4.3}$$

for all $m \leq n \leq N_{max}$, and where m is the number of point in the initial experimental design, $p_0 = \min(1, 20/d)$, n is the iteration number, and N_{max} is the maximum number of allowed evaluations for the optimization. As a result, the probability of perturbation for each variable decreases as n grows.

DYCORS was tested on problems up to 200 dimensions and a watershed calibration problem[73], where DYCORS had superior performances. Since the problems of model calibration involve parameter sets with high dimension, DYCORS is integrated into this new DYCORS-PEST framework.

4.2.4 PEST

In this study, The most widely used method for groundwater calibration is PEST [23]. The core of PEST is the Gauss-Marquardt-Levenberg (GML) [53] algorithm, which is a derivative-based local optimization method. Doherty and co-workers have made series of improvements to PEST [22, 25, 24] and the authors have offered many training sessions, which have contributed to the propagation of its use. Many features are added to PEST, like the Automatic User Intervention (including truncated singular value decomposition method) that instructs the algorithm to not perturb some most insensitive parameters for a certain number of iterations. PEST is very efficient to find the local optimal when the objective function is convex. PEST requires a significant amount of user input for the algorithm parameters such as maximum increment for derivative computation, step size, stopping criterion, etc.. However, it can be applied to many existing simulation models without accessing to models' source code, thus allowing simple calibration setup with an arbitrary model. Although the algorithm PEST will stop when a local optimum is found, the user can manually restart the algorithm at different starting points.

Gradient-based methods can be easily trapped into local optimum and cannot necessarily find the global optimum solution. To avoid the drawbacks of PEST, Stochastic RBF [72], a global optimization algorithm, is combined with PEST to solve computationally expensive groundwater calibration problems [87]. However, when the problem is highly multimodal, the hybrid method is still not sufficiently efficient to find the global optimum. In this study, we propose the new framework to enhance the local optimizer PEST by integrating PEST into DYCORS [73].

4.2.5 DYCORS-PEST

DYCORS-PEST involves a memetic search, which is a combination of global and local search methods. It has two main phases:

1. high dimensional global optimization search with DYCORS.

2. gradient-based local optimization with PEST

Phase 1 utilizes DYCORS algorithm because it has been shown to be able to efficiently find good solutions for high-dimensional, computationally expensive multi-modal black-box problems when computation budget is limited. As we discussed in Subsection 4.2.3, DYCORS differ from earlier algorithms in that the number of perturbed dimensions is random and the expected number of perturbation decreases with iteration. This dynamical dimensioned search feature has been shown to be efficient for higher dimensional problems [83]. The DYCORS-PEST framwork is shown in Algorithm 4.2 and the details of the Phase 1 is given in Algorithm 4.3 and works as DYCORS.

Algorithm 4.2 DYCORS-PEST

Input:

I-1 A real-valued black-box function f defined on a hypercube $\mathcal{D} = [a, b] \subseteq \mathbb{R}^d$.

I-2 N_{max}: The maximum number of allowed function evaluations.

I-3 A response surface model

I-4 A set of initial evaluation points $S_0 = \{x_1, \ldots, x_{n_0}\}$determined by a randomly generated Latin Hypercube design.

I-5 N_{cand}: The number of candidate points randomly generated in each iteration.

I-6 A function $\varphi(n)$ defined for all positive integers $n_0 \leq n \leq N_{max} - 1$ whose values are in $[0, 1]$.

I-7 The tolerance for the number of consecutive failed iterations τ_{fail} and the threshold for the number of consecutive successful iterations $\tau_{success}$.

I-8 The initial step size σ_{init} and the minimum step size σ_{min}.

I-9 T_{max}: The maximum number of times the step size can be reduced before starting the local search.

1: Initialization. Set $\sigma_0 = \sigma_{int}$, $C_{fail} = 0$, $C_{success} = 0$, $T_{shrink} = 0$.
2: Initial point evaluation and initial surrogate. Set $n = n_0$, $f_{best} = f(x_{best})$, where x_{best} is the best point found so far. Build the initial response surface $s_0(x)$ based on the initial n_0 evaluated points.
3: while $n < N_{max}$ **do**
4: **Global search phase**;
5: **Local search phase**; The best point found by Global search phase is denoted by x^G_{best}. Use PEST starting from x^G_{best} on the true objective function to further improve the solution
6: end while

Output: Best solution found so far: x_{best} and f_{best}.

Algorithm 4.3 Global Search Phase

1: while Global termination condition is not satisfied **do**

2: Compute the probability of perturbing a coordinate: $p_{select} = \varphi(n)$.

3: Generate N_{cand} candidate points around x_{best}: generate $\Omega_n = \{y_{n,1}, \ldots, y_{n,N_{cand}}\}$by perturbing each variable with probability p_{select}. The perturbation magnitude is an $N(0, \sigma_n)$ random variable.

4: Select next iterate x_{n+1}through the function **Select_evaluation_point(**Ω_n**,**s_n**(x),**S_n**)**.

5: Compute the expensive function value $f(x_{n+1})$.

6: Update the best solution: update the set of evaluated points $S_{n+1} = S_n \cup \{x_{n+1}\}$; update x_{best}, f_{best} and response surface.

7: Update counters: if $f(x_{n+1}) < f_{best}$, reset $C_{success} = C_{success} + 1$ and $C_{fail} = 0$; otherwise reset $C_{fail} = C_{fail} + 1$ and $C_{success} = 0$.

8: Adjust step size: $[\sigma_{n+1}, C_{success}, C_{fail}] =$ **Adjust_Step_Size**$(\sigma_n, C_{success}, \tau_{success}, C_{fail}, \tau_{fail})$. $T_{shrink} = T_{shrink} + 1$ if $\sigma_{n+1} < \sigma_n$.

9: Reset $n = n + 1$.

10: end while

The global stopping criterion which is either the maximum number of function evaluations (in which case Algorithm 4.2 stops after Step 4), or $T_{shrink} = T_{max}$(in which case Algorithm4.3 Step 8). The global search phase in Algorithm 4.3 shows that after building an initial response surface in Algorithm 4.2 Step 2, candidate points are created by adding a random multivariate perturbation to the current best point (Algorithm 4.3, Step 3). $P_{select} = \varphi(n)$ (Algorithm 4.3, Step 2), which is the probability that any one dimension is perturbed, is computed in order to generate random perturbations. If no variable of x_{best} is selected for perturbation, we select one variable at random. By perturbing the selected dimensions through adding a random variable $N(0, \sigma_n)$, N_{cand} candidate points are generated. Then in the next step (Algorithm 4.3, Step 4), the point with a minimum score which is computed as the weighted sum of the surrogate surface value $s(x_{cand})$ and a metric that is based on the distance between x_{cand} and the set of previously evaluated points S_n is selected for expensive function evaluation. After updating the best solutions, response surface model and counters in Algorithm 4.3 Steps 6 and 7, the variance σ_n (for $N(0, \sigma_n)$) is adjusted in Algorithm 4.3 Step 8 to speed up the convergence. The term T_{shrink} in Step 8 is a measure of the iterations done without improvement and when T_{shrink} reaches the value T_{max}, the global search stops and Phase 2 starts. More details for generating the candidate points, adjusting $\varphi(n)$, **Select_evaluation_point** and **Adjust_Step_Size** should refer to Regis and Shoemaker [73].

In Phase 2, we start a local search with a widely used local search package PEST on the true objective function starting from x^G_{best} which denotes the best solution found after the global phase in an attempt to further improve the solution. As we discussed in Subsection 4.2.4, PEST is a widely used local optimization algorithm especially in the field of water resources. PEST is very efficient to find the local optimal when a good starting point is provided, so continuing the local search with PEST after the global phase ensures a better starting point than using PEST alone. The performance of PEST heavily depends on x^G_{best}. If x_{best} is in the vicinity of the global minimum, PEST converge to and stop at a global optima. We must note that PEST is derivative-based algorithm, so in each iteration at least d function evaluations to numerically compute the derivatives and then it needs one or more additional function evaluations to find the point for starting the next iteration. As a result, we must use a global variable to count expensive function evaluations rather than counting the algorithm's iterations. If there are function evaluation budget left after PEST stops, the algorithm goes back to the global search phase.

4.3 Model Description

4.3.1 The Problem of Model Calibration

We consider calibration of a model of transient groundwater flow in an unconfined, heterogeneous, isotropic aquifer and we assume the groundwater is incompressible with constant density and viscosity. The equations used in this study to describe groundwater flow are

$$\frac{\partial}{\partial x}(hK\frac{\partial h}{\partial x}) + \frac{\partial}{\partial y}(hK\frac{\partial h}{\partial y}) \pm Q = \mu\frac{\partial h}{\partial t}\ x, y \in \Omega, t = 0 \tag{4.4}$$

$$h(x,y,t)|_{t=0} = h(x,y,0)\ x,y \in \mathit{\Omega}, t = 0 \tag{4.5}$$

$$h(x,y,t)|_{\mathit{\Gamma}_1} = h(x,y,t)\ x,y \in \mathit{\Gamma}_1, t \geq 0 \tag{4.6}$$

$$hK\frac{\partial h}{\partial \overrightarrow{n}}|_{\mathit{\Gamma}_2} = q(x,y,t)\ x,y \in \mathit{\Gamma}_2, t \geq 0 \tag{4.7}$$

where t is time (T); x, y are Cartesian coordinates (L); h is the hydraulic head (L), $h(x,y)$; K is the value of hydraulic conductivity (L/T); μ is specific storage coefficient of the aquifer (1/L); Q is the fluid sinks/sources term (1/T); Ω is the model domain; $\mathit{\Gamma}_1$ is Dirichlet boundary; $\mathit{\Gamma}_2$ is Neumann boundary; $\overrightarrow{n}$ denotes the normal to the boundary $\mathit{\Gamma}_2$; $h(x,y,0)$ indicates the initial water table; $h(x,y,t)$ is the water table on Dirichlet boundary at time time t; and $q(x,y,t)$ represents lateral flux of Neumann boundary. Since the direction and rate of groundwater flow is determined by spatial or temporal variations in some hydrologic and hydro-geological parameters (e.g. K and μ), to apply the groundwater flow models, the knowledge of these hydrologic and hydro-geological parameters is required. Therefore, as one of the first steps in modeling study, field measurements of these parameters, such as pumping tests, are essentially point measurements providing an estimation of parameters for the area near the observation wells. However, the data from field measurements can only represent a small part of the study area in many cases because of the limited number of observation wells. Hence estimations of spatially distributed parameters with whole aquifer model rather than using point measurements become necessary. Hence the field measurements are used to establish the ranges of each parameter in each zone and the optimization-driven calibration is used to estimate the parameters in each zone within this range.

Automatic calibration with optimization algorithms is a way to obtain the best values of model input parameters. In most cases, the objective function in the optimization is to minimize an error function which defines the discrepancy between model outputs and the observations. Generally, calibration of parameters can be formulated as a box-constrained minimization problem, where the objective function is a error function and decision variables are the model parameters to be calibrated. are bounded. To compute the objective function for each set of potential parameter vector, one the groundwater numerical models simulation will be performed. As discussed in the introduction, we mainly focus on the cases where the groundwater model is computationally expensive to simulate, which may take from many minutes to many hours to compute just one simulation in serial or parallel. As a result, only very limited number of function evaluations are allowed with a limited computational budget.

4.3.2 Study Cases

The study groundwater aquifer in this paper is called Miyun-Huai-Shun watershed basin, which is one of the major water supply resources of Beijing city. In the following the Miyun-Huai-Shun aquifer will be referred to as Miyun aquifer. As shown in Figure 4.1, Miyun aquifer is located in the northeast of Beijing city, China. The main aquifer area is 456 km2 with three boundaries: eastern, western and south. According to previous studies of these three boundaries, they can be regarded as relatively impervious boundaries. Given that the horizontal dimension of this aquifer ranges from tens of kilometers while the depth varies only from tens to hundreds of meters, the groundwater problem in this aquifer has been simplified as a two-dimensional, unconfined, heterogeneous, isotropic, transient flow system. The model is governed by a partial differential equation as Eq.4.4, and the boundary conditions can be expressed as Eq.4.5-4.7. This groundwater problem is solved by Finite element method (FEM) [90] and has been applied to several engineering projects with different regimes [90, 88].

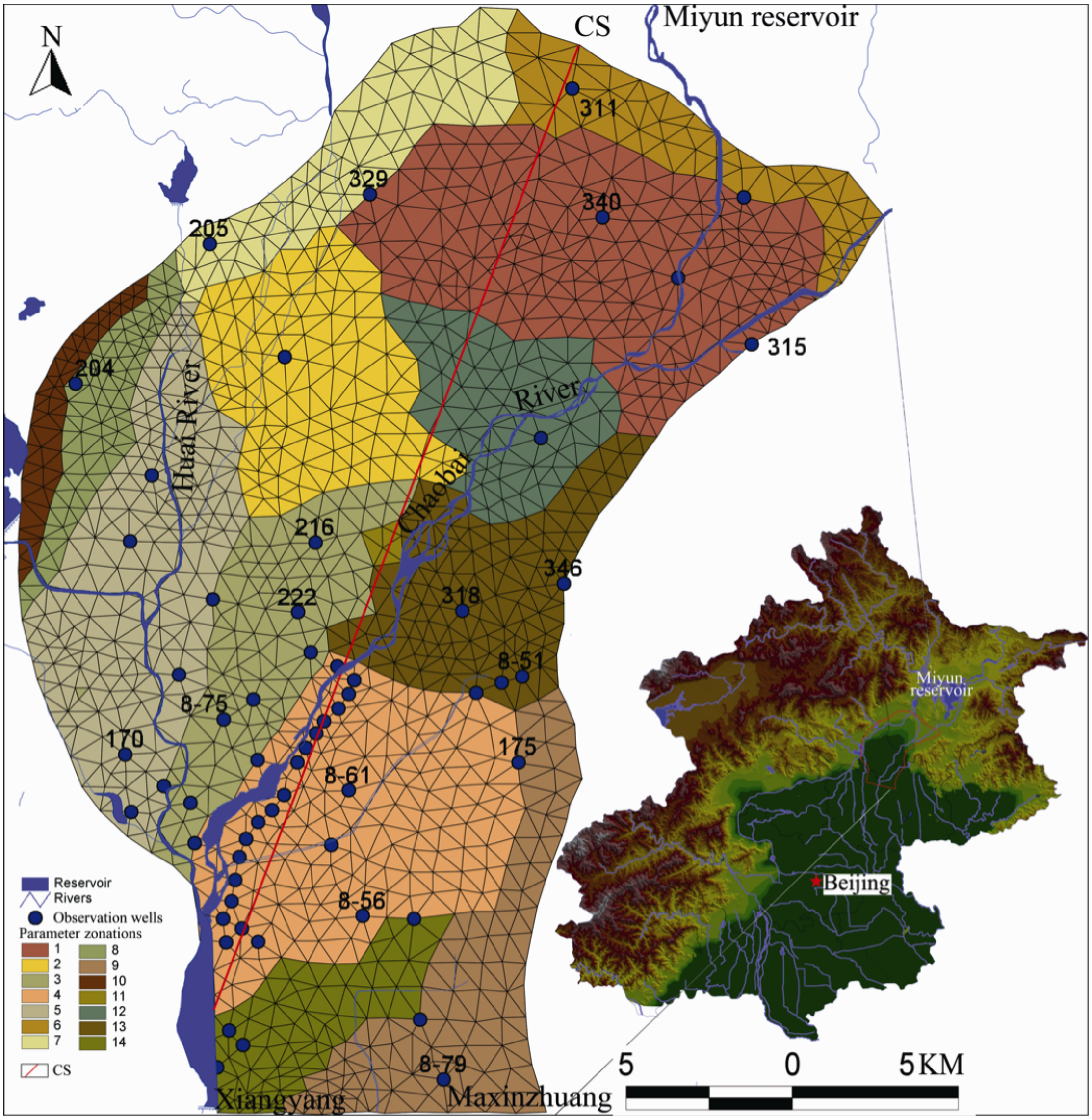


Figure 4.1: Site Map: Miyun-Huai-Shun aquifer site and surroundings. Location with hydraulic head observation wells of the study area, zones of model for parameters [90]

In our study, according to the lithological information and hydro-geologic characteristics from previous studies and research, the entire model aquifer can be divided into 14 zones in horizontal direction as shown by different colors in Figure 4.1. The model parameters required to be determined by the optimization procedure vary according to different zones, but the parameters are assumed to be constant in the same zone. In this study, two sets of model parameters, hydraulic conductivity (K) and specific yield (μ), were selected for the automatic calibration, since these two parameter sets are comparatively more important for groundwater model. Thus two parameters are taken into account in each zone, a total of ($14\times2=$) 28 model parameters need to be calibrated in the problem. The observed values of water levels were obtained from 46 observation wells scattered in the entire aquifer. As shown in Figure 4.1, the solid blue points demonstrate the 46 observation wells. Therefore, the objective of automatic groundwater model calibration process in this study was to change the input parameter vectors (K and μ) until model outputs were close enough to the observed water levels of 46 observation wells.

The groundwater system discussed in this paper is modeled with Finite Element Method (FEM), and triangles grid were used to discretize the aquifer domain. According to the theory of FEM, up to some limit the finer the grid is, the more accurate FEM performs. To show the performance of the new optimization algorithms in different grid size FEM models, two different scaled meshes were applied for the entire model area. The 'Coarse Grid' has 478 nodes and 871 triangular elements. The individual grid area ranges from 0.059 km2 to 1.71 km2. The 'Fine Grid' has 1239 nodes and 2337 elements (Figure 4.1) with a range of individual grid are from 0.032 km2 to 0.5 km2. Since the 'Coarse Grid' has much fewer model grids, it saves a lot of computation cost per simulation. The 'Fine Grid', on the other hand, has much finer grids, so it requires more computation time each model simulation.

In order to demonstrate the effectiveness of our algorithm, we include both 'Coarse Grid' and 'Fine Grid' in the study cases and compare performance of the new algorithm to Stochastic RBF, which has shown its superiority in early study. In Case 1, the "Coarse Grid" groundwater model to calibration 28 parameters. In Case 2, we applied the "Fine Grid" groundwater model with all 28 parameters calibration as well. Both of the two cases are based on the real observed data from 46 observation wells and the decision variable boundaries are defined according to the previous aquifer study of hydro-geologic parameters. The available pumping tests demonstrate that the ranges of hydraulic conductivity and specific yield are from 120 m/d to 270 m/d, 0.12 to 0.24, respectively. However, we used slightly broader ranges than pumping tests since the pumping tests were taken in limited areas and the ranges from those tests might not be exact enough. The aquifer parameter bounds are shown in Table 4.1.

Table 4.1: Range of Calibrated Parameters for the Miyun-Huai-Shun Groundwater Aquifer

Zone	$K_i(m/d)$		μ_i	
i	K_i^{lower}	K_i^{upper}	μ_i^{lower}	μ_i^{upper}
1	50	800	0.10	0.35
2	50	800	0.01	0.30
3	50	800	0.10	0.30
4	20	800	0.10	0.30
5	50	500	0.10	0.30
6	50	800	0.10	0.30
7	50	800	0.10	0.30
8	30	500	0.10	0.30
9	30	800	0.10	0.35
10	50	500	0.10	0.35
11	20	500	0.01	0.35
12	50	800	0.01	0.35
13	50	800	0.10	0.35
14	20	500	0.10	0.35

For the purposes of this study, simulation model output is denoted as $h_{i,j}^{sim}(K, \mu)$, which specifies the hydraulic head at well j and simulation time period i for given parameter sets K, µ. The value $h_{i,j}^{obs}$ represents the corresponding observed hydraulic heads from the observation wells. The whole aquifer domain is divided into N zones, thus each component of the model parameter is associated with one zone. The objective function seeks to find a good match between the observed and simulated hydraulic heads by minimizing their squared difference. There are a number of functions that can be used to specify the objective function, i.e. total squared residual error, R^2, root mean squared error, maximum absolute error and Nash-Sutcliffe index (NSE) [60]. In this study, a nonlinear least squares function is implemented to evaluate the goodness-of-fit measures of groundwater model calibration. The objective function for the optimization algorithm is to find the parameter sets that minimize SSE. The objective function formulated by total squared residual error (SSE) is defined as follows

$$\underset{K,\mu}{\text{Minimize}} \qquad SSE(K,\mu) = \sum_{j=1}^{T}\sum_{i=1}^{N}[\omega_{i,j}(h_{i,j}^{obs} - h_{i,j}^{sim}(K,\mu))^2] \tag{4.8}$$

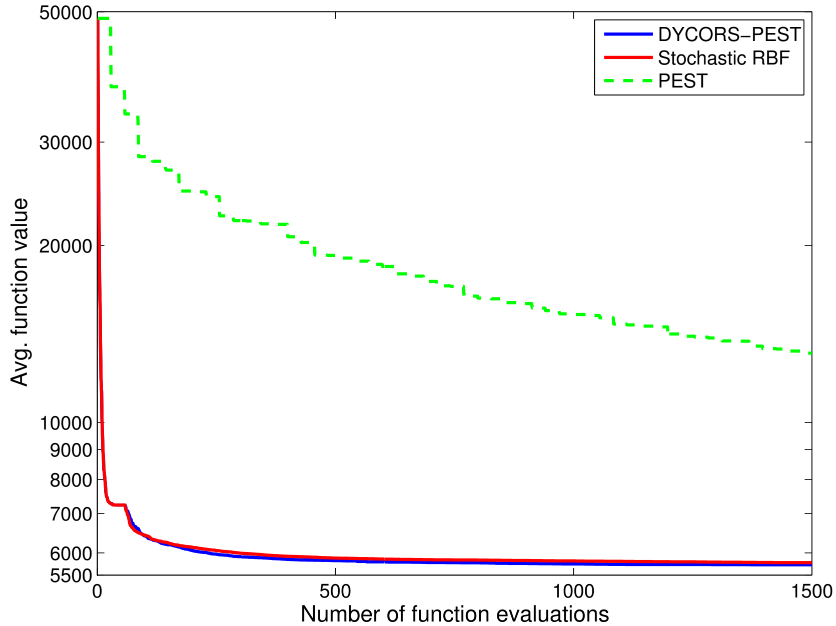
where, T is the total simulation period of time; N is the total number of wells; $SSE(K,\mu)$ is the sum of squared error between observed and simulated hydraulic heads given the values of parameter sets (K and μ) to be calibrated in the groundwater numerical model. The lengths of these two decision vectors are equal to the number of zones of the aquifer area. Groundwater models can be nonlinear, non-convex, non-smooth and even multimodal function of parameter values, so the corresponding inverse problems are very complicated to solve and the objective function has multiple local minima. The optimization processes requires a repeated simulation of a forward groundwater model in order to compute simulated hydraulic head using decision variables from parameter sets K and μ.

4.4 Numerical Results

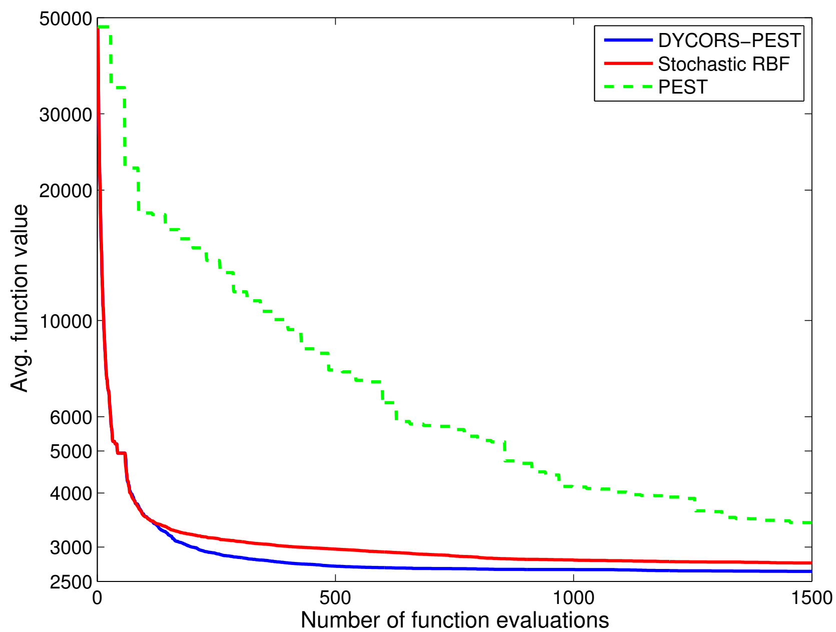
The goal of this study is to integrate a local optimizer PEST with global optimization algorithm (DYCORS). Thus the experimental runs were designed to test the new method against two other methods (PEST and Stochastic RBF), which have been applied to the same groundwater problems. Because of the multi-modal behavior of combinatorial optimization problems multiple trials for each of the algorithms were done. Multiple trials also helps to account for stochastic nature of of all three algorithms. In both of the cases, each algorithm was run 20 times for up to 1500 function evaluations. For both DYCORS-PEST and Stochastic RBF, symmetric Latin hypercube design, which consists of $2(d+1)$ points. The starting point for PEST is obtained from the first point of Latin hypercube initial design since PEST requires one initial set.

To compare the optimization algorithms, we examined mainly two characteristics of the new method: (1) efficiency in giving good objective function for a given number of function evaluations; (2) variability of solutions in multiple trials.

The performance of all three algorithms is compared in Figure 4.2 by using progress plots: average best objective function value versus the number of function evaluations. Average best objective function value indicates the average over 20 trials of the best objective function solution obtained in the given number of evaluations. The lowest curves are the best since the goal of optimization is to reduce the differences between the observed data and simulated data, and hence have the lowest objective value. Thus, the smaller the average objective values are, the better optimization results we get. From Figure 4.2, PEST is clearly the algorithm with the worst performance by looking at the significant distance between the green dashed curve (PEST) and the other two curves (solid red curve denotes Stochastic RBF; solid blue curves is DYCORS-PEST) in both Case 1 and Case 2. In order to visualize the performance of DYCORS and Stochastic RBF, PEST is removed from the progress plot as shown in Figure 4.3. In both cases, DYCORS-PEST is clearly superior to Stochastic RBF as it has the fastest drop and lowest average SSE at all the number of function evaluations. As described in earlier sections, Case 2 uses the 'Fine Grid' groundwater model which results a more global problem thus this case is a more difficult optimization problem than Case 1 ('Coarse Grid'). Figure 4.2 shows that the advantages of DYCORS-PEST in Case 2 is more obvious than in Case1 given Case 2 solves a harder problem.

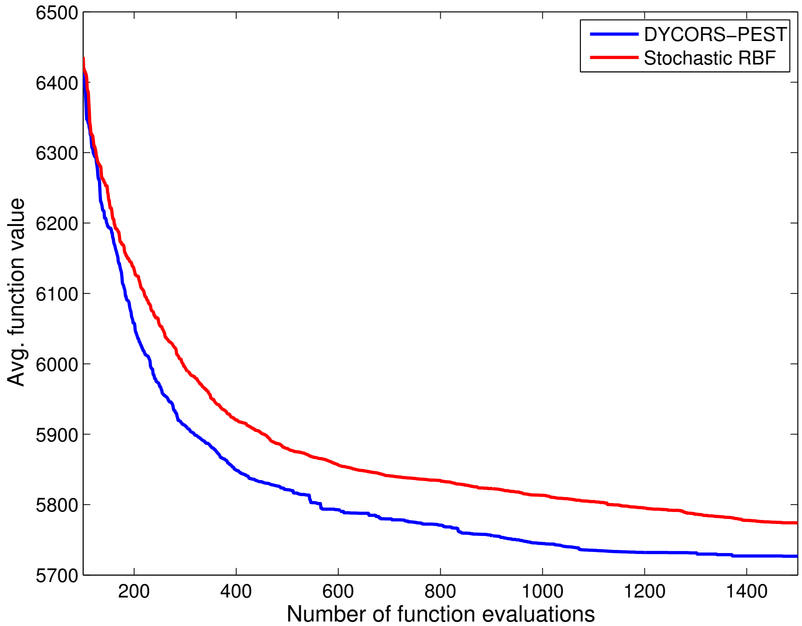


(a) Case 1 (Coarse Grid)

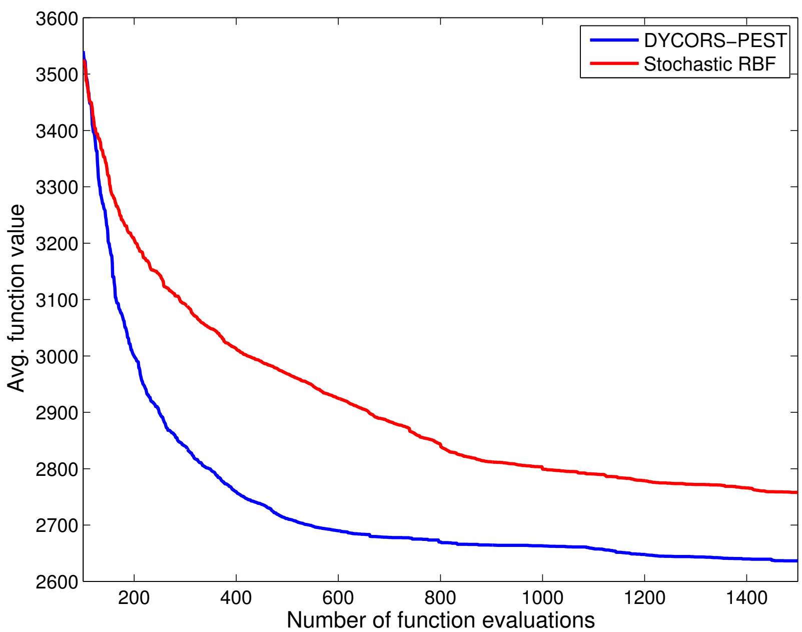


(b) Case 2 (Fine Grid)

Figure 4.2: Objective function value averaged over 20 trials vs. number of function evaluations, including PEST; a) Case 1, b) Case 2



(a) Case 1 (Coarse Grid)



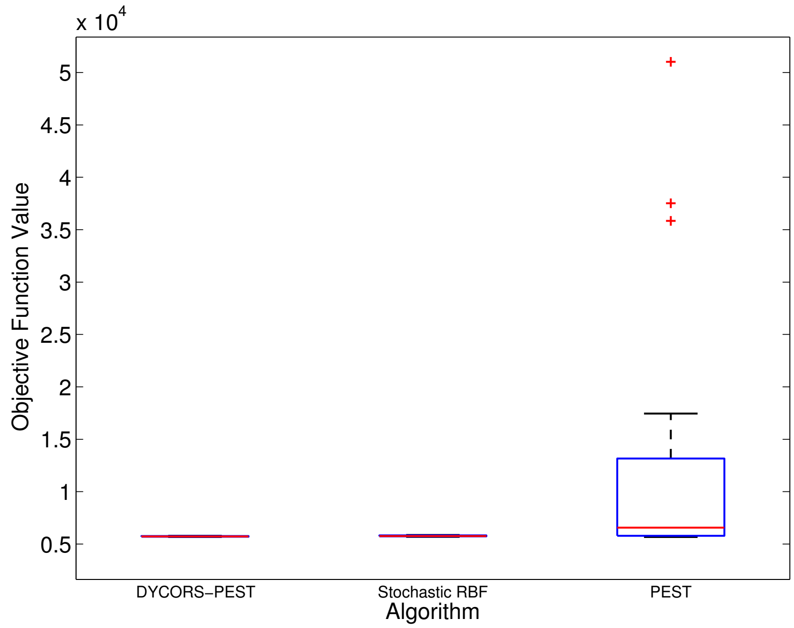
(b) Case 2 (Fine Grid)

Figure 4.3: Objective function value averaged over 20 trials vs. number of function evaluations, excluding PEST; a) Case 1, b) Case 2

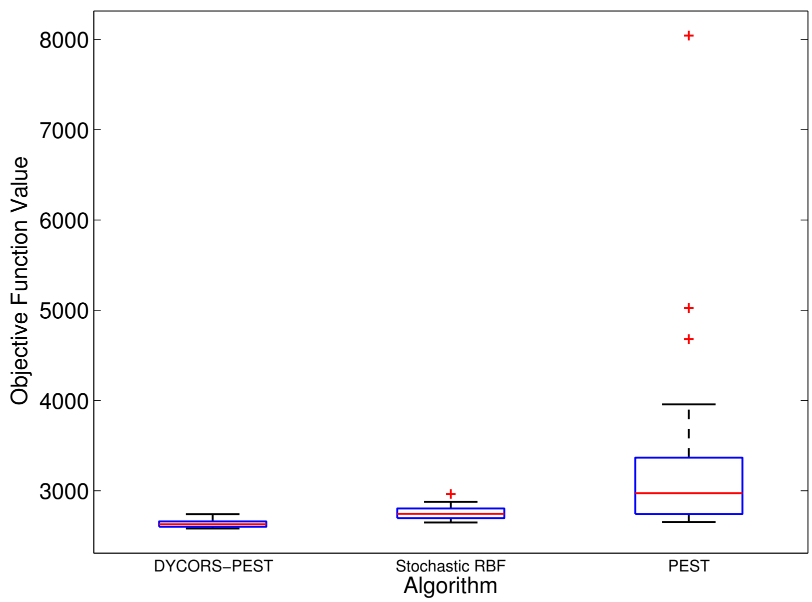
Although Figure 4.2 and 4.3 show the efficiency of the algorithms, the robustness of algorithms cannot be demonstrated only by the average values of objective function values. In a real calibration practice, probably only one algorithm will be applied to calibrate because the objective functions for environmental models are usually extremely expensive to calculate. Therefore, it is particularly important to choose an algorithm that can produce better results consistently, which implies the mean is low and the variance is small. A smaller variability with a low mean of objective function value suggests a more reliable algorithm for producing a good calibration in any given trial. Thus, an algorithm that produces a good solution consistently is clearly a superior choice to an algorithm that has a sizable chance of producing a poor solution. The mean and standard deviation of objective function value for the best solution at the end of 200, 500, 1000 and 1500 function evaluations for both Cases and all the algorithms are shown in Table 4.2, where the best results are marked by bold. The reported results show that DYCORS-PEST has the lowest mean and standard deviation among all three algorithms with different number of function evaluations in both cases. In order to demonstrate the variability of these algorithms, the best solutions at the end of 1500 function evaluations of each algorithm are plotted into box plots as shown in Figure 4.4. The box plot shows the median, interquartile range, and outliers based on the 20 trials for each algorithm. Figure 4.4a) and b) denote box plots for Case 1 and Case 2, respectively. PEST clearly is the worst among all three algorithms as indicated by its large spreads. Figure 4.5 shows box plots with only DYCORS-PEST and Stochastic RBF, where DYCORS-PEST has the smaller spread (more reliable) when compared to Stochastic RBF for the SSE objective function formulation. In addition, both PEST and Stochastic RBF have outliers while there is no outlier in DYCORS as shown in Figure 4.4. However, the outlier value of PEST is substantially large in both Cases because PEST is not a global optimization algorithm and it can be easily trapped into local minima without moving to the global optimal solution if the initial solutions assigned to PEST are close to local minima.

Table 4.2: Mean and Standard Deviation of objective function (SSE) over 20 trials with 200, 500, 1000 and 1500 function evaluations for Case 1 and Case 2. Best result of all algorithms is marked by bold.

Problem	Algorithm	Statistics	200 eval.	500 eval.	1000 eval.	1500 eval.
Case 1	DYCORS-PEST	Mean	**6057.9**	**5821.4**	**5744.9**	**5726.7**
		SD.	**98.45**	**69.23**	**47.96**	**38.16**
	Stochastic RBF	Mean	6133.6	5879.7	5813.4	5774.2
		SD.	167.35	67.06	66.62	52.74
	PEST	Mean	24698	19242	15269	13120
		SD.	16069	14911	14309	12946
Case 2	DYCORS-PEST	Mean	**2999.0**	**2711.6**	**2663.2**	**2636.6**
		SD.	**142.82**	**58.48**	**44.02**	**45.24**
	Stochastic RBF	Mean	3205.5	2968.1	2799.2	2757.9
		SD.	203.39	151.89	100.44	81.48
	PEST	Mean	15340	7690	4127	3416
		SD.	10579	5490.7	2060	1274.4

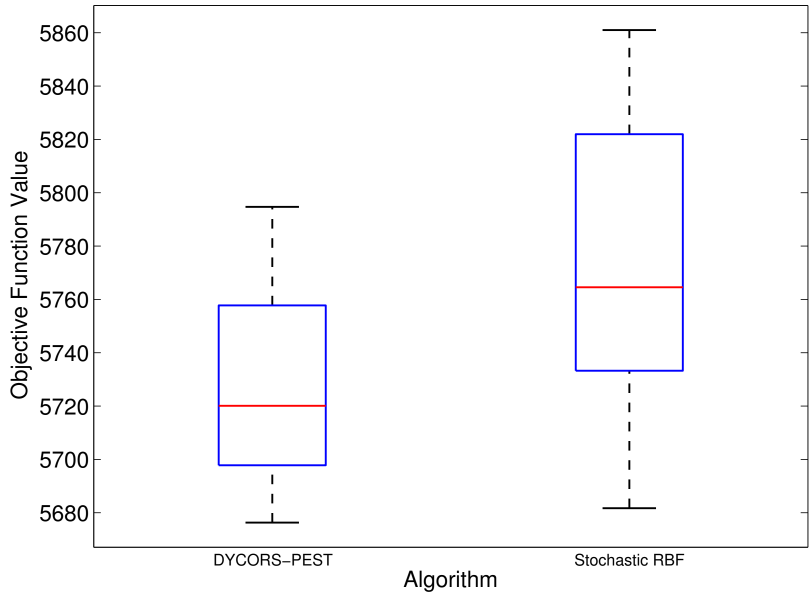


(a) Case 1 (Coarse Grid)

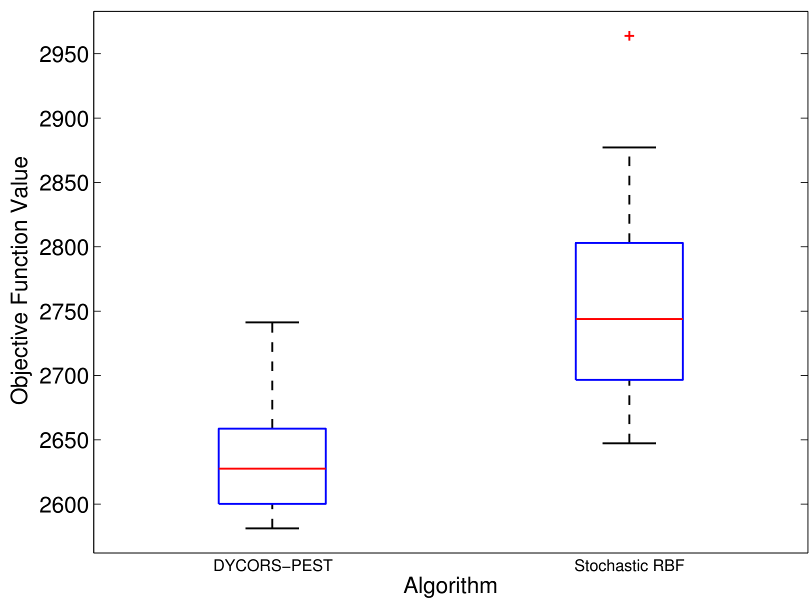


(b) Case 2 (Fine Grid)

Figure 4.4: Box plot of best solution for each algorithm based on 20 trials after 1500 function evaluations, including PEST; a) Case 1, b) Case 2



(a) Case 1 (Coarse Grid)



(b) Case 2 (Fine Grid)

Figure 4.5: Box plot of best solution for each algorithm based on 20 trials after 1500 function evaluations, excluding PEST; a) Case 1, b) Case 2

In order to ensure a fair comparison between the tested algorithms, this study initiates all the algorithms from the same set points. Pairwise two sample statistical tests were performed for significant difference in the means of the objective function values for the best objective function value by each algorithm. Table 4.3 show the p-value for each pairwise test. A p-value is the smallest value of the type-I error (i.e. incorrectly rejecting the null hypothesis when it is true) such that the observed results would be sufficient to reject the null hypothesis (which in this cases is that the algorithms are the same). A low p-value suggests a strong evidence for rejection the null hypothesis. The results indicate that for both cases DYCORS-PEST produces significantly different (lower) mean at a 5% significant level from all the other tested algorithms, thereby providing strong evidence of superior algorithm performance. The test results between Stochastic RBF and PEST indicate that Stochastic RBF has significant different mean than PEST when $\alpha = 5\%$.

Table 4.3: Hypothesis testing for differences in means at the final function evaluation of Case 1 and Case 2 (1500th simulation) with significant at the $\alpha = 5\%$. * after the p-value denotes that the algorithm in the row is better than the one in the row with significance level at $\alpha = 5\%$.

Problem	Algorithm	DYCORS-PEST	Stochastic RBF	PEST
Case 1	DYCORS-PEST	1.000	0.0023*	0.0095*
	Stochastic RBF		1.000	0.0268*
	PEST			1.000
Case 2	DYCORS-PEST	1.000	< 0.001*	0.0148*
	Stochastic RBF		1.000	0.0154*
	PEST			1.000

4.5 Conclusions

In this study, we introduced the algorithm DYCORS-PEST, which is a surrogate model based optimization algorithm for computationally expensive calibration problems. DYCORS-PEST is essentially an integration method by combing a global optimization algorithm DYCORS introduced by Regis and Shoemaker [73]with a local optimization method PEST [23]. DYCORS-PEST extends DYCORS by another local search phase, which is a gradient based local optimization on the true objective function to improve the accuracy of the solution.

We evaluated the performance of DYCORS-PEST on two 28-parameter groundwater calibration study cases based on Miyun Aquifer with real observation data. Both of the two cases are multi-modal, nonlinear and computationally expensive groundwater calibration problems. Case 2 uses a finer model grid thus the problem is more difficult than Case 1. Numerical experiments showed that DYCORS-PEST was superior to PEST and Stochastic RBF in both cases within 1500 function evaluations (see Figure 4.2 and Figure 4.3). This result further illustrates that DYCORS-PEST is an effective tool for calibrating parameter for computationally expensive groundwater models with high dimension. The reliability of DYCORS-PEST as indicated by the box plots also provided clear evidence of superiority of this algorithm over other methods. DYCORS-PEST applied in this study can be extended to any high dimensional calibration problem that requires computationally expensive simulations for cost evaluation within a limited number of function evaluations.

Chapter 5

Conclusions

This dissertation focuses on development and implementation of computationally efficient optimization algorithms for groundwater management and calibration of computationally expensive models. The application of GOMORS, a new multi-objective optimization algorithm, was conducted to solve the computationally groundwater management problems with multiple conflicting objectives. A new mixed-integer optimization algorithm, SO-MI, was implemented to minimize the computational expense of fixed cost problems with applications to groundwater remediation designs. We developed a new methodology DYCORS-PEST to solve high dimensional computationally expensive groundwater calibration problems.

In Chapter 2, we have shown that the surrogate based multi-objective optimization algorithm, GOMORS, does outperforms others in a statistically significant manner, especially for with limited computation budget. We employed three difference performance metrics, hypervolume, IGD and GD, to assess the effectiveness of four multi-objective optimization algorithms. NSGA-II, the most widely used multi-objective optimization algorithm as the baseline in the comparison, has the worst performance. The other two popular algorithms, AMALGAM and BORG, also have very promising performance. However, our results show that GOMORS is more efficient given very limited computation simulation times for problems that are computationally expensive.

In Chapter 3, we applied a novel mixed-integer optimization algorithm SO-MI on two groundwater pump-and-treat system design problems. SO-MI is a surrogate model based algorithm designed for computationally expensive nonlinear mixed-integer black-box optimization problems. The key feature of this algorithm is that it utilizes surrogate models to guide the search thus save the expensive function evaluation budget, and is able to find accurate solutions with relatively few function evaluations. We also evaluated the performances of two other popular mixed-integer algorithms, Genetic Algorithms and NOMAD. One of the study cases is a hypothetical case involving one chemical contaminant with 64 decision variables (32 integer and 32 continuous) and the other study case is modified from a real EPA study case, which deals with two different contaminants with 18 decision variables (8 integer and 10 continuous). Thus, the real application is a more computationally expensive and complicated problem. The results indicate that SO-MI is able to find significantly better solutions than the other two algorithms under limited computation budget (1000 function evaluations in this study) for both study cases and SO-MI has more advantages when it is applied to more complicated problem.

In Chapter 4, a new methodology DYCORS-PEST, integrating local search method PEST with response surface based global optimization method DYCORS was developed for groundwater model calibration problems. The suggested DYCORS-PEST method tries to use a memetic approach to improve the accuracy of the solution in which we use a local optimization search around a subset of the previously evaluated points. Starting with a surrogate based global search algorithm can efficiently explore the problem domain and locate a good starting point for the local search. PEST is a well-known efficient local search algorithm in Water Resources field. With the information from the global search phase, PEST is able to quickly find the local minimum. Another feature of the global search method in DYCORS-PEST is that the number of decision variables being perturbed is dynamically adjusted in each iteration in order to be more effective for higher dimensional problems. The study compared the suggested DYCORS-PEST with Stochastic RBF, another surrogate based global search algorithm and PEST with applications to two 28-parameter groundwater calibration problems. Our results indicate this new method outperforms Stochastic RBF and PEST for high dimension computationally expensive groundwater calibration problems.

New Methods GOMORS, SO-MI and DYCORS-PEST are shown to be distinctly better than previous algorithms on water resources problem in the sub area of Groundwater Hydrology. Although this study focused on groundwater management and calibration models, the results are just as relevant to all environmental simulations of a computationally demanding model. In further research, the suggested methodologies should be extended for much larger scaled water resources problems and to the problems from other application areas using traditional optimization methods.

Bibliography

[1] KC Abbaspour, R Schulin, and MT van Genuchten. Estimating unsaturated soil hydraulic parameters using ant colony optimization. *Advances in Water Resources*, 24(8):827–841, 2001.

[2] M A Abramson, C Audet, G Couture, Jr J E Dennis, S Le˜Digabel, and C Tribes. The NOMAD project. Technical report.

[3] Mark A Abramson and Charles Audet. Convergence of mesh adaptive direct search to second-order stationary points. *SIAM Journal on Optimization*, 17(2):606–619, 2006.

[4] Mark A Abramson, Charles Audet, James W Chrissis, and Jennifer G Walston. Mesh adaptive direct search algorithms for mixed variable optimization. *OL*, 3(1):35–47, 2009.

[5] Mark A Abramson, John E Dennis, Charles Audet, and Québec Groupe d'études et de recherche en analyse des décisions Montréal. *Filter Pattern Search Algorithms for Mixed Variable Constrained Optimization Problems*. Montréal : Groupe d'études et de recherche en analyse des décisions, 2004.

[6] D.P. Ahlfeld, J.M. Mulvey, and G.F. Pinder. Designing optimal strategies for contaminated groundwater remediation. *Advances in Water Resources*, 9(2):77–84, 1986.

[7] Taimoor Akhtar. Efficient Multi-objective Optimization of Computationally Expensive Models with Application to Watershed Model Calibration. *PhD Thesis*, 2012.

[8] Alaa H Aly and Richard C Peralta. Comparison of a genetic algorithm and mathematical programming to the design of groundwater cleanup systems. *Water Resour. Res.*, 35(8):2415–2425, August 1999.

[9] M.M. Aral. *Aquifer Parameter Prediction in Leaky Aquifers*. Elsevier Science, 1985.

[10] M Atiquzzaman, S Y Liong, and X Yu. Alternative decision making in water distribution network with NSGA-II. *... of water resources planning and ...*, 2006.

[11] M. Babbar and B.S. Minsker. Groundwater remediation design using multiscale genetic algorithms. *J. Water Resour. Plann. Manage.*, 132:341, 2006.

[12] David Becker, Barbara Minsker, Robert Greenwald, Yan Zhang, Karla Harre, Kathleen Yager, Chunmiao Zheng, and Richard Peralta. Reducing Long-Term Remedial Costs by Transport Modeling Optimization. *Ground Water*, 44(6 Understanding):864–875, November 2006.

[13] Elias G Bekele and John W Nicklow. Multi-objective automatic calibration of SWAT using NSGA-II. *Journal of Hydrology*, 341(3-4):165–176, August 2007.

[14] M.D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, 2003.

[15] Jes s Carrera, Andr s Alcolea, Agust n Medina, Juan Hidalgo, and Luit J Slooten. Inverse problem in hydrogeology. *Hydrogeol J*, 13(1):206–222, February 2005.

[16] Jen-Min CHENG and William W G Yeh. A Proposed Quasi-Newton Method for Parameter-Identification in a Flow and Transport-System. *Advances in Water Resources*, 15(4):239–249, 1992.

[17] Carlos A Coello Coello, Gary B Lamont, and David A van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, September 2007.

[18] A R Conn, K Scheinberg, and Luís N Vicente. Geometry of interpolation sets in derivative free optimization. *Math. Program.*, 111(1-2):141–172, December 2006.

[19] Teresa B Culver and Christine A Shoemaker. Dynamic Optimal Control for Groundwater Remediation With Flexible Management Periods. *Water Resour. Res.*, 28(3):629–641, March 1992.

[20] Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, July 2001.

[21] Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T Meyarivan. A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. In *Lecture notes in ...*, pages 849–858. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.

[22] J. Doherty and J.M. Johnston. METHODOLOGIES FOR CALIBRATION AND PREDICTIVE ANALYSIS OF A WATERSHED MODEL1. *JAWRA Journal of the American Water Resources Association*, 39(2):251–265, 2003.

[23] John Doherty. *PEST: Model-Independent Parameter Estimation, user's manual, 5th ed.* Brisbane, Australia, April 2010.

[24] John Doherty. *Addendum to the PEST Manual*. Brisbane, Australia, June 2012.

[25] John Doherty and David Welter. A short exploration of structural noise. *Water Resour. Res.*, 46(5), May 2010.

[26] Jason L Dorn and S Ranji Ranjithan. Evolutionary multiobjective optimization in watershed water quality management. In *EMO'03: Proceedings of the 2nd international conference on Evolutionary multi-criterion optimization*. Springer-Verlag, April 2003.

[27] Q Y Duan, V K Gupta, and S Sorooshian. Shuffled complex evolution approach for effective and efficient global minimization. *J. Optim. Theory Appl.*, 76(3):501–521, 1993.

[28] Q Y Duan, S Sorooshian, and VK GUPTA. Optimal Use of the Sce-Ua Global Optimization Method for Calibrating Watershed Models. *Journal of Hydrology*, 158:265–284, 1994.

[29] Andreas Efstratiadis and Demetris Koutsoyiannis. One decade of multi-objective calibration approaches in hydrological modelling: a review. *Hydrological Sciences Journal–Journal des Sciences Hydrologiques*, 55(1):58–78, 2010.

[30] Mark Erickson, Alex Mayer, and Jeffrey Horn. Multi-objective optimal design of groundwater remediation systems: application of the niched Pareto genetic algorithm (NPGA). *Advances in Water Resources*, 25(1):51–65, 2002.

[31] S Finsterle. iTOUGH2 user's guide. *LBNL-40040*, page 130, 2007.

[32] F Giacobbo, M Marseguerra, and E Zio. Solving the inverse problem of parameter estimation by genetic algorithms: the case of a groundwater contaminant transport model. *Ann Nucl Energy*, 29(8):967–981, 2002.

[33] T. R. Ginn and J. H. Cushman. Inverse methods for subsurface flow: A critical review of stochastic techniques. *Stochastic Hydrology and Hydraulics*, 4(1):1–26, March 1990.

[34] David Edward Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, January 1989.

[35] Steven M Gorelick, Clifford I Voss, Philip E Gill, Walter Murray, Michael A Saunders, and Margaret H Wright. Aquifer Reclamation Design: The Use of Contaminant Transport Simulation Combined With Nonlinear Programing. *Water Resour. Res.*, 20(4):415–427, 1984.

[36] David Hadka and Patrick Reed. Borg: An Auto-Adaptive Many-Objective Evolutionary Computing Framework. *Evolutionary Computation*, pages 1–31, February 2012.

[37] Samuel Kamel Hanna. *An iterative Monte Carlo technique for estimating conditional means and variances of transmissivity and hydraulic head fields*. PhD thesis, The University of Arizona., 1995.

[38] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.

[39] M Heidari and SR Ranjithan. A hybrid optimization approach to thf estimation of distributed parameters in two-dimensional confined aquifers. *J Am Water Resour as*, 34(4):909–920, 1998.

[40] Mary C Hill and Claire R Tiedeman. *Effective Groundwater Model Calibration: With Analysis of Data, Sensitivities, Predictions, and Uncertainty*. Wiley-Interscience, 1 edition, January 2007.

[41] MC Hill, RL Cooley, and DW Pollock. A controlled experiment in ground water flow model calibration. *Ground Water*, 36(3):520–535, 1998.

[42] Amy B Chan Hilton and Teresa B Culver. Groundwater Remediation Design under Uncertainty Using Genetic Algorithms. *J Water Res Pl-Asce*, January 2005.

[43] Y. Hyun and K.K. Lee. Model identification criteria for inverse estimation of hydraulic parameters. *Ground Water*, 36(2):230–239, 1998.

[44] DR Jones, M Schonlau, and WJ Welch. Efficient global optimization of expensive black-box functions. *J Global Optim*, 13(4):455–492, 1998.

[45] H. Karahan and M.T. Ayvaz. Groundwater parameter estimation by optimization and dual reciprocity finite differences method. *Journal of Porous Media*, 8(2), 2005.

[46] Halil Karahan and M Tamer Ayvaz. Forecasting Aquifer Parameters Using Artificial Neural Networks. *Journal of Porous Media*, 9(5), 2006.

[47] Halil Karahan and M Tamer Ayvaz. Simultaneous parameter identification of a heterogeneous aquifer system using artificial neural networks. *Hydrogeol J*, 16(5):817–827, August 2008.

[48] J B Kollat and P M Reed. Comparing state-of-the-art evolutionary multi-objective algorithms for long-term groundwater monitoring design. *Advances in Water Resources*, 29(6):792–807, June 2006.

[49] J B Kollat, P M Reed, and T Wagener. When are multiobjective calibration trade-offs in hydrologic models meaningful? *Water Resour. Res.*, 48(3):W03520, March 2012.

[50] L.K. Kuiper. A comparison of several methods for the solution of the inverse problem in two dimensional steady state groundwater flow modelling. *Water Resour. Res.*, 22(5):705–714, 1986.

[51] S Le˜Digabel. Algorithm 909: NOMAD: Nonlinear Optimization with the MADS algorithm. *ACM Transactions on Mathematical Software*, 37(4):1–15, 2011.

[52] AC LIN and WWG YEH. Identification of Parameters in an Inhomogeneous Aquifer by Use of Maximum Principle of Optimal Control and Quasi-Linearization. *Water Resour. Res.*, 10(4):829–838, 1974.

[53] D W Marquardt. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *J Soc Ind Appl Math*, 11(2):431–441, 1963.

[54] DC MCKINNEY and MD LIN. Genetic Algorithm Solution of Groundwater-Management Models. *Water Resour. Res.*, 30(6):1897–1906, 1994.

[55] DC MCKINNEY and MD LIN. Approximate Mixed-Integer Nonlinear-Programming Methods for Optimal Aquifer Remediation Design. *Water Resour. Res.*, 31(3):731–740, 1995.

[56] D. McLaughlin and L.R. Townley. A reassessment of the groundwater inverse problem. *Water Resour. Res.*, 32(5):1131–1161, 1996.

[57] B.S. Minsker and C A Shoemaker. Dynamic optimal control of in-situ bioremediation of ground water. *Journal of water resources planning . . .*, 1998.

[58] Pradeep Mugunthan, Christine A Shoemaker, and Rommel G Regis. Comparison of function approximation, heuristic, and derivative-based methods for automatic calibration of computationally expensive groundwater bioremediation models. *Water Resour. Res.*, 41(11), 2005.

[59] Juliane Müller, Christine A Shoemaker, and Robert Piché. Computers & Operations Research. *Computers and Operation Research*, 40(5):1383–1400, May 2013.

[60] J.E. Nash and JV Sutcliffe. River flow forecasting through conceptual models part I—A discussion of principles. *Journal of Hydrology*, 10(3):282–290, 1970.

[61] TN OLSTHOORN. Effective Parameter Optimization for Groundwater Model Calibration. *Ground Water*, 33(1):42–48, 1995.

[62] M. Powell. The NEWUOA software for unconstrained optimization without derivatives. *Large-scale nonlinear optimization*, pages 255–297, 2006.

[63] M J D Powell. UOBYQA: unconstrained optimization by quadratic approximation. *Math. Program.*, 92(3):555–582, May 2002.

[64] M J D Powell. Developments of NEWUOA for minimization without derivatives. *IMA Journal of Numerical Analysis*, 28(4):649–664, February 2008.

[65] MJD Powell. The Theory of Radial Basis Function Approximation in 1990. pages 105–210. Oxford University Press, USA, January 1992.

[66] MJD Powell. Recent research at Cambridge on radial basis functions. *International Series of Numerical Mathematics*, pages 215–232, 1999.

[67] Patrick Reed, Barbara Minsker, and David E Goldberg. Designing a competent simple genetic algorithm for search and optimization. *Water Resour. Res.*, 36(1):3757–3761, December 2000.

[68] Patrick M Reed, David M Hadka, Jonathan D Herman, Joseph R Kasprzyk, and Joshua B Kollat. Evolutionary multiobjective optimization in water resources: The past, present, and future. *Advances in Water Resources*, 51(C):438–456, January 2013.

[69] PM Reed and BS Minsker. Striking the balance: Long-term groundwater monitoring design for conflicting objectives. *J Water Res Pl-Asce*, 130(2):140–149, 2004.

[70] R G Regis and C A Shoemaker. Local Function Approximation in Evolutionary Algorithms for the Optimization of Costly Functions. *IEEE Trans. Evol. Computat.*, 8(5):490–505, October 2004.

[71] R G Regis and C A Shoemaker. A stochastic radial basis function method for the global optimization of expensive functions. *INFORMS Journal on Computing*, 19(4):497–509, 2007.

[72] R G Regis and C A Shoemaker. A Stochastic Radial Basis Function Method for the Global Optimization of Expensive Functions. *INFORMS Journal on Computing*, 19(4):497–509, January 2007.

[73] Rommel G Regis and Christine A Shoemaker. Combining radial basis function surrogates and dynamic coordinate search in high-dimensional expensive black-box optimization. *Eng Optimiz*, 45(5):529–555, 2013.

[74] I Remson, G M Hornberger, and F J Molz. *Numerical methods in subsurface hydrology with an introduction to the finite element method*. Wiley, New York [etc.], 1971.

[75] XL Ren and B Minsker. Which groundwater remediation objective is better: A realistic one or a simple one? *J Water Res Pl-Asce*, 131(5):351–361, 2005.

[76] Charles S Sawyer, David P Ahlfeld, and Alan J King. Groundwater Remediation Design Using a Three-Dimensional Simulation Model and Mixed-Integer Programming. *Water Resour. Res.*, 31(5):1373–1385, 1995.

[77] Christine A Shoemaker, Rommel G Regis, and RYAN C FLEMING. Watershed calibration using multistart local optimization and evolutionary optimization with radial basis function approximation. *Hydrological Sciences Journal*, 52(3):450–465, June 2007.

[78] Abhishek Singh and Barbara S Minsker. Uncertainty-based multiobjective optimization of groundwater remediation design. *Water Resour. Res.*, 44(2), February 2008.

[79] Thangjam Somchand Singh and Dibakar Chakrabarty. Multi-objective optimization for optimal groundwater remediation design and management systems. *Geosci J*, 14(1):87–97, April 2010.

[80] N Z Sun. *Inverse Problems in Groundwater Modeling*. Theory and Applications of Transport in Porous Media. Springer, 1 edition, 1994.

[81] Ne-Zheng Sun, Ming-Chin Jeng, and William W G Yeh. A proposed geological parameterization method for parameter identification in three-dimensional groundwater modeling. *Water Resour. Res.*, 31(1):89–102, 1995.

[82] Y Tang, P Reed, and T Wagener. How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration? *Hydrology and Earth System Sciences*, 10(2):289–307, May 2006.

[83] Bryan A Tolson and Christine A Shoemaker. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resour. Res.*, 43(1), January 2007.

[84] David A van Veldhuizen and Gary B Lamont. Evolutionary computation and convergence to a pareto front. pages 221–228, 1998.

[85] David A van Veldhuizen and Gary B Lamont. Multiobjective Evolutionary Algorithm Test Suites. *SAC*, pages 351–357, 1999.

[86] Jasper A Vrugt and Bruce A Robinson. Improved evolutionary optimization from genetically adaptive multimethod search. *Proceedings of the National Academy of Sciences of the United States of America*, 104(3):708–711, 2007.

[87] Ying Wan. Comparison of Stochastic Radial Basis Function and PEST for Automatic Calibration of Computationally Expensive Groundwater Models with Application to Miyun-Huai-Shun Aquifer. *Master Thesis*, 2013.

[88] Guangqian Wang, Jiahua Wei, and Yuandong Zhang. Analysis of the Sustainability of the Development of a Small Phreatic Aquifer in Northern China. *Water International*, 29(4):467–474, December 2004.

[89] Herbert F Wang and Mary P Anderson. *Introduction to Groundwater Modeling: Finite Difference and Finite Element Methods (Series of Books in Geology)*. W H Freeman & Co (Sd), May 1982.

[90] J H Wei. *Groundwater Geographical Information System-Integration, Visualization and Case Study*. PhD thesis, China, 2001.

[91] S.M. Wild, R G Regis, and C A Shoemaker. ORBIT: Optimization by radial basis function interpolation in trust-regions. *SIAM Journal on Scientific Computing*, 30(6), 2009.

[92] S Q Yan and B Minsker. Optimal groundwater remediation design using an Adaptive Neural Network Genetic Algorithm. *Water Resour. Res.*, 42(5), 2006.

[93] WG Yeh. Review of Parameter Identification Procedures in Groundwater Hydrology. *Water Resour. Res.*, 22(2):95–108, 1986.

[94] JH Yoon and CA Shoemaker. Comparison of optimization methods for ground-water bioremediation. *J Water Res Pl-Asce*, 125(1):54–63, 1999.

[95] Xuesong Zhang, Raghavan Srinivasan, and Michael Van Liew. On the use of multi-algorithm, genetically adaptive multi-objective method for multi-site calibration of the SWAT model. *Hydrol. Process.*, 24(8):955–969, April 2010.

[96] C. Zheng and P. Wang. Parameter structure identification using tabu search and simulated annealing. *Advances in Water Resources*, 19(4):215–224, 1996.

[97] C. Zheng and P P Wang. MT3DMS: a modular three-dimensional multispecies transport model for simulation of advection, dispersion, and chemical reactions of contaminants in groundwater systems; documentation and user's guide. 1999.

[98] Chunmiao Zheng and P Patrick Wang. An integrated global and local optimization approach for remediation system design. *Water Resour. Res.*, 35(1):137–148, 1999.

[99] E Zitzler, L Thiele, M Laumanns, C M Fonseca, and V G da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evol. Computat.*, 7(2):117–132, April 2003.