Development of Efficient Pollution Source Identification Model Using ANN-GMS-GA Based Simulation-Optimization Approach

Thesis

submitted in partial fulfillment of the requirements

of the Degree of

DOCTOR OF PHILOSOPHY

by

Triptimoni Borah

(Roll No. 11610417)

Department of Civil Engineering

Indian Institute of Technology Guwahati

Guwahati–781039

Dedicated to

My

Beloved Father

STATEMENT

I do hereby declare that the matter embodied in this thesis is the result of analytical work carried out by me in the Department of Civil Engineering, Indian Institute of Technology Guwahati, Guwahati, Assam, India.

In keeping with the general practice of reporting scientific evaluation, due acknowledgement have been made whenever the work described is based on the findings of literature review.

Triptimoni Borah

Guwahati 781039

Date:

CERTIFICATE

This is to certify that the work contained in the thesis entitled "**Development of Efficient Pollution Source Identification Model Using ANN-GMS-GA Based Simulation-Optimization Approach"** submitted by Mrs. Triptimoni Borah to the Indian Institute of Technology Guwahati, for the award of the degree of Doctor of Philosophy in Civil Engineering is a record of bonafide research work carried out by her under my supervision and guidance. The thesis work, in my opinion, has reached the requisite standard fulfilling the requirements for the degree of Doctor of Philosophy.

The research work contained in this thesis have not been submitted in part or full to any other University or Institute for award of any degree or diploma.

Guwahati
Date:

(Rajib Kumar Bhattacharjya)
Associate Professor
Department of Civil Engineering
Indian Institute of Technology Guwahati
Guwahati 781039

Acknowledgement

I express my sincere gratitude towards my supervisor, Dr. R.K. Bhattacharjya, Associate Professor, Department of Civil Engineering for his thorough guidance, constant encouragement and thorough motivation to pursue this research program, which made it a wonderful learning experience. His easy approachability and constant inspiration have induced a positive influence on me to do my best and for that, I am forever grateful. I am really indebted to him for the countless patient hours he spent in reviewing the thesis.

I am thankful to the members of my doctoral progress committee: Prof. S. Talukdar, Prof. A.K. Sarma and Prof. J.K. Deka for their various constructive suggestions, which have helped in improving various aspect of my research work.

I am also thankful and grateful to one and all that helped me in my research work. Thanks are due to Mr Parth Mazumder, Dr. A. Borsaikia, Ms Jonali Saikia, Dr. Malaya Chetia, Ms Swapnali Barman, Ms Runti Choudhury, Dr. Wazir Alam, Ms Bhagyashree Duwara and many unnamed for their support and cooperation which was quite significant.

I would like to express my affection and admiration to my family and parents who have wholeheartedly supported me during the course of this study. I cannot find words to express my thanks and gratitude to them. I would like to thanks my husband, Bishnu, daughter, Baishali, and son, Bishal, for their constant support for completion of this research work.

I would also take this opportunity to remember of the many people who came in my life during the last three years, either by design or by accident. They influenced me in many ways and gave me lot of experience.

Triptimoni Borah

Contents

Abstract

Identification of unknown pollution sources is an important and challenging task for the engineers working on pollution management of a groundwater aquifer. The locations and the transient magnitude of contaminant sources can be identified using inverse optimization techniques. In this approach, an error function is formulated which can be minimized using an optimization algorithm. The error function is the difference between simulated and observed contaminant concentration at observation locations. The observed concentration is measured in the field. On the other hand the simulated concentration can be calculated using aquifer simulation model. As such, there is a need to incorporate the aquifer simulation model with the optimization model. Thus technique is also known as simulation-optimization approach as aquifer simulation model is incorporated with the optimization model. The performance of the source identification model is highly related to the aquifer simulation model. Incorporation of sophisticated numerical simulation model will give better performance, but the model will be more computationally expensive. On the other hand, the model will be computationally less expensive, if approximate simulation model like ANN model is incorporated with the optimization model. However, in this case, there will be a reduction in the predictive performance of the model. Keeping this in view, this study develops four improved methodologies for identification of unknown groundwater pollution sources. In the first approach, the groundwater modeling system (GMS) is linked with optimization model for solving source identification problem. The incorporation of GMS with the optimization model enables to solve bigger real world pollution source identification problems. The main challenge of this approach is the linking of the external aquifer simulator, GMS with the optimization model. This has been overcome by executing GMS in Matlab environment. The main drawback of the approach is that the approach is computationally expensive. For reducing the computational time, the second approach uses artificial neural networks (ANN) model to simulate the flow and transport processes of the aquifer. The ANN model is then externally linked with the optimization model. This approach drastically reduces the computational time of the simulation-optimization model. The problem which was solved in few days can now be solved in few hours. However, most of the time, it yields only the near optimal solution. Therefore, in the third approach, a hybrid optimization model is presented which initially solves the problem using ANN based

simulation-optimization model. The solution obtained by the ANN based model is then used as the initial solution for the GMS based model. This approach is computationally more efficient than the GMS based model and also more accurate than the ANN based model. However, this model is not computationally efficient as compared to ANN based simulation-optimization model. Therefore, for further reduction in computational time, another methodology is presented incorporating both numerical and approximate simulation models. This methodology is computationally better than the GMS based simulation-optimization model while maintaining predicting capability at per GMS based model. The inverse optimization models are solved using two non-classical optimization techniques, *i.e.* Genetic algorithm and Direct search method. The efficiency and field applicability of the models are demonstrated using two illustrative study areas. The study reveals that the performance of Genetic algorithm is better than Direct search method and the fourth model, ANN-GMS-GA based simulation-optimization model is more efficient in predicting the unknown groundwater pollution sources.

Keywords: pollution source identification, MODFLOW, MT3DMS, ANN, feed forward back propagation, Simulation, Optimization

List of Figures

List of Tables

Nomenclature

b_k	Binary number (0 or 1)
b_1	Biases in the hidden layer
b_2	Biases in the output layer
C^k	Dissolved concentration of species k
C_s^k	Concentration of the source or sink flux for species k
$\boldsymbol{C}$	Output vector (concentration)
D_{ij}	Hydrodynamic dispersion coefficient tensor
E_p	Change of weights
F_i	Fitness value of i^{th} string
k_{xx}	Hydraulic conductivity along the x coordinates axes
k_{yy}	Hydraulic conductivity along the y coordinates axes
k_{zz}	Hydraulic conductivity along the x coordinates axes
θ	Porosity of the subsurface medium
$\propto_L$	Longitudinal dispersivity
$\propto_T$	Transverse dispersivity
δ_{pk}	Error at a single output unit
$\emptyset_1$	Transfer function for the neurons in hidden layer
$\emptyset_{\mathbf{2}}$	Transfer function for the neurons in the output layer
h	Potentiometric head
OC_t^n	Observed concentration
OC_o^k	Observed concentration at well location *o* for k^{th} time steps
$\overline{OC}_0^k$	Mean of observed concentration at well location for k^{th} time steps
P	Pumping well
q_s	Volumetric flow rate per unit volume of aquifer representing fluid sources (positive) and sinks (negative)

R	Coefficient of correlation
$\sum R_n$	Chemical reaction term
SC_t^n	Simulated concentration
SC_s^k	Simulated concentration at observed location o for k[th] time steps
$\overline{SC}_s^k$	Simulated concentration at observed location o for k[th] time steps
S1….S5	Contaminant sources
N	Total number of observation wells
n	denotes the total number of population
n_x	Number of patterns
η	Porosity
η	Learning rate parameter
q	Represents length of string
Ss	specific storage of the porous material
S	Storativity
S	input vector (source values)
T	Total time steps
t	time
v_i	Seepage or linear pore water velocity
V	weight matrix for the synaptic connections between hidden and output layer
W	volumetric flux per unit volume representing sources and/ or sink of water
W1…. W8	Well number 1 to 8
W	Weight matrix for the synaptic connections between input and hidden layer
x_i	Distance along the respective Cartesian coordinate axis
x^u	Upper bound of the particular variable
x^l	Lower bound of the particular variable

Abbreviation

AARE	Average absolute relative error
AC	Actual concentration
ADE	Advective-Dispersion Equation
ARE	Absolute relative error
ANN	Artificial Neural Networks
BP	Back propagation
CRLS	Constrained robust least square
DE	Differential evolution
DS	Direct search
EC	Estimated concentration
ES	Evolutionary strategies
FFBP	Feed forward back propagation
FRM	Fitness reduction method
GA	Genetic algorithms
GMS	Groundwater modeling system
GP	Genetic programming
GPS	Generalized pattern search algorithm
HN	Number of hidden neurons
HS	Harmony search
LHS	Latin hypercube sampling
LM	Levenberg-Marquardt
MADS	Mesh adaptive search algorithm
MOC	Method of characteristics
MINOS	Modular In-core Nonlinear Optimization System
MRE	Minimum relative entropy
MSE	Mean square error

NNLS	Non negative least square estimate
ORGA	Optimum resources allocation guideline
PDF	Probability distribution function
PSA	Particle swarm algorithm
QR	Quasi-reversible
SA	Simulated annealing
SD	Standard deviation
SL	Source location
SBSM	Search bound sampling method
SUTRA	Saturated unsaturated transport
SLP	Successive linear programming
TR	Tikhonov regularization
TS	Threshold statistics
VEGA	Vector-evaluated genetic algorithm
WOS	Without sorting
WS	With sorting

Chapter 1

Introduction

1.1 General

Identification of unknown pollution sources in a groundwater aquifer is an important and challenging task for the hydrogeologist working on pollution management of an aquifer. The location and magnitude of unknown groundwater pollution sources can be identified using inverse optimization technique (Gorelick *et al*., 1983; Yeh,1986; Datta *et al*., 1989; Wagner, 1992; Aral and Guan, 1996; Mahar and Datta, 1997, 2000, 2001; Aral *et al*.,2001; Ahlfeld *et al*., 1986; Datta and Chakrabarty, 2003 *etc*.). For obtaining the unknown pollution sources, the inverse optimization model minimizes the error between observed and simulated concentration of contaminant at observation locations. The simulated concentrations can be obtained using the aquifer simulation model, which basically solves the governing partial differential flow and transport equations. As such, it is necessary to incorporate the aquifer simulation models with the optimization model for identifying unknown pollution sources. The simulation model can be incorporated with the optimization model using embedding optimization technique (Aguado and Remson, 1974; Gorelick, 1983; Alley *et al*., 1976; Willis and Newman, 1977; Molx and Bell, 1977; Elango and Rouve, 1980; Peralta and Datta, 1990; Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al*., 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2001; McPhee and Yeh, 2008; Datta *et al*., 2009), response matrix approach (Gorelick, 1979; Willis, 1979; Gorelick and Remson, 1982b; Gorelick *et al*., 1983; Aral and Guan, 1996) and by using linked simulation-optimization approach (Gorelick *et al.*, 1984; Tsai *et al.*, 2003; Datta and Chakrabarty, 2003; Singh and Datta, 2006; Bhattacharjya and Datta 2009). The embedded technique incorporates the finite difference or finite element approximation of the governing equations as equality constraints with the optimization model. Embedded optimization approach has been widely used by many researchers for solving source identification problems as well as for solving other groundwater management problems (Alley *et al.*, 1976; Willis and Newman, 1977; Molx and Bell, 1977; Elango and Rouve, 1980; Datta and Peralta, 1986; Ahlfeld *et al.*, 1986; Yazicigil and Rasheeduddin, 1987; Peralta and Datta, 1990; Peralta *et al.*, 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2000; Aral *et al.*, 2001; McPhee and Yeh, 2008; Datta *et al.,* 2009 *etc.*). Though the embedded technique was widely used by many researchers, but application of the technique is only limited to small scale groundwater management problem. It has several limitations for large scale groundwater management problems as large number of constraints need to be incorporated in the optimization problem. Solution of such optimization problem is quite complex and time consuming too. The response matrix approach is based on the principle of superposition and linearity. This method is reported as unsatisfactory for highly nonlinear system (Rosenwald and Green, 1974), and therefore not suitable for large scale groundwater management problems. In the linked simulation-optimization technique, the groundwater simulation model is externally coupled with the optimization model for solving the groundwater inverse problem or groundwater management problems (Gorelick, 1979; Willis, 1979; Gorelick and Remson, 1982; Gorelick *et al.,*1983; Skaggs and Kabala, 1994; Datta *et al.,* 1989; Aral and Guan, 1996; Datta and Chakrabarty, 2003; Bhattacharjya and Datta, 2005; Singh and Datta, 2006; Bhattacharjya and Datta, 2007; Bhattacharjya and Datta, 2009; Datta *et al.*, 2011 *etc*.). Datta and Chakrabarty (2003) used linked simulation optimization approach for solving pollution source identification model. They have linked the groundwater simulation model, SUTRA with the large scale optimization system solver MINOS for solving source identification problem. The advantage of this approach is that the optimization model has fewer decision variables and thus can be solved easily using suitable optimization algorithms. Moreover, sophisticated groundwater simulation models, like SUTRA, GMS, FEEFLOW, PMWIN *etc*. which have the capability to simulate complex real world groundwater situation can be used. Through the optimization model handles only few decision variables, the approach is computationally very extensive as repetitive iterations between the optimization model and the simulation model is necessary for obtaining the optimal solution.

Bhattacharjya and Datta (2009) reported that the simulation-optimization approach may need up to few days computational time for solving a medium level field scale groundwater management problem if numerical simulation model is used. In order to achieve computational efficiency, some researchers have used artificial neural networks model to identify unknown source characteristics. Singh *et al.* (2004) used artificial neural network model for identification of groundwater pollution sources when all the concentration data are available. They also solved the source identification problem using ANN when concentration measurement data was not available over an initial length of time (Singh and Datta, 2007). Some other application of artificial neural network for groundwater management can be found in Ranjithan et al., 1993; Aly and Peralta, 1999; Rao et al., 2004; Yan and Minsker, 2006; Bhattacharjya and Datta, 2009; Kourakos and Mantoglou, 2009; Dhar and Datta, 2009. The computational time can also be reduced by replacing the expensive numerical groundwater simulation model by an approximate simulation model. The most commonly used approximate simulation models are the regression model, artificial neural networks (ANN) model, genetic programming, etc. (Bhattacharjya et al., 2005). Alley (1986) and Lefkoff and Gorelick (1990) used the regression model for approximating aquifer processes within the management model. Bhattacharjya and Datta (2009) presented ANN-GA based simulation-optimization model for solving coastal aquifer management model. They have replaced the numerical groundwater simulation model by an approximate ANN model. This approach drastically reduces the computational time of the simulation-optimization model. The problem which was solved in few days can now be solved in few hours. The main disadvantage of the approach is that it may not yield the exact optimal solution of the problem as approximate simulation model is used in place of the exact numerical simulation model. Most of the time, it obtains the near optimal solution of the problem. Now we have two extreme situations, if we use the actual numerical simulation model, we may get the exact optimal solution of the problem but it will be very expensive in terms of computational time requirement. On the other hand, if we use the approximate simulation model like regression model or the ANN model, it may not yield the exact optimal solution of the problem. As such, to achieve efficiency in both accuracy and computational time, there is a need to develop better approaches for solving the groundwater source identification problem.

An optimization algorithm is necessary for solving the inverse optimization model. Gradient based classical optimization techniques have been implemented by various researchers (Alley *et al.,* 1976; Willis, 1976; Willis and Newman, 1977; Gorelick and Remson, 1982; Gorelick *et al*., 1983; Molx and Bell, 1977; Elango and Rouve, 1980; Peralta and Datta, 1990; Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al*., 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2001; McPhee and Yeh, 2008; Datta *et al.,* 2009). However, the algorithms are not suitable for non-linear non-convex problem. For such problems, the classical methods yield only the local optimal solution of the problem. The local optimal solution can be avoided by running the model for several times using different initial solutions. This this is a time consuming approach and does not give any guarantee that the global optimal solution will be obtained.

In order to overcome the difficulties of classical optimization techniques, global search techniques, such as genetic algorithms, direct search method, evolutionary strategies, *etc.* can be used to solve the inverse optimization problem. These algorithms work with the functions value only and does not require the derivative information of either the constraint functions or the objective function. These method are also known as zeroth order method as it does not use the derivative of the function. In this study, we have evaluated the performance of two non-classical optimization algorithms, direct search method (DS) and genetic algorithms (GA) in solving the inverse optimization problem. It has been observed that performance of genetic algorithms is better than the direct search method. Thus a genetic algorithm has been recommended for solving the problem.

Genetic algorithm is considered as one of the most popular and robust non-classical optimization algorithms to handle non-linear non-convex problem. The algorithm is based on the principle of natural evolution of species and is guided by the survival of fittest principle of nature. Genetic algorithm has been used extensively by different researchers for solving groundwater management models (Hilton and Culver, 2005; Mahinthakumar and Sayeed, 2005; Singh and Datta, 2006; Jin *et al.,* 2009; Bhattacharjya and Datta, 2009). This is a population based algorithm. Unlike the classical optimization methods, the algorithm starts with a set of initial solutions, called population. The populations pass though the natural operators, *i.e.* selection, crossover and mutation to search for better individual. The iterative process, which is known a generation continues till termination criteria are not satisfied.

For finding the optimal source locations and magnitude, the groundwater simulation models have to be incorporated with the genetic algorithm. The simulation models provide the information needed for calculating the fitness of all individual population. As such, in each generation, the simulation models have to be called for a numbers of times equal to the population size used in genetic algorithm. As a result, a huge number of simulation calls is necessary to obtain the optimal solution of the problem. For example, for population size of hundred and maximum iteration of hundred, tenthousand simulation calls will be necessary. Thus the algorithm is computationally very intensive, if computationally extensive simulation model, such as GMS is used. On the other hands, the model will be very efficient with respect to computational time requirement, if simplified approximated situation model, like ANN is used in the model. Therefore, if we use the groundwater simulation software GMS to simulate the aquifer, the computational time requirement will be very high, but solution will be more accurate. On the other hand, if we use the ANN based approximate simulation model, the computational time requirement will be very low, but the solution obtained may not be the global optimal solution and may only be a near optimal solution.

The efficiency in computational time and in predating capability can be achieved by using hybrid optimization technique. In this technique, ANN based simulation-optimization model is initially used to obtain a near optimal solution. The near optimal solution is then used as the initial solution for the numerical simulation based optimization model. This approach is computationally more efficient than the complete numerical simulation based simulation optimization model. For further improvement in computational time, another modified simulation-optimization approach is also proposed where fitness function of the population are calculated by utilizing either by GMS, i.e. the numerical simulation model or by ANN simulation model. This is a genetic algorithms based methodology. In this methodology, the fitness values of few best individuals of the population are calculated using GMS model and fitness of the other individuals are calculated using the ANN model. This has further reduced the computational time of the model and also obtains the global optimal solution as obtained by the numerical simulation based simulation-optimization model.

The performance of the proposed methodology is evaluated using two illustrative study areas. Initially, the illustrtaive study area considered by Mahar and Datta (2000) is used to demonstrate the efficiency and applicability of the proposed model. The model is then applied to a large aquifer system to show its field applicability. The evaluation of results show that the performance of this methodology is encouriging even when applied to a complicated source identification problem.

1.2 Groundwater pollution sources

The groundwater aquifers may be contaminated by manmade and natural sources. Some of the common sources of pollution are: leakages from septic tanks and landfills, dumping of industrial waste and municipal waste, dumping of toxic chemical waste, radioactive disposal, mining and mine drainage, application of fertilizer and pesticides to agricultural field, *etc*. Further, groundwater may also be polluted by natural pollution sources like arsenic, lead, fluoride *etc*. Further the contamination may be occurred at a particular location due to single source or by multiple sources. The groundwater pollution source which is extended in limited areas is termed as point source pollutant. Some of the point sources are leakage from sewer line, waste injection wells, landfill leaching, *etc.* Nonpoint sources may come from many distributed sources. It may be occurred due to groundwater and surface water interactions, natural leaching as well as rainfall and snowfall moving over and below the ground. The surface runoff carries the natural and manmade pollutants and deposits it in rivers and lakes. The nonpoint pollution sources also occur due to excessive use of fertilizer, herbicides, pesticides *etc*. In this study we have only considered the point source pollutant.

1.3 Objectives of the thesis

It has been observed that lot of studies have been reported by various researchers to solving the problem of unknown pollution sources in groundwater aquifer. The performance of the model is highly related to the numerical simulation model used in the simulation-optimization model. In this study, the commercial groundwater simulation system GMS is used as it has the capability to simulate complicated groundwater aquifer system. However, use of statistical numerical simulation model makes the simulation-optimization model computationally very expensive. Thus the main objective of this research work is to develop an efficient source identification model to identify unknown pollution sources of a complicated large scale groundwater aquifer. The objectives of this research work are as follows:

i) Development of an aquifer simulation model using Groundwater Modeling System (GMS).
ii) Development of pollution source identification model using GMS based simulation-optimization model. The inverse optimization model is solved using genetic algorithms (GA) and direct search (DS) method.
iii) Development of artificial neural networks (ANN) model for simulating the aquifer process.
iv) Development of pollution source identification model using ANN based simulation-optimization model. In this case also, the inverse optimization model is solved using genetic algorithms (GA) and direct search (DS) method.
v) Performance evaluation of Direct Search method and Genetic Algorithms in predicting unknown groundwater pollution sources.
vi) Development of hybrid model for identifying unknown pollution sources to reduce the computational time of the simulation-optimization model
vii) Development of GMS-ANN and GA based simulation-optimization model for efficient prediction of unknown pollution sources.
viii) Performance evaluation of the models using illustrative study areas.

1.4 Organization of the thesis

There are seven chapters in this thesis including the introductory chapter 1. In chapter 2, we have presented literature review of the studies conducted worldwide on identification of groundwater pollutions sources. In chapter 3, we have presented the methodology for simulating the flow and transport processes in aquifer by using GMS model. In this chapter, we have also developed simulation-optimization models using GMS. The optimization model is solved using genetic algorithms (GA) and direct search (DS) method. The comparison between Genetic Algorithms and Direct Search methods are also presented in this chapter. In chapter 4, we have presented ANN based simulation-optimization methodology for pollution source identification problem. The performance of ANN simulator is evaluated for error free and erroneous data. The chapter 5 deals with the development of hybrid-optimization model where solution obtained by ANN based simulation-optimization model is considered as the initial solution of GMS based optimization model. The efficiency and applicability of the model is also discussed in this chapter. In chapter 6, we have developed ANN-GMS and GA based simulation-optimization methodology where flow and transport processes have been done by using both GMS and ANN model. The relative efficiency of the methodology over the GMS based, ANN based and Hybrid-Optimization is also discussed in this chapter. The summary, conclusions and recommendations for future work of the study are presented in chapter 7.

Chapter 2

Literature Review

2.1 General

The requirement of freshwater is increasing day by day due to the alarming growth of population and substantial industrial and agricultural development. Due to the increased demands of freshwater, overexploitation of groundwater becomes unavoidable in many parts of the world, specifically for the regions away from the surface water sources. As a result of overexploitation, groundwater table has been depleted substantially in many parts of the world. Apart from depletion of groundwater table, the overexploitation of groundwater has also exaggerated the problem of groundwater contamination as more and more contaminants reach the well point due to the increased flow towards the well. The groundwater may be contaminated by various point and nonpoint sources of pollutants. Some of the common groundwater pollution sources are: intensive use of fertilizers, herbicides, pesticides, disposal of human and industrial waste, natural pollution sources of minerals like fluoride, arsenic, iron, *etc*. This leads in increase in concentration of contaminants at the well locations. Once groundwater gets polluted, it is difficult and uneconomical to restore its quality again. Moreover, in many situations, the sources of pollution are not known explicitly. As such, there is a need to develop methodologies to identify the unknown groundwater pollution sources so that appropriate management strategies can be adopted for sustainable exploitation of the aquifer (National Research Council 1990). The contamination of groundwater has been addressed by numerous researchers from different corners of the world (Deninger, 1970; Kleinecke, 1971; Emsellem and de Marsily, 1971; Maddock, 1972; Neuman, 1973; Frind and Pinder, 1973; Segol *et al.,* 1975; Nutbrown, 1975; Hefez, 1975; Alley *et al.,* 1976; Navarro, 1977; Molx and Bell, 1977; Elango and Rouve, 1980; Gorelick and Remson, 1982; Heidari, 1982; Gorelick, 1982; Gorelick and Remson, 1982; Yeh *et al.,* 1983; Willis, 1983; Tung and Kolterman, 1985; Datta and Peralta, 1986; Wanakule et al., 1986; Peralta and Kowalski, 1986; Yazicigil and Rasheeduddin, 1987; Reilly et al., 1987; Ahlfeld et al., 1988; Ahlfeld, 1990; Ahlfeld and Sawyer, 1990; Wang and Ahlfeld, 1994; Ahlfeld and Heidari, 1994; Gharbi and Peralta, 1994; Datta and Dhiman, 1996; Mahar and Datta, 2000, 2001; Das and Datta, 2000; Montas *et al.,* 2000; Mugunthan and Shoemaker, 2004; Nunes *et al*., 2004; Yeh *et al*., 2007; Dhar and Datta, 2007; Chandalavada and Datta, 2008; Kollat *et al*., 2011; Dhar and Datta ,2009; Azghadi and Kerachian, 2010; Chandalavada *et al*., 2011; Prakash and Datta, 2013).

The groundwater pollution source can be characterized by its location, magnitude, and duration of activity. The pollutant observed at an observation well may be caused by the pollution sources of more than one location. Further, at each potential location, magnitude of source flux may vary and may also operate with different disposal period. Thus, there is a huge combination of possible source characteristics in terms of its location, magnitude, and disposal periods (Srivastava and Singh, 2014). This is really a difficult task to identify the true combination of pollution sources from the innumerable combinations.

Unknown groundwater pollution sources can be identified using inverse optimization technique. In this technique an error function is defined which minimizes the absolute differences between simulated and actual contaminant concentration at few observation locations. The actual concentration can be observed at the field. On the other hand, the simulated concentration can be obtained using aquifer simulation model. Therefore it is necessary to incorporate the aquifer simulation model with the optimization model. It can be viewed that for calculation of objective function value, it is necessary to simulate the aquifer process. Thus the methodology is known as simulation-optimization problem. The incorporation of aquifer simulation model with the optimization model makes the problem a very difficult one. The process is complex and time consuming too.

The aquifer simulation model basically solves the partial differential groundwater flow and transport equations. The equations are generally solved using numerical techniques as analytical solution exists only for simplified situations. The most popular numerical methods generally used to simulate the flow and transport processes in aquifer are the finite difference and finite element techniques. Some of the applications of finite difference and finite element techniques are Remson *et al.,* 1971; Wang and Anderson, 1982; Huyakorn and Pinder, 1983; Voss, 1984. The groundwater simulation model can be incorporated with the optimization model by using embedded optimization approach, response matrix approach and by using linked simulation optimization approach (Gorelick, 1983). In the embedded approach, the discretized finite difference or finite element equations are added as equality constraints to the optimization problem. The incorporation of the discretized equations as equality constraints to the optimization model is equivalent to the simulation of the aquifer process.The response matrix approach is based on the principle of superposition and linearity (Gorelick 1982). In this method, the unit responses are obtained using an external groundwater simulation model and the response matrix is used in the optimization model to calculate the objective function value. In the linked simulation-optimization approach, an external groundwater simulation model is incorporated with the optimization model. For calculation of objective and constraint functions value, the groundwater simulation model needs to call by the optimization problem.

An optimization algorithm is necessary to solve the inverse optimization problem. Classical gradient based optimization methods have been utilized by many researchers (Alley *et al.,* 1976; Willis, 1976; Willis and Newman, 1977; Gorelick *et al.*, 1983; Molx and Bell, 1977; Elango and Rouve, 1980; Peralta and Datta, 1990). Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al.,* 1991; Wang and Ahlfeld, 1994). Before the invention of fast computer system, it was difficult to solve the non-linear optimization problem. In such a situation, the objective and constraint functions were formulated as a linear function and the problem was solved using linear programming methods (Lee and Aronafsky, 1958; Gorelick *et al.*,1984; Neuman,1973; Yeh, 1975; Kleinecke, 1971; Hefez *et al*., 1975; Gorelick and Remson,1982). The linear method used for solving the groundwater management model as a series of constraints has been described by Aquado and Remson, 1974, 1980; Aquado *et al.*,1974, 1977; Alley *et al.,* 1976; Willis,1976; Remson and Gorelick, 1980; Molx and Bell, 1977; Elango and Rouve, 1980. With the development of fastest computer, non-linear optimization methods become more popular in solving the groundwater inverse problem (Gorelick *et al*., 1979, 1984; Colarullo *et al*., 1984; Ahlfeld *et al.*, 1988; Lefkoff and Gorelick, 1986; Finney *et al.,* 1992; Karatzas and Pinder, 1993; Mantoglou, (2003,2004); Rao *et al.,* 2004. Groundwater quantity management models based on embedding technique and nonlinear programming were proposed by many researchers including Alley *et al.,* 1976; Willis and Newman, 1977; Molx and Bell, 1977; Elango and Rouve, 1980; Peralta and Datta, 1990; Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al.,* 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2001; McPhee and Yeh, 2008; Datta *et al.,* 2009. Hence several researchers are working on embedding method though it has some limitations. The gradient based classical methods however have lots of limitations. Most of the time, it is difficult to calculate the gradient of the objective and constraint functions. Sometime, it is even the functions are non-differentiable. Further, classical methods are highly sensitive to the starting initial points and it is developed under the assumption of unimodality of the functions. In order to avoid the difficulties of the classical methods, researchers also used non-classical gradient less algorithms to solve the inverse problem (Ritzel *et al.*, 1994; Wang and Zheng, 1998; Aral *et al*., 2001; Hsiao and Chang, 2002; Singh and Datta, 2007).

The main objective of this chapter is to survey the chronological development occurred with respect to the solution of unknown groundwater pollution source identification models. The rest of the chapter has been organized as (1) source identification model using embedded approach, (2) source identification problem using response matrix approach, (3) source identification model using linked simulation-optimization approach, (4) source identification model using linear programming approach, (5) source identification model using non-linear programming approach, (6) source identification model using non-classical optimization techniques, (7) conclusions of the chapter.

2.2 Source identification model using embedded technique

As mentioned earlier, in the embedded approach, the finite difference or finite element approximation of the governing equations as embedded as equality constraints with the management model along with the other physical and managerial constraints. One of the initial applications of embedded approach was presented by Aguado and Remson (1974). They have solved groundwater flow management model using embedding optimization technique. During the same time, Futagamiet *et al. (*1976) also used embedded optimization approach to obtain the optimal discharges from the various types of outfall to meet water quality requirements, and also the distribution patterns of several water qualities in the water basin. They used finite element approximation of the convection-diffusion equation as constraints to the optimization model. Later, the embedding technique was used in steady state pollutant source management problems (Willis, 1976; Gorelick and Remson, 1982), and also in the transient pollutant source management problems (Willis, 1977; Willis, 1979). The embedded technique generally gives better result as it uses the nonlinear governing equations directly within the management model, however the dimension of the problem is considerably very large, even for a small study area. As a result the embedded technique needs more computer storage as well as more central processing unit (CPU) time. However, in case of large numbers pumping wells and for steady state management policies, the embedding technique requires less computer memory and processing time than the response matrix approach (Peralta and Datta, 1990). Moreover with the invention of high speed computers in the recent years, the requirement of large computational time and computer storage have become less constraining. Gorelick (1983) in his pioneer works mentioned that the embedded technique has numerical limitations for large scale aquifer systems, especially when aquifer properties are highly heterogeneous. Similar observation has also been made by Tung and Kolterman (1985). Another limitation of the approach is that reformulation of the optimization is necessary to any change in the governing equation or in the discretization scheme. Mahon *et al.* (1989) reported stability of the embedded approach on study area of 20,900 sq. km. In spite of these limitations, embedded approach has been applied by many researchers in solving many groundwater management problems (Alley *et al*., 1976; Willis and Newman, 1977; Molx and Bell, 1977; Elango and Rouve, 1980; Gorelick and Remson, 1982; Peralta and Datta, 1990; Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al.*, 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2001; McPhee and Yeh, 2008; Datta *et al.,* 2009).

Mahar and Datta (1997) applied embedded approach for identifying unknown groundwater pollution sources. They have developed a three steps optimization methodology for designing optimal monitoring network as well as for simultaneous identification of groundwater pollution sources. They have evaluated their model on small aquifer of 1.04 sq. kilometer. Mahar and Datta (2001) also formulated two different embedded optimization models for identification of pollution sources. These models incorporate both the flow and transport equations as equality constraints. In the first model the hydraulic head distribution is not specified but the flow and transport parameters are assumed to be known. Therefore, the discretized forms of the transport equation are nonlinear constraints in this model. The second optimization model is developed for simultaneous estimation of hydraulic conductivity, effective porosity, longitudinal dispersivity, and transverse dispersivity of aquifer parameters along with the identification of unknown pollution sources. This model is applicable to the case when hydraulic head distributions as well as the aquifer parameters are unknown. They have solved their model using large scale optimization solver MINOS (Murtagh and Saunders, 1993).

2.3 Source identification model using response matrix

Response matrix approach was one of the most popular methods for solving groundwater management problems. The approach is based on the principle of superposition and linearity. Thus the approach assumes that the groundwater aquifer is a linear system. In this approach, unit response of the system is obtained in the form of response matrix.The unit response describes the influence of unit simulator. The response matrix is constructed using an external aquifer simulation model and the matrix is to be computed before execute the optimization model. An analytical solution or a numerical simulation model is generally used for generating these response coefficients. As the method is applicable to linear system only, this approach is not suitable for large-scale groundwater problems which involve nonlinearities. This approach is useful when the system is approximately linear and boundary conditions are homogeneous. This approach gives erroneous results for highly nonlinear system (Rosenwald and Green, 1974). Gorelick, 1979 and Willis, 1979 have demonstrated the dynamic management of pollutant sources using response matrix approach. Gorelick (1982) used response matrix approach for solving groundwater management models. He constructed concentration response matrix using U.S. Geological Survey solute transport simulation model. The management model was applied to a complex hypothetical groundwater system. The optimization model was solved using linear programming problem. In the same year, Gorelick and Remson (1982) used the response matrix approach for solving pollution source identification model. They used a numerical model to simulate transient solute transport process in the aquifer and the model was used to develop the concentration response matrix. The objective of the problem was to obtain the set of potential sources which makes the simulated concentration at observation locations closer to the local groundwater solute concentration data. They used linear programming method to solve the identification problem.

Gorelick *et al.* (1983) applied response matrix approach for obtaining unknown groundwater pollution sources and magnitudes. They used least squares regression and linear programming technique to solve the optimization model. Datta *et al.* (1989) developed expert systems for identifying groundwater pollution sources incorporating optimal statistical pattern recognition techniques. They also used response matrix approach to incorporate the simulation model with the optimization model. They reported that the performance of the model was found efficient when applied to missing observation data. Skaggs and Kabala (1994) used response matrix approach to incorporate the simulation model with the optimization model. They used Tikhonov regularization to recover the release history of groundwater contaminant plume in a one-dimensional groundwater system. Wagner (1992) presented a source identification model with simultaneous estimation of aquifer parameters using response matrix approach. The have solved the optimization problem using nonlinear maximum likelihood estimation. Aral and Guan (1996) have used genetic algorithms and response matrix technique to identify sources of groundwater pollution. They have concluded that the results obtained by their model is more accurate than the results obtained by using linear programming technique. The response matrix approach is used by many researchers for solving variety of groundwater problem (Heidari, 1982; Willis and Liu, 1984; Peralta and Kowalski, 1986; Yazicigil *et al.* 1987; Galeati and Gambolati, 1988; Datta and Dhiman, 1996).

2.4 Source identification model using linked simulation optimization

The groundwater simulation model can also be externally linked to the management model (Emach and Yeh, 1998, Finney *et al.*, 1992, Bhattacharjya and Datta, 2009, Datta *et al.,* 2011). In this approach, the optimization model calls the simulation model when it requires any information from simulation model. Thus the search process of the optimization model is performed by iterating between simulation and optimization model. At every stage of iteration, the simulation model provides information to the optimization model to reach the optimal solution. The approach has several advantages as compared to the embedded and response matrix approaches and can be effectively applied to large groundwater problem. As the simulation model is used as a separate module, therefore modifications, if any in the simulator can be easily incorporated. The dimensionality of the problem also reduces significantly in this approach.The main drawback of this approach is that, it requires numerous repetitive iterations between the simulation and optimization models to arrive at an optimal solution. Further, the simulation of flow and transport processes in groundwater aquifer are highly nonlinear and time consuming, therefore large computational time is required to achieve the optimal solution.

Gorelick *et al*. (1984) developed a nonlinear simulation-optimization model to determine optimal design of reclamation schemes for contaminated groundwater systems. They combined finite element based groundwater flow and transport simulation model, SUTRA (Voss, 1984) with a nonlinear optimization procedure MINOS (Murtagh and Saunders, 1993) to solve the formulated problem. The developed methodology was applied for both steady and transient contaminant migration problems. Tsai *et al*. (2003) proposed two global-local optimization methods by using linked simulation-optimization approach for parameter structure identification in three dimensions, the sequential global-local optimization and the hybrid global-local optimization. They formulated a sequential global-local optimization scheme that includes a genetic algorithm, a quasi-Newton method and local search is developed to minimize the fitting residual. Then, a hybrid global-local method of hybrid genetic algorithm is developed to calculate the parameter structure error. Aral *et al.* (2001) formulated a linked simulation optimization model for identification of unknown groundwater contaminant source location and release history. They considered simple as well as complex scenarios. The results were promising even with large measurement errors. In this study a new combinatorial approach is introduced, which is identified as progressive genetic algorithm and proposed for solution of the nonlinear optimization model. They showed numerically that the proposed method is a robust tool for solution of source identification or other similar highly nonlinear optimization problem (Aral and Guan, 1996; Guan and Aral, 1999).

Datta and Chakrabarty (2003) formulated a methodology by using a classical optimization-based linked simulation-optimization approach, in which the flow and transport simulation models are externally linked to the optimization model to solve the source identification problem. This approach can be used to solve large-scale identification problems, although the linking procedure is extensive and computationally complex especially when gradient-based optimization technique is utilized. Singh and Datta (2006) proposed a GA based linked simulation optimization model, which is also referred as global optimization method. The performance of the developed methodology is demonstrated for combinations of various potential source characteristics, locations, magnitudes, and release periods. GA is used for solving the optimization model in this study. They concluded that search algorithm of GA is in many ways similar to directed random search procedure. This similarity is very useful in formulating and solving a linked optimization-simulation model. Unlike other classical optimization techniques, such linking does not require complex intermediate steps, like computation of the Jacobian matrix. Therefore, they used GA to solve the linked simulation-optimization model. With the increase in the number of potential sources, the complexity of the source-identification problem increases. Comparatively higher identification errors are observed with erroneous concentration measurements. The main advantage of the proposed methodology is the possibility of linking the GA-based optimization model with an external flow and transport simulation model. Datta *et al.* (2009) presented an improved methodology over the methodology suggested by Mahar and Datta (1997). The developed methodology combines both a linked simulation-optimization model using an external independent simulator for source identification, and an integer programming based optimization model for dynamic optimal monitoring network design. The potential applicability of the methodology is demonstrated using an illustrative study area. The performance of the methodology using spatiotemporal pollutant concentration measurements are evaluated by solving two illustrative study areas. They reported that the methodology is computationally more efficient than the embedding approach. Ayvaz (2010) presented a pollution source identification model using linked-simulation optimization approach based on heuristic harmony search (HS) algorithm. In their model the locations and release histories of the pollution sources are treated as the explicit decision variables and determined through the optimization model. The proposed model is applied to one simple and one complicated study areas with different geometrics, measurement errors and different solution parameter sets. Evaluation results show that the model can be applied to solve inverse pollution source identification problem. Datta *et al.* (2011) demonstrated the successful use of classical nonlinear optimization algorithm to solve the unknown groundwater pollution source characterization problem using linked simulation-optimization approach. Two optimization models, OSIM1 and OSIM2 are developed. The computational efficiency of these models is evaluated using two linked simulation-Optimization method using nonlinear optimization algorithms. The proposed methodology is potentially applicable to large scale study areas and does not have the computational limitations that have been faced by the embedded approach. Performance of the proposed source identification model using spatiotemporal pollutant concentration measurements are evaluated by solving illustrative problems. They formulated two different optimization models and the relative importance of the model formulations is demonstrated in terms of computational efficiency. With the limited performance evaluation, they have reported that the proposed models have potential applicability to solve a problem of fairly large study area with multiple unknown pollution sources. Ayaz *et al.* (2014) developed an ANN based linked-simulation optimization model for complete identification of unknown groundwater pollution sources. The model contains two parts as optimization model and an ANN model. Using the spatial and temporal data, they formulated an objective function which minimized the observed and simulated concentration to identify the unknown groundwater pollution sources. In the formulation of the objective function, they used lag time which is not known. The main advantage of the study is that it requires only upper half of the breakthrough curve and is capable of predicting source parameter when the time is not known.

2.5 Source identification model using linear programming

The Linear programming technique has been extensively used by many researchers for solving groundwater management models due to its easy formulation and application. The linear programming technique was mostly developed during the World War II and it was used for military planning purpose. It has been reported that linear programming technique was used for groundwater management in the year 1958. After this publication, linear programming technique was used in many groundwater management analysis. Most of the groundwater management models presented prior to 1983 represented only linear systems. Neuman (1973) formulated a linear programming problem coupled with a finite element model of steady state groundwater flow to determine a discrete set of alternative solutions. Yeh (1975) conducted a study for comparing various methods of source identification problem including linear programming for a simple one dimensional problem. Some other significant application of linear programming programme are: Kleinecke, 1971; Neuman,1973; Frind and Pinder, 1973; Hefez *et al.,*1975; Gorelick and Remson, 1982; Aquado and Remson, 1974, 1980; Aquado *et al.,* 1974, 1977; Alley *et al.,* 1976; Willis, 1976; Remson and Gorelick, 1980, 1982; Molx and Bell, 1977; Elango and Rouve, 1980; Futagami *et al*., 1976; Wills, 1976; Gorelick and Remson, 1982, *etc*.

One of the first applications of linear programming problem on identification of unknown groundwater pollution sources was Gorelick *et al*., 1983. They presented two approaches for identifying groundwater pollutant source locations and magnitude. They combined groundwater solute transport simulation model with least square regression and linear programming to identify locations and magnitudes of unknown aquifer pollutant sources. The objective function minimizes deviation between simulated and measured non reacting solute concentration data. The model was applied to a hypothetical area for both steady and transient conditions. The steady state model identified unknown pipe leak and magnitudes. The transient model identified several annual disposal fluxes in the aquifer based on concentration histories collected at observation wells.

Apart from groundwater management models, linear programming has also been applied to the problem of groundwater parameter identification. Kleinecke (1971) used linear programming formulation and attempted to determine both the transmissity and storage coefficient by minimizing the difference between predicted and observed groundwater heads over time. Becker and Yeh (1972) as well as Marino and Yeh (1973) used least square optimization for solution of parameter identification problem in unsteady open channel flow and finite leaky aquifer system respectively. They found satisfactory results in both the situations. Fried and Pinder (1973) developed a model using Galerkin finite element method for solving the inverse problem for aquifer transmissivity. Neuman and Yakowitz (1979) proposed a statistical approach to the inverse problem of parameter estimation. They explained the transmissivity determination from pumping tests or other measurements at the specific locations in the aquifer. Wagner and Gorelick (1986) presented multiple regression techniques to estimate aquifer parameters, and a linear source for a one dimensional hypothetical column system. They mentioned that estimation of linear source term was found to be highly sensitive. Karatzas and Pinder (1993) formulated a concave minimization problem using linear programming. Konikow and Bredehoeft (1978) used the solute transport simulation model MOC (Konikow and Bredehoeft, 1978) to compute contaminant distribution in post optimal stage. Sidauruk *et al.* (1998) presented an inverse model using correlation coefficient optimization methodology to identify the groundwater contaminant sources and parameters. They developed the models for instantaneous and continuous point sources by minimizing the correlation coefficient of the linear regression. Zheng and Wang (1999) presented a method that integrated Tabu search with linear programming. Tokgoz *et al.* (2002) used linear and mixed-integer programming for optimal dewatering schemes for excavation of collector line. A successive linear programming (SLP) for groundwater flow management problem was used by Ahlfeld *et al.* (2008) in unconfined aquifer. Some of the researchers used linear programming technique for groundwater management model in coastal aquifer (Tracy, 1998; Nishikawa, 1998; Azaiez and Hariga, 2001; Sethi *et al*., 2002; Karterakis *et al*., 2007).

2.6 Source identification model using non-linear programming

The groundwater pollution management problems or the contaminant source identification problems are basically nonlinear in nature. As such, the problem has been converted to linear one to make it suitable to solve using linear programming techniques. Therefore for solving groundwater quality management models, it is necessary to determine the groundwater flow field prior to the solution of the management model (Gorelick *et al.,* 1984). With the development of high speed computer, nonlinear techniques have been used by various researchers for management of groundwater aquifer problems (Gorelick *et al*., 1979, 1984; Colarullo *et al.*, 1984; Ahlfeld *et al.*, 1988; Lefkoff and Gorelick, 1986; Finney *et al*., 1992; Karatzas and Pinder, 1993; Mantoglou, 2003; Rao *et al*., 2004). Aguado and Remson (1974) were the first to present groundwater flow management models using embedding technique. The identification of coupled source location and release history is also a complex nonlinear optimization problem. The nonlinear programming techniques were also used by many researchers including Alley *et al.*, 1976; Willis and Newman, 1977; Molx and Bell, 1977; Elango and Rouve, 1980; Peralta and Datta, 1990; Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al*., 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2001; McPhee and Yeh, 2008; Datta *et al*., 2009; Gorelick and Remson, 1982; Willis, 1976; Willis, 1979. One of the most notable contributions is the application of embedded optimization technique for solving groundwater management problems. The embedded technique generally gives better result as it uses the nonlinear governing equations directly within the management model, but dimension of the problem is considerably large, even for a small study area. The embedded technique needs more computer storage as well as central processing unit time. Moreover, embedded technique has numerical limitations for large scale aquifers systems, especially with considerable heterogeneity in aquifer properties.

The classical non-linear methods need the calculation of Jacobian matrix and most of the time it is necessary to calculate the Jacobian matrix using numerical method. Often the numerical calculation of the Jacobian matrix is the most time consuming part of a large scale nonlinear optimization model. Ahlfeld *et al.* (1986) realized that numerical calculation of Jacobian is computationally very expensive. They presented a method for improving solution efficiency of nonlinear programming by estimating the Jacobian analytically. The Jacobian was estimated by using the adjoin sensitivity method. The applicability of the method was demonstrated using illustrative study area. Wagner and Gorelick (1986) presented nonlinear multiple-regression methodology for estimating the parameters which characterize the transport of contaminants in aquifers and streams. They used least square regression procedure for estimating transport parameters. They also demonstrated the importance of statistical methods. Gailey and Gorelick (1993) presented a groundwater quality management model to examine performance of various single well pumping schemes for capture of dissolve contaminants. They linked the groundwater simulation model to a nonlinear optimization code NPSOL (Gill *et al*., 1986). Wagner (1992) used nonlinear maximum likelihood estimation combined with groundwater flow and contaminant transport simulation to solve the source identification problem. The goal of this study was to identify the plume contaminant and also to minimize the expected cost of plume contaminant. They examined four different model formulations, ranging from simple deterministic optimization based on the expected values of the uncertain parameters to full stochastic programming with recourse. The optimal designs obtained using the deterministic approach is significantly more expensive. These designs incur significant recourse costs because they do not account for the heterogeneity and uncertainty of the hydraulic conductivity field. Mahar and Datta (2000) also used a nonlinear optimization to estimate the magnitude, locations, and duration of groundwater pollution sources under transient conditions. The optimization models for identification of sources presented in this study require the measured values of concentrations and hydraulic heads at specified locations. The data obtained from arbitrarily located existing wells may not be sufficient enough to result in reasonably good and efficient identification of sources. In this study, an optimal monitoring network design model and a source identification model is developed and also tested the utility of the developed methodology.

The groundwater pollution source identification problem has been proposed by Meyer and Brill 1988; Mahar and Datta 1997, 2000, 2001; Michalak and Kitanidis, 2004; Datta *et al.*, 2009. Mahar and Datta (1997, 2000, and 2001) used embedding technique with nonlinear optimization for unknown groundwater pollution sources. An important issue for real world situation is the identification of unknown groundwater pollution source which is closely related with parameter estimations. The identification of groundwater pollution source is the difficult and challenging task for scientist and water resource manager. The groundwater quality monitoring network design is also important part in groundwater management model because it is difficult to identify the pollution sources without a scientifically designed efficient monitoring network. To know the groundwater quality, we have to study the physical, chemical and biological properties of groundwater and for monitoring network design, it is essential to know the selection of sampling sites and sampling frequency.

Lots of work has been carried out on pollution source identification problem and quality monitoring network design. The last three decades the groundwater quality monitoring has increased significantly and the researchers have proposed different monitoring network design methodologies. The major contribution of the researchers Bagtzoglou *et al.*, 1992; Wagner, 1992; Skaggs and Kabala, 1994; Aral and Guan, 1996; Mahar and Datta, 1997, 2000, 2001; Aral *et al.*, 2001; Atmadja and Bagtzoglou, 2001; Singh *et al.*, 2002; Michalak and Kitanidis, 2004; Singh and Datta, 2006, 2007; Chadalavada and Datta, 2008; Datta *et al*., 2009; Dhar and Datta, 2009. Bagtzoglou *et al.* (1992) developed random walk based backward tracking model which provides crucial information for design of monitoring and data collection network in support of groundwater pollution source identification efforts.

Willis and Finney (1985) presented a transient nonlinear management model for an unconfined aquifer system using quasi-linearization and optimal control theory. The main objective of the management model was to identify optimal pumping pattern necessary to satisfy as exogenous water demand. The formulated nonlinear optimization model was solved using a quasilinearization based optimization algorithm and by using MINOS. The results obtained from quasilinearization based optimization algorithm were identical to the result obtained by using MINOS. Ahlfeld *et al.* (1986) developed a nonlinear simulation-optimization model to determine optimal strategies for contaminated groundwater remediation. The nonlinear programming algorithms available in MINOS were used to solve the designed problem. The management model was solved for both steady and transient conditions. Datta *et al.* (2009) describe to identify optimal dynamic monitoring network design and identification of unknown groundwater pollution sources. The methodology consists of three distinct steps. The preliminary identification of unknown sources, based on limited concentration data from existing arbitrarily located wells provides the initial rough estimate of the sources fluxes. Recently Amirabdollahian and Datta (2013) presented a comprehensive review on the contaminant source identification models. They presented reviews of source identification and monitoring network design methodologies. An integration of source identification and monitoring network design is also discussed. Lastly, some of the issues which need future attention in this area are presented.

2.7 Source identification models using non-classical methods

The algorithms such as linear programming, non-linear programming, quasi-linear programming, integer programming, stochastic programming *etc.* are used to solve the source identification problems. But the computational time required to obtain the desired accuracy may be enormous and solution results may not be as expected. Therefore some researchers used nontraditional optimization algorithms such as genetic algorithms (GA), artificial neural network (ANN), simulated annealing (SA), particle swarm algorithms (PSA), differential evolution (DE), ant colony, *etc*. to reduce the computational time and also improved efficiency of the model. These optimization methods are conceptually different from the traditional programming techniques. These methods are based on certain biological, molecular, and neurological phenomena. The genetic algorithms are based on the principles of natural genetics and natural selection. In neural network based methods the problem is modeled as a network consisting of several neurons and the network is trained suitably using optimization algorithms. Simulated annealing is based on the simulation of thermal annealing of critically heated solids. The simulated annealing is a stochastic method that can find the global minimum with a high probability and are naturally applicable for the solution of discrete optimization problems.

Genetic algorithm (GA) has been used for solving groundwater related problems. Classical optimization techniques, *e.g.* linear programming, nonlinear programming, mixed integer programming, were extensively used for groundwater management models by many researchers. The main disadvantage with most of classical methods is their dependence on gradient search technique. Most of the time, these gradients are calculated numerically. The numerical estimation of gradients is often the most time consuming part of an optimization. Moreover, numerical calculation of gradients sometime may lead to severe errors. Another disadvantage of classical optimization techniques is that many times it is inefficient in avoiding local optimal solutions, especially when the optimization problem is non-convex and response surface is highly irregular. The other limitations of classical methods are: point-to-point search, necessity of initial guesses, deterministic transition rule, assumption of unimodality, *etc.* (Deb, 2001). Instead of this, some researchers used genetic algorithms (GA) which is becomes a powerful and robust tool for function optimization.The study of genetic algorithms (GA) was originated in mid 1970's by John H. Holland, his colleagues, and his students at the University of Michigan. These algorithms are computationally simple but powerful in their search for improvement after each generation (Goldberg, 1989). Genetic algorithms (GA) are novel heuristic search techniques based on the mechanics of natural selection and natural genetics, which combines artificial survival of the fittest concept with genetic operations abstracted from nature (Holland, 1975). The basic techniques of GA are designed to simulate mechanism of population genetics and natural rules of survival in pursuit of the ideas of adaptation. One of the great advantages of GA is that it does not require differentiability of either objective functions or constraint functions (Goldberg, 1989). The constraints handling capacity of GA is also better than classical optimization techniques because of population based approach (Deb, 2001). GA is referred to as global optimization method (Deb, 2001). The heuristic search technique, Genetic algorithm (GA) may be used as a tool for solving the optimum management model, because of its relative efficiency in identifying global optimal solutions especially for nonlinear non-convex problems. The GA-based optimization approach is also suitable for externally linking the numerical simulation model within the optimization model. As the iteration between simulation model and optimization is very large in this case, the computational time requirement will be depended on the type of aquifer simulation model. The technique has been used for many fields after its introduction by Holland in 1975 such as, groundwater management model, pipe network optimization, groundwater parameter determination, seawater intrusion management model, aquifer clean up time, groundwater resource planning problem (Hsiao and Chang, 2002), optimal identification of unknown groundwater pollution sources (Singh and Datta, 2007), *etc.* Morshed and Kaluarachchi (2000) reviewed applications of genetic algorithms in solving groundwater optimization problems. They explore three methodologies to enhance GA which are fitness reduction method (FRM), search bound sampling method (SBSM), and optimum resources allocation guideline (ORGA). They solve a nonlinear groundwater problem using genetic algorithms coupled with the enhancement methods. The results obtained by genetic algorithms were compared with the results obtained by gradient based nonlinear programming technique. In addition, they also studied sensitivity of the enhancement methods to various genetic algorithms parameters. They concluded that FRM enhances efficiency of genetic algorithm in handing constraints, SBSM enhances accuracy of genetic algorithms in solving problem with fixed costs, and the ORGA enhances the reliability of genetic algorithm by providing some convergence guarantee for a given computational resources. The accuracy of the genetic algorithms increases marginally from near-optimal to global-optimal when FRM and SBSM are applied together.

Ritzel *et al.*, (1994) applied genetic algorithms (GA) to solve two objectives steady state groundwater pollution contaminant problem. The two objectives were: maximize reliability and minimize cost of hydraulic contaminant system. They have used both vector-evaluated GA (VEGA) and Pareto GA algorithm to solve the multiple objective problems. Wang and Zheng (1998) coupled the flow model and transport model with GA and applied to hypothetical study area for groundwater remediation. They coupled the groundwater flow model (MODFLOW) and the solute transport model (MT3DMS) with GA and applied to a hypothetical and a field-scale problem for groundwater remediation. They used pump-and-treat method, which involves installing and operating a set of extraction wells so that the contaminated ground water is hydraulically contained and can be pumped out for subsequent treatment. Aral *et al.* (2001) formulated a methodology for contaminant source identification problem as a nonlinear optimization method. They presented a GA based methodology which is naming as progressive genetic algorithms (PGA) for identifying the location, sources and release history of the aquifer. They demonstrated this methodology for relatively smaller aquifer. In this methodology, PGA combined the groundwater simulation model with GA to transfer nonlinear optimization model into a series of approximate optimization problem. Singh and Datta (2006) used genetic algorithm based simulation-optimization approach for optimal identification of unknown groundwater pollution sources. Flow transport simulation model is externally linked to the GA-based optimization model. The performance evaluation results presented for the three different scenarios. The concentration measurement data are simulated by using the numerical simulation model MOC. These concentration values are determined for different locations and time, specified boundary conditions, pumping and recharge rates, and parameter values. These simulated spatial and temporal pollutant concentration measurement values are used to evaluate the fitness function value of the GA. Singh and Datta (2007) used genetic algorithm based simulation-optimization approach for optimal identification of unknown groundwater pollution sources. In this study, the pattern matching capability of an ANN is exploited to identify unknown sources of pollution from partial breakthrough curves. Data for training the ANN are simulated using a groundwater flow and contaminate transport numerical simulation model. The trained network was then utilized for the identification of pollution sources for specified concentration observation data at the given locations. Yeh *et al.* (2006) used factorial kriging and genetic algorithm for optimal monitoring network design to get sufficient and non-redundant information of monitoring variables. Sreekanth and Datta (2011) have shown the potential applicability of genetic programming (GP) in groundwater problems as surrogate models.

Artificial neural network (ANN) has been used for solving pollution source identification problems. An artificial neural network, which is considered as a universal approximator, mimics the function of human brain by acquiring knowledge through learning process. Mathematically, an ANN is often viewed as a universal approximator (ASCE Task Committee, 2000; Zurada, 1999). Since the early nineties, ANN have been successfully used in hydrology-related areas such as rainfall-runoff modeling, stream flow forecasting, groundwater modeling system, water quality, water management, water management policy, precipitation forecasting, hydrologic time series and reservoir operations (ASCE Task Committee, 2000). A neural network is characterized by its architecture that represents the pattern of connection between nodes, its method of determining is the connection weights, and the activation function (Fausett, 1994). Some patterns are very complex in nature. The learning process involves finding of an optimal set of weights for synaptic connections between artificial neurons of the networks. The ability to gather knowledge through process of learning, like a human brain, from sufficient input patterns makes it possible to apply ANN to solve large scale real world problems. Training is a process of determining the synaptic weights of an ANN networks for getting desired output. There are two types of training processes: supervised and unsupervised training. Supervised training estimates the synaptic weights of the network based on specified input and output patterns. In supervised training, both input and output data are required. On the other hand, in unsupervised training, the networks perform classification of the inputs without using any specified target patterns. Once ANN is trained, the relationship between input and output is encoded in the network. Then it can be used to predict output based on the information fed to the input nodes. The flow and transport processes in groundwater can be approximate by Regression analysis and artificial neural network (ANNs) model. The ANN model is considered as a better approximator than regression analysis. ANN is considered to have magical problem solving capabilities in approximating input-output responses (Beale and Jackson, 1991). It is observed from the literature review that ANNs may provide better alternative to the conventional numerical computational techniques in groundwater problems. Main advantages of ANNs can be stated as ease in application, reduced data requirement in some cases lesser computational burden and improved result accuracy than the conventional methods (Jain and Deo, 2004). But the success of ANN application depends both on the quality and quantity of data available (ASCE, 2000). A number of studies used ANN as surrogates for groundwater models in coupled simulation-optimization (Ranjithan *et al.*, 1993; Aly and Peralta, 1999; Rao *et al.*, 2004; Yan and Minsker, 2006; Bhattacharjya and Datta, 2009; Kourakos and Mantoglou, 2009; Dhar and Datta, 2009). Ranjithan *et al.* (1993) presented a neural network based screening tool for identifying critical realization from a large set of hydraulic conductivity realization. Some of the researcher used regression analysis for approximating an aquifer process. Alley (1986) and Lefkoff and Gorelick (1990) used the regression analysis for approximating an aquifer processes within the management model. Singh and Datta (2004) formulated an ANN-based methodology to solve the groundwater pollution sources and the corresponding parameter estimation. The performance of the trained and selected ANN models was evaluated primarily for the idealized case of error free concentration measurement and then for the more realistic case of erroneous concentration measurement data. They concluded that the performance of the ANN model was reasonably well even with large concentration measurement errors. Singh and Datta (2004) presented a methodology for characterizing of unknown pollution sources using ANNs. The data required for training are simulated using numerical groundwater flow and contaminant transport simulation model. The trained network was then used for identification of pollution sources for specified concentration observation data at given locations.They have reported that if the number of potential sources are increased, the complexity of the problem increases and error are also very large when large concentration measurement error are incorporated together with multiple source locations. Singh and Datta (2007) developed an ANN based methodology to identify unknown groundwater pollution sources when concentration observation are missing. The groundwater flow and transport numerical simulation model is utilized to generate the necessary patterns for training the ANN. Performance evaluation results show that the back-propagation based ANN model is essentially capable of extracting hidden relationship between patterns of available concentration measurement values and the corresponding sources characteristics. Srivastava and Singh (2014) presented a methodology for identification of unknown pollution sources in complex real groundwater problem. The developed methodology utilized a flow and transport simulation model to simulate aquifer response in terms of concentration measurements corresponding to specified source fluxes. In this study, uniform random generation and Latin hypercube sampling (LHS) method of random generation are used to generate source fluxes. These source fluxes are used in groundwater flow and transport simulation model to generate necessary data for ANN model building processes. They reported that among two processes, LHS technique was found more promising.

Several researchers have attempted to develop pollution source identification problem by using different methodologies. One of the methods for pollution source identification is the run forward simulation and checks the solution with the available observed data. Bagtzoglou *et al.* (1992) presented Advective-Dispersion Equation (ADE) backward in time using random walk particle methods. They presented the model in such a way that the advective part of the model is reversed while the dispersive part is left unchanged in heterogeneous media. They applied the methodology to the flow field and observed using geostatistical techniques that the values of standard deviation for the probability distribution function (PDF) distribution are extremely high relative to the actual pdf values. In the recent literatures it is observed that researchers are much more interested in Tikhonov regularization (TR) and minimum relative entropy inversion (MRE) methodologies. Skaggs and Kabala (1994) recovered the release history using Tikhonov regularization method in a one-dimensional homogeneous system. They observed the effect of measurement errors, parameter estimation errors, and numerical instability on the performance of their models using high noise levels and found that moderate noise level data gave good results as compared to high noise level data. After that Skaggs and Kabala (1995) presented a quasi-reversible (QR) solution of the convection-dispersion equation to identify release history of the contaminated plume. They found that QR methodology is more accurate to identify plume’s recent evolution history than the plume’s release history. It is observed that QR methodology required less computational effort than Tikhnov regulation technique. Liu and Ball (1999) presented a model to estimate contaminant plume history from observed contaminant concentrations of a field-scale groundwater remediation experiment at Dover Air Force Base. They assumed that the contaminant source history to be a function of time with unknown form and the results provided the important information regarding the contaminant release history at sites of groundwater contamination and cleanup. Neupauer *et al.* (2000) developed a model for recovering the release history of the conservative contaminant in a one-dimensional groundwater system. They evaluated the relative effectiveness of Tikhonov regularization and minimum relative entropy inversion using hypothetical release history function and contaminated plume. Milnes and Perrochet (2007) presented a theoretical framework to identify single point source location for heterogeneous multidimensional systems using the concept of mass transfer theory. In this study forward and backward flow field is applied, yielding complementary information as to forward and backward advective-dispersive mass fraction. The methodology is applied to 2D heterogeneous example of ideal conditions assuming the entire pollution plume is known. Again the model is applied to more realistic case where only few observation points exist.

Michalak and Kitanidis (2004) developed a geostatistical approach for recovery of antecedent distribution of contaminant at a given point back in time. In this study geostatistical approach combined with adjoin state method to recover the release history in heterogeneous media. The geostatistical approach have addressed by lot of researchers for estimating the release history of the groundwater pollution (Neupaner *et al.*, 2000; Snodgrass and Kitanidis, 1997; Michalak and Kitanidis, 2002; Michalak and Kitanidis, 2003). Sun *et al.* (2006) used constrained robust least square approach (CRLS) for source release history and the global optimization solver is used for location search. Sun *et al.* (2006) showed that CRLS gave better performance than its non-robust estimator *i.e.* non negative least square estimate (NNLS).

2.8 Summary and critical appraisal of literature review

The reviewed literature indicates that lot of works have been carried out on pollution source identification problem worldwide. Several methods have been adopted for optimal identification of unknown groundwater pollution sources using pollutant concentration and other measurement data. It has been observed that different methodologies have been developed to incorporate the simulation model with optimization model. However, application of some methods, mainly response matrix approach and embedded approach are limited to small aquifer system only. Linked simulation-optimization methodology has emerged as one of the most potential methods to incorporate the simulation model with the optimization model. The method can be applied to large scale groundwater aquifer system. It has also been reported that linked simulation optimization model is computationally less efficient than the other methods. Some researchers also used approximate simulation models to reduce the computational time. However, incorporation of approximate simulation model has reduced the predicting efficiency of the model. As such there is a need to develop new methodology in order to achieve efficiency both in quality of the solution and in computational time. In the successive chapters, effort has been given to develop new methodologies to overcome the present difficulties.

Chapter 3

Development of GMS Based Simulation-Optimization Models for Identification of Unknown Groundwater Pollution Sources

3. General

The literature review done in the previous chapter reveals that, lot of works have been carried out worldwide using different approaches to identify unknown groundwater pollution sources. Out of these approaches, linked simulation optimization approach has been immersed as one of the most promising algorithm for solving groundwater simulation-optimization model. The performance of the simulation-optimization model is highly related to the groundwater simulation model used in the model. Used of highly sophisticated numerical simulation model would generally produce better prediction of the unknown groundwater pollution sources. In this chapter, we have developed an improved methodology by linking user friendly commercial groundwater simulation software groundwater modeling system (GMS) with the Matlab-based optimization model. The efficiency and field applicability of the model is demonstrated using two illustrative study areas.

3.1 Methodology Development

3.1.1. Development of simulation-optimization methodology

The unknown groundwater pollution sources can be identified using inverse optimization technique. For finding the unknown pollution sources, the inverse optimization model minimizes the difference between simulated and the observed concentration of contaminants at observation wells. The error function can be written as,

$$Minimize\ \ err = \sum_{t=1}^{T}\sum_{n=1}^{N}(OC_t^n - SC_t^n)^2 \quad (3.1)$$

The err is a function of magnitude of pollution sources. OC_t^n is the observed concentration at n[th] well location for t^{th} time steps; SC_t^n is the simulated concentration at n[th] observed location for t^{th} time steps; N is total number of observation wells and T is the total time steps.

The observed concentration is the concentration measured at the observation wells. The simulated concentration can be obtained using the aquifer simulation model, which basically solves the governing partial differential flow and transport equations. As such, the aquifer simulation model has to be executed for calculating the objective function value. This can be done by linking the aquifer simulation model with the optimization model. As simulation model is linked with the optimization model, the methodology is known as simulation-optimization model. Various software are available for simulation of aquifer process. Some of them are GMS, SUTRA, FEFLOW, *etc*. We have used groundwater simulation system (GMS) to simulate the flow and transport processes in aquifer. The simulation-optimization methodology can be explained using the schematic diagram presented in Fig. 3.1.

The algorithm starts with the supply of initial guesses of contaminants at probable source

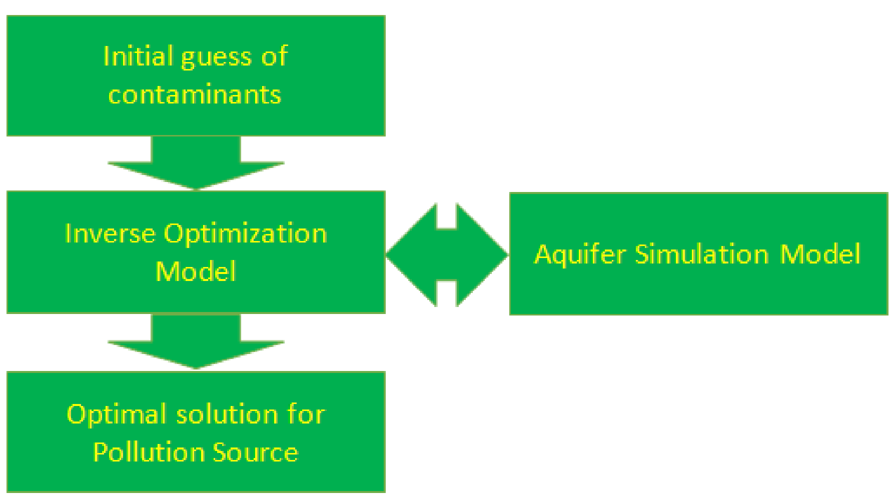


Fig. 3.1: Schematic representation of Simulation-Optimization methodology

locations. The source values are supplied to the inverse optimization model. For calculating objective function value for the given sources, the inverse model calls the aquifer simulation model. The source values are then updated by the inverse model. The iterations between optimization model and aquifer simulation model continue till termination criteria are not satisfied.

An optimization algorithm is necessary for solving the inverse optimization model. Gradient based classical optimization techniques have been implemented by various researchers (Alley *et al.,* 1976; Willis, 1976; Willis and Newman, 1977; Gorelick and Remson, 1982; Gorelick *et al.*, 1983; Molx and Bell, 1977; Elango and Rouve, 1980; Peralta and Datta, 1990; Yazicigil and Rasheeduddin, 1987; Datta and Peralta, 1986; Peralta *et al*., 1991; Wang and Ahlfeld, 1994; Mahar and Datta, 1997; Mahar and Datta, 2001; McPhee and Yeh, 2008; Datta *et al.,* 2009). However, the algorithms are not suitable for non-linear non-convex problem. For such problems, the classical methods yield only the local optimal solution of the problem. The local optimal solution can be avoided by running the model several times using different initial solutions. However, this is a time consuming approach and also does not give any guarantee that the global optimal solution will be obtained. The problem of classical optimization algorithms can be overcome using gradient less not classical global search methods. The most popular gradient search algorithms are direct search method, genetic algorithms, differential evolution technique, particle swarm algorithm, ant colony algorithm, *etc*. In this study, we have initially used direct search (DS) method and genetic algorithms (GA) for solving the inverse optimization problem. The efficiency of these two algorithms has been evaluated using illustrative study areas.

Direct search optimization method does not use the derivative information of either the objective function or the constraint functions. The value of the objective function and the constraint functions are only used to guide the search process. In every iteration, the algorithm searches a set of points in and around the current best solution. The algorithm starts with an initial solution supplied by the user. A set of solutions is then generated around the initial solutions. Fitness value which represents the goodness of a solution is then calculated for each generated solution. For calculating the fitness values, the algorithms has to call the groundwater simulation software GMS. As such the GMS has to be linked with the direct search algorithm.

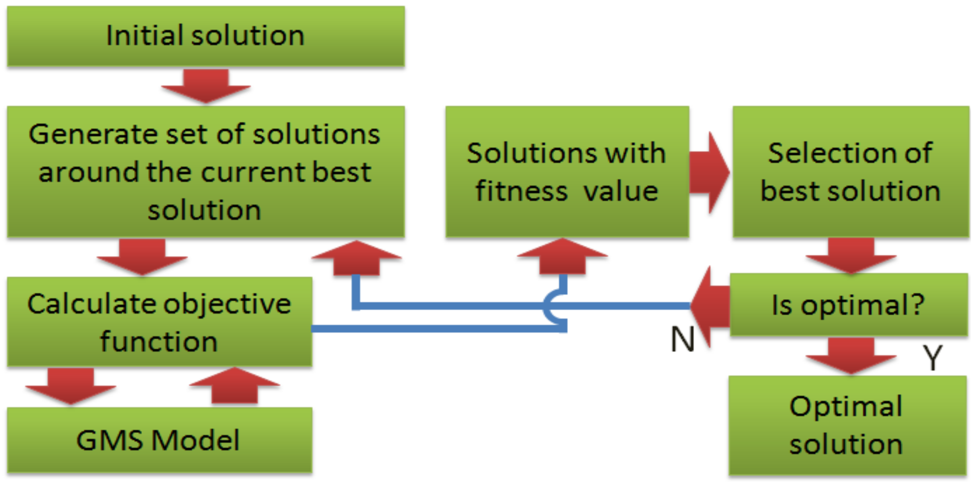


Fig. 3.2: Flow chart of GMS-DS algorithm

The methodology can be represented as GMS-DS methodology. Fig. 3.2 shows the flowchart of the GMS-DS methodology.

Similar to Direct search method, Genetic algorithms is also a gradient less search algorithm. It is considered as one of the most popular and robust non-classical optimization algorithm to handle the non-linear non-convex problem. The algorithm is based on the principle of natural evolution of species and is guided by the survival of fittest principle of nature. The algorithm is a population based algorithm. Unlike the classical optimization methods, the algorithm starts with a set of solutions, called population. The population passes though the natural operators, i.e. selection, crossover and mutation to search for better individual. The iterative process, which is known a generation continues till termination criteria are not satisfied. Genetic algorithm can be applied for solving the inverse source identification model. For finding the optimal source locations and magnitude, the groundwater simulation models have to be incorporated with the genetic algorithm. The simulation models provide the information needed for calculating the fitness of all individual population. The GA based simulation-optimization algorithm starts with the generation of initial population. The initial population is generated randomly between specified upper and lower bounds. The populations are then sent to the objective function sub-routine for calculation of objective function value of each individual of the population. The next step is to check the termination criteria. If it is satisfied, the iteration will be stopped and optimal solution will be displayed. Otherwise, the population will pass through the three genetic operators, i.e. reproduction, crossover and mutation. We may also use elitism operator in this step. The new populations generated by the genetic operators are then sent to the objective function sub-routine for calculation of objective function value. The iteration process will continue till termination criteria are not satisfied. This method uses GMS to simulation the groundwater aquifer. The methodology can be described using the flowchart given in Fig. 3.3. As such the approach can be named as GMS-GA model. The performance of the model is evaluated using two illustrative study areas. The integration of GMS software with GA based optimization model enables us to solve very complicated real world source identification problem as GMS has the capability to simulate more complex groundwater aquifer system.

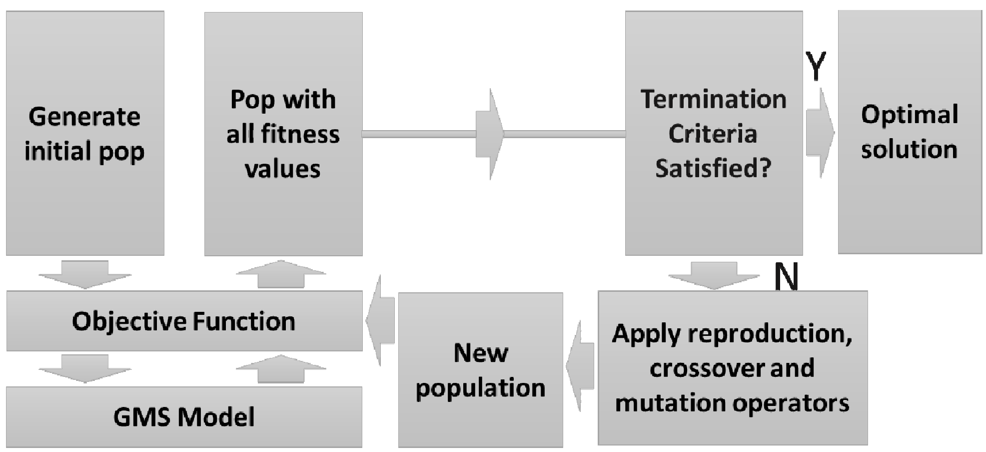


Fig. 3.3: Flow chart of GMS-GA algorithm

3.1.2 Development of simulation model using GMS

As mentioned, GMS has been used to simulate the flow and transport processes in the aquifer. GMS has a GIS based graphical user interface to prepare the input files required for the MODFLOW and MT3DMS simulation. Initially, we have used GMS for preparing the input files for the illustrative study areas considered by in this study. Once the input files are ready, the simulation can be performed by executive the *exe* file of MODFLOW and MT3DMS. GMS has the option to run the *exe* files and also has the option to display the simulation results, *i.e.* post processing of the results. The *exe* files can also be run by using other platforms. In this study, we have used Matlab to run the *exe* files and also for post processing the output data. Further, the input files, specifically the source values can be modified easily using Matlab code. The process of aquifer simulation implemented in

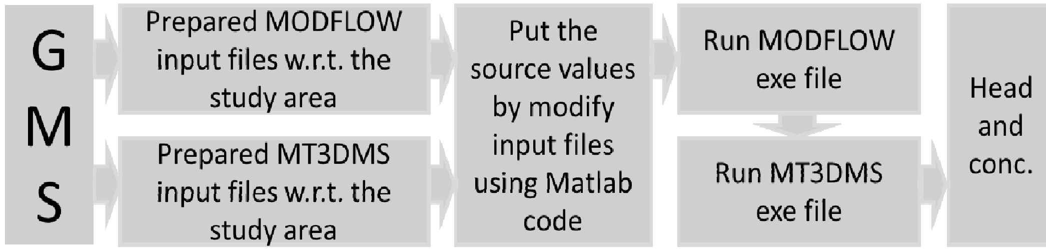


Fig. 3.4: Flow chart showing procedure to execute MODFLOW and MT3DMS models in Matlab environment

Matlab environment has been shown in Fig. 3.4. A Matlab code is developed to open and change the input files of MT3DMS. MT3DMS is then run in Matlab environment which was used to calculate the objection function value of the source identification optimization model.

3.1.2.1 MODFLOW

MODFLOW is a computer program that numerically solves three dimensional ground water flow equation for a porous medium using finite difference method. The main objectives in designing MODFLOW were to design a program that can be readily modified, is simple to use and maintain, can be executed on variety of computers with minimal changes and has the ability to manage large data sets required when running large problems. MODFLOW was designed upon modularization approach. The modular structure of MODFLOW consists of a main program and a series of highly-independent subroutines called modules. The modules are grouped in packages. Each package deals with a specific feature of the hydrologic system which is to be simulated such as flow from wells or flow into drains or with a specific method of solving linear equations which describe the flow system such as the Strongly Implicit Procedure or Preconditioned Conjugate Gradient. The division of MODFLOW into modules permits the user to examine specific hydrologic features of the model independently. This also facilitates development of additional capabilities because new modules or packages can be added to the program without modifying the existing ones.

GMS includes a comprehensive graphical user interface to the groundwater model MODFLOW. GMS supports MODFLOW as pre and post-processor. The input files for MODFLOW are generated by GMS. When GMS is used to generate input files for MODFLOW, it saves data in a number of formatted text file which are linked with HDF5 file. Output files of MODFLOW may be binary files or formatted text files. These files are read by MODFLOW when MODFLOW is launched from the GMS menu. The output for MODFLOW is then imported to GMS for post processing. The groundwater flow equation (3.2) used in MODFLOW can be written as,

$$\frac{\partial}{\partial x}\left[k_{xx}\frac{\partial h}{\partial x}\right]+\frac{\partial}{\partial y}\left[k_{yy}\frac{\partial h}{\partial y}\right]+\frac{\partial}{\partial z}\left[k_{zz}\frac{\partial h}{\partial z}\right]+W=S_s\frac{\partial h}{\partial t} \tag{3.2}$$

Where k_{xx}, k_{yy} and k_{zz} are the values of hydraulic conductivity along the x, y, and z coordinate axes (L/T); h is the potentiometric head (L); W is volumetric flux per unit volume representing sources and/ or sink of water, where negative values are extractions, and positive values are injections (T^{-1}); Ss is the specific storage of the porous material (L^{-1}); and; t is time (T). MODFLOW solves the flow equation using finite difference method.

3.1.2.2 MT3DMS

MT3DMS is a modular three-dimensional transport model that can simulate advection, dispersion and chemical reactions of dissolved constituents. Here MS stands for multi-species structure for accommodating add-on reaction packages. The computer program of the MT3DMS transport model uses modular structure similar to MODFLOW. Like MODFLOW, the MT3DMS model consists of a main program and a large number of highly independent subroutines called module, which are grouped into series of packages. Each of these packages deals with the single aspect of transport simulation. The similarity between MT3DMS and MODFLOW in program structure and design is intended to facilitate the use of MT3DMS transport model in conduction with MODFLOW. MT3DMS uses output head and cell-by-cell flow data computed by MODFLOW. Therefore, before doing MT3DMS simulation there should be a MODFLOW simulation. In MODFLOW simulation, time is divided into stress period, during which all external stress parameter (*i.e.*, source/sink) are constant. Stress period are in turn divided into time steps if simulation is transient. The hydraulic head and fluxes at each time step are solved by flow model and used by the transport model.

In MT3DMS simulation, each time step used in obtaining flow equation is further divided into a number of smaller time increments called transport step, in which hydraulic head and fluxes are assumed constant. Generally, flow and transport model should have same number of stress period. Because flow time step need to be specified in the transport model, so that the velocity field and source\sink information can be updated in the transport model properly. When flow model is steady state with only one stress period and one time step then only MT3DMS code allow the flow and transport model to have different number of stress period. Like MODFLOW, input files for MT3DMS can also be generated by GMS. These files are read back by MT3DMS when it is launched from GMS menu. The output for MT3DMS is then imported to GMS for post processing.

The transport equation (3.3) can be written as,

$$\frac{\partial(\theta C^k)}{\partial t} = \frac{\partial}{\partial x_i}\left(\theta D_{ij}\frac{\partial C^k}{\partial x_j}\right) - \frac{\partial}{\partial x_i}\left(\theta v_i C^k\right) + q_s C_s^k + \sum R_n \tag{3.3}$$

Where, C^k is the dissolved concentration of species k (ML^{-3}); θ is the porosity of the subsurface medium, dimensionless; t is the time (T); x_i is the distance along the respective Cartesian coordinate axis (L); D_{ij} is the hydrodynamic dispersion coefficient tensor (L^2T^{-1}); v_i is the seepage or linear pore water velocity (LT^{-1}); q_s is the volumetric flow rate per unit volume of aquifer representing fluid sources (positive) and sinks (negative) (T^{-1}); C_s^k is the concentration of the source or sink flux for species k, (ML^{-3}); $\sum R_n$ is the chemical reaction term ($ML^{-3}T^{-1}$). MT3DMS solves the transport equation using finite difference method.

3.1.3 Optimization methods

In this study, we have used direct search optimization method and genetic algorithms for solving the inverse optimization problem. The optimization toolbox available in Matlab is used to implement direct search algorithm. We have used our own Matlab code for implementing genetic algorithms.

3.1.3.1 Direct Search

Direct search optimization method does not use any derivative information of the objective function or the constrained function, only its values are used to guide the search process. It requires only the objective function values for finding the minimum and hence is often called the nongradient method. The direct search methods are also known as zeroth-order methods since they use zeroth-order derivatives of the function. All the unconstrained minimization methods are interactive in nature and hence they start from an initial trial solution and proceed toward the minimum point. The direct search can be very useful when objective globally converge to the minimal value of the objective function (Perttunen *et al*. 1993). Unfortunately, this global converge may come at the expense of a large and exhaustive search over the domain.

Unlike more traditional optimization methods that use information about the gradient or higher order derivatives to search for an optimal point. A direct search algorithm searches a set of points around the current point, looking for one where the value of the objective function is lower than the value at the current point. We can use direct search to solve problems for which the objective function is not differentiable, or is not even continuous. Pattern direct search only uses function comparisons, it is easy to implement and use. In matlab toolbox, Global Optimization Toolbox functions include three direct search algorithms called the generalized pattern search (GPS) algorithm, the generating set search (GSS) algorithm, and the mesh adaptive search (MADS) algorithm. All are *pattern search* algorithms that compute a sequence of points that approach an optimal point. At each step, the algorithm searches a set of points, called a *mesh,* around the *current point*, the point computed at the previous step of the algorithm. The mesh is formed by adding the current point to a scalar multiple of a set of vectors called a *pattern*. If the pattern search algorithm finds a point in the mesh that improves the objective function at the current point, the new point becomes the current point at the next step of the algorithm.

This study utilizes pattern search algorithm available in matlab toolbox to identify the groundwater pollution sources in well defined groundwater aquifer. To identify the unknown groundwater pollution sources, the optimization formulation uses the objective of minimizing the weighted sum of squares of the difference between model determined and actual concentration. At the time of solving the optimization model, the external GMS simulation model is linked as an independent model to the optimization method in Matlab environment for calculating objective function value.

3.1.3.2 Genetic Algorithms (GA)

Genetic Algorithm (GA) is considered to be an efficient optimization algorithm for solving non-linear non-convex problems. It is a search procedure based on the mechanics of natural selection and natural genetics. It follows Darwin's survival of the fittest principle. Based on that principle, the intelligent search method of genetic algorithms finds the best and fittest solution of the problem. As a robust and efficient algorithm, it is now becoming very much popular in engineering optimization problems for its wide range of precise search and capability of solving complex non-linear problems. It is most appropriate for large complex nonlinear model where finding the global optimal solution is a difficult task by using other conventional optimization methods. Genetic algorithms work only with the function values of the objective function and the constraint functions. It does not require any other auxiliary information of the objective and constraint functions. The primary differences of genetic algorithms with the other classical algorithms are, genetic algorithms work with the coding of the parameter set not the parameters themselves; it searches from a population of points not a single point; it uses the probabilistic transition rules not the deterministic rules. The basic elements of natural genetics use in genetic search procedure are reproduction, crossover and mutation. Genetic algorithm has been used extensively by different researchers for solving groundwater management models (Ritzel *et al.,* 1994, Wang and Cheng, 1997; Huang and Mayer, 1997; Aly and Peralta, 1999; Morshed and Kaluarachchi, 2000; Aral *et al.,* 2001; Hsiao and Chang, 2002; Maskey *et al.,* 2002; Singh and Datta, 2007; Bhattacharjya and Datta, 2009).

3.1.3.3 Working principle

The operation of genetic algorithm begins with a population of random strings representing design or decision variable. First, genetic algorithm creates predefined number of populations of solution within specified bound of the variable. Each of the members of the population is called individual. The decimal value of the decision variable is given by the following linear mapping rule.

$$x = x^l + \frac{x^u - x^l}{2^q - 1} \sum_{k=0}^{q} 2^k b_k \tag{3.4}$$

Where, x^u, x^l are the upper and lower bound of the particular variable, b_k is a binary number (0 or 1) and q represents length of string, depends upon desired solution accuracy. In general, a fitness function is derived from objective function and fitness value of each individual is determined by using decimal value obtained in equation (3.4). The population is then operated by three main genetic operators, *i.e.* reproduction, crossover and mutation to create a new population of points. The procedure is continued until the termination criterion is met. Fig. 3.5 shows the flowchart of simple genetic algorithms.

The three genetic operators are discussed below.

3.1.3.4 Reproduction

Reproduction is the first operator applied on a population. It selects good string from the population in a probabilistic manner according to the fitness values and forms a mating pool. Selecting strings according to their fitness values means that string with a higher fitness value have the higher probability of contributing one or more off spring to the next

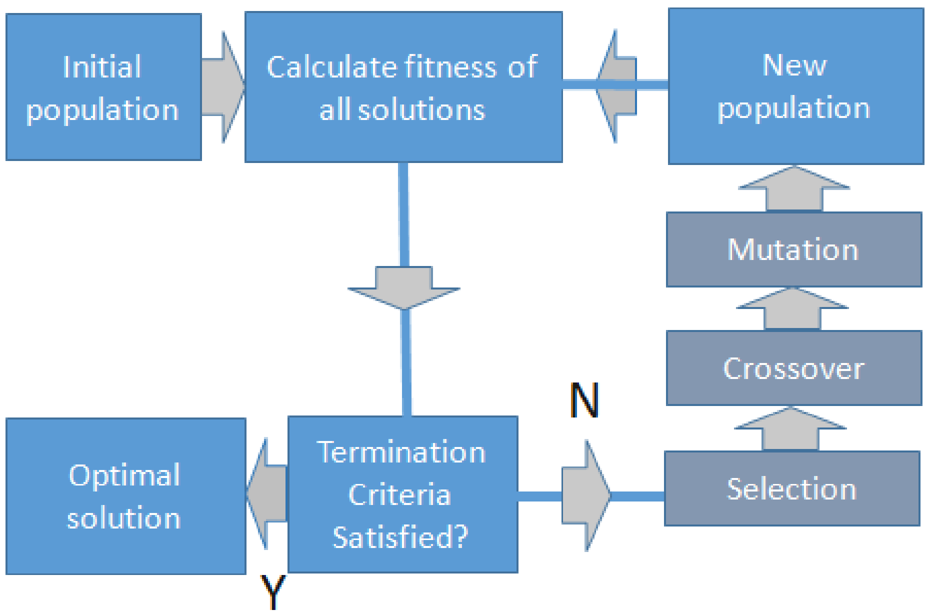


Fig. 3.5: flowchart of simple genetic algorithms

generation.

The probability of selecting any string for mating pool is

$$P_i = \frac{F_i}{\sum_{j=1}^{n} F_j} \qquad (3.5)$$

Where, F_i is the fitness value of i^{th} string and n denotes the total number of population. The probable number of copies in mating pool of i^{th} string is,

$$N_i = \frac{F_i}{\breve{F}} \qquad (3.6)$$

Where, $$\breve{F} = \sum_{i=1}^{n} \frac{F_i}{n} \qquad (3.7)$$

Thus in reproduction, good string in a population is probabilistically assigned a large number of copies and a mating pool is formed. It is worth mentioning here that no new strings are formed by the reproduction operator.

3.1.3.5 Crossover

In reproduction, good strings in a population are probabilistically assigned a larger number of copies and a mating pool is formed. But no new strings are formed in the reproduction phase. In crossover stage, new strings are formed by exchanging information among strings of mating pool. In this stage new strings are formed by exchanging information among the randomly selected strings of mating pool. The two participating string in the crossover operation are known as parent strings and the resulting strings are known as children string. Fig. 3.6 shows the schematic representation of single point crossover operation. The single point crossover operator is performed by randomly selecting a crossing site along the string and by exchanging all the bits on the right side of the crossing site. A better child string may be produced by the crossover operator. However, it will depend on whether an appropriate site of crossing is chosen or not. Since the knowledge of appropriate site is usually not known beforehand, a random site is often chosen. If good strings are produced by crossover operation then in the next mating pool there will be more number of copies of them, otherwise they will not survive too long because reproduction will select against those strings in subsequent generations. In order to preserve some of the good strings that are already present in the mating pool, not all the strings are used in crossover operation. If crossover probability of pc is used then 100*pc percent strings in the population are used in the crossover operation and remaining percentage are remains as they are in the current population.

3.1.3.6 Mutation

Like crossover operator, mutation operator is also used for the search of new string. Mutation operator makes random but, small changes, to the encoded solution.

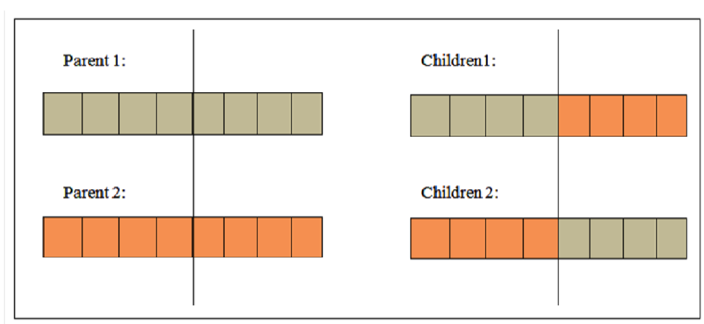


Fig. 3.6: Schematic representation of crossover probability

This prevents the falling of all solutions into a local optimum and extends the search space of the algorithm. This is divergence in nature while crossover operator is convergence in nature. In case of binary coded genetic algorithms, the mutation operator changes 1 to 0 and vice versa with a small mutation probability. For example, for the string below (Fig. 3.7), if optimum solution requires 1 in the extreme left or right bit then neither reproduction nor crossover operation can create 1 in that position. But mutation can introduce 1 in that position. Thus mutation operator alters a string locally to create a good string. Fig. 3.7 shows the example of a mutation. The inclusion of mutation introduces some probability (P_m) of turning 0 into 1.

3.1.3.7 Limitation of Genetic Algorithm

Although because of its simplicity and classiness, genetic algorithm have proven

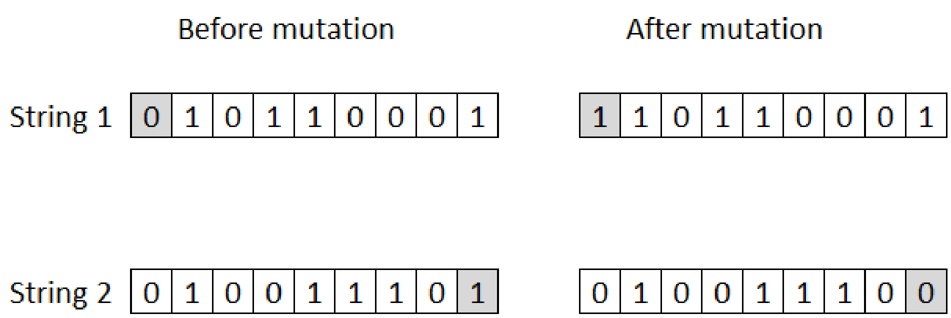


Fig.3.7: Mutation of a chromosome

themselves as efficient problem solving strategy, yet they cannot be considered as universal remedy. Due to its probabilistic and deterministic nature, it may yields different solutions on different runs. Another drawback of genetic algorithm is that, the result obtained using GA may not be the true optimal solution but it may be close to global optimal solution. As such, GA can be used initially to obtain near global solution. Thereafter, classical optimization algorithm can used to obtain the true optimal solution, using solution of GA as initial point. The significance of using solution of GA as initial point of classical optimization algorithm is to search local optima surrounding the initial point.

3.2 Study Area

The performance of the developed model has been evaluated using two illustrative study areas. We have used the study area considered by Mahar and Datta (2000) to evaluate the relative efficiency of the model. This is a rectangular shaped confined aquifer. To show the field applicability of the developed model, another bigger study area with irregular boundary is also considered in this study.

3.2.1 First study area

The problem considered by Mahar and Datta (2000) is initially considered here for evaluating the performance of the proposed models (Fig. 3.8). The hypothetical confined

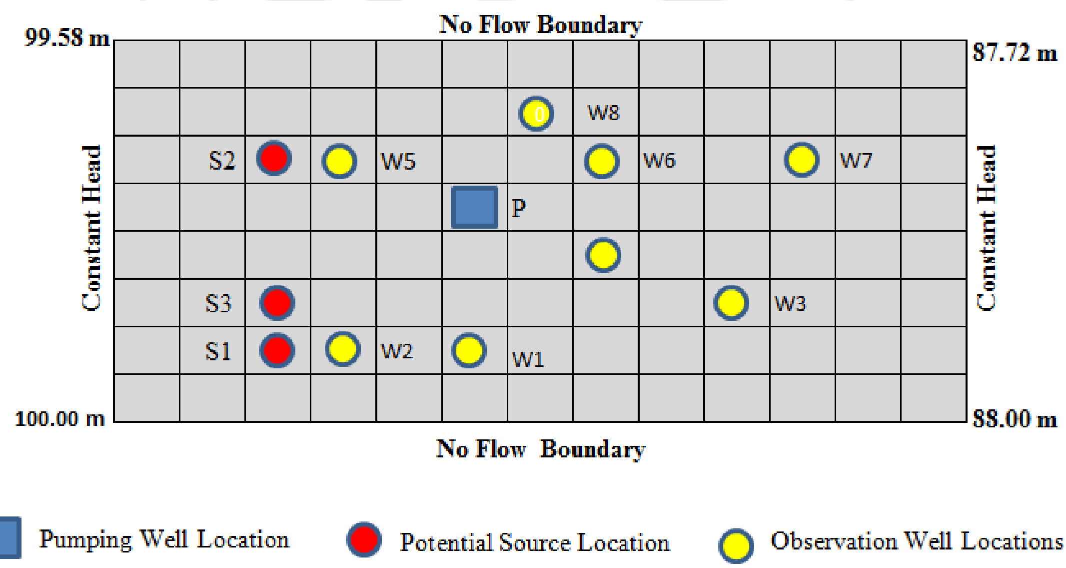


Fig. 3.8: First Illustrative Study Area

aquifer has an area of 1.04 km^2 (1.3 km× 0.8 km) and is homogeneous and isotropic. Bear (1972) discussed various types of boundary conditions related to flow and transport problems in aquifer. In this study area, the boundary condition of the aquifer is considered as time invariant. The aquifer has fixed head boundary condition in the east and in the west sides, whereas the north and the south boundaries of the aquifer have no flow boundary. In the east side of the aquifer, the hydraulic head is varying from 100.00m to 99.58m. On the other hand, in the west side of the aquifer, the hydraulic head is varying from 88.00m to 87.72m. The aquifer is simulated for single layer in the computational grid. There are 8 rows and 13 columns. The study area is discritized into 100m x 100m. Inside the aquifer, the initial head is considered as 100 m. The hydraulic parameters used

Table: 3.1 Flow and transport parameter values for illustrative study area

Parameter	value	Parameter	value
K_{xx} (m/s)	0.0002	b (m)	30.5
K_{yy} (m/s)	0.0002	Δx (m)	100.0
η	0.25	Δy (m)	100.0
$\propto_L$(m)	40.0	Δt	3.0
$\propto_T$ (m)	9.6	S	0.002

Table: 3.2 Source Flux at different locations for first illustrative study area

Source Flux of First study area (gm/sec)				
	Time Step			
Source	1	2	3	4
S1	47.00	15.00	37.00	0
S2	30.00	58.80	0	35.00
S3	0	0	0	0

for simulating the flow and transport processes are shown in Table 3.1. There are three pollutant sources in the aquifer which have been designated as S1, S2 and S3. The concentration values at the three locations are shown in Table 3.2. The source S1 and S2 are active where as the source S3 is inactive throughout the simulation period. The initial contaminant concentration for the entire aquifer is 100 ppm. The parameters used in the simulation are; horizontal hydraulic conductivity ($K_{xx} = K_{yy}$) of 0.0002 m/s, porosity (η) of 0.25, longitudinal dispersivity ($\propto_L$) of 40.0m, transverse dispersivity ($\propto_T$) of 9.6 m and storativity (S) of 0.002. There are eight observation wells which have been designated as W1, W2, W3, W4, W5, W6, W7 and W8. The aquifer has only a single pumping well (P) which is located at the centre of the study area. The pumping rate shown in Table 3.3. The flow and transport simulation are made for 5 years considering time interval of 3 months. At the time of simulating the transport process, out of three pollutant sources, only two are considered as active while other one is taken as inactive.

Table: 3.3 Pumping rate of water at pumping locations

Pumping rate (m^3/day)			
Time step	First study area	Time step	First study area
1	273.02	11	163.29
2	163.29	12	327.45
3	327.45	13	273.02
4	163.29	14	163.29
5	273.02	15	381.02
6	327.45	16	217.72
7	163.29	17	273.02
8	273.02	18	163.29
9	381.02	19	327.45
10	217.72	20	217.72

3.2.2 Second Study area

The proposed model is applied in a bigger area as shown in Fig. 3.9 to show the field applicability of the model. The size of the second study area is approximately 17.346 km^2. The west and south sides of the aquifer are bounded by the rivers. As such, constant head boundary is considered in these two sides. The other two boundaries, *i.e.*, the north and the east sides have no flow boundary. The boundary condition of the aquifer is considered as time invariant. There are five contaminant sources in the aquifer which have been designated as S1, S2, S3, S4 and S5. The concentration values assigned for these five locations are shown in Table 3.4. The initial contaminant concentration for the entire aquifer is 200 ppm. There are four pumping wells in the study area which has been marked as P1, P2, P3 and P4. The pumping rate assigned for different stress periods are shown in Table 3.5. There are eight observation wells which are designated as W1, W2, W3, W4, W5, W6, W7 and W8. The flow and transport simulation are done for 5 years considering time step of 3 months. The hydraulic parameters for this study area are

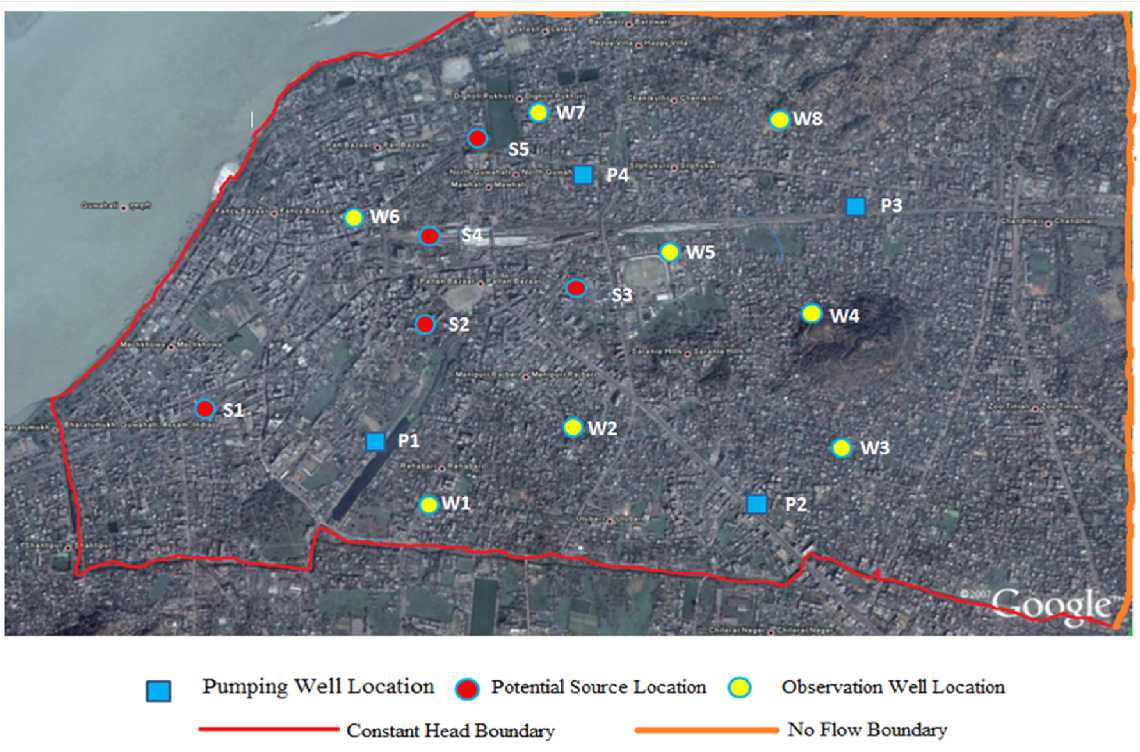


Fig. 3.9: Illustrative Second Study Area

Table 3.4: Source flux at different locations for Second illustrative Study area

Source Flux of Second Study area (gm/sec)					
	Time Step				
Source	1	2	3	4	5
S1	908.42	1130.50	653.35	902.13	721.25
S2	644.02	1023.87	1139.88	781.09	889.77
S3	0	0	0	0	0
S4	0	1024.16	652.05	1117.45	889.77
S5	987.08	0	0	1104.82	639.93

considered as same as the first study area.

3.3 Results and Discussion

3.3.1 GMS simulation results for first study area

Table 3.5: Pumping rate of water in second study area from Pumping location

Time Step	Discharge rate (m^3/day)	Time Step	Discharge rate (m^3/day)
1	327.024	11	272.52
2	163.512	12	218.016
3	218.016	13	327.024
4	381.528	14	163.512
5	109.008	15	381.528
6	327.024	16	163.512
7	272.520	17	272.520
8	163.512	18	218.010
9	381.528	19	327.024
10	109.008	20	272.520

We have used groundwater simulation system (GMS) to simulate the flow and transport processes in aquifer. A comparative analysis is initially carried out to evaluate the results obtained by GMS model with the results obtained by Mahar and Datta (2000). The solution obtained by GMS is indicated by the word ‘GMS’. Mahar and Datta (2000) used optimization based simulation technique for solving the flow and transport equations. They solved the optimization model using large scale optimization solver MINOS. After simulation, the concentration values at eight observation wells are obtained for different time steps. The concentrations are measured for 5 years at a time interval of 3 months at eight observation well locations at W1, W2, W3, W4, W5, W6, W7 and W8. Fig. 3.10 shows the break through curve generated by GMS and Mahar and Datta (2000) for eight number of observation wells. The figures show the plot between concentration (ppm) versus time for two methodologies. It has been seen from the figures that the solution obtained by the GMS is similar to the solution obtained by Mahar and Datta (2000). From these figures, it can be concluded that the both results are comparable. The comparison of results demostrate the validity of GMS methodology in simulating the flow and transport for the first study area.

A number of 2D transient transport simulation is also performed using MODFLOW and MT3DMS in first illustrative study area. The simulation model simulates the physical processes in aquifer and show the distribution of pollutants in the aquifer. Therefore 20 time steps for the entire simulation period. It is observed that the distribution of pollutant concentration at each and every time step is different from the previous step. The Fig. 3.11 shows the distribution of pollutant concentration in the entire aquifer for four time steps. The first plot of the figure shows the concentration distribution after 90 days. It can be observed from the figure that pollutant is moving towards to the pumping well due to continious pumping. Thus the concentration is increasing towards the pumping well. The second plot of the figure shows the pollutant distribution at 450 days. The concentartion distribution after the 450 days also shows that more and more pollutants are moving towards the pumping well due to continious pumping of water. The concentration is increasing towards the pumping well. The third plot of the figure shows the pollutant distribution at 900 days. The concentartion distribution at 900 days shows that pollutant concentration is decreasing near the pumping well. This is because that the sources were active only up to one year. The fourth plot of the figure shows the pollutant distribution at 1800 days. The concentration countour drawn at 1800 days shows that the plume has almost disappeared from the aquifer as the contaminant sources were not active for longer period. The analysis of the results shows that the results are intuitively as expected.

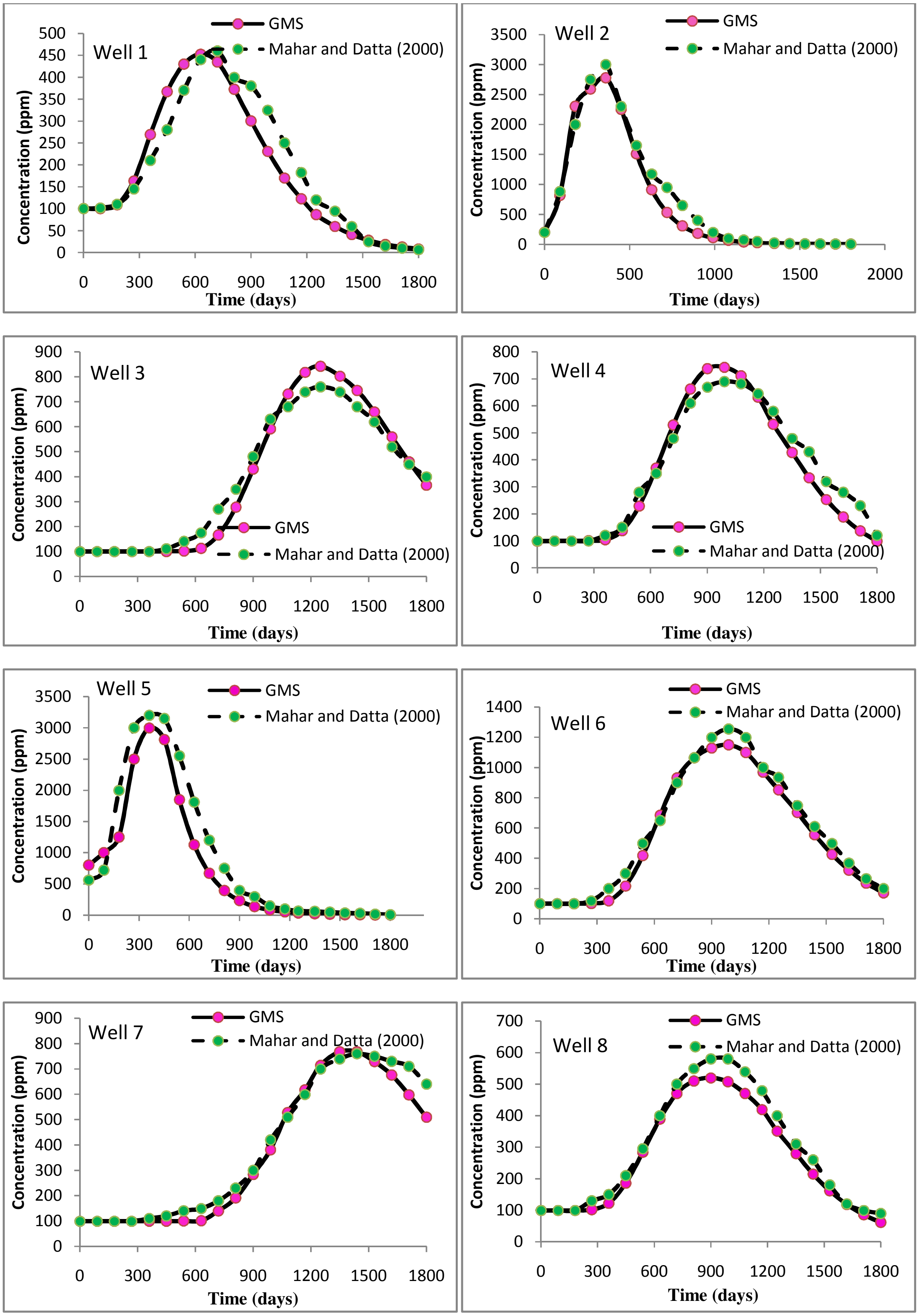
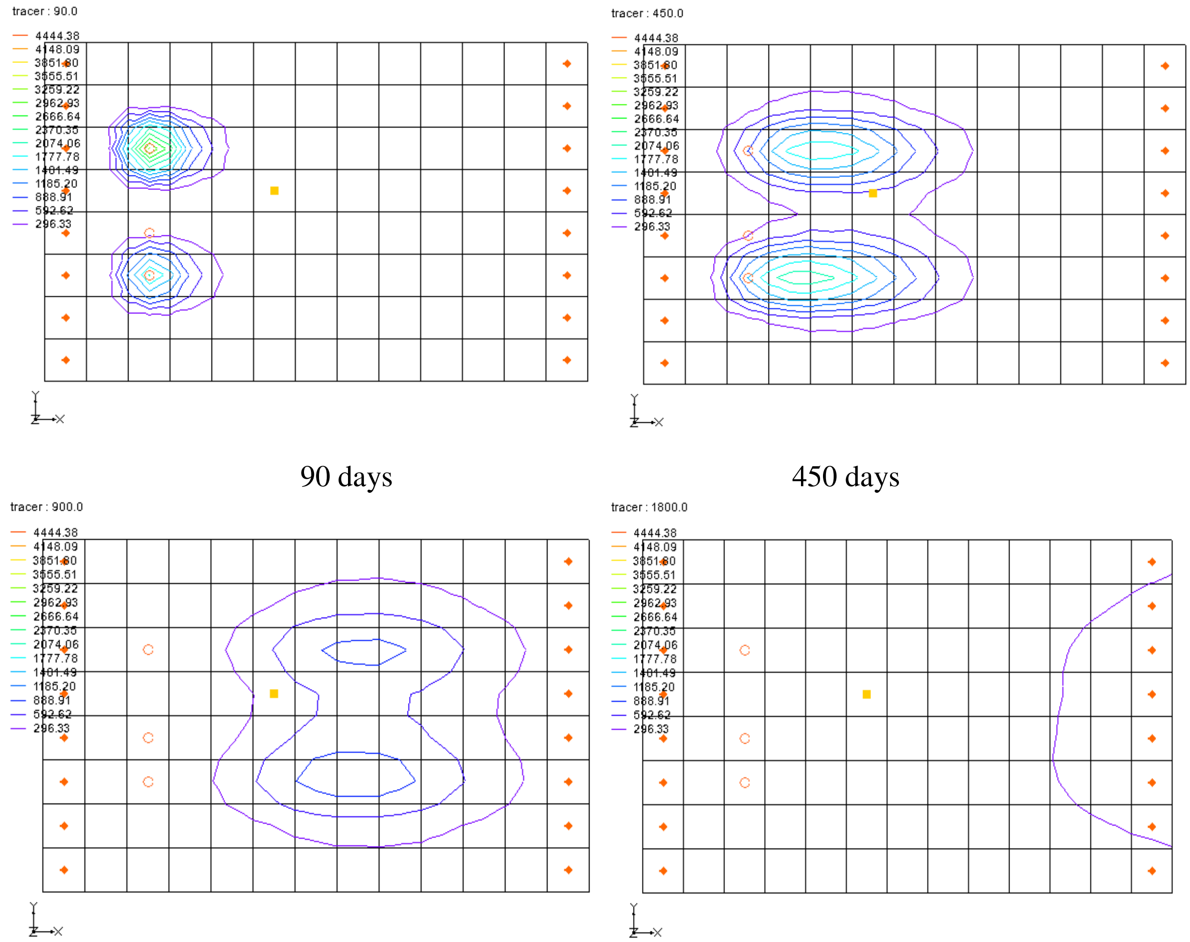


Fig. 3.10: Break through curve for study first study area



900 days   1800 days

Fig. 3.11: Concentration distribution in the first study area for four time steps

3.3.2 GMS simulation results for second study area

The flow and transport simulations are also performed for the second study area using GMS. The parameters used in the second study area are same as that of used in the first study area. The pumping rate and source flux are shown in Table 3.4 and Table 3.5 respectively. In this study area, the concentrations are measured for 20 time steps at 8 observation well locations. The concentration are measured at eight numbers of observation well locations after simulating the flow and transport processes by GMS methodology. The numerically simulated concentration by GMS methodology at the observation well locations are plotted as function of time. Fig. 3.12 shows the break

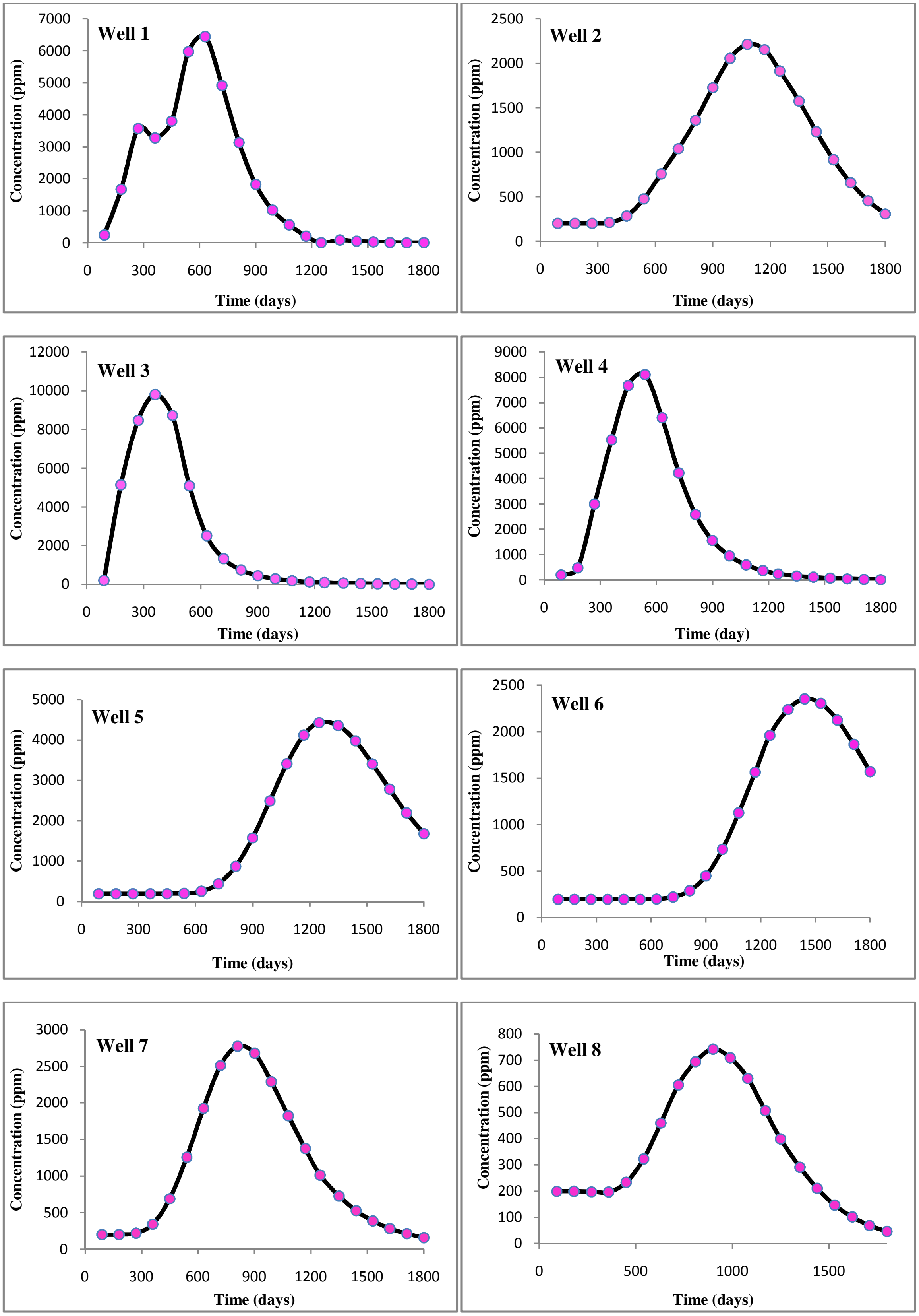
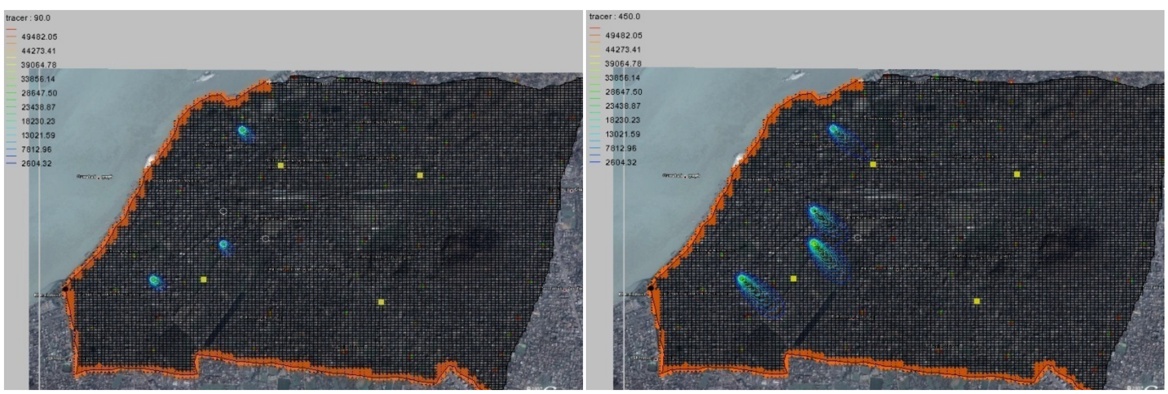


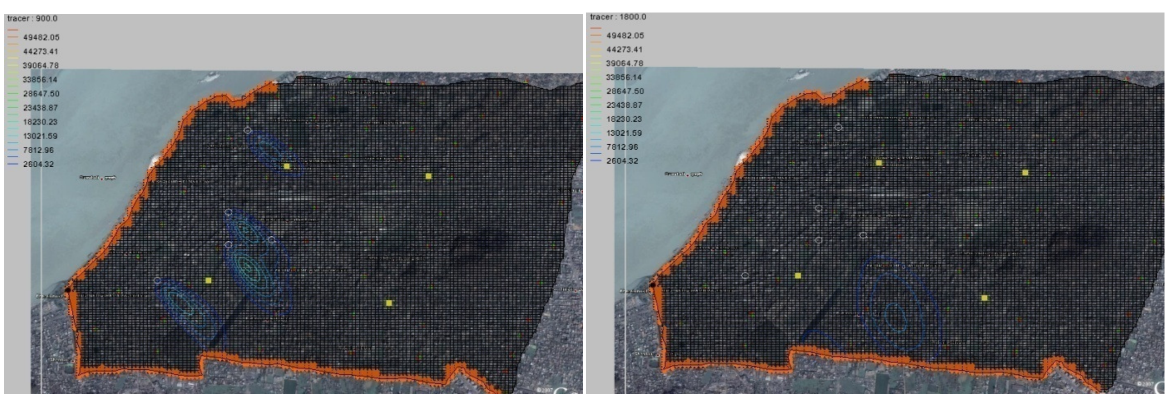
Fig. 3.12: Break through curve by GMS for second study area

throgh curves generated by GMS model for eight observation well locations. It is observed that the breakthrough curves are intuitively as expected.

The 2D transport simulation is also performed by GMS for transient flow and transport for second illustrative study area. In this study area also, simulation processes is done for 20 time steps which is further divided 10 number of transoport steps. Thus we can observe the pollutant distribution for 200 time step. It is observed that the pollutant concentration is different at each and every time step due to continious pumping of water. Fig. 3.13 shows the distribution of concentration in the entire aquifer for the four time steps, *i.e.* at 90 days, 450 days, 900 days and 1800 days. The first figure shows the



90 days 450 days



900 days 1800 days

Fig. 3.13: Concentration distribution contour in the second study area for four time steps

concentration contour at 90 days. It can be observed that concentration is increasing towards the pumping well. During this time step, the sources in the aquifer were active, therefore the concentration is increasing towards the pumping well. The concentartion distribution at 450 days shows that more and more pollutants are moving towards the pumping well due to continious pumping of water. After that, concentartion is decreasing near the pumping well. This is because of that the sources were active only up to five time steps. The contour plot at 1800 days shows that there is no contaminant in the entire aquifer.

3.3.3 Performance evaluation of GMS-DS model for first study area

The inverse model described earlier is used for estimating the unknown pollution source of the first study area. The direct search optimization function available in Matlab toolbox is used to solve the inverse problem. The parameters and stopping criteria used in the direct search optimization model are: mesh tolerance of 1e-6, maximum iteration of 10,000, maximum function evaluation of 10,000, time limit of infinity, X tolerance of 1e-6 and function tolerance of 1e-6. The unknown source flux obtained by GMS-DS methodology is compared with the source flux value obtained by Mahar and Datta (2000) and also with the actual concentration. Table 3.6 shows the results obtained using the

Table. 3.6: Actual and Estimated concentration at three point sources

Time Step	Source Loc.	Actual Conc. (gm/sec)	Predicted Conc. (gm/sec)	
			Mahar and Datta (2000)	GMS-DS
1	S1	47.00	45.10	45.14
	S2	30.00	29.34	29.37
	S3	0.00	0.20	0.21
2	S1	15.00	18.17	18.15
	S2	58.8	55.05	55.07
	S3	0.00	0.00	0.00
3	S1	37.00	30.44	30.94
	S2	0.00	2.44	2.54
	S3	0.00	0.00	0.00
4	S1	0.00	1.84	1.77
	S2	35.00	33.62	34.19
	S3	0.00	0.19	0.18

GMS-DS methodology along with the results obtained by Mahar and Datta (2000) and actual concentration. This comparison shows that the results obtained by GMS-DS model are comparable with the results obtained by Mahar and Datta (2000). For example, the concentration estimated for the first time step by GMS-DS model at S1 source is 45.14 gm/sec whereas that is estimated by Mahar and Datta (2000) is 45.10 gm/sec and the actual concentration at source S1 is 47 gm/sec. Similarly, the concentration estimated for the sources S2 and S3 by GMS-DS model for the first time step are 29.37 gm/sec and 0.21 gm/sec respectively, whereas that are estimated by Mahar and Datta (2000) are 29.34

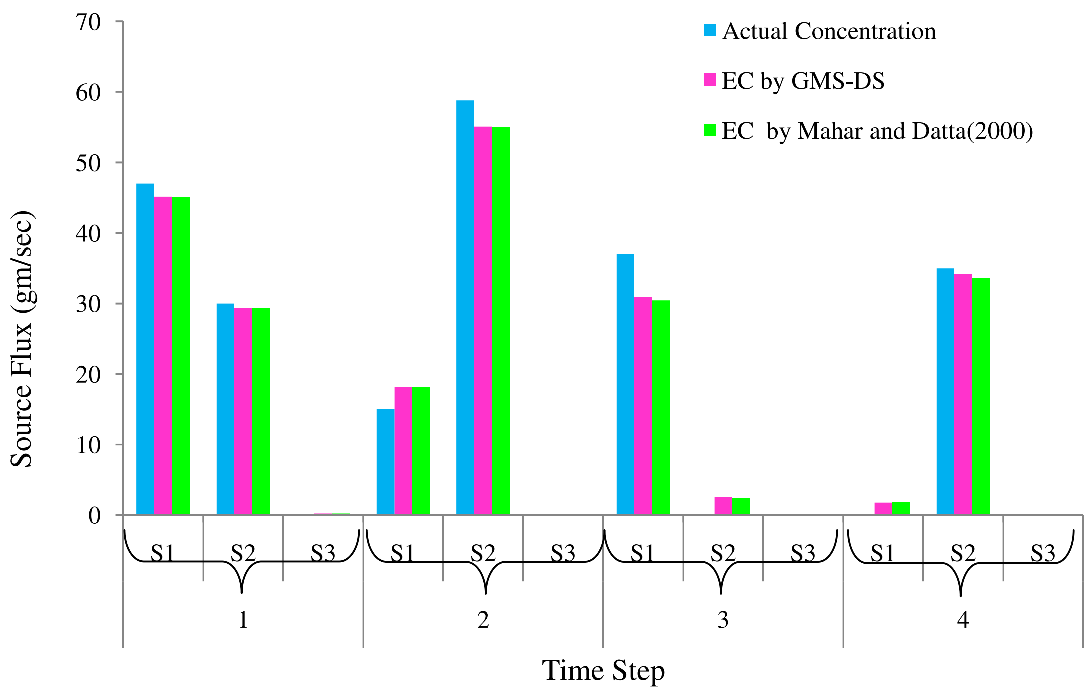


Fig. 3.14: Comparison of source fluxes for first illustrative study area

gm/sec and 0.20 gm/day respectively. The actual concentrations are 30 gm/sec and 0gm/sec respectively. This shows that the concentration predicted by GMS-DS is closer to the actual source values and also comparable with the result obtained by Mahar and Datta (2000). For the other sources also similar trend has been observed in all the time steps. Fig. 3.14 shows the comparison of concentration at three source locations obtained by GMS-DS model along with the concentration obtained by Mahar and Datta (2000) and the actual concentration. It may be observed from the figure that the results obtained by the proposed methodology are closer to that obtained by the Mahar and Datta (2000). A study is also carried out to find out the relative efficiency of GMS-DS model over the method used by Mahar and Datta (2000) using relative error. Table 3.7 shows that relative error for the concentration obtained by the GMS-DS methodology and the methodology suggested by Mahar and Datta (2000). The relative error is calculated with respected to the actual concentration. It can be observed that the relative error of GMS-DS method is lesser than that of Mahar and Datta (2000). For example, for the first time step at S1 source, relative error achieved by GMS-DS methodology is 3.96 %, whereas that of obtained by Mahar and Datta (2000) is 4.04%. Again the relative error for second and third time steps at S1 source location are 21% and 16.36% respectively whereas that are obtained by Mahar and Datta (2000) model are 21.13% and 17.72% respectively. This shows that the performance of GMS-DS model is better than Mahar and Datta (2000). Similarly, for S2 and S3 source locations also, the relative errors of GMS-DS model are lesser than the relative errors achieved by Mahar and Datta (2000). It clearly shows that the performance of the GMS-DS methodology is better than the method proposed by Mahar and Datta (2000). Fig. 3.15 shows the comparison of the two methodologies in terms of relative errors as bar diagram.

Table. 3.7: Comparison of relative error for first study area

Time Step	Source Loc.	Relative Error (%)	
		Mahar and Datta (2000)	GMS-DS
1	S1	4.04	3.96
	S2	2.20	2.11
	S3	-	-
2	S1	21.13	21.00
	S2	6.38	6.33
	S3	-	-
3	S1	17.73	16.36
	S2	-	-
	S3	-	-
4	S1	-	-
	S2	3.94	2.29
	S3	-	-
AVERAGE		4.61	4.33

3.3.4 Performance of the GMS-GA model for first illustrative study area

The inverse model is also solved using genetic algorithms to obtain unknown pollution sources of first study area. The parameters used in GA based optimization model are: population size of 200, generation of 200, crossover fraction of 0.9, mutation fraction of 0.003 and elitism fraction of 0.1. The unknown source flux obtained by GMS-GA model

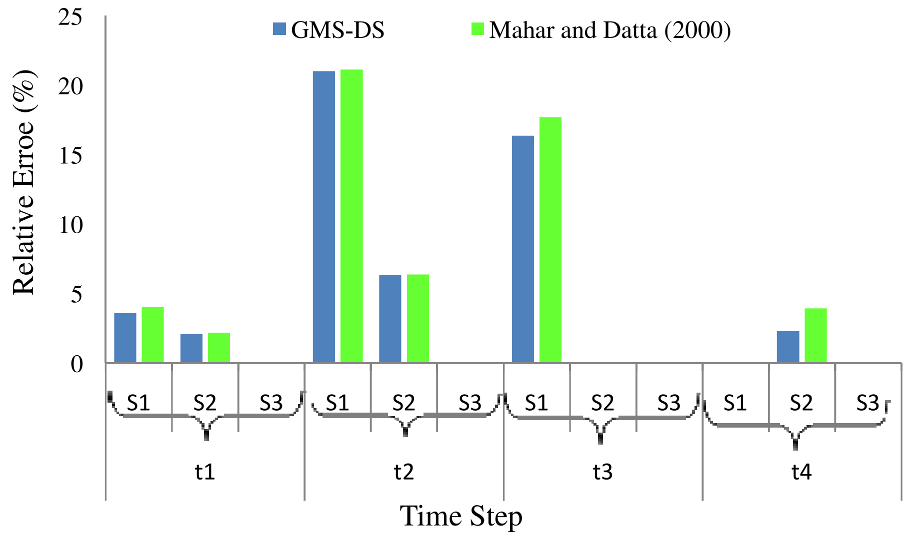


Fig.3.15: Comparison of relative error (%) between GMS-DS model and Mahar and Datta (2000) for first illustrative study area.

is compared with the source flux value obtained by GMS-DS, Mahar and Datta (2000) and also with the actual concentrations. Table 3.8 shows the results obtained using the GMS-GA methodology along with the results obtained by GMS-DS, Mahar and Datta (2000) and actual concentration. This comparisons show that the result obtained by GMS-

Table 3.8: Actual and Estimated concentration by three models at three point sources

Time Step	Source Loc.	Actual Conc. (gm/sec)	Predicted Conc. (gm/sec)		
			Mahar and Datta (2000)	GMS-DS	GMS-GA
1	S1	47.00	45.10	45.14	45.16
	S2	30.00	29.34	29.37	29.37
	S3	0.00	0.20	0.21	0.23
2	S1	15.00	18.17	18.15	18.16
	S2	58.8	55.05	55.07	55.13
	S3	0.00	0.00	0.00	0.00
3	S1	37.00	30.44	30.94	30.98
	S2	0.00	2.44	2.54	2.54
	S3	0.00	0.00	0.00	0.00
4	S1	0.00	1.84	1.77	1.93
	S2	35.00	33.62	34.19	34.21
	S3	0.00	0.19	0.18	0.18

GA model is slightly superior to the results obtained by GMS-DS, Mahar and Datta (2000) and are more closed to the actual concentration. For example, the actual concentration of source S1 at time step 1 is 47 gm/sec. The concentration predicted by GMS-GA model and GMS-DS model are 45.16 gm/sec and 45.14 gm/sec respectively. The concentration predicted by Mahar and Datta (2000) is 45.10 gm/sec. It shows that the source concentration predicted by the GMS-GA is slightly better than that of obtained by GMS-DS model and Mahar and Data (2000). Similar trend has also been observed in the other sources. Fig. 3.16 shows the comparison of these methodologies using bar diagram. The comparison results clearly indicate that GMS-GA is slightly better than

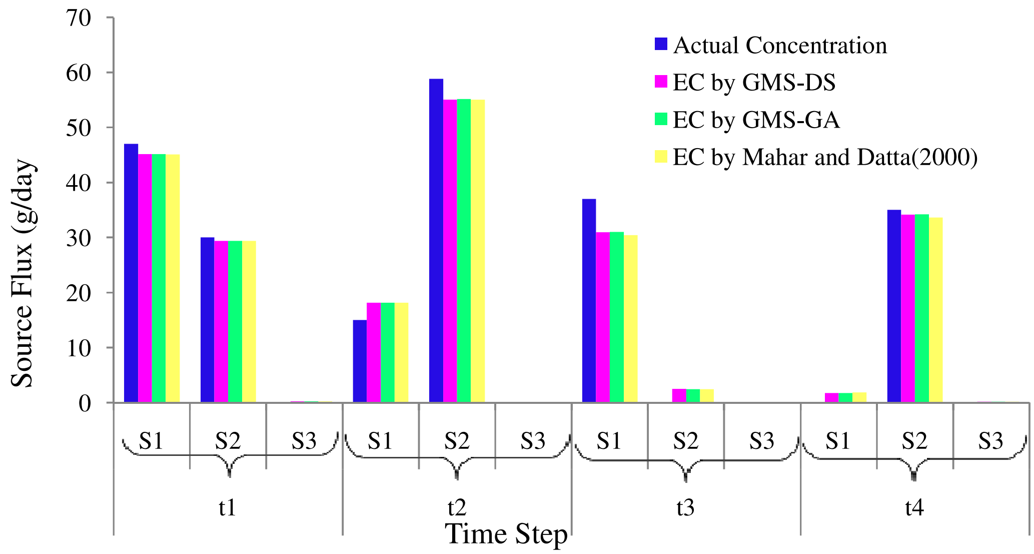


Fig. 3.16: Comparison of source fluxes for first illustrative study area

the results obtained by Mahar and Datta (2000) and GMS-DS models.

A study is also carried out to find the relative efficiency of the GMS-GA model with GMS-DS and Mahar and Datta (2000). Table 3.9 shows that relative error achieved by GMS-GA and GMS-DS models and Mahar and Datta (2000). The relative error is calculated with respect to the actual concentration. It can be observed that the relative error of the GMS-GA and GMS-DS methods are lesser than Mahar and Datta (2000). For example, for the first time step, relative error achieved by the GMS-GA model for S1 source is 3.91% and that of achieved by GMS-DS methodology is 3.96 %. The relative error achieved by Mahar and Datta (2000) is 4.04%. This clearly indicates that the GMS-GA methodology is better than the other two models. Similarly for second, third and fourth time steps also, the relative errors achieved by GMS-GA model are lesser than the relative errors achieved by GMS-DS and Mahar and Datta (2000). It is also observed in the table that average relative error achieved by GMS-GA model is 4.32 % whereas that is obtained by GMS-DS and Mahar and Datta (2000) are 4.33 % and 4.61% respectively. It clearly shows that the performance of GMS-GA is better than GMS-DS and Mahar and Datta (2000) models. Fig. 3.17 shows the comparison of relative errors achieved by three models as bar diagram. This evaluation of the results shows that GMS-GA model could be promising one for solving groundwater pollution source identification problem.

Table 3.9: Comparison of relative error for first study area

Time Step	Source Loc.	Relative Error (%)		
		Manar and Datta 2000	GMS-DS	GMS-GA
1	S1	4.04	3.96	3.91
	S2	2.20	2.11	2.10
	S3	-	-	-
2	S1	21.13	21.00	21.07
	S2	6.38	6.33	6.24
	S3	-	-	-
3	S1	17.73	16.36	16.27
	S2	-	-	-
	S3	-	-	-
4	S1	-	-	-
	S2	3.94	2.29	2.26
	S3	-	-	-
AVERAGE		4.61	4.33	4.32

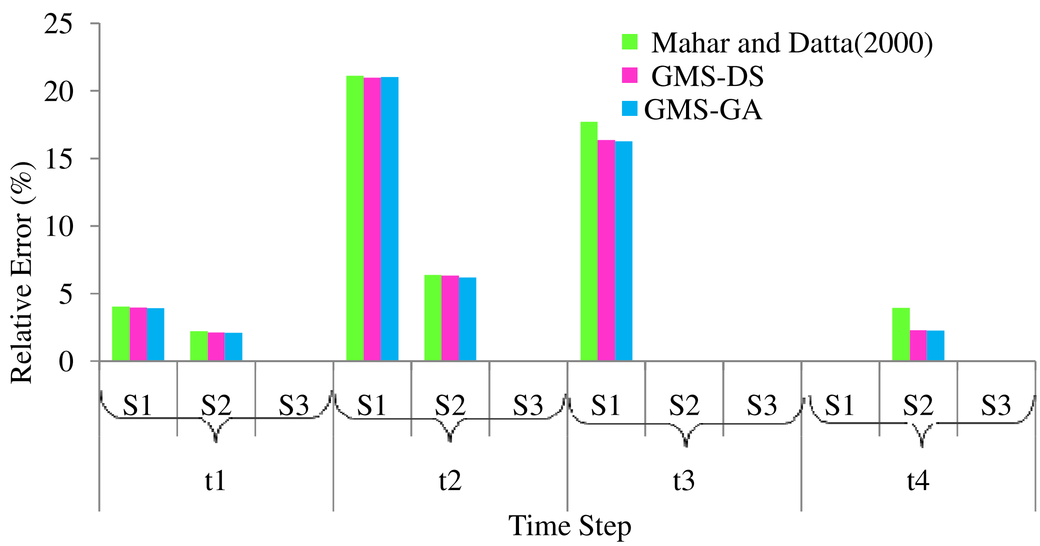


Fig. 3.17: Comparison of Relative error for first illustrative study area

3.3.5 Performance of the GMS-DS model for second study area

To check the field applicability and performance of the proposed large aquifer system, the model is also applied to second illustrative study area. The unknown source flux obtained by GMS-DS model is compared with the actual concentration. The concentration values obtained by GMS-DS model are shown in Table 3.10. It can be observed in the table that the concentration predicted by GMS-DS model is closer to the actual concentration. For example, the actual concentration at source S1 at first time step is 908.42 gm/sec. The concentration predicted by GMS-DS model is 1008.12 gm/sec. The actual concentration at S2, S3, S4 and S5 are 644.01 gm/sec, 0 gm/sec, 0 gm/sec and 987.07gm/sec

Table 3.10: Actual and Estimated concentration for second study area at five point sources

Time Step	Source Location	Actual Conc. (gm/sec)	GMS-DS (gm/sec)	Relative Error (%)
1	S1	908.42	1008.12	10.97
	S2	644.02	805.58	25.05
	S3	0	36.28	-
	S4	0	0	-
	S5	987.08	711.88	27.88
2	S1	1130.50	1008.38	10.81
	S2	1023.87	1151.43	12.45
	S3	0	0	-
	S4	1024.16	763.95	25.41
	S5	0	24.10	-
3	S1	653.35	732.92	12.18
	S2	1139.88	1147.11	0.64
	S3	0	67.51	-
	S4	652.05	735.55	12.8
	S5	0	97.50	-
4	S1	902.13	1068.65	18.5
	S2	781.09	809.98	3.69
	S3	0	0	-
	S4	1117.45	1080.37	3.32
	S5	1104.82	906.99	17.9
5	S1	721.25	647.13	10.25
	S2	889.77	761.36	14.43
	S3	0	70.76	-
	S4	889.77	795.29	10.62
	S5	639.93	691.36	8.03
AVERAGE				8.23

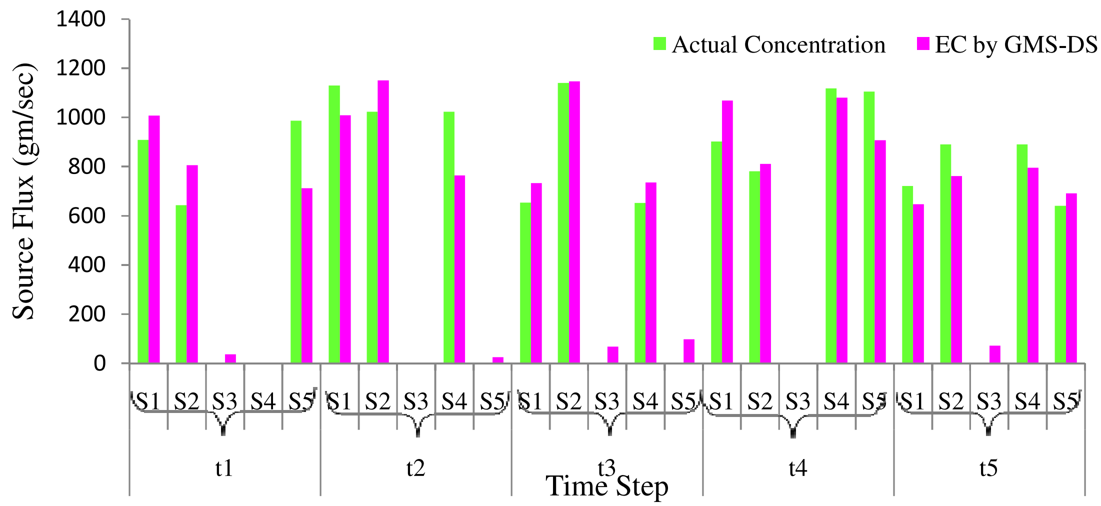


Fig. 3.18: Comparison of concentration obtained by using GMS-DS model with actual concentration for second illustrative study area

respectively for first time step whereas that are estimated by GMS- DS model are 805.58 gm/sec, 36.28 gm/sec, 0 g/day and 711.88 gm/sec respectively. Fig. 3.18 shows the comparison of actual concentration and concentration predicted by GMS-DS model using bar diagram. We have also evaluated performance of the model using relative error. Table 3.10 shows the relative error with respect to the actual concentration for all the source locations. As shown in the table, the relative errors achieved by GMS-DS model are varying between 0.64% and 27.88%. The average value of relative error is 8.33%. Considering the complexity of the problem, the prediction made by GMS-DS model is in acceptable range.

3.3.6Performance of the GMS-GA methodology in second study area

To check the applicability of GMS-GA model on a large aquifer system, the model is applied to the second illustrative study area. The concentration value obtained by GMS-GA is compared with GMS-DS model. Table 3.11 shows the concentration at source locarions obtained using GMS-DS and GMS-GA methodologies along with the actual concentration. It can be observed in the table that the concentration predicted by the GMS-GA model is closer to the actual concentration. For example, the actual concentration at source S1 at first time step is 908.42 gm/sec. The concentration predicted by GMS-GA model is 990.72 gm/sec and the concentration predicted by GMS-DS is

Table 3.11: Actual and Estimated concentration for second study area at five point sources

Time Step	Source Location	Actual Conc. (gm/sec)	Predicted Conc. (gm/sec) GMS-DS	GMS-GA
1	S1	908.42	1008.12	990.72
	S2	644.02	805.58	771.30
	S3	0	36.28	36.19
	S4	0	0	0
	S5	987.08	711.88	748.60
2	S1	1130.50	1008.38	1010.78
	S2	1023.87	1151.43	1149.70
	S3	0	0	0
	S4	1024.16	763.95	765.87
	S5	0	24.10	23.97
3	S1	653.35	732.92	730.83
	S2	1139.88	1147.11	1133.27
	S3	0	67.51	66.74
	S4	652.05	735.55	733.36
	S5	0	97.50	97.07
4	S1	902.13	1068.65	1068.66
	S2	781.09	809.98	807.35
	S3	0	0	0
	S4	1117.45	1080.37	1083.70
	S5	1104.82	906.99	910.59
5	S1	721.25	647.13	656.26
	S2	889.77	761.36	775.26
	S3	0	70.76	70.64
	S4	889.77	795.29	797.95
	S5	639.93	691.36	690.61

1008.10 gm/sec. For the other sources also the concentration predicted by GMS-GA is better than GMS-DS model. Fig. 3.19 shows the comparison of these methodologies using bar a diagram. It can be observed that the concentration predicted by GMS-GA model is better than the concentration predicted by GMS-DS model. Table 3.12 shows the relative error achieved by GMS-GA and GMS-DS methodologies. It can be observed that the relative error achieved by GMS-GA is lesser than GMS-DS model. For the first time step, relative error achieved for S1and S2 sources by GMS-GA are 9.06%, 19.76%. The relative error achieved by GMS-DS model are 10.97 %, 25.05% respectively. It is also noted in the table that average relative error achieved by GMS-GA model is 7.97 % whereas that is obtained by GMS-DS model is 8.23 %. This clearly indicates that the

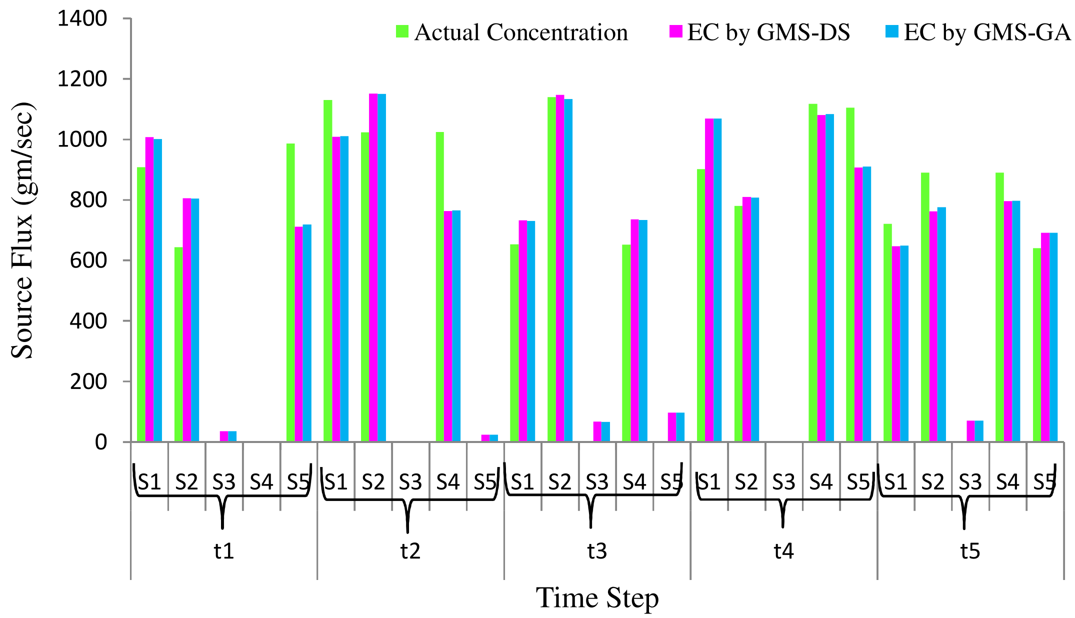


Fig. 3.19: Source Fluxes obtained by using GMS-DS, GMS-GA and comparison with actual value for second illustrative study area

GMS-GA model is better than GMS-DS model, i.e. the performance of genetic algorithm (GA) is better than the direct search (DS) algorithm in solving the inverse optimization problem. Fig. 3.20 shows the comparison of the relative errors of GMS-GA and GMS-DS models.

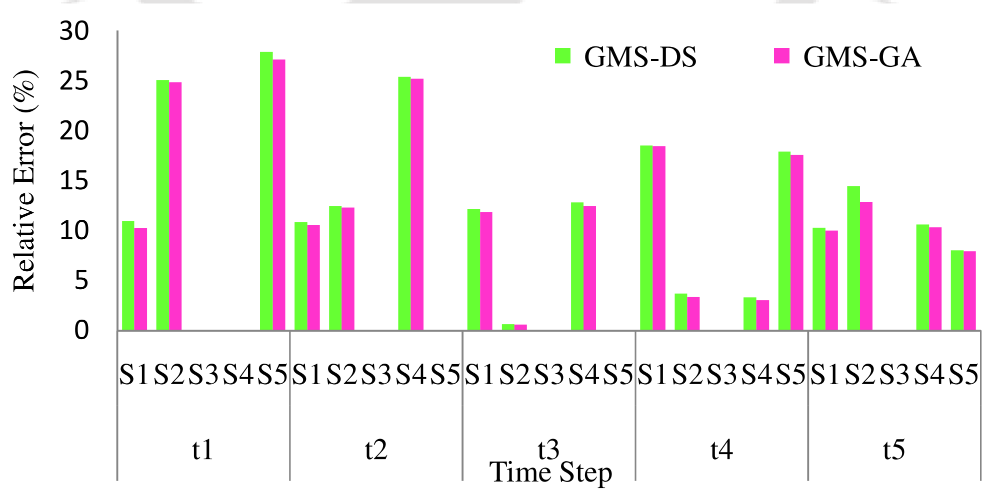


Fig. 3.20: Comparison of relative error (%) achieved by GMS-DS and GMS-GA models for second study area

Table 3.12: Comparison of relative error of GMS-DS and GMS-GA models for second study area

Time Step	Source Location	Relative Error (%)	
		GMS-DS	GMS-GA
1	S1	10.97	9.06
	S2	25.05	19.76
	S3	-	-
	S4	-	-
	S5	27.88	24.16
2	S1	10.81	10.59
	S2	12.45	12.29
	S3	-	-
	S4	25.41	25.22
	S5	-	-
3	S1	12.18	11.86
	S2	0.64	0.58
	S3	-	-
	S4	12.8	12.47
	S5	-	-
4	S1	18.5	18.46
	S2	3.69	3.36
	S3	-	-
	S4	3.32	3.02
	S5	17.9	17.58
5	S1	10.25	9.01
	S2	14.43	12.87
	S3	-	-
	S4	10.62	10.32
	S5	8.03	7.92
	AVERAGE	8.23	7.97

3.4 Evaluation of GMS-GA and GMS-DS models with respect to computational time and objective function value

The performance of GMS-DS model and GMS-GA model are also evaluated in terms of computational time and objective function values achieved at the optimal solution. Table 3.13 shows the comparison of the objective function value and computational time requirement for GMS-GA and GMS-DS methods for both the study areas. It may be observed from the table that for the first study area, the time required for solving the problem by GMS-GA model is 2 days 1hour 15 minutes and 24 seconds. On the other

Table 3.13: Objective function value and computational time required by GMS-GA and GMS-DS models

Study Area	Model	Objective function value	Computational Time (DD:HH:MM:SS)
First	GMS-GA	1.34E-06	02:01:15:24
	GMS-DS	1.56E-06	02:03:03:20
Second	GMS-GA	1.34E-06	05:01:54:45
	GMS-DS	4.22E-05	05 03 00 32

hand the time required by GMS-DS model is 2 days 3 hours 3 minutes and 20seconds. This has shown that GMS-GA model is computationally little bit efficient than the GMS-DS model. For the second study area, time required by GMS-GA model to solve the problem is 5 days 1 hour 54 minutes and 45 seconds. On the other hand the time required by GMS-DS model is 5 days 3 hours and 32 seconds. In this case also GMS-GA model is computationally more efficient than GMS-DS model. From the objective function value point of view, both the models have high predicting capability in identification of unknown groundwater pollution sources but GMS-GA model show slightly better results than the GMS-DS model.

3.5 Conclusions

In this chapter we have developed a new simulation-optimization methodology by liking groundwater simulation system GMS with the Matlab based optimization model. The optimization model is solved using direct search (DS) method and also using genetic algorithms (GA). As GMS has the capability to simulate complicated real world groundwater simulation model, the developed methodology can be applied to obtain the unknown pollution sources of real world aquifer system. The performance of the model is evaluated by using an illustrative study area which was used by Mahar and Datta (2000). The evaluation of the results show that both the models, i.e. GMS-DS and GMS-GA are slightly better than the model proposed by Mahar and Datta (2000). Further, the GMS-GA is slightly better than GMS-DS model. The model is also applied to a large scale aquifer system. In this study area also, GMS-GA model evolves as a slightly better than GMS-DS model. This limited performance evaluation shows that methodologies could be promising one for solving real world source identification problem. The main disadvantage of the models is that they are computationally very extensive. It takes up to few days to solve a relatively medium scale source identification problem. The computational time required to solve the problem is directly related to the computational time required by the aquifer simulation model. The computational time can be reduced by using simplified aquifer simulation models. The next chapter deals with the development of source identification model using approximate simulation model.

Chapter 4

Development of ANN Based Simulation-Optimization Models for Identification of Unknown Groundwater Pollution Sources

4. General

In the previous chapter, GMS is linked with optimization model for solving groundwater pollution source identification problem. GMS has the capability to simulate complicated real world groundwater simulation model. As such, the linking of GMS with the optimization model has facilitated to solve complicated real world source identification problem. Performance of the methodology is also better than the embedded optimization approach. However, the approach is computationally extensive. It takes up to few days to solve a source identification problem of a medium scale groundwater aquifer. The computational time can be reduced by using approximate simulation model. In this chapter, we have developed a simulation-optimization model using ANN as the approximate simulator of groundwater aquifer process. The ANN model is linked externally with the inverse optimization model which has been solved using direct search method and genetic algorithms. The performance of the model is evaluated using the two illustrative study areas.

4.1 Methodology Development

4.1.1 Development of simulation-optimization methodology

In order to achieve efficiency in computational time, ANN model is proposed to simulate flow and transport processes in aquifer in place of GMS model. The ANN model is linked with the inverse optimization model for identifying the unknown groundwater pollution sources. Initially, the inverse model is solved using direct search (DS) optimization

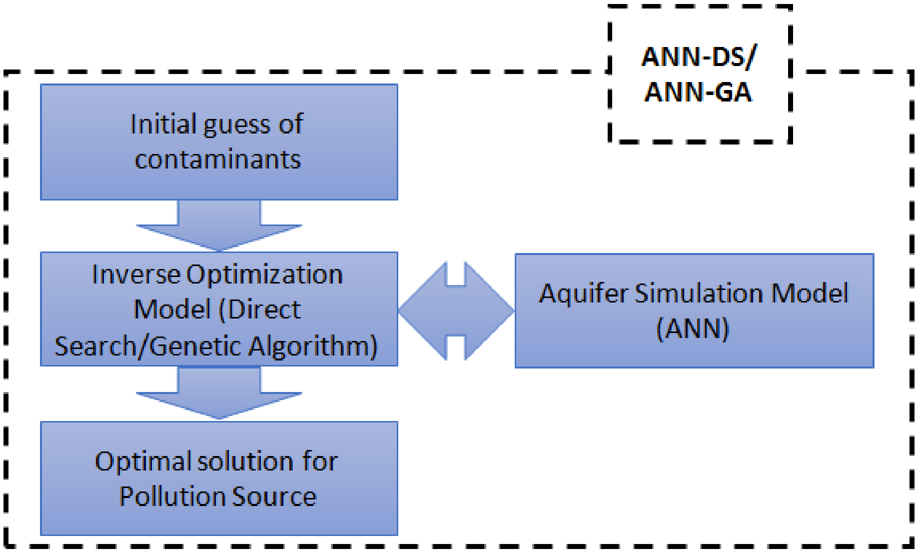


Fig. 4.1: Schematic representation of ANN-DS and ANN-GA Model

algorithm. The approach can be named as ANN-DS mode. The model is also solving using genetic algorithms (GA). This approach is designated as ANN-GA model. Fig. 4.1 shows the schematic representation of ANN-DS and ANN-GA methodologies. The ANN model is developed using ANN toolbox available in Matlab which is then used to

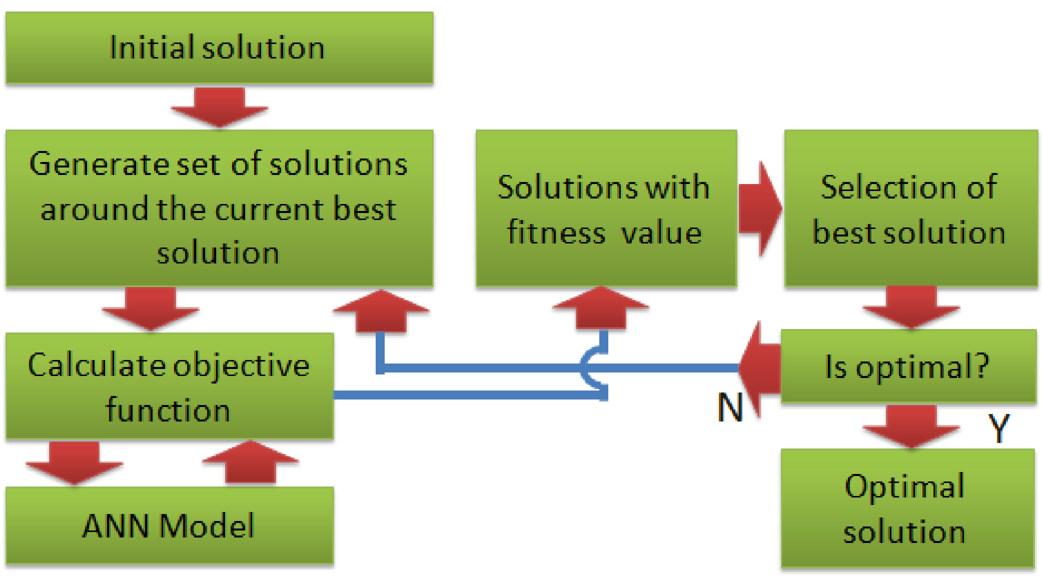


Fig. 4.2: Flow chart of ANN-DS algorithm

calculate the objection function value of the source identification optimization model. Fig. 4.2 shows the flowchart of ANN-DS simulation-optimization model. The ANN simulator is repeatedly called by the optimization model to evaluate the error function. The iterative process is continued until termination criteria are not satisfied.

The model is also solved using Genetic Algorithms. Fig. 4.3 shows the flow chart of ANN-GA based simulation-optimization model. The GA based simulation-optimization algorithm starts with the generation of initial population. The initial population is generated randomly between specified upper and lower bounds. The populations are then sent to the objective function sub-routine for calculation of objective function value of each individual of the population. The objection function considered here is an error function which minimizes the difference between the observed and simulated concentration of contaminants at observation locations. The simulated concentration is calculated using ANN simulator. The solutions are then check for termination criteria. If the termination criteria are satisfied, the optimal solution will be displayed. Otherwise, the solution will pass through the genetic operators to obtain new solutions. This iterative

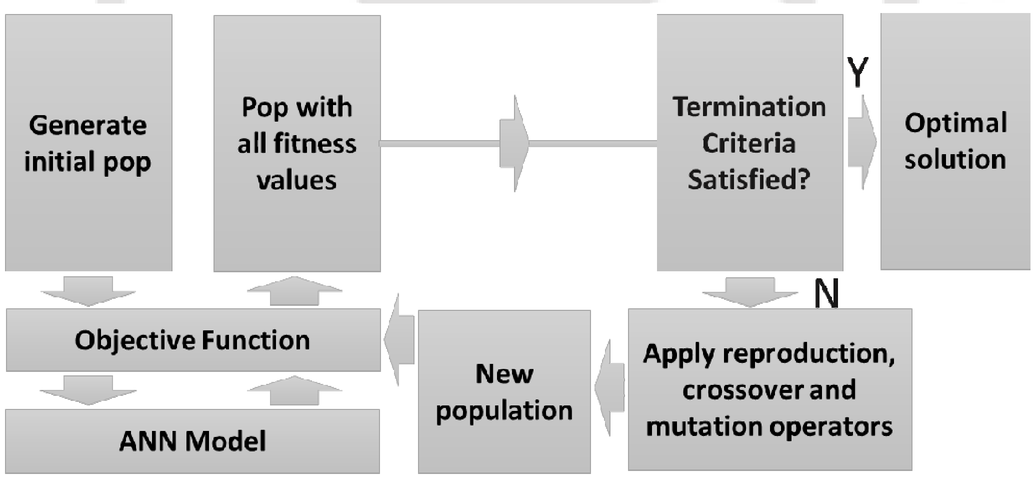


Fig. 4.3: Flow chart of ANN-GA algorithm

process will continue until termination criteria are not satisfied.

4.1.2 Development of ANN model

In order to achieve efficiency in computational time, Artificial Neural Networks (ANN) model is used in place of GMS model for simulating the flow and transport processes in aquifer. We have used single hidden layer standard feed-forward network and the network is trained using back propagation algorithm. The data required for training the ANN model is generated by using GMS model. For solving the source identification problem, we need to find out the simulated concentration at the observation locations. The objective of the simulation model is to predict the concentration at observation wells. Thus the output from the ANN model is the concentration at the observation wells at different time steps. The pollutants sources in the aquifer are the input to the ANN model.

There are eight observation wells in both the study areas. We have developed eight ANN models, each model predicts the concentration at a particular well only. This is done here as the performance of the ANN model is not acceptable when all the observation wells are considered in a single model.

4.1.2.1 Development of ANN model for first study area

An ANN model is developed to simulate the flow and transport processes for the first study area (Fig. 3.8). The ANN model has three neuron layers such as input, output and hidden layers. There are three pollutants sources in the aquifer and the aquifer is simulated for a period of five years at an interval of three months. The input to the ANN model is the entire source fluxes of five year which is equal to 60. The output of the ANN model is the concentration at the observation wells. There are eight observation wells. As such, total number of output from the ANN model is equal to 160. Around 2000 data are generated using the GMS model for training and validation of the ANN model. As the number of input and output are very large, it is become difficult to train the ANN model in standard workstation. Thus, we initially tried to develop the ANN model using only 500 patterns. The sigmoidal transfer function is used in the hidden neurons and linear transfer function is used in the output neurons. The network is trained using Levenberg Marquardt algorithm. With 60 inputs and 160 outputs, the ANN architecture can be represented as 60-HN-160. Where HN is the number of hidden neurons. We tried to train the ANN model with different values of hidden neurons. However, model performance was not as expected. Three different statistical criteria are used to evaluate the performance of the ANN model. They are average absolute relative error (AARE), threshold statistics (TS) and coefficient of correlation (R). For the best combination of 60-

Table 4.1: Performance of the ANN Model for first study area

Output	AARE	R	Output	AARE	R
O11	7.50	0.006	O51	76.67	0.04
O12	15.70	0.05	O52	18.93	0.95
O13	35.41	0.001	O53	70.12	0.99
O14	516.22	0.16	O54	61.31	0.99
O15	1707.50	0.006	O55	55.24	0.99
O21	116.97	0.38	O61	95.52	0.21
O22	68.96	0.22	O62	91.10	0.94
O23	93.79	0.13	O63	8.29	0.98
O24	1017.41	0.60	O64	22.82	0.99
O25	484.90	0.01	O65	36.46	0.99
O31	87.59	0.002	O71	87.59	0.03
O32	53.00	0.08	O72	2.48	0.007
O33	61.73	0.28	O73	41.02	0.97
O34	2.63	0.69	O74	10.41	0.97
O35	19.84	0.39	O75	11.53	0.89
O41	536.79	0.01	O81	767.23	0.31
O42	57.87	0.39	O82	45.13	0.16
O43	10.14	0.97	O83	3.93	0.77
O44	21.29	0.99	O84	28.80	0.45
O45	36.10	0.04	O85	10.35	0.19

100-160 ANN structure, Table 4.1 shows the performance of the ANN model on the basis of R and AARE values. The R and AARE values for observation well 1, at first and second time steps are 0.006, 7.50 and 0.05, 15.70 respectively. It is observed that, for the other outputs also, the model performance is very poor. Thus it can be conclude that, for this study area, this ANN simulator cannot be used to simulate the flow and transport processes of the aquifer.

In order to improve the performance of ANN model, we have developed eight different ANN models for predicting concentration at observation wells, *i.e.* each ANN model will predict the concentration of a particular observation well only. The problem is now much smaller with input to the ANN model is the source fluxes at source locations, which is equal to 60. The output from the ANN model is the concentration at the observation wells, which is equal to 20. The ANN architecture for the first study area is therefore 60-HN-20. Where HN is the number of hidden neurons. The optimal number of hidden neurons required has been calculated using trial and error method. An experiment is also conducted to obtain the other parameters of the ANN models. The network is trained using Levenberg Marquardt algorithm. Around 2000 data are generated using the GMS model for training and validation of the ANN model. Out of the 2000 patterns, 60% is used to train the networks and 20% each is for validation and prediction of the model.

4.1.2.2 Development of ANN model for second study area

In the second study area (Fig. 3.9), there are five pollution sources and simulation is performed for five years at an interval of three months. As such, total input to the ANN model is equal to 100. The outputs from the ANN model are the concentrations for the entire simulation period at an observation well. As such, the total output of the ANN model is equal to 20. The ANN architecture for the second study area is 100-HN-20, where HN is the hidden neurons. For this study area also, an experiment has been conducted to find out the number of hidden neurons and other model parameters for optimal performance of the model. Based on the experimental study we have used 65 hidden neurons for the second study area. The sigmoidal transfer function is used in the hidden neurons and linear transfer function is used in the output neurons. The network is trained using Levenberg Marquardt algorithm. In this study area also, around 2000 data are generated using the GMS model for training and validation of the ANN model. Out of the 2000 patterns, 60% is used to train the networks and 20% each is for validation and prediction of the model.

4.1.3 Artificial Neural Network (ANN)

We used artificial neural network (ANN) as an approximate simulator of aquifer processes. ANN is considered as a universal approximator which mimics the function of the human brain by acquiring knowledge through the process of learning. During the process of learning, ANN finds optimal weights for the synaptic connections between the artificial neurons of the network. Once the ANN is trained, the model can be used to predict the output based on the information fed to the input nodes. Artificial neural networks are important alternatives to the traditional methods of data analysis and modeling. The major aspects of the artificial neural networks are:

- ✓ Set of processing units called neurons
- ✓ State of activation for the inputs to the network.
- ✓ Output function for each unit.

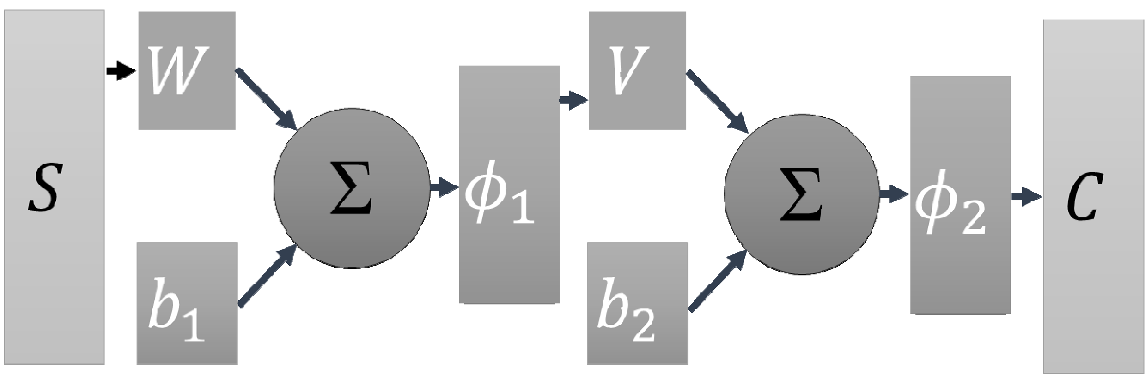


Fig. 4.4 ANN architecture

- ✓ Pattern of connectivity among various units.
- ✓ Learning rule whereby patterns of connectivity are modified by experience.

Artificial neural network (ANN) is a recent tool in the field of function mapping which off late has attracted scientists and technologist from the field of Water Resource Engineering to model complex input output relationship. The ability to gather knowledge through the process of learning, like a human brain, from sufficient input patterns make it possible to apply the ANN scale to real world problems (Bhattacharjya et al, 2009). To date many kinds of neural network architectures have been developed. One of the popular neural network architectures is the feed forward network model. Feed forward ANNs allow signal to travel one way only, from input to output. There is no feedback (loops) i.e., the output of any layer does not affect the same layer. Feed forward ANNs tend to be straight forward networks that associate inputs with outputs. A feed forward network has a sequence of layers consisting of a number of neurons in each layer. The output of neurons of one layer becomes input to neurons in succeeding layer. The first layer, input layer, receives data from the outside world and feed to hidden layers. The last layer, output layer, sends information to users. By this way, non-linear relationship is developed in the network. After choosing the network architecture, the network is trained by providing the data in the form of several input output pairs. During the training process, the network adjusts its weights to minimize the error between predicted and actual output. This network requires much time for training. The feed forward back-propagation has one-way connections from input to output layers. They are most commonly used for prediction, pattern recognition, and nonlinear function fitting. Supported feed-forward networks include feed-forward backpropagation, cascade-forward backpropagation, feed-forward input-delay backpropagation, linear, and perception networks (Hassoun, 1999). The Artificial Neural Network is discussed in details in ASCE Task Committee (2000), Hassoun (1999), Rao *et al*. (2004), *etc.*

This study uses feed-forward neural network. Fig. 4.4 shows the schematic representation of the ANN structure. The input output relation for the network can be written as,

$$\boldsymbol{C} = \emptyset_2[\boldsymbol{V}.\emptyset_1(\boldsymbol{W}.\boldsymbol{S} + \boldsymbol{b_1}) + \boldsymbol{b_2}] \tag{4.1}$$

Where, $\boldsymbol{S}$ is the input vector (source values), $\boldsymbol{C}$ is the output vector (concentration), $\boldsymbol{W}$ is the weight matrix for the synaptic connections between input and hidden layer, $\boldsymbol{V}$ is the weight matrix for the synaptic connections between hidden and output layer, $\boldsymbol{b_1}$ is the biases in the hidden layer, $\boldsymbol{b_2}$ is the biases in the output layer, $\emptyset_1$ is the transfer function for the neurons in hidden layer and $\emptyset_2$ is the transfer function for the neurons in the output layer.

4.1.3.1 Advantages of Artificial Neural Network

The ANN modeling has a number of distinct advantages. First, no mathematical algorithm is required to build the model. The networks simply learn from the sample data and generate a black-box-type relationship. Conventional modeling requires mathematical algorithms to describe these uncertainties, where a neural network simply learns the process based on past experience shown through the data (DibiKe and Solomatine, 1999).

Second, the ANN modeling approach is fast and flexible. Since there is no need to build a physical model and generally no special laboratory tests are required, even a complicated neural network model can be completed relatively quickly once the data are collected. In addition, if some changes in the treatment process are necessary, the network can be quickly adjusted to the new processes through model training, in which the new data describing the new processes are added to the network's learning procedure (Kisi and Asce, 2004). Third, the ANN can handle nonlinear relationship well, due to its inherent non-linear data structure and computation process (ASCE Task Committee, 2000). The capacity for modeling the nonlinear relationship makes the neural network modeling well suited for forecasting. Fourth, neural networks tend to be inherently fault-tolerant, as their data structure is loosely organized and there is no boundary limit on the input parameters. Depending on the significance of the parameters in the neural network architecture, the impact of the false input can be either blocked within the data structure or reduced in magnitude due to the nonlinear calculations inside the data structure. This fault- tolerance feature is critical for the real-time process control (ASCE Task committee, 2000).

4.1.3.2 Limitation of Artificial Neural Network System

Currently, ANNs are usually simulated in software and thus limited by the power of our computers. Even quite simple networks can take weeks to train. Their long evolutionary history gives human brains a big advantage over ANNs – a lot of structure (e.g. modularity) and knowledge is innate, and does not need to be learned. Other factors (e.g. learning rates) have also been optimized over many generations. One can simulate evolution for ANNs, but again can run into resource limitations. Another problem is that choosing appropriate input, output and internal representations for ANNs can be far from straightforward. There remains considerable uncertainty over what representations are used in real brains (e.g. invariant representations). Often it makes sense to model processes at a higher level of abstraction. We know that our brains are neural networks and that they can manipulate symbols, but training an ANN is unlikely to be the best way to go about building an artificial intelligence system that have good reason to believe is most easily formulated in terms of manipulating symbols (James and David, 1991).

4.1.3.3 Learning rule for Feed-Forward Back-propagation (FFBP)

Generally feed forward networks are trained using a back-propagation (BP) learning algorithm. Back propagation refers to the propagation of error back through the network from output layer through hidden layers towards input layer. The calculation of output is carried out, layer by layer, in the forward direction. Then, the weights of hidden layers are adjusted by propagating the error back through the network (DibiKe and Solomatine, 1999). The neural network structure, in this study possessed a three-layer learning network consisting of an input layer, hidden layer and an output layer. Back propagation needs feed forward neural network, where in interlayer connections feed the neuron's output into the neurons in the next forward layer. Training of the artificial feed forward neural network by using back propagation algorithm involves two passes. In the forward pass the input signals propagate from the network input to the output. In the reverse pass, the calculated error signals propagate backward through the network, where they are used to adjust weights. The calculation of output is carried out, layer by layer, in the forward direction. In the reverse pass, the weights of the output layer are adjusted first, since the target value of each output neuron is available to guide the adjustment of associated weights. Then the weights of hidden layers are adjusted by propagating the error back through the network (ASCE Task committee, 2000). In fact, the development of backpropagation is one of the main reasons for the renewed interest in artificial neural networks. The input-output pairs are used to train a network until the network can approximate a function (Haykin, 1999). Backpropagation provides a computationally efficient method for changing the weights in a feed-forward network (ASCE Task Committee, 2000).

The general mathematical forms of the output units in FFBP network is as follows:

$$net_{pj}^{h} = \sum_{i=1}^{N} w_{ji}^{h} x_{pi} + \theta_j^h \quad (4.2)$$

where input vector, $x_p = (x_{p1,} x_{p2,......} x_{pN})^t$, is applied to the input layer of the network. The input units distribute the values to the hidden, w_{ji}^h is the weight on the connection from the ith input unit, and θ_j^h is the bias term (this term is a weight on a connection that has its input value always equal to 1).

Assume that the activation of this node is equal to the net input; then, the output of this node is

$$i_{pj} = \int_j^h (net_{pj}^h) \quad (4.3)$$

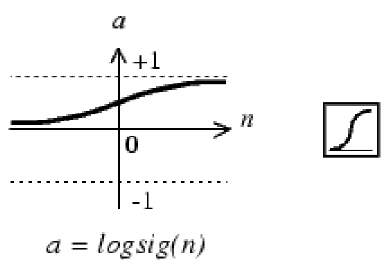
The equations for the output nodes are

$$net_{pk}^{o} = \sum_{j=1}^{L} w_{kj}^{o}\, i_{pj} + \theta_k^{o} \qquad (4.4)$$

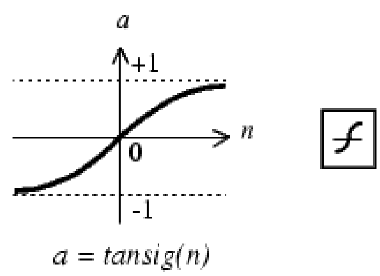
$$o_{pk} = \int_k^{o}(net_{pk}^{o}) \qquad (4.5)$$

where the "o" superscript refers to the quantities on the output layer.

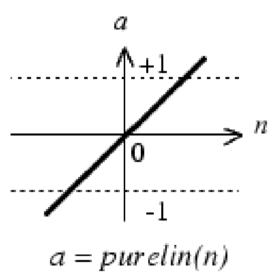
4.1.3.4 Levenberg-Marquidect Algorithm



Log-Sigmoid Transfer Function



Tan-Sigmoid Transfer Function



Linear-Transfer Transfer Function

Fig. 4.5 Transfer function of feed forward neural network

The Lvenberg-Marquardt (LM) Algorithm is an iterative technique that locates the minimum of a function that is expressed as the sum of squares of nonlinear functions. It has become a standard technique for nonlinear least-squares problems and can be thought of as a combination of steepest descent and the Gauss-Newton method. In mathematics and computing, the Levenberg–Marquardt algorithm is also known as the damped least-squares method.

4.1.3.5 ANN transfer function

The tansig, logsig and purelin transfer function are most commonly used for developing the ANN model. It is the ratio of the output of a system to the input of a system. Mathematically the transfer function is a function of complex variables. Fig. 4.5 shows the transfer function of the feed-forward neural network.

4.1.3.5.1 Neuron Model (tansig, logsig, purelin)

Multilayer networks often use the log-sigmoid transfer function logsig. The function logsig generates outputs between 0 and 1 as the neuron's net input goes from negative to positive infinity. Alternatively, multilayer networks may use the tan-sigmoid transfer function tansig. Occasionally, the linear transfer function purelin is used in backpropagation networks. If the last layer of a multilayer network has sigmoid neurons, then the outputs of the network are limited to a small range. If linear output neurons are used the network outputs can take on any value. In backpropagation, it is important to be able to calculate the derivatives of any transfer functions used. The three transfer functions described here are the most commonly used transfer functions for backpropagation, but other differentiable transfer functions can be created and used with backpropagation if desired.

4.1.3.5.2 Update of Output-Layer Weights

From the delta rule, the error at a single output unit to be $\delta_{pk} = (y_{pk} - O_{pk})$, where the subscript "$p$" refers to the p^{th} training vector, and 'k' refers to the k^{th} output unit.

The error that is minimized by sum of the squares of the errors for all output units

$$E_p = \frac{1}{2}\sum_{k=1}^{M} \delta_{pk}^2 \quad (4.6)$$

To determine the direction in which to change the weights, calculate the negative of the gradient of $E_p, \nabla E_p$, with respect to the weights, w_{kj}. Then we can adjust the values of the weights such that the total error is reduced. Each component is considered separately to make it simple. So from equation (4.6)

$$E_p = ½\sum\left(y_{pk} - O_{pk}\right)^2 \quad (4.7)$$

and $$\frac{\partial E_p}{\partial w_{kj}^o} = -(y_{pk} - O_{pk}) \frac{\partial \int o}{\partial (net_{pk}^o)} \frac{\partial (net_{pk}^o)}{\partial w_{kj}^o} \tag{4.8}$$

We have used equation (4.5) for the output value, o_{pk}, and the chain rule for partial derivation. The last factor in equation (4.8) is

$$\frac{\partial (net_{pk}^o)}{\partial w_{kj}^o} = \left(\frac{\partial}{\partial w_{kj}^o} \sum_{j=1}^{L} w_{kj}^o \, i_{pj} + \theta_k^o \right) = i_{pj} \tag{4.9}$$

Combining Eqs. (4.8) and (4.9), we have the negative gradient

$$-\frac{\partial E_p}{\partial w_{kj}^o} = \left(y_{pk} - O_{pk}\right) \int_k^o (net_{pk}^o) i_{pj} \tag{4.10}$$

As far as the magnitude of the weight change is concerned, we take it to be proportional to the negative gradient. Thus, the weights on the output layer are updated according to

$$w_{kj}^o(t+1) = (t) + \nabla_p w_{kj}^o(t) \tag{4.11}$$

where $$\nabla_p w_{kj}^o = \eta (y_{pk} - o_{pk}) \int_k^o (net_{pk}^o) i_{pj} \tag{4.12}$$

The factor η is called the learning rate parameter.

There are two forms of the output function that are of interest here:

$$\int_k^o (net_{jk}^o) = net_{jk}^o$$

(4.13)

$$\int_k^o (net_{jk}^o) = (1 + e^{-net_{jk}^o})^{-1} \tag{4.14}$$

The function (4.14) defines the linear output unit. The function (4.14) is called a sigmoid or logistic function. The choice of output function depends on how to choose to represent the output data. If we want the output unit to be binary, we should use a sigmoid output function. In other cases, either a linear or a sigmoid output function is appropriate. In the first case, $\int_k^o = 1$; in the second case, $\int_k^o = \int_k^o (1 - \int_k^o) = o_{pk}(1 - o_{pk})$. For these two cases, we have

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \eta (y_{pk} - o_{pk}) i_{pj} \tag{4.15}$$

For the linear output,

$$w_{kj}^{o}(t+1) = w_{kj}^{o}(t) + \eta\left(y_{pk} - o_{pk}\right)o_{pk}(1 - o_{pk})i_{pj} \quad (4.16)$$

4.1.3.5.3 Network sizing, Weights and Learning Parameters

Generally, three layers are sufficient to forming a network. Sometimes it seems to be easier to solve more than one hidden layer. In this case, easier means that the network learn faster. The size of the input layer is usually dictated by the nature of the application. Determining the number of units to use in the hidden layer is not usually as straightforward, as it is for the input and output layers. The main idea is to use few hidden- layer units as possible, because each unit adds to the load on the CPU during simulations. If the network fails to converge to a solution, it may be that more hidden nodes are required (ASCE Task Committee, 2000). Weights should be initialized to small, random, that appear in the equations for the input to a unit. It is common practice to treat this bias value as another weight, which is connected to a fictitious unit that always has an output of 1.

So $$net_{pk}^{o} = \sum_{j=1}^{L} w_{kj}^{o}\, i_{pj} + \theta_k^{o} \quad (4.17)$$

By making the definitions, $\theta_k^{o} \equiv w_{k(L+1)}^{o}$ and $i_{p(L+1)\equiv 1}$, we can write

$$net_{pk}^{o} = \sum_{j=1}^{L+1} w_{kj}^{o}\, i_{pj} \quad (4.18)$$

So θ_k^{o} is treated just like a weight, and it participates in the learning process as a weight. Another possibility is simply to remove the bias terms altogether, their use is optional. Selection of a value for the learning rate parameter η has a significant effect on the network performance. Usually, η must be small number. A value of η means that the network will have to make a large number of iterations. Increasing η as the network error decreases, will often help to speed convergence by increasing the step size as the error reaches a minimum. The performance of the FFBP was found to be superior to conventional statistical and stochastic methods in the continuous flow series prediction (ASCE Task Committee, 2000). The FFBP algorithm has some drawbacks such as the local minima problem. In the review study of the ASCE Task Committee (2000), other ANN methods such as conjugate gradient algorithms, the radial basis function, the cascade correlation algorithms and recurrent neural networks were explained to overcome this local minima problem.

4.1.4 Criteria for Model Evaluation

We have applied three statistical criteria for evaluating the performance of the ANN model. Of the several numerical indicators, the three important ones selected for the present study were the average absolute relative error (AARE) (Bhattacharjya *et al.,* 2007), threshold statistics for an absolute relative error level of x% (TSx) (Jain and Kumar, 2006) and coefficient of correlation (R) (Bhattacharjya et al., 2007).

4.1.4.1 Average absolute relative error (AARE)

The average absolute relative error is a measure of error in estimating pollutant concentration. The AARE is defined as the average relative error in forecasting pollutant concentration (Jain et al., 2001).The AARE can be estimated using the following equation

$$\text{AARE} = \frac{1}{N}\sum_{1}^{N}\left|\frac{OC_o^k - SC_s^k}{SC_s^k}\right| \text{ X } 100 \qquad (4.1)$$

Here, OC_o^k is the observed concentration at well location o for k^th time steps by GMS model ; SC_s^k is the simulated concentration at observed location o for k^th time steps by ANN model; $\overline{OC_0^k}$ is the mean of observed concentration at well location for k^th time steps by GMS model; $\overline{SC_s^k}$ is the mean of simulated concentration at observed location o for k^th time steps by ANN model; n_x is the number of data points for which ARE is less than x%, N is total number of data points computed.

4.1.4.2 Absolute relative error (ARE)

The absolute relative error is a statistical parameter which is considered as performance statistics to quantify the performance of the predicted model (Jain *et al.,* 2004). The absolute relative error can be written as

$$\text{ARE} = \left|\frac{OC_o^k - SC_s^k}{SC_s^k}\right| \text{ X } 100 \qquad (4.20)$$

4.1.4.3 Coefficient of Correlation (R)

The R value close to 1 indicates good agreement between the observed and predicted values (ASCE Task Committee on Definition of Criteria for Evaluation of Watershed

Models 1993). MSE is a network performance function. It measures the network's performance according to the mean of squared errors.

$$R=\frac{\sum_1^N (OC_O^k-\overline{OC}_0^k)(SC_S^k-\overline{SC}_S^k)}{\sqrt{\sum_1^N (OC_O^k-\overline{OC}_0^k)^2 (SC_O^k-\overline{SC}_0^k)^2}} \tag{4.21}$$

4.1.4.4 Threshold Statistics (TSx)

Thresholds statistics were computed for ARE levels of 5%, 10%, 20%, 30 %, 40 % and 50% in this study. It is noted that lower AARE values and higher TSx values would indicate good model performance (Jain and Kumar, 2006). Correlation coefficient values close to 1.0 indicate good model performance. (Jain and Kumar, 2006). The TS and AARE statistics measure the effectiveness of a model in terms of its ability to accurately predict data from a calibrated model (Jain et al. 2001). The TSx can be expressed

$$\text{TS}x=\frac{n_x}{N}\text{ X }100\% \tag{4.22}$$

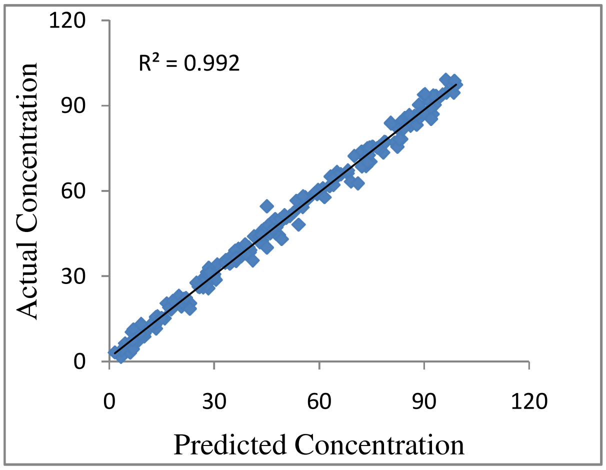
Where n_x is the number of pattern whose ARE value is less than x%, N is the total number of pattern.

4.2 Results and Discussion

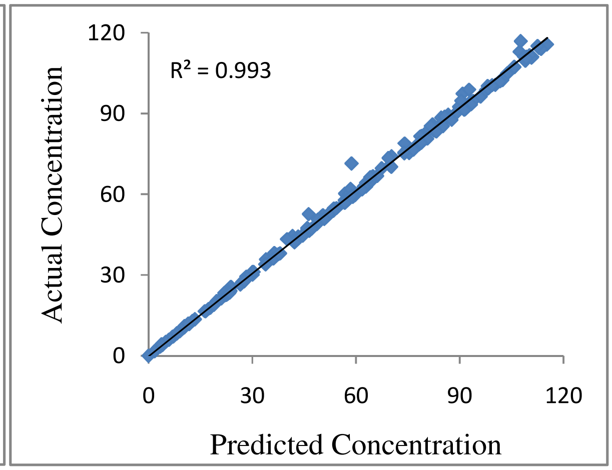
4.2.1 Performance of the ANN model for first study area

The performance of the ANN model as an approximate simulator of the flow and transport processes in grounwater aquifer is evaluated on the basis of three stastical criteria, *i.e.* average absolute relative error (AARE), threshold statistics (TS) and coefficient of correlation (R). The performance of the model is also evaluated for error free and erroneous data.

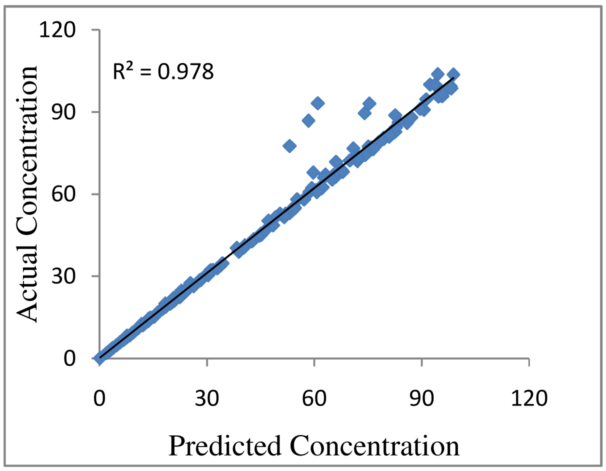
Fig. 4.6 to Fig. 4.13 show the scatter plots between ANN computed and GMS simulated outputs for first study area. The x-axis of the plot represents the predicted concentration by the ANN model and y-axis represents the concentration simulated by the GMS model. Fig. 4.6 shows the scatter plots for observation well 1 at five time steps. It can be observed that the linear fit line makes an angle approximatly equal to 45^{o} with the x-axis and coefficient of correlation value is 0.9922. This shows that the ANN prediction is quite



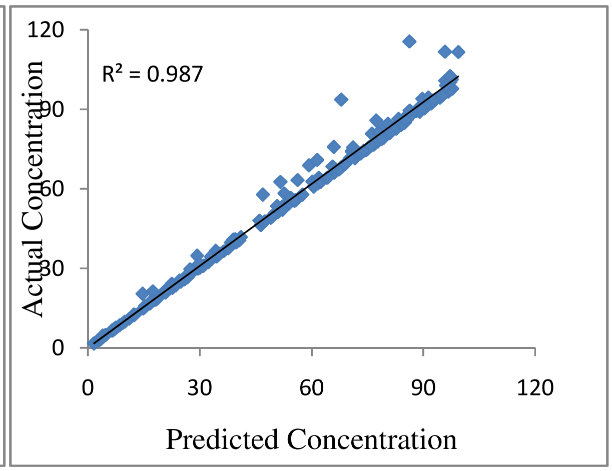
Output 11



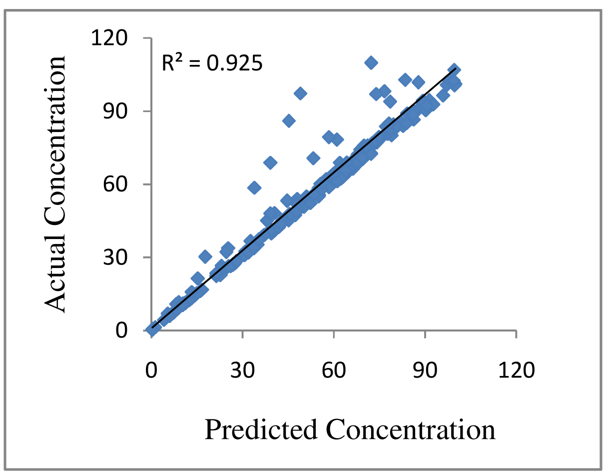
Output 12



Output 13

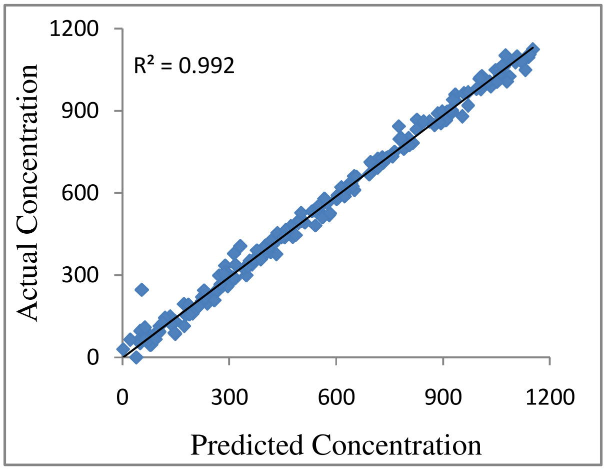


Output 14

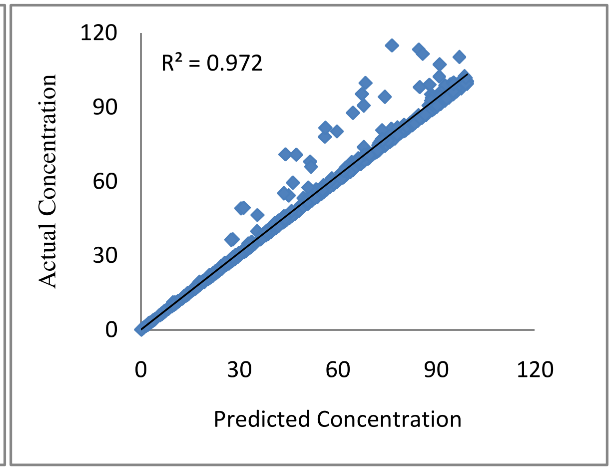


Output 15

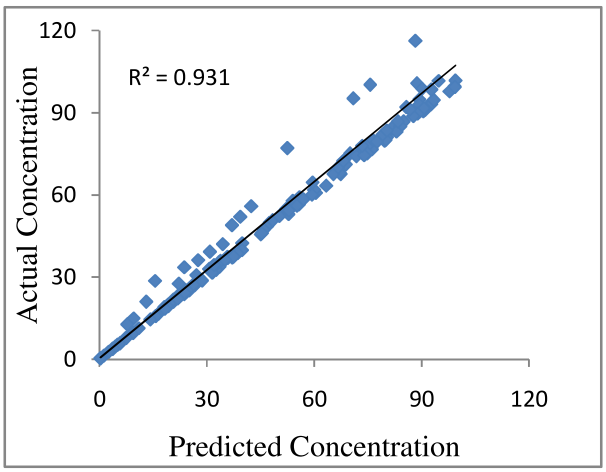
Fig. 4.6: Scatter Plot for Well 1 for first study area



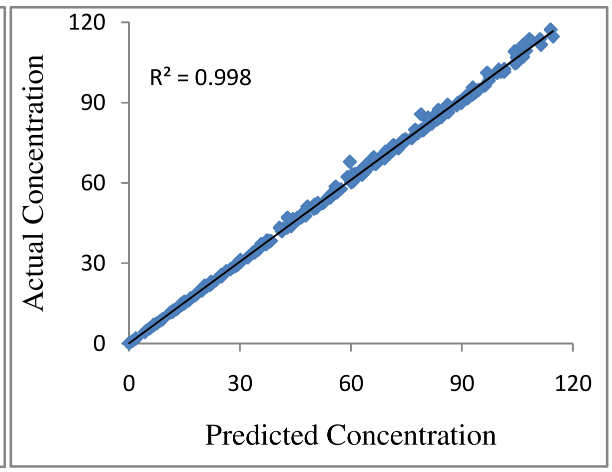
Output 21



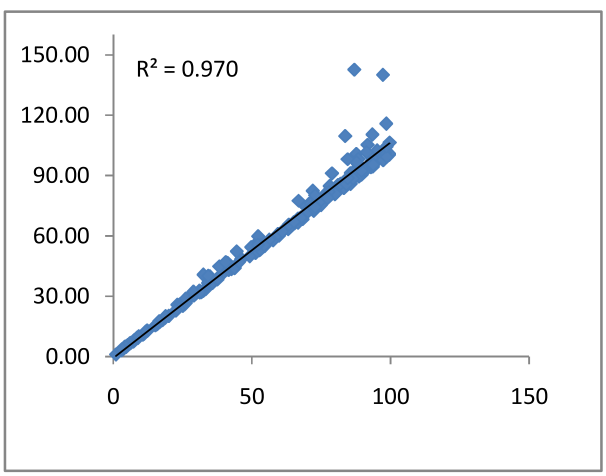
Output 22



Output 23

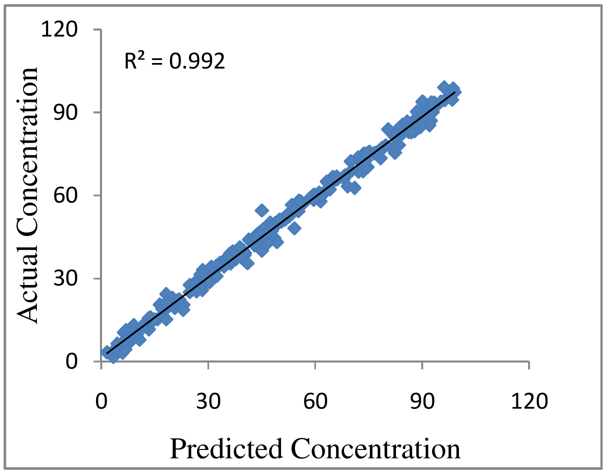


Output 24

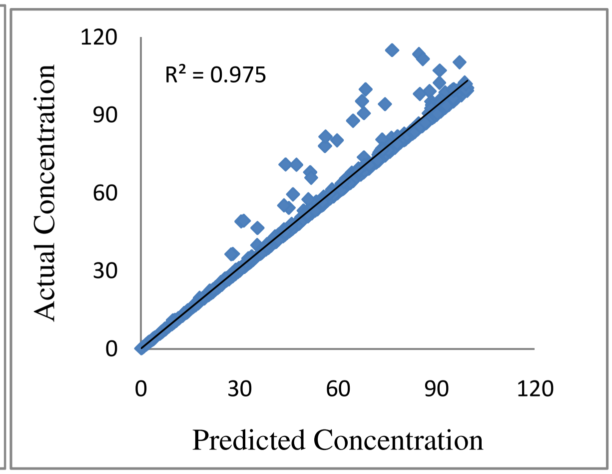


Output 25

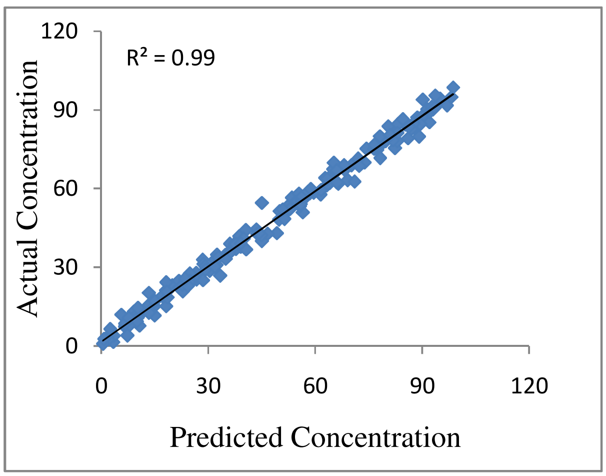
Fig. 4.7: Scatter Plot for Well 2 for first study area



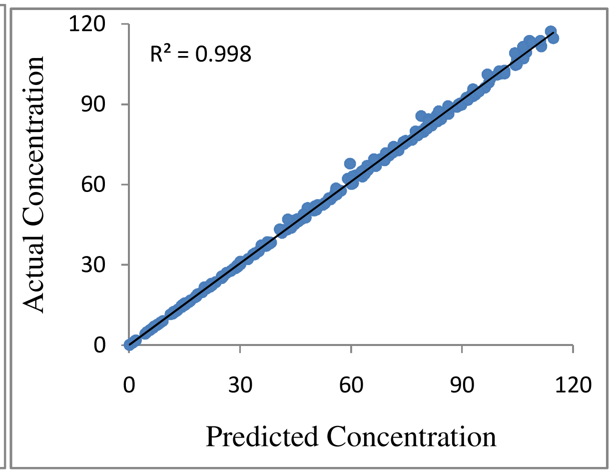
Output 31



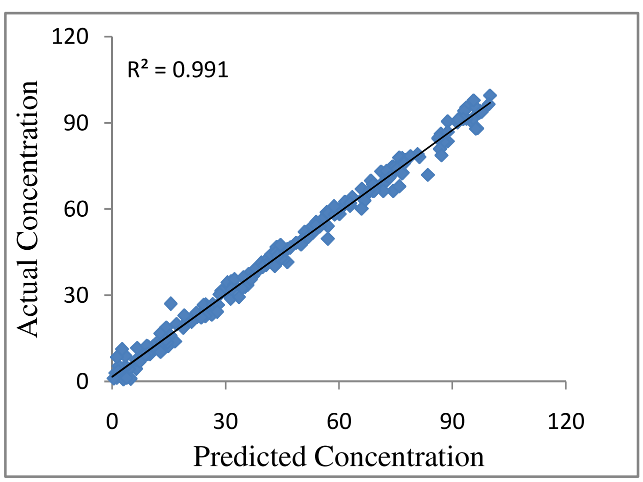
Output 32



Output 33

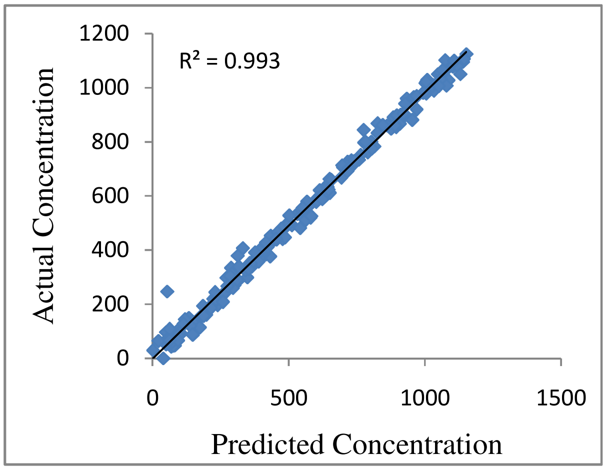


Output 34

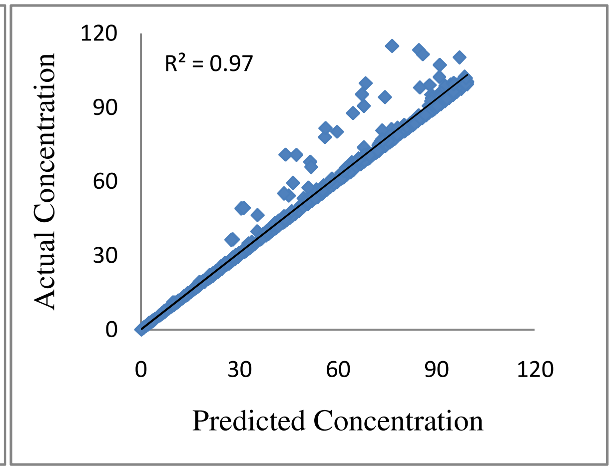


Output 35

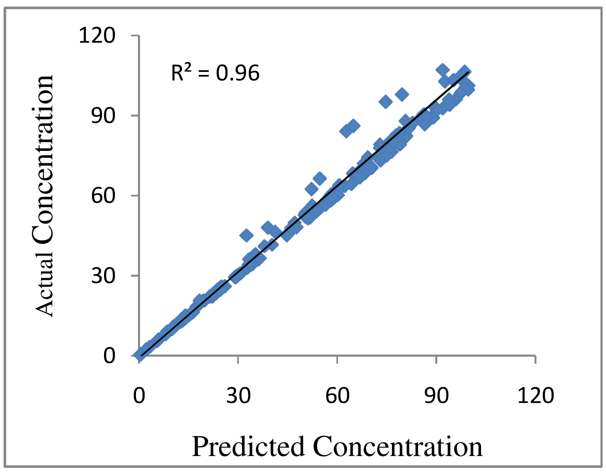
Fig. 4.8: Scatter Plot for Well 3 for first study area



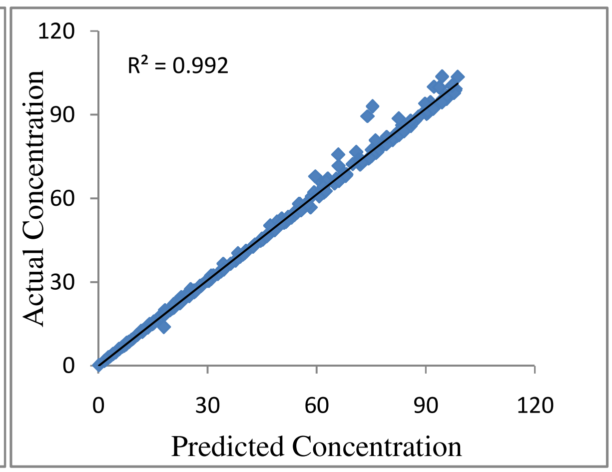
Output 41



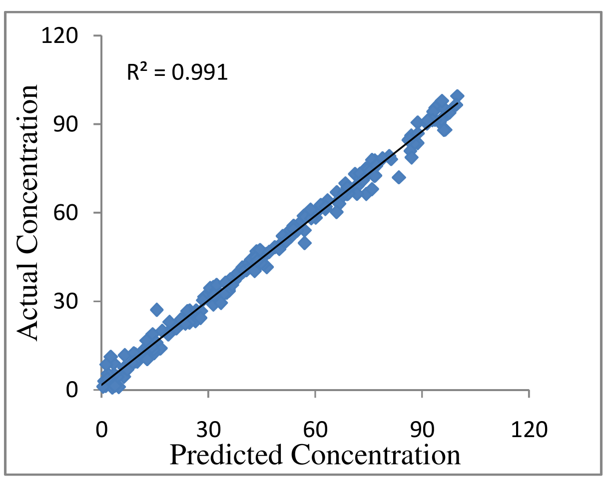
Output 42



Output 43

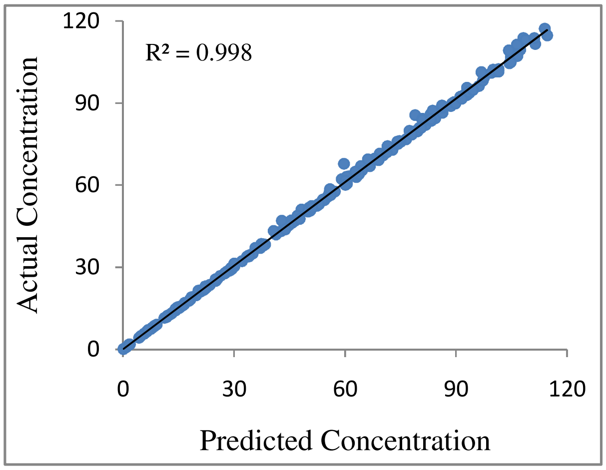


Output 44

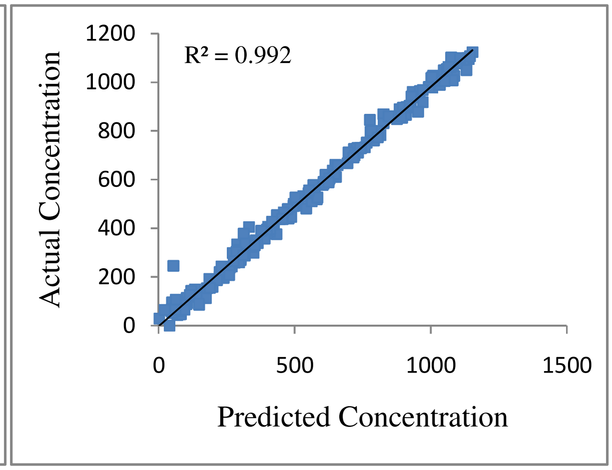


Output 45

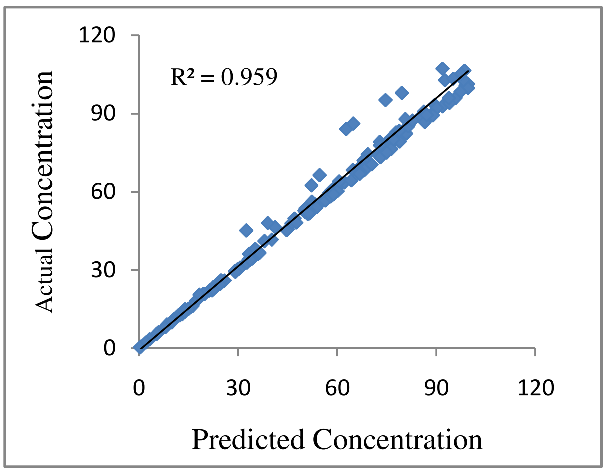
Fig. 4.9: Scatter Plot for Well 4 for first study area



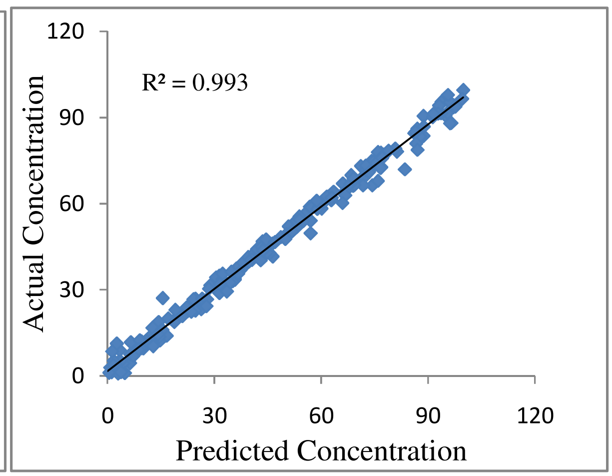
Output 51



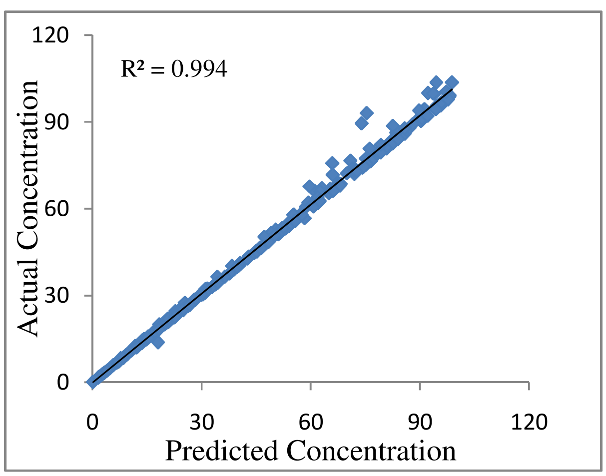
Output 52



Output 53

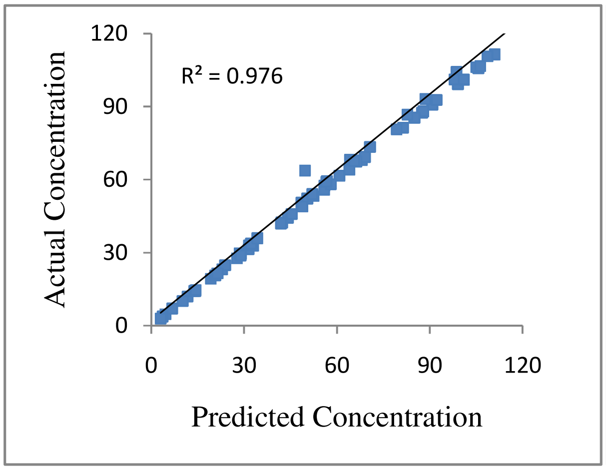


Output 54

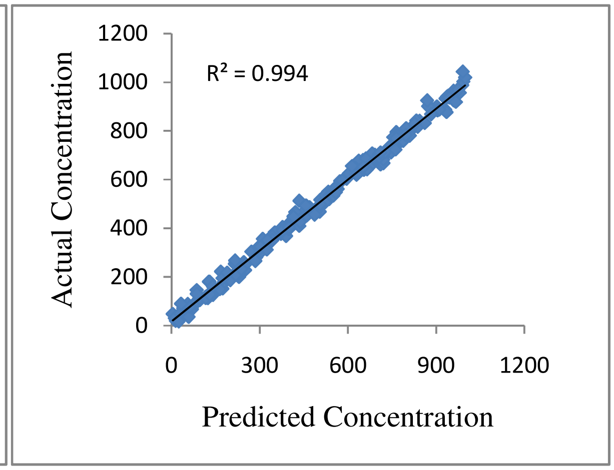


Output 55

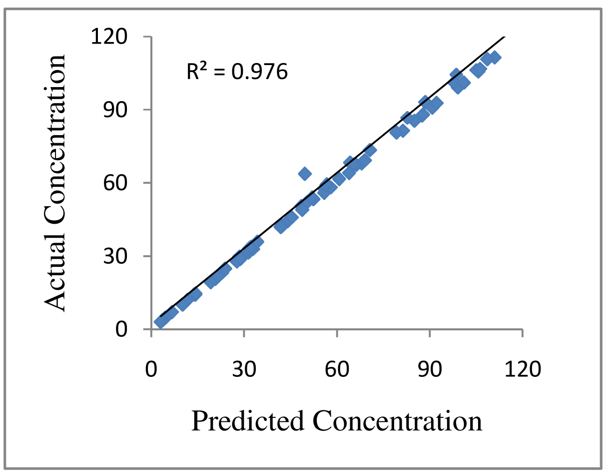
Fig. 4.10: Scatter Plot for Well 5 for first study area



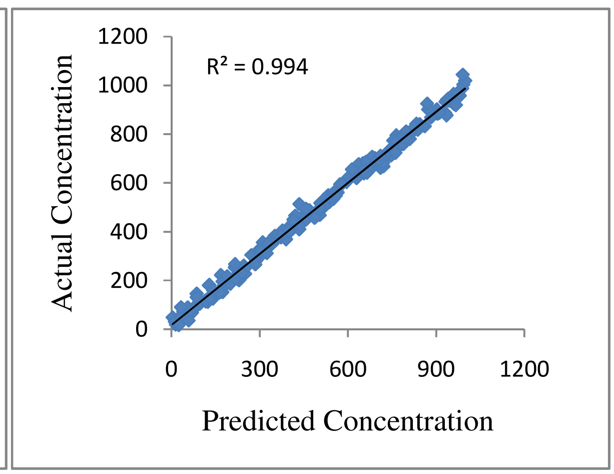
Output 61



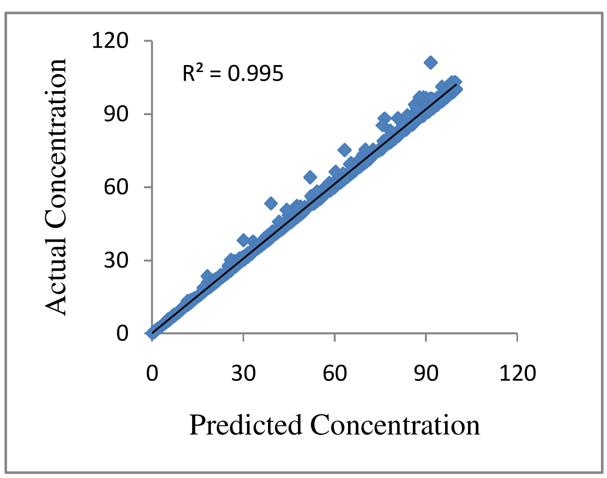
Output 62



Output 63

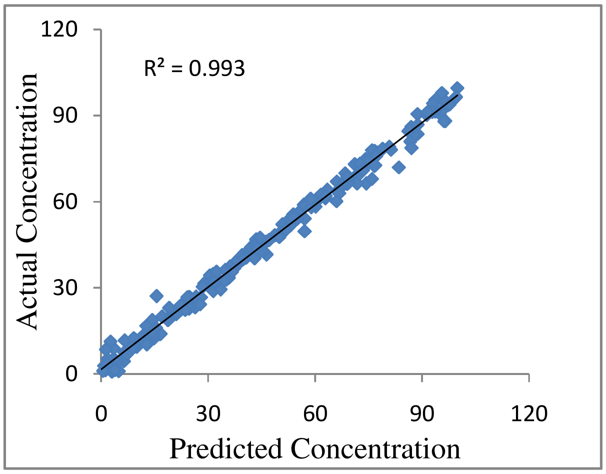


Output 64

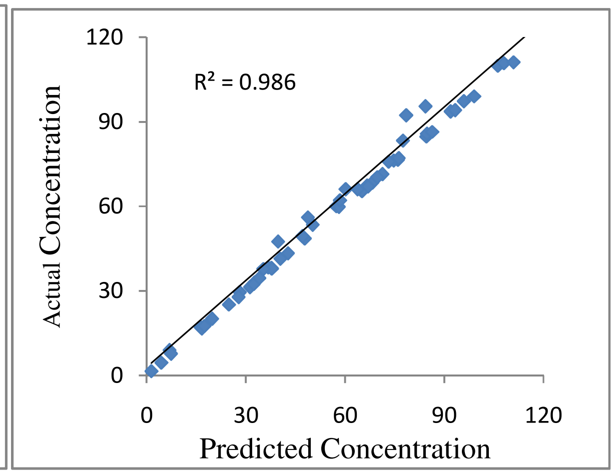


Output 65

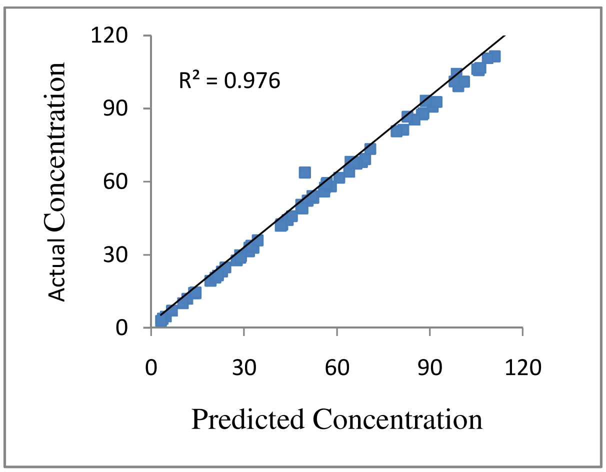
Fig. 4.11: Scatter Plot for Well 6 for first study area



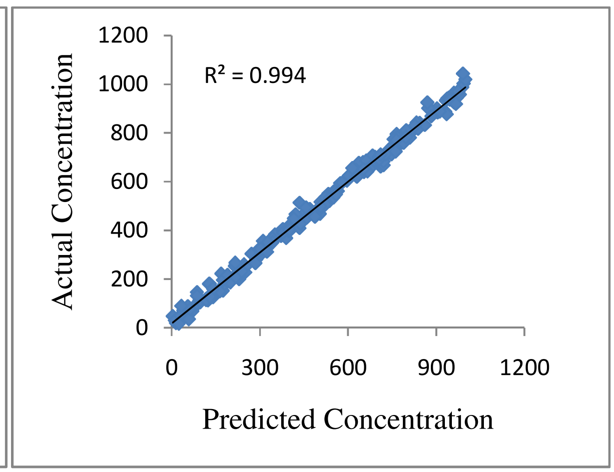
Output 71



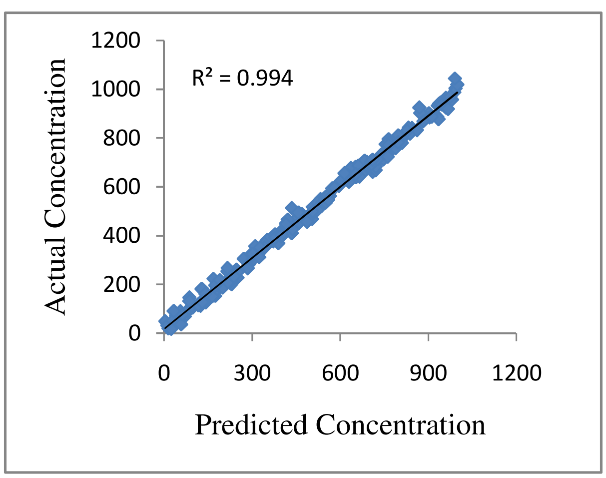
Output 72



Output 73

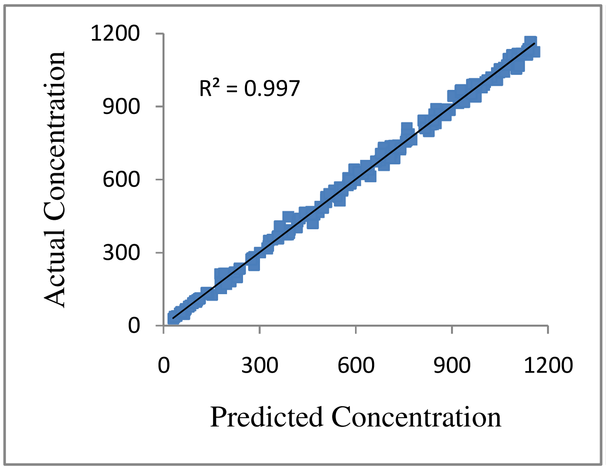


Output 74

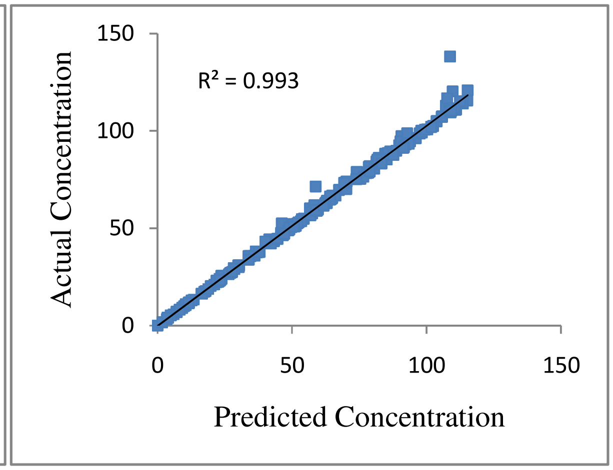


Output 75

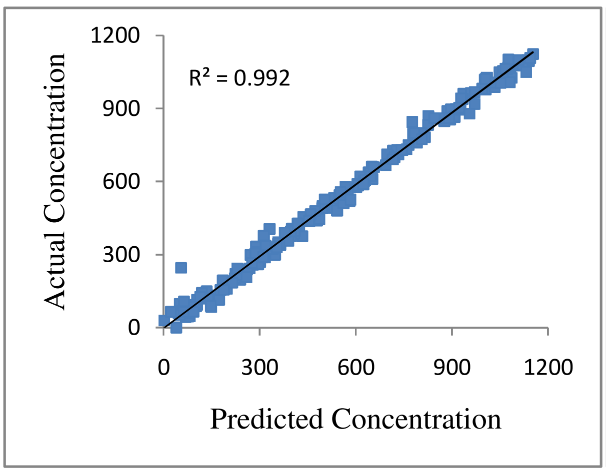
Fig. 4.12: Scatter Plot for Well 7 for first study area



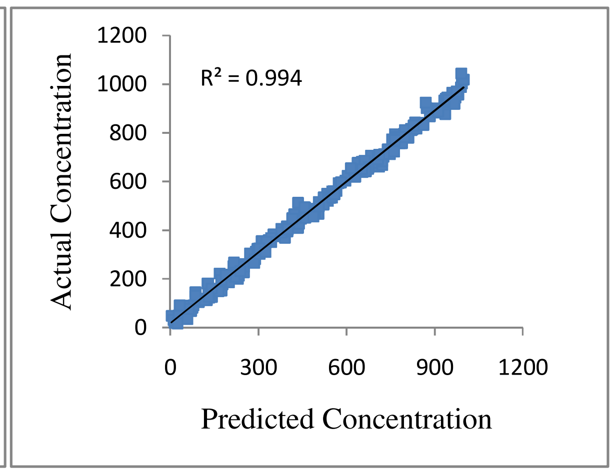
Output 81



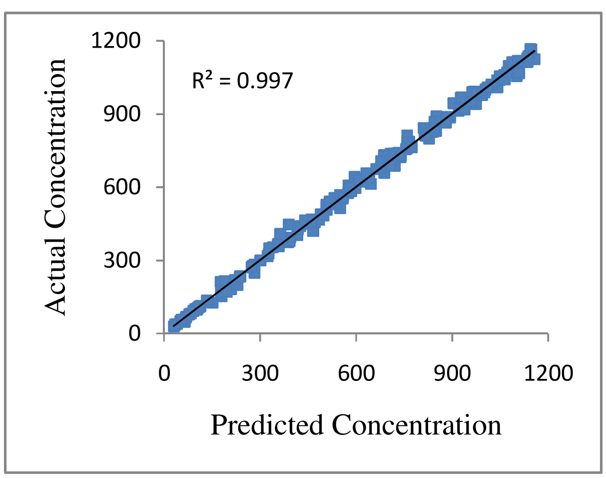
Output 82



Output 83



Output 84



Output 85

Fig. 4.13: Scatter Plot for Well 8 for first study area

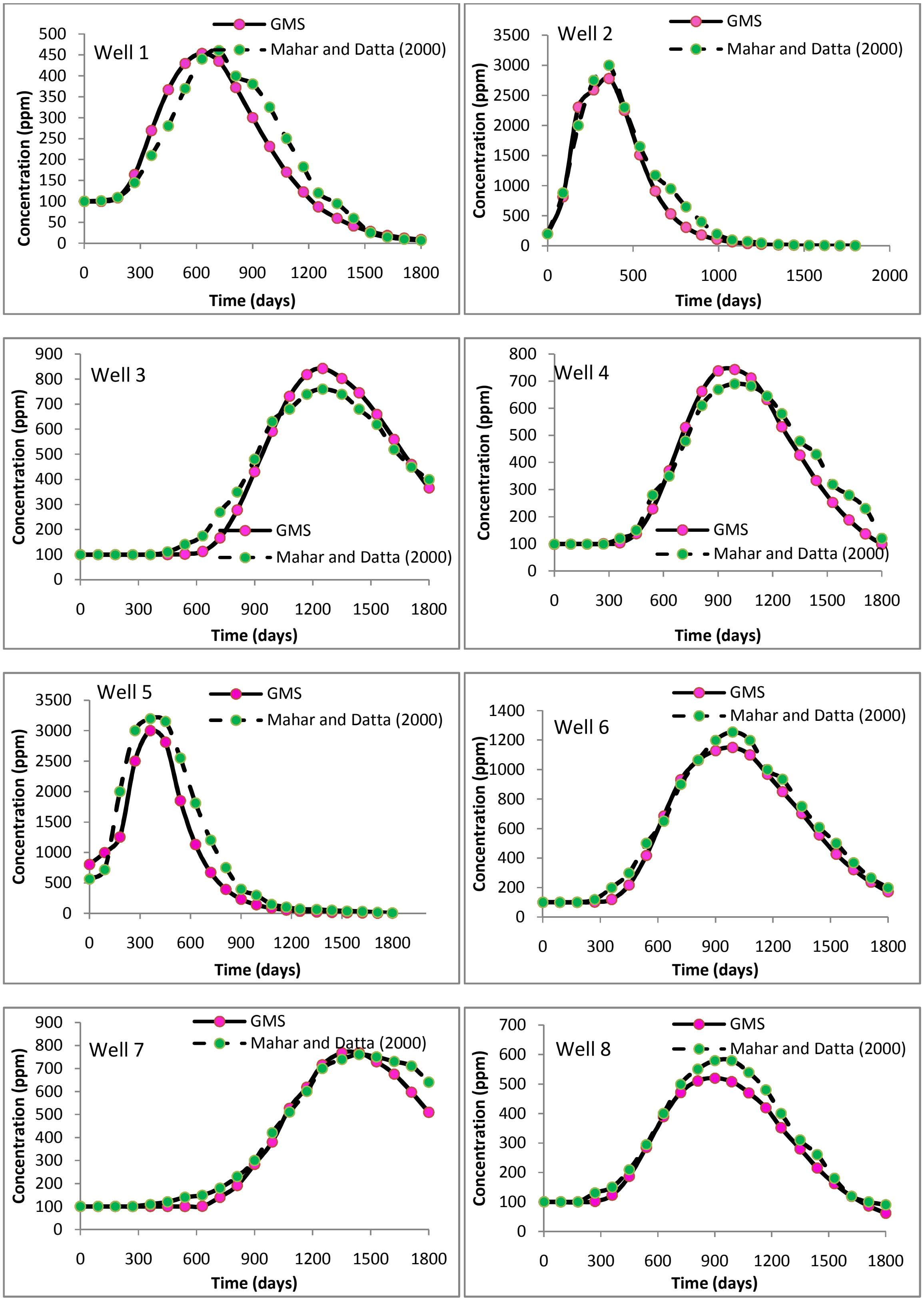


Fig. 4.14: Shows the breakthrough curve generated for eight observation locations of first study area

good. Fig. 4.7 to Fig. 4.13 show the scatter plots for observation wells 2, 3, 4 , 5, 6, 7 and 8. It can be observed from the figures that, the ANN prediction and GMS simulation results are almost similar for these wells. These show that that ANN model has high predicting capablity in simulating the flow and transport processes in groundwater aquifer. Fig. 4.14 shows the breakthrough curve generated for eight observation locations of first study area by using ANN and GMS models. It is observed in the figure that the breakthrough curves are comparable and follow the same pattern.

4.2.2.1 Performance of ANN Model with error free data

Initially, the performance of the model is evaluated for error free data, i.e. the model is trained using the data generated using the GMS model. The performance of the model is evaluated on using average absolute relative error (AARE), threshold statistics (TSx) and coefficient of correlation (R) criteria. Table 4.2 shows the value of AARE, R and TSx for different output at different times. The AARE and R values for the first observation well at first time step is 3.93 and 0.985 respectively. For the second time step, the AARE and R values are 2.26 and 0.997 respectively. These values are quite encouraging and show the good predicting capability of the ANN model. For the other observation wells also, the value of AARE and R values are quite encouraging and in acceptable region (ASCE Task Committee, 2000). The highest and lowest value of AARE is 7.11 and 1.33 with an average value of 3.050. For the co-efficient of co-relation (R), the highest and lowest value of R is 0.998 and 0.943 respectively with an average value of 0.984. The performance of the model is also superior in terms of TSx statistics. The highest value of TS5 is 97.80% and lowest value is 70.04% with an average value of 88.47%. It means that on an average ARE values for that 88.47% pattern are less than 5%. The highest value of TS10, TS20, TS30, TS40 and TS50 are 99.6%, 100%, 100%, 100% and 100% and lowest values are 84%, 90%, 95.80%, 98%, and 99.80% respectively. The average values of TS10, TS20, TS30, TS40 and TS50 are 94.47%, 96.98%, 98.87%, 99.64% and 99.90% respectively. This shows that ARE values for that 88.47% patterns are less than 5%; 94.47% patterns are less than 10%; 96.98% patterns are less than 20%; 98.87% patterns are less than 30%; 99.64% patterns are less than 40% and 99.90% patterns are less than 50%. Thus, it can be concluded that for the first study area, the performance of the ANN model is quite encouraging and the ANN model can be used to simulate the flow and transport processes of the aquifer.

Table 4.2: Performance of the ANN model trained with error free data

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	3.93	0.985	83.20	93.40	95.20	99.40	99.80	100.00
O12	2.26	0.997	93.20	96.20	99.80	100.00	100.00	100.00
O13	2.28	0.997	87.60	97.00	99.60	99.80	100.00	100.00
O14	2.87	0.994	85.00	95.20	98.40	99.80	100.00	100.00
O15	1.39	0.999	98.00	99.60	100.00	100.00	100.00	100.00
O21	5.54	0.98	75.80	88.80	92.60	97.20	98.80	100.00
O22	3.63	0.991	89.20	92.60	94.20	98.20	100.00	100.00
O23	2.34	0.996	93.00	96.40	97.80	99.60	100.00	100.00
O24	1.44	0.999	96.60	99.60	100.00	100.00	100.00	100.00
O25	6.30	0.974	79.20	86.60	90.40	96.00	99.00	100.00
O31	1.32	0.979	72.80	84.00	94.00	96.40	98.00	100.00
O32	1.449	0.998	96.00	99.60	100.00	100.00	100.00	100.00
O33	3.33	0.985	89.60	92.80	95.60	98.60	100.00	100.00
O34	2.841	0.993	88.00	94.00	98.00	99.60	98.60	100.00
O35	2.596	0.994	91.40	95.00	97.80	99.60	100.00	100.00
O41	1.62	0.99	95.05	97.40	100.00	100.00	100.00	100.00
O42	1.34	0.994	96.40	99.60	100.00	100.00	100.00	100.00
O43	4.90	0.943	79.60	90.40	92.40	96.40	98.80	100.00
O44	1.33	0.998	99.40	100.00	100.00	100.00	100.00	100.00
O45	2.69	0.985	92.40	94.40	97.20	99.60	100.00	100.00
O51	1.36	0.998	98.40	99.60	100.00	100.00	100.00	100.00
O52	2.19	0.992	91.20	97.80	99.40	100.00	100.00	100.00
O53	3.81	0.979	91.60	93.40	94.40	96.60	99.00	100.00
O54	3.37	0.98	90.80	93.40	94.80	97.60	99.40	100.00
O55	5.64	0.97	77.80	88.40	90.80	95.80	98.80	100.00
O61	2.90	0.986	89.00	94.60	98.40	99.40	100.00	100.00
O62	1.449	0.997	93.20	99.20	100.00	100.00	100.00	100.00
O63	3.22	0.98	90.00	93.40	96.00	98.60	100.00	100.00
O64	7.11	0.926	70.04	85.80	90.00	96.00	98.60	100.00
O65	2.571	0.987	91.80	95.20	98.60	100.00	100.00	100.00
O71	2.94	0.986	86.80	94.40	98.20	99.60	100.00	100.00
O72	4.89	0.968	79.80	87.20	90.80	96.80	99.20	100.00
O73	1.334	0.998	97.80	99.40	100.00	100.00	100.00	100.00
O74	6.687	0.95	71.60	89.40	95.00	97.00	98.20	99.80
O75	5.533	0.959	75.20	91.60	97.00	98.80	99.40	100.00
O81	1.776	0.996	91.60	98.40	99.80	100.00	100.00	100.00
O82	2.254	0.995	93.60	97.80	99.40	100.00	100.00	100.00
O83	3.201	0.976	91.00	93.80	95.60	99.00	100.00	100.00
O84	1.442	0.998	97.20	99.60	100.00	100.00	100.00	100.00
O85	2.956	0.996	89.00	93.80	98.00	99.20	100.00	100.00
Average	3.05	0.9847	88.47	94.47	96.98	98.87	99.64	99.99

4.2.1.2 Performance of ANN model with erroneous data

The performance of the ANN model is also evaluated using erroneous data, *i.e.* error associated with the pattern due to the imprecise measurement. This error is a random error. Therefore, the training patterns are perturbed by adding normally distributed random error with standard deviation SD and mean zero. We have added error with SD 0.05, 0.10, 0.15 and 0.20. The magnitude of error is more for higher value of SD. Table 4.3 shows the performance of the ANN model when the model is trained with normally distributed error generated for SD values of 0.05. In this case, the highest and the lowest AARE values are 8.95 and 1.78 respectively with an average of 3.754. This value is small and acceptable range. The highest and the lowest value of R is 0.998 and 0.926 respectively with an average value of 0.978. The AARE and R values for all the output are shown in Table 4.3. The performance of the model with erroneous data with SD 0.05 is also evaluated using TSx statistics for x value of 5, 10, 20, 30, 40 and 50. The average values of TS5, TS10, TS20, TS30, TS40 and TS50 are 82.52%, 91.42%, 94.04%, 95.86%, 96.92% and 97.30% respectively. It means ARE value of 82.52% pattern are less than 5%; 91.42% pattern are less than 10%; 94.04% pattern are less than 20%; 95.86% pattern are less than 30%; 96.92% pattern are less than 40% and 97.30% pattern are less than 50%. Table 4.3 also shows the TSx statistics for all the outputs.

Tables 4.4, 4.5, 4.6 show the statistical evaluation of the ANN model for SD 0.10, 0.15 and 0.20 respectively. The average value of AARE for SD value of 0.05, 0.10, 0.15 and 0.20 are 3.754, 4.406, 5.304 and 6.072 respectively. Similarly, the average value of R for SD value of 0.05, 0.10, 0.15 and 0.20 are 0.978, 0.968, 0.9508 and 0.940 respectively. It can be observed that though the performance of the model has deteriorated with the increase in noise level, still the performance of the model is in acceptable range. With respect to the TSx statistics also, the performance of the model is in acceptable range even when the trained with noisy data.

Fig. 4.15 shows the ANN model performance in terms of AARE when the training patterns are perturbed with different level of noise. For the error free data, the AARE is 3.05. This AARE value increases to 3.754, when the model is trained with SD of 0.05 and mean of zero. The figure shows that the AARE value has an increasing trend, however the increase in AARE value is not very significant. This shows that the model performance does not degrade much when trained with noisy data.

Table 4.3: Performance of the ANN Model Trained with erroneous data with SD=0.05

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	4.14	0.969	89.80	93.20	96.00	98.20	99.00	100.00
O12	3.62	0.978	91.60	94.80	95.20	97.20	99.80	100.00
O13	2.92	0.986	91.20	97.80	99.40	100.00	100.00	100.00
O14	3.12	0.982	89.20	92.40	96.40	98.60	100.00	100.00
O15	1.94	0.998	90.40	97.60	99.80	100.00	100.00	100.00
O21	6.06	0.963	68.20	89.60	91.00	94.20	97.00	99.20
O22	4.27	0.97	81.40	92.20	95.00	97.60	99.60	100.00
O23	2.84	0.982	82.20	95.00	98.80	100.00	100.00	100.00
O24	1.97	0.996	91.00	97.60	99.20	100.00	100.00	100.00
O25	7.23	0.942	67.20	88.20	91.60	95.00	97.20	98.80
O31	1.97	0.992	90.00	97.20	100.00	100.00	100.00	100.00
O32	1.86	0.994	89.20	98.60	100.00	100.00	100.00	100.00
O33	3.89	0.976	91.80	93.20	97.00	99.20	100.00	100.00
O34	3.43	0.985	93.2	94.40	96.00	99.80	100.00	100.00
O35	2.881	0.993	89.20	96.40	98.40	99.00	100.00	100.00
O41	1.792	0.998	89.20	98.00	100.00	100.00	100.00	100.00
O42	1.921	0.998	90.20	98.00	99.80	100.00	100.00	100.00
O43	5.32	0.975	70.20	86.80	95.80	98.20	99.40	100.00
O44	1.98	0.998	90.00	98.20	99.80	100.00	100.00	100.00
O45	3.56	0.982	89.40	92.40	95.00	98.00	99.40	100.00
O51	1.78	0.997	92.40	96.40	98.60	100.00	100.00	100.00
O52	2.905	0.995	82.20	94.20	99.20	99.80	100.00	100.00
O53	4.78	0.95	87.60	93.40	94.40	96.60	99.00	100.00
O54	4.31	0.95	88.80	93.40	94.80	97.60	99.40	100.00
O55	6.97	0.93	65.20	88.40	90.80	95.80	98.80	100.00
O61	3.61	0.986	89.00	91.60	95.20	98.20	99.00	100.00
O62	2.68	0.997	93.20	99.20	100.00	100.00	100.00	100.00
O63	4.872	0.959	86.20	91.40	94.40	98.60	99.20	100.00
O64	8.95	0.926	60.6	85.80	90.00	91.60	95.40	98.00
O65	3.271	0.987	91.80	95.20	98.60	100.00	100.00	100.00
O71	3.421	0.986	86.80	94.40	98.20	99.60	100.00	100.00
O72	5.342	0.968	79.80	87.20	90.80	96.80	99.20	98.20
O73	2.19	0.998	97.80	99.40	100.00	100.00	100.00	100.00
O74	7.543	0.949	56.20	81.20	88.80	91.00	95.80	97.60
O75	6.33	0.969	67.20	90.80	94.00	95.20	96.60	97.20
O81	2.15	0.995	90.80	95.20	97.00	99.00	100.00	100.00
O82	2.821	0.986	88.20	94.60	97.40	99.80	100.00	100.00
O83	4.226	0.966	80.40	93.60	94.20	97.40	100.00	100.00
O84	1.789	0.998	89.80	98.40	100.00	100.00	100.00	100.00
O85	3.525	0.988	84.80	92.80	95.00	98.00	99.40	100.00
Average	3.754	0.97825	82.52	91.42	94.04	95.86	96.92	97.30

Table 4.4: Performance of the ANN Model Trained with Erroneous Data with SD=0.10

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	4.89	0.962	87.00	92.40	95.20	98.00	99.20	100.00
O12	3.781	0.979	89.20	91.60	96.20	99.80	100.00	100.00
O13	3.155	0.986	90.00	95.20	98.60	99.00	100.00	100.00
O14	3.67	0.982	88.40	91.20	95.00	98.20	100.00	100.00
O15	2.556	0.997	93.20	97.60	99.20	100.00	100.00	100.00
O21	6.431	0.962	62.00	77.40	86.40	91.60	96.00	98.80
O22	4.853	0.967	77.80	89.00	92.00	96.60	98.20	100.00
O23	3.55	0.982	81.80	93.40	96.60	99.80	100.00	100.00
O24	2.25	0.991	90.20	95.00	98.60	100.00	100.00	100.00
O25	7.79	0.941	59.80	73.00	87.60	92.00	96.40	97.20
O31	2.15	0.994	88.20	96.00	98.80	100.00	100.00	100.00
O32	2.354	0.98	86.00	96.40	98.00	100.00	100.00	100.00
O33	4.561	0.961	79.20	88.80	93.20	96.00	98.00	100.00
O34	4.11	0.966	81.2	89.00	93.80	97.20	99.00	100.00
O35	3.95	0.972	85.00	92.00	96.40	97.80	100.00	100.00
O41	1.822	0.996	90.00	98.60	100.00	100.00	100.00	100.00
O42	2.33	0.993	89.80	97.60	99.40	100.00	100.00	100.00
O43	6.412	0.977	68.80	84.20	94.60	97.40	98.60	100.00
O44	3.56	0.982	89.40	92.40	95.00	98.00	99.40	100.00
O45	4.109	0.977	75.00	92.40	94.20	98.20	99.40	100.00
O51	2.43	0.994	92.40	96.40	98.60	100.00	100.00	100.00
O52	3.09	0.985	91.80	94.40	96.00	98.80	100.00	100.00
O53	5.22	0.95	72.20	84.40	94.40	96.60	98.20	99.00
O54	4.87	0.97	83.60	93.40	94.80	97.60	99.40	100.00
O55	7.11	0.93	68.80	88.40	90.80	95.80	98.80	100.00
O61	3.99	0.98	84.00	91.60	94.00	99.40	100.00	100.00
O62	2.78	0.989	93.20	96.00	98.40	100.00	100.00	100.00
O63	4.95	0.96	82.00	91.60	93.40	97.00	99.00	100.00
O64	8.89	0.919	65.40	79.80	84.20	90.60	92.20	95.80
O65	3.924	0.981	90.20	93.40	95.00	97.40	99.00	100.00
O71	3.776	0.986	86.80	94.40	97.20	98.00	99.60	100.00
O72	5.892	0.94	71.20	79.80	87.00	91.60	95.00	98.40
O73	4.89	0.963	75.00	85.00	89.60	95.20	97.00	100.00
O74	7.8	0.893	68.00	79.20	85.00	91.80	93.40	95.60
O75	6.911	0.92	71.20	78.80	89.00	92.80	94.00	96.20
O81	3.663	0.97	91.60	94.20	97.00	98.00	99.00	100.00
O82	3.67	0.995	91.00	94.00	97.60	98.20	100.00	100.00
O83	5.786	0.949	80.40	85.00	91.60	95.00	97.00	98.20
O84	3.67	0.97	90.20	99.60	100.00	100.00	100.00	100.00
O85	4.651	0.969	81.00	93.80	98.00	99.20	100.00	100.00
Average	4.406	0.968	80.05	88.21	92.21	94.95	96.25	97.07

Table 4.5: Performance of the ANN Model Trained with erroneous data with SD=0.15

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	5.21	0.955	85.20	89.00	91.00	96.00	98.20	99.80
O12	4.384	0.961	87.00	89.00	93.00	97.80	99.20	100.00
O13	3.79	0.979	89.00	93.40	95.40	97.80	98.00	99.20
O14	4.01	0.969	86.80	89.20	91.20	96.00	98.40	100.00
O15	2.86	0.986	91.00	95.60	98.20	100.00	100.00	100.00
O21	6.93	0.93	59.60	75.00	83.20	89.20	95.00	97.00
O22	5.562	0.952	75.00	87.20	89.20	93.40	97.60	99.20
O23	3.97	0.977	78.80	92.00	93.80	97.60	100.00	100.00
O24	2.71	0.988	89.00	93.00	94.60	99.60	100.00	100.00
O25	8.22	0.922	54.80	68.20	85.00	89.00	93.80	98.80
O31	2.43	0.987	87.00	93.20	97.00	99.20	100.00	100.00
O32	2.751	0.979	83.40	92.60	97.00	100.00	100.00	100.00
O33	5.32	0.959	77.80	85.00	91.00	95.20	97.80	100.00
O34	4.89	0.961	79.8	88.00	90.80	96,80	98.20	100.00
O35	4.62	0.97	83.20	91.80	95.00	97.00	99.80	100.00
O41	2.945	0.991	83.40	94.60	99.20	99.80	100.00	100.00
O42	2.089	0.994	90.40	97.80	99.80	100.00	100.00	100.00
O43	7.115	0.967	69.40	84.20	88.60	95.00	98.00	100.00
O44	5.788	0.936	77.80	88.00	90.80	96.00	99.00	100.00
O45	6.059	0.97	68.80	85.20	94.80	97.00	98.00	100.00
O51	3.15	0.988	89.20	93.20	95.80	98.60	100.00	100.00
O52	3.55	0.973	90.00	93.00	95.40	98.00	99.60	100.00
O53	6.02	0.93	69.20	81.40	85.40	91.00	93.00	95.00
O54	4.87	0.96	78.00	84.00	91.80	95.20	97.00	99.00
O55	9.22	0.86	58.60	64.00	85.40	89.00	91.00	93.00
O61	4.23	0.97	81.20	83.40	91.00	95.40	98.00	100.00
O62	4.871	0.959	75.00	87.20	90.20	96.60	98.00	100.00
O63	5.015	0.94	70.00	75.00	83.20	87.60	95.20	98.00
O64	9.762	0.851	54.20	60.60	81.20	85.00	87.00	91.00
O65	4.425	0.959	78.00	84.00	90.20	93.00	95.60	97.40
O71	5.174	0.93	75.00	78.20	87.80	90.00	94.00	97.00
O72	7.823	0.929	68.00	81.00	85.00	89.00	92.80	95.20
O73	5.352	0.93	77.00	81.80	87.60	95.20	97.20	99.20
O74	8.232	0.91	62.00	69.80	76.20	91.00	93.00	95.00
O75	7.44	0.92	69.00	72.00	79.80	94.20	97.60	98.40
O81	4.887	0.961	78.20	89.60	92.20	97.00	99.00	100.00
O82	5.762	0.953	71.60	79.20	89.20	92.00	97.00	99.00
O83	8.534	0.909	72.00	79.00	81.20	85.40	91.40	95.00
O84	5.885	0.931	68.20	78.60	85.20	89.00	96.00	99.00
O85	6.331	0.939	67.00	82.00	87.00	91.00	93.20	98.60
Average	5.3048	0.9508	74.36	82.17	87.80	93.38	94.56	96.20

Table 4.6: Performance of the ANN Model Trained with erroneous data with SD=0.20

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	5.87	0.949	83.00	88.00	89.80	93.40	97.00	99.60
O12	4.89	0.955	84.20	88.00	89.80	95.00	96.40	99.80
O13	4.15	0.961	85.00	91.00	93.00	96.80	97.20	99.00
O14	4.67	0.958	79.80	83.00	87.00	94.40	97.20	100.00
O15	3.29	0.97	86.40	93.00	95.00	99.20	100.00	100.00
O21	7.29	0.921	52.00	71.00	79.80	85.00	92.00	96.40
O22	5.93	0.944	69.80	81.20	86.40	91.00	95.00	99.00
O23	4.471	0.967	74.00	89.00	91.80	96.40	99.00	100.00
O24	3.09	0.981	85.40	91.00	93.20	97.60	99.20	100.00
O25	8.76	0.921	49,80	62.00	79.00	85.60	92.00	97.00
O31	2.98	0.98	85.00	91.00	95.40	97.00	99.20	100.00
O32	3.11	0.961	81.00	89.40	95.00	99.80	100.00	100.00
O33	5.99	0.94	61.40	79.00	85.40	91.20	96.60	99.20
O34	5.64	0.959	62	81.20	85.40	93.20	97.80	99.00
O35	5.112	0.962	69.80	83.40	89.40	95.00	97.80	100.00
O41	3.255	0.965	91.00	93.80	95.40	98.00	99.60	100.00
O42	2.574	0.984	90.40	95.00	97.60	99.60	100.00	100.00
O43	8.87	0.954	62.80	79.00	85.20	92.40	97.20	99.80
O44	6.69	0.96	70.20	86.20	90.20	95.80	98.20	100.00
O45	6.925	0.957	73.00	85.40	89.40	95.40	98.40	100.00
O51	4.05	0.976	83.60	92.40	94.40	98.20	99.20	100.00
O52	4.231	0.97	83.20	92.00	93.60	98.00	99.20	100.00
O53	6.97	0.94	68.00	85.00	89.00	92.00	97.40	99.60
O54	5.73	0.95	72.20	80.00	89.60	91.20	95.00	97.00
O55	9.882	0.89	56.20	62.00	81.00	85.00	88.00	91.00
O61	5.867	0.93	79.00	81.00	85.00	91.00	93.00	97.00
O62	5.233	0.939	68.20	81.40	88.60	91.40	97.60	99.00
O63	6.671	0.911	62.00	68.00	81.00	85.00	93.00	97.00
O64	9.88	0.84	53.00	64.00	71.00	79.00	84.00	89.00
O65	5.541	0.93	75.00	79.20	85.60	89.20	93.00	97.00
O71	6.233	0.927	73.20	75.60	85.00	87.00	91.00	96.20
O72	8.554	0.913	65.20	75.40	81.40	85.00	89.00	93.00
O73	6.923	0.93	71.00	79.00	87.00	89.00	95.00	97.00
O74	9.551	0.893	61.20	71.60	83.00	89.20	91.00	93.00
O75	8.14	0.907	67.80	71.20	75.60	89.00	92.00	95.00
O81	5.324	0.939	69.80	80.20	85.00	93.00	95.00	98.00
O82	6.22	0.923	63.20	69.00	82.40	88.00	95.20	97.80
O83	9.554	0.897	61.20	71.00	84.60	89.40	92.80	94.00
O84	6.98	0.934	69.00	76.60	85.20	87.20	92.00	98.00
O85	7.788	0.927	65.50	79.20	83.00	85.00	92.20	97.80
Average	6.072	0.940	70.47	78.89	84.99	89.61	93.04	95.50

Fig.4.16 shows the variation of R when ANN model is trained with different level of noise. The value of R is 0.987 when trained with error free data. The value decreases to

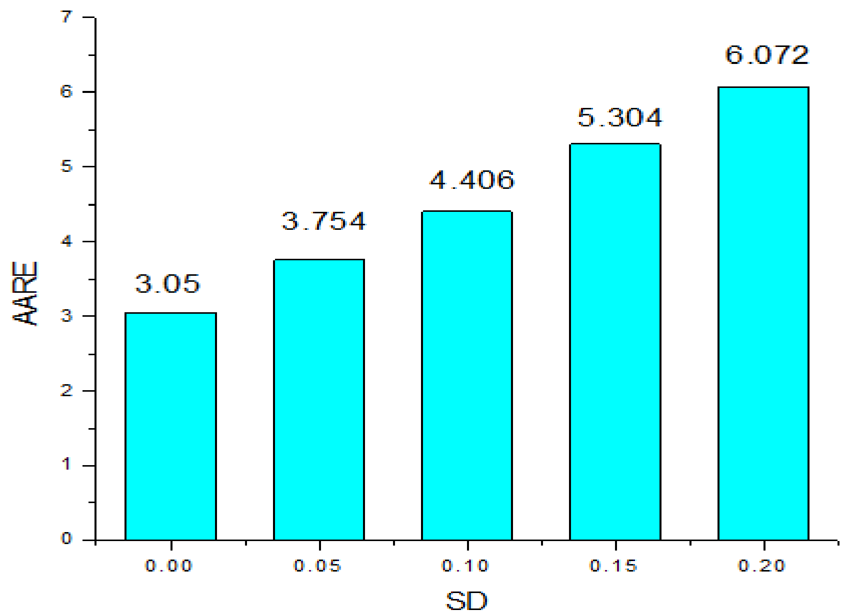


Fig. 4.15: Variation of AARE with increase in Noise level

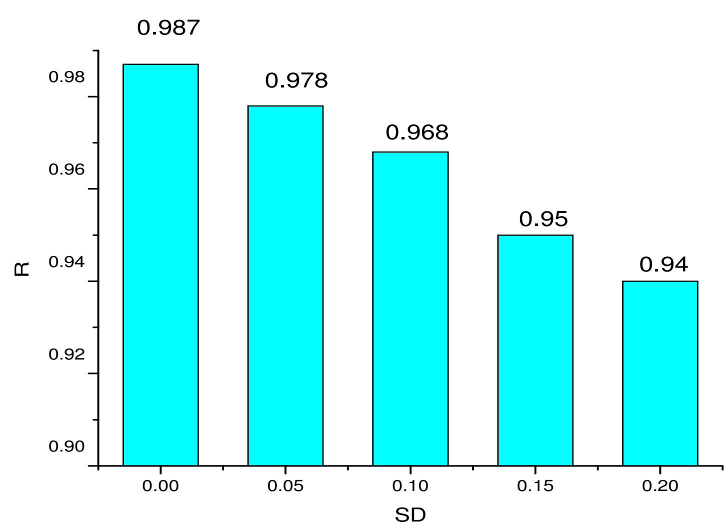


Fig. 4.16: Variation of R with increase in Noise level

0.978, when the model is trained with noise of SD of 0.05 and mean of zero. The value is decreases to 0.968, 0.95 and 0.94 when the model is trained with SD of 0.10, 0.15 and

0.20 respectively. In this case also, it can be seen that the model performance does not degrade much when ANN model is trained with noisy data.

Fig.4.17 shows the performance of ANN model in terms of TSx statistics when the training patterns are perturbed with different level of noise. From the figure, it is clear that the model is very sensitive with respect to the perturbed error for TS5 statistics. The average value of TS5 is varying from 88.47% to 70.47%. It means that 88.47% of

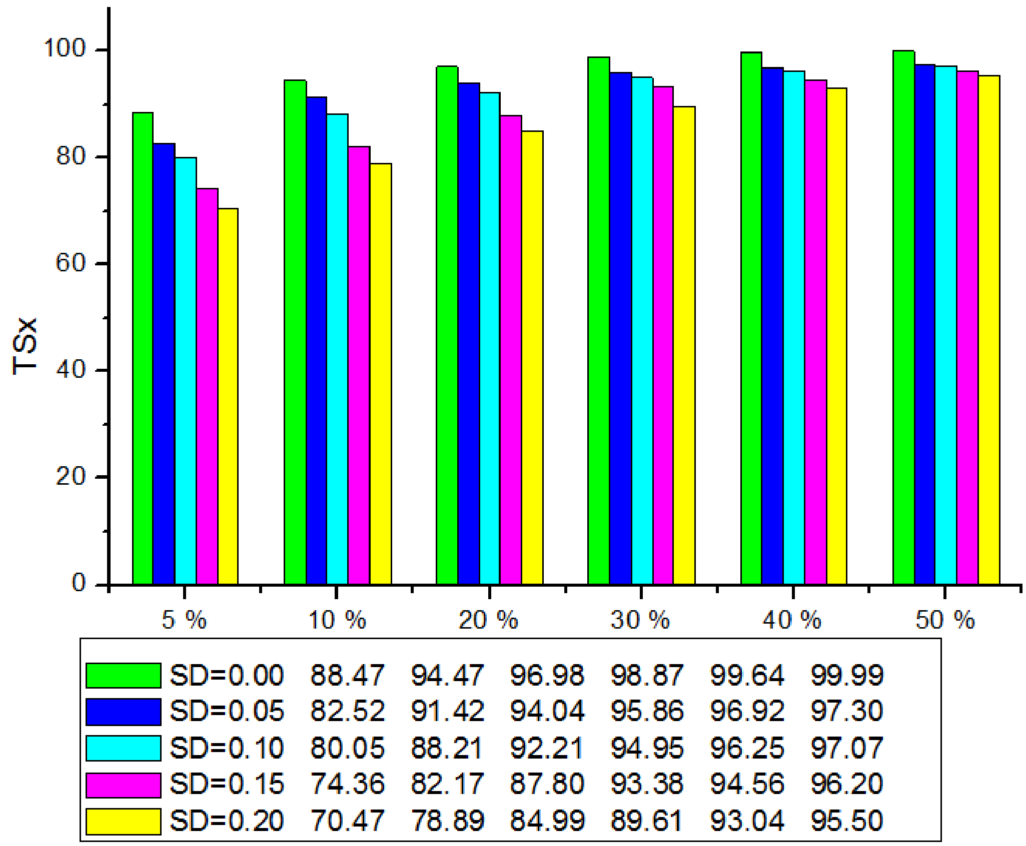


Fig. 4.17: Performance of the ANN model in Terms of TS statistics value with erroneous data

predicted concentration had an ARE value less than 5% when the model is trained with an error free data. This value is decreases to 82.52%, when the training patterns are perturbed with random error of SD of 0.05 and mean zero. This value is further decreases to 80.05%, 74.36%, and 70.47% when the training pattern are perturbed with different level of noise, i.e. SD of 0.10, 0.15, and 0.20 respectively. In case of TS10, 94.47% of predicting concentration has an ARE value lesser than 10% when the model is trained with error free data. TS10 value decreases when the training patters are perturbed with random error generated using SD of 5%, 10%, 15% and 20%. The TS10 value is decreases to 91.42%, when the training patterns are perturbed with SD is equal to 0.05 and mean zero. The value is decreases to 88.21%, 82.17%, 78.89% when training pattern are perturbed with different level of noise. This shows that the model is sensitive to noise in terms of TS10. The model is also sensitive for TS20. TS20 is varying from 96.98% to 84.99%. For the error free data, TS20 value is 96.98%. This value is further decreases to 94.04%, 92.21%, 87.80% and 84.99% when the model is trained with SD is equal to 0.05, 0.10, 0.15 and 0.20. The ANN model is less sensitive for TS30, TS40 and TS50. For example, for the error free data, TS30 is 98.87%. This value is decreases to 89.61% when the training patterns are perturbed with SD is equal to 0.20 and mean zero. In respect to TS50, the model is not very sensitive.

4.2.4 Transfer function and optimization algorithm used in ANN model

Another task is the selection of transfer function and optimization algorithm for the ANN model. An experiment is conducted to find out the transfer functions to be used in the neurons of hidden and output layers. The best transfer function is evaluated on the basis

Table 4.7: Performance of different transfer function for the ANN model

Transfer function Hidden Layer	Transfer function Output Layer	MSE	Computational Time (Sec)	Iteration
tansig	*tansig*	0.092	327	36
tansig	*logsig*	0.066	186	20
logsig	*tansig*	0.107	251	27
logsig	*logsig*	0.07	206	22
logsig	*purelin*	0.0093	218	23
purelin	*tansig*	0.00927	149	16
purelin	*logsig*	0.0172	308	33
purelin	*purelin*	0.017	102	11
tansig	*purelin*	0.0011	195	21

of mean square error (MSE) of the training pattern. The transfer functions tried here are *tansig*, *logsig* and *purelin*. Table 4.7 shows the MSE, computational time and iteration required for different combination of transfer functions for the hidden and output layers. From the table, it is clear that *tansig* and *purelin* transfer functions for hidden layer and output layer give the best combination for network. The *tansig* transfer function for hidden layer and *purelin* transfer function for output layer give the lowest value of MSE, which is 0.0011. For this combination, the computational time required for simulation as well as iteration are also low as compared to other combination of transfer functions.

Table 4.8: Performance of different optimization Algorithms for the ANN model

Training Algorithms	MSE	Computational Time (Sec)	Iteration
traingd	0.398	99	1000
traingdm	0.404	97	1000
traingda	0.0137	28	290
traingdx	0.067	15	115
trainrp	0.002505	111	1000
traincgf	0.000237	118	833
traincgp	0.000168	147	1000
trainscg	0.000226	78	812
trainoss	0.00107	47	359
trainlm	0.000047	235	36

Another experiment is also conducted to select the best optimization algorithm available in matlab toolbox to train the network. The training algorithms available in Matlab are *traingd*, *traingdm*, *traingda*, *traingdx*, *trainrp*, *traincgf*, *traincgp*, *trainscg*, *trainoss* and *trainlm*. Table 4.8 shows the MSE, computational time and iteration values for different training algorithms. It is noted that *trainlm* optimization algorithm gives the lowest value of MSE. As such, we have used *trainlm* algorithm to train the ANN model with *tansig* transfer function in the hidden neuraons and *purelin* transfer function in the output neurons.

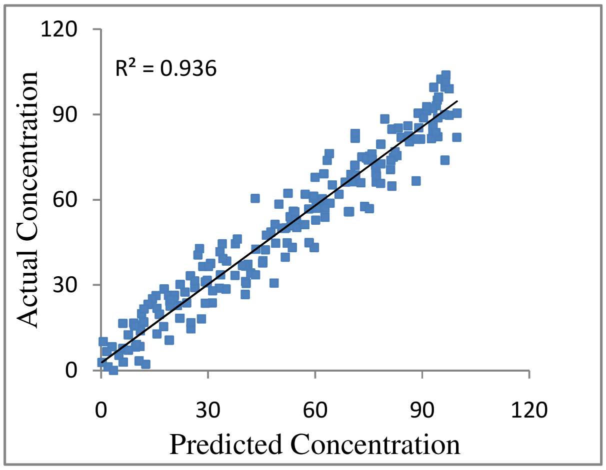
4.2.5 Performance of the ANN model for second study area

For the second study area also, we have evaluated the performance of the ANN model using average absolute relative error (AARE), threshold statistics (TS) and coefficient of correlation (R). Fig. 4.18 to Fig. 4.25 show the scatter plots for eight observation wells. It can be observed that the ANN prediction and GMS simulation results are almost the same. From these figures, it is clear that the actual and predicted concentration are almost same for each output. Therefore, it can be concluded that the developed ANN model has high predicting capability in simulating the flow and transport processes of groundwater aquifer.

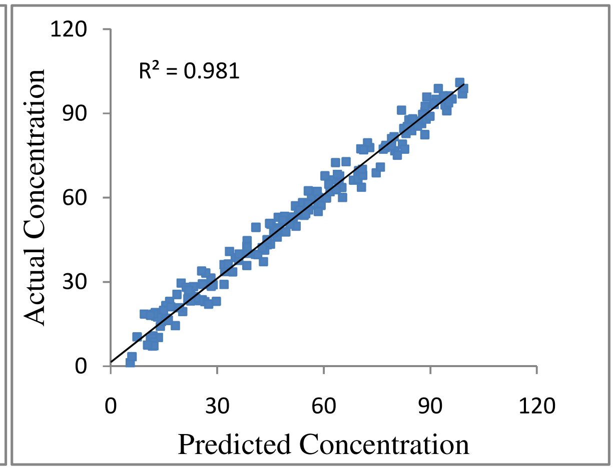
The comparison between the ANN simulation and the GMS simulation are shown in terms of breakthrough curve in Fig. 4.26. It can be seen from the figure that the both the curves are comparable and the results are intuitively as expected.

4.2.6 Performance of ANN Model with error free data

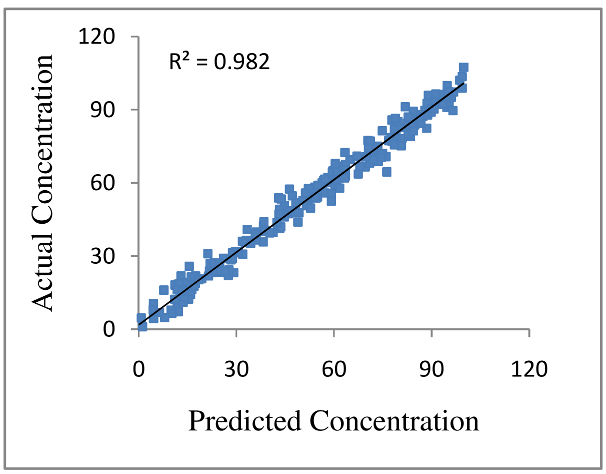
For the second study area, we have also checked the performance of the ANN model on the basis of average absolute relative error (AARE), threshold statistics (TSx) and coefficient of correlation (R) criteria. Table 4.9 shows that average value of AARE and R value are 2.99 and 0.9831 when trained with error free data. The performance of the model is also evaluated for TSx statistics. The highest and lowest value of AARE is 7.45 and 1.11 with an average value of 2.99, which is small and acceptable range. These show that, the performance of the model is superior in terms of AARE value. The highest and lowest value of R is 0.998 and 0.94 respectively with an average value of 0.9831. The R value is in higher side and show better prediction capabilities of the ANN model. The performance of the model is also superior in terms of TSx statistics. The highest value of TS5 is 99.40% and lowest value is 61.40% with an average value of 87.65%. It means ARE values for that 87.65% pattern are less than 5%. The highest value of TS10, TS20, TS30, TS40 and TS50 are 100% and lowest values are 79%, 85.40%, 91.20%, 96.60%, and 98.80% respectively. The average values of TS10, TS20, TS30, TS40 and TS50 are 94.60%, 97.31%, 98.82%, 99.54% and 99.93% respectively. It means ARE values of 87.65% pattern are less than 5%; 94.60% pattern are less than 10%; 97.31% pattern are



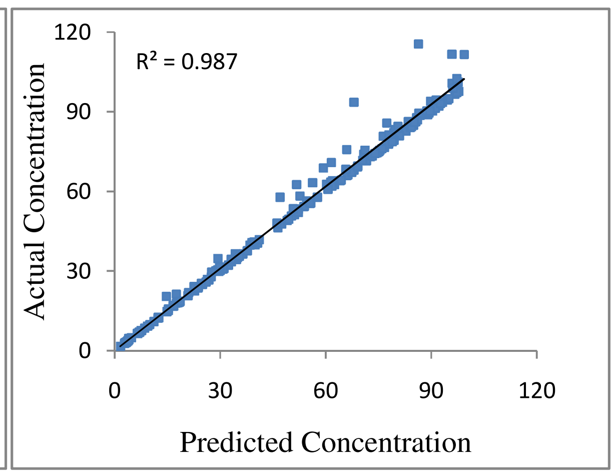
Output 11



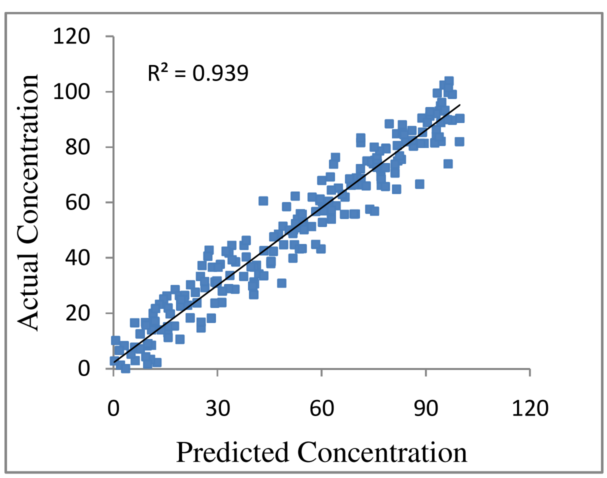
Output 12



Output 13

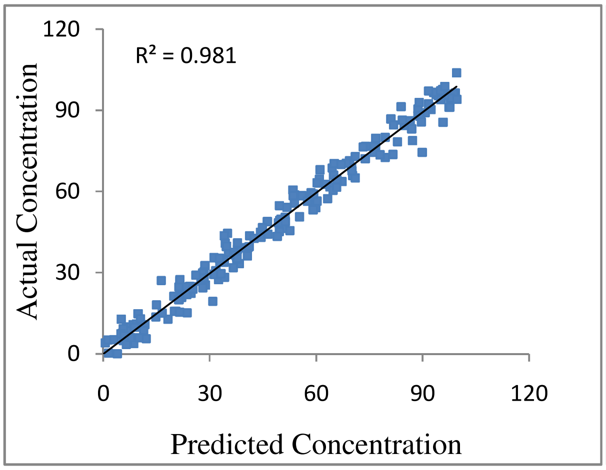


Output 14

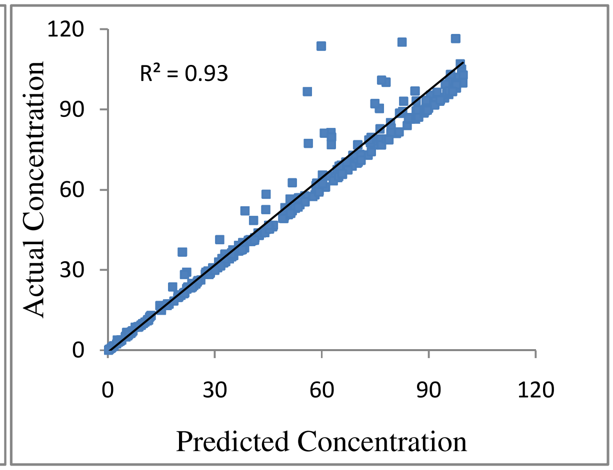


Output 15

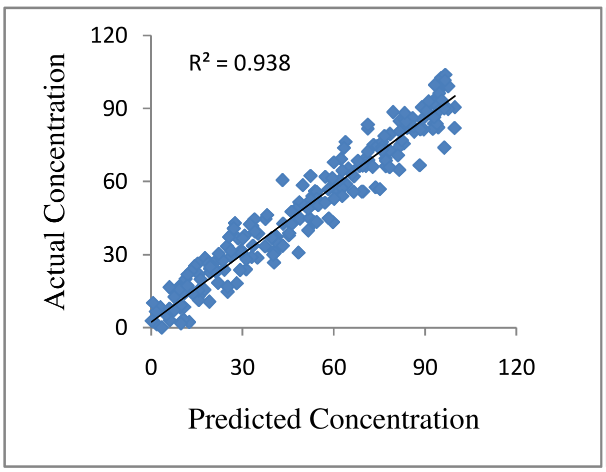
Fig. 4.18: Scatter Plot for Well 1 for second study area



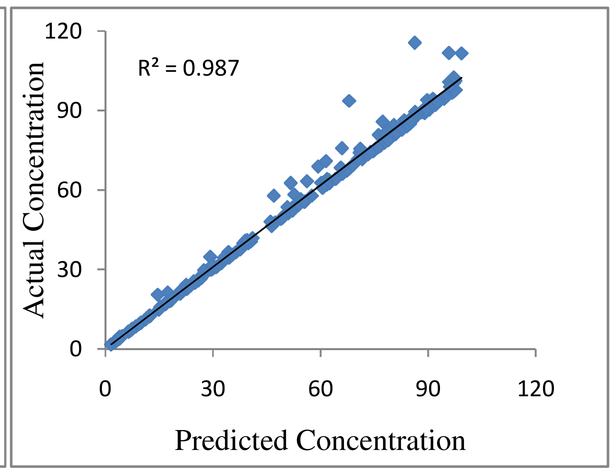
Output 21



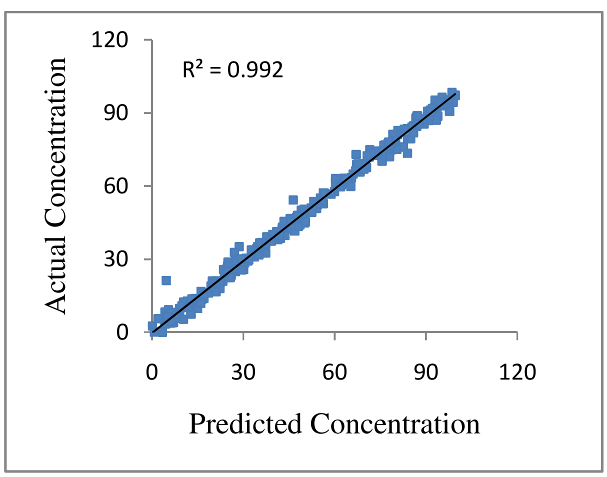
Output 22



Output 23

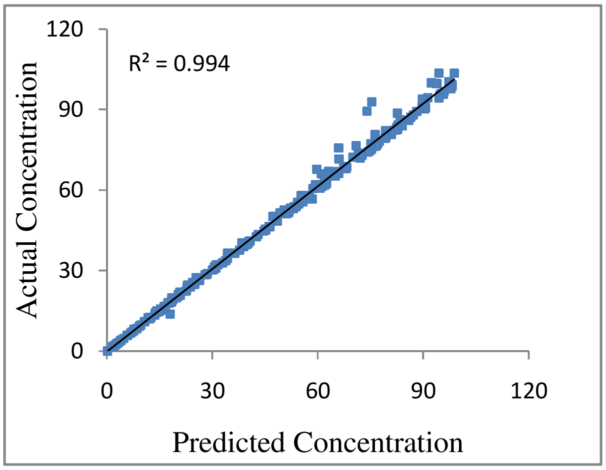


Output 24

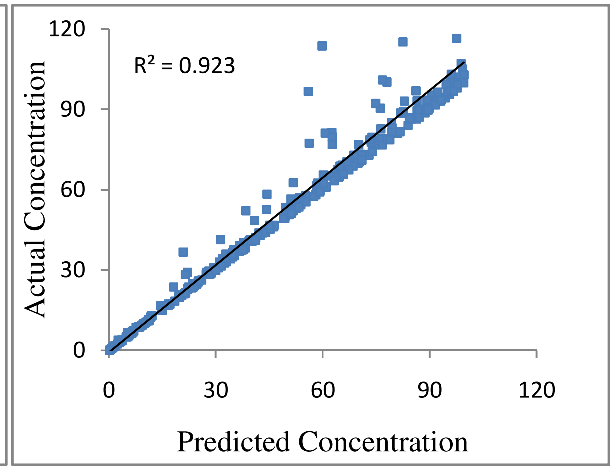


Output 25

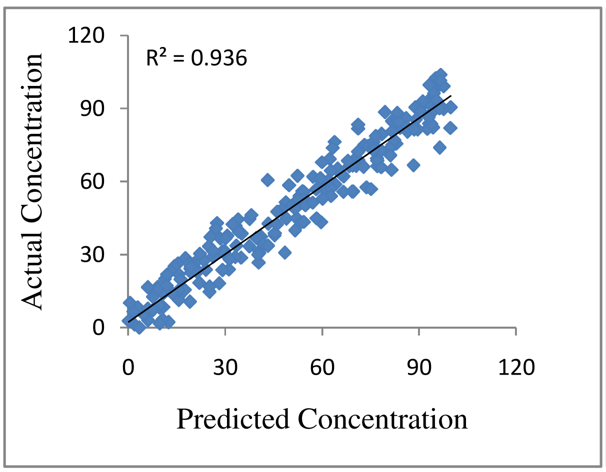
Fig. 4.19: Scatter Plot for Well 2 for second study area



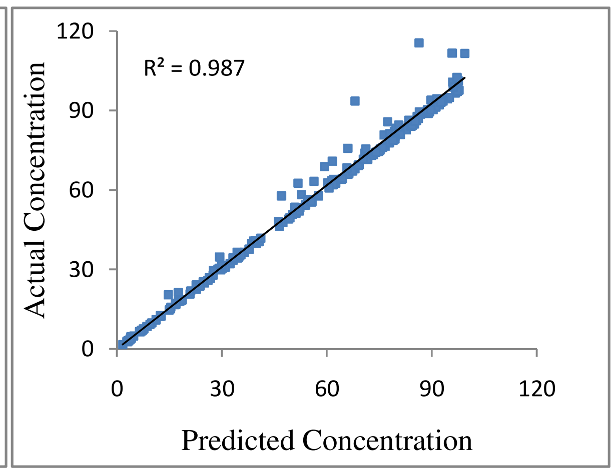
Output 31



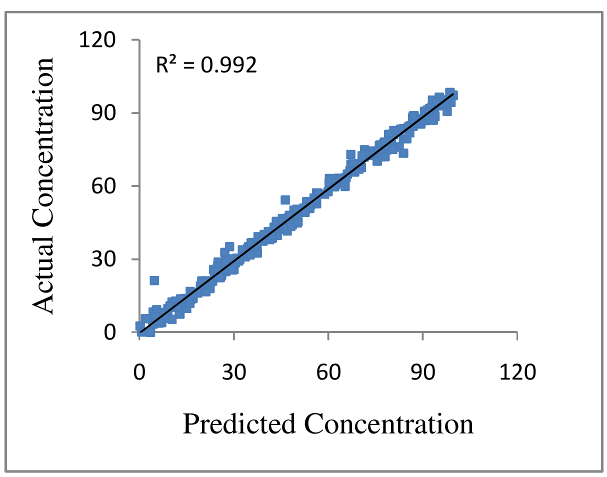
Output 32



Output 33

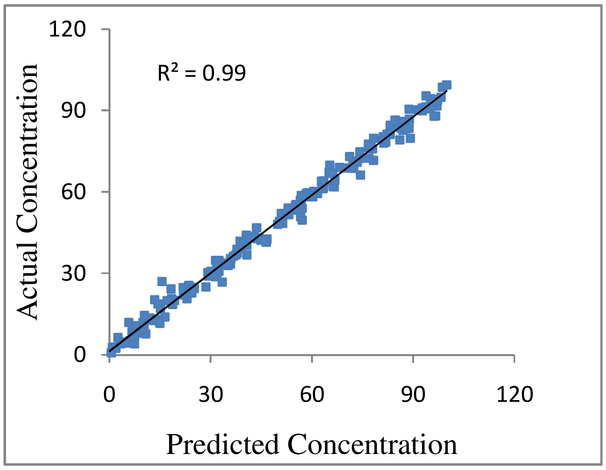


Output 34

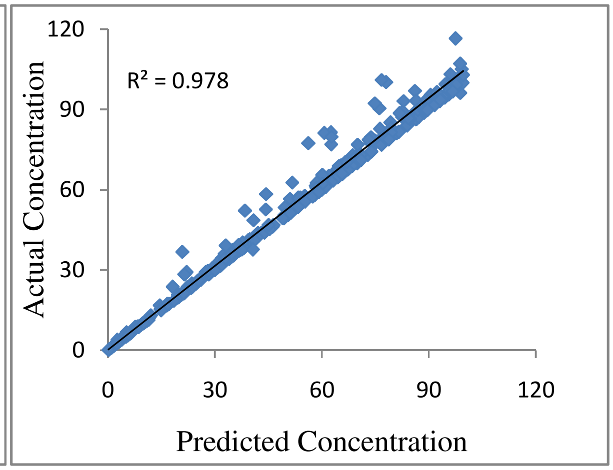


Output 35

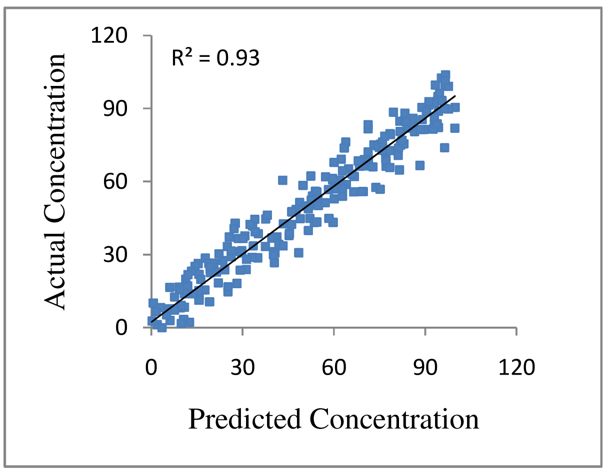
Fig. 4.20: Scatter Plot for Well 3 for second study area



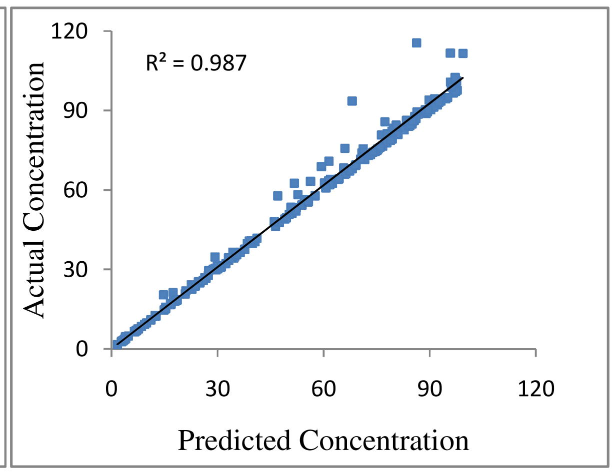
Output 41



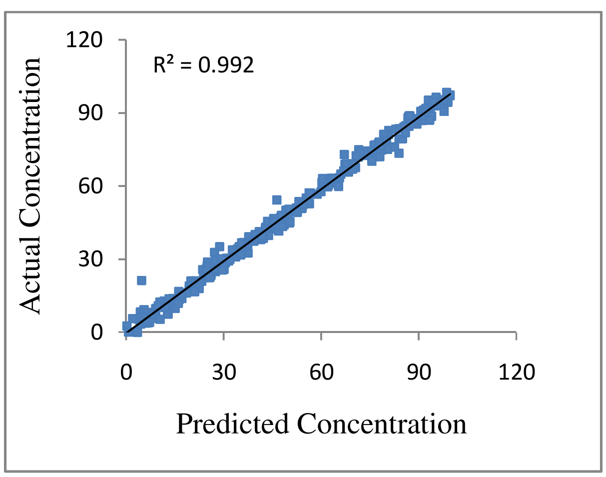
Output 42



Output 43

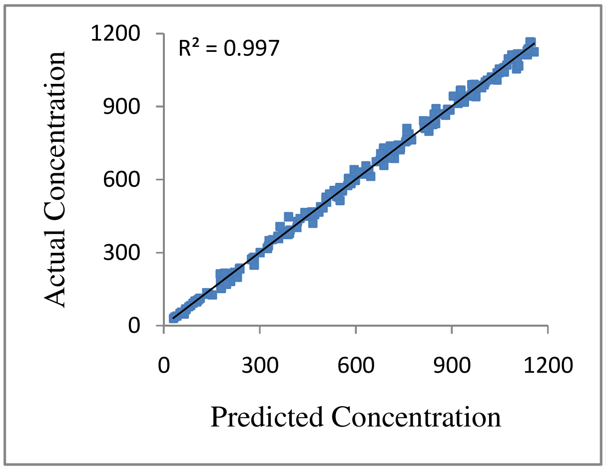


Output 44

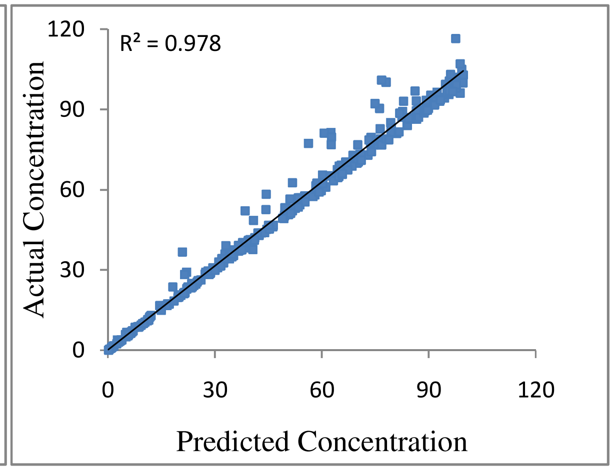


Output 45

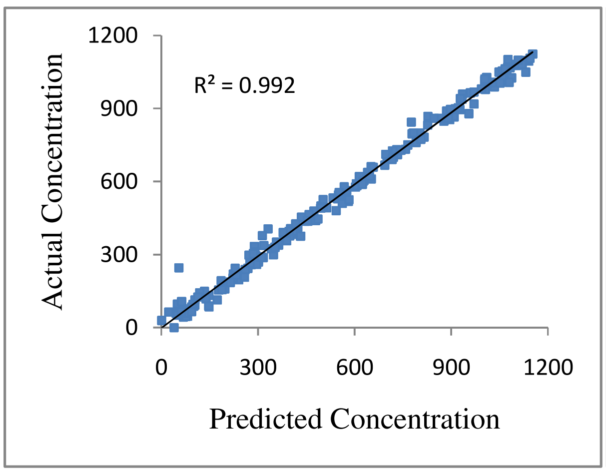
Fig. 4.21: Scatter Plot for Well 4 for second study area



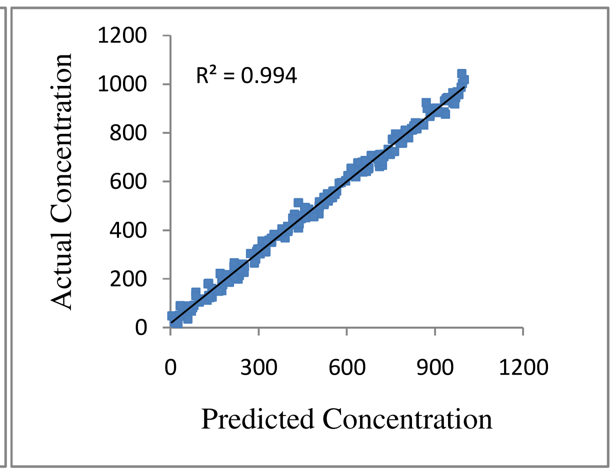
Output 51



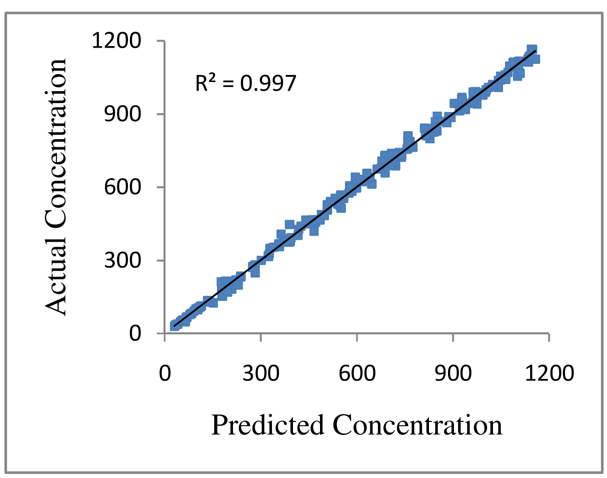
Output 52



Output 53

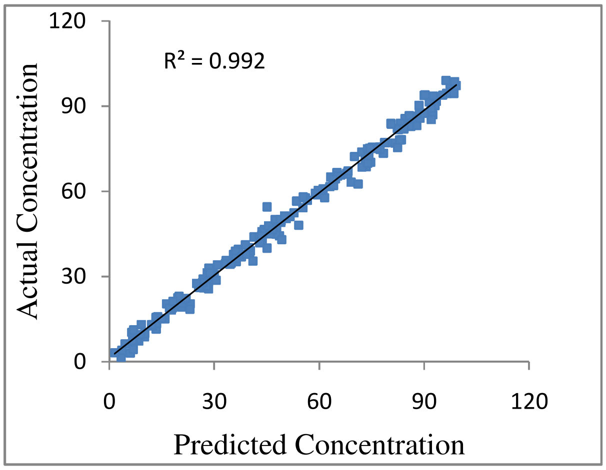


Output 54

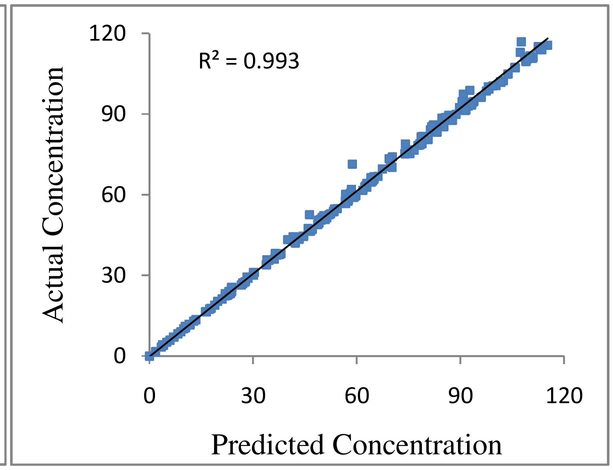


Output 55

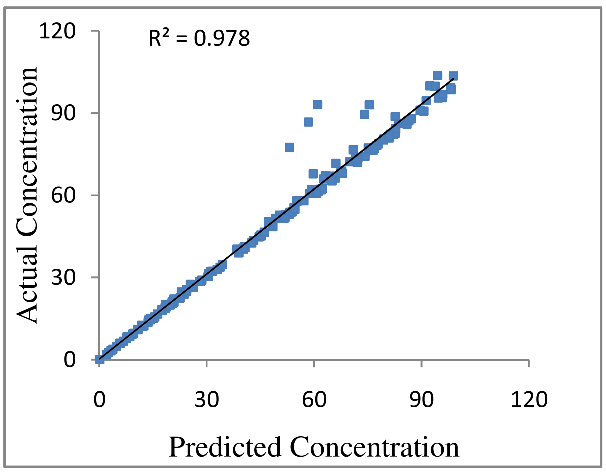
Fig. 4.22: Scatter Plot for Well 5 for second study area



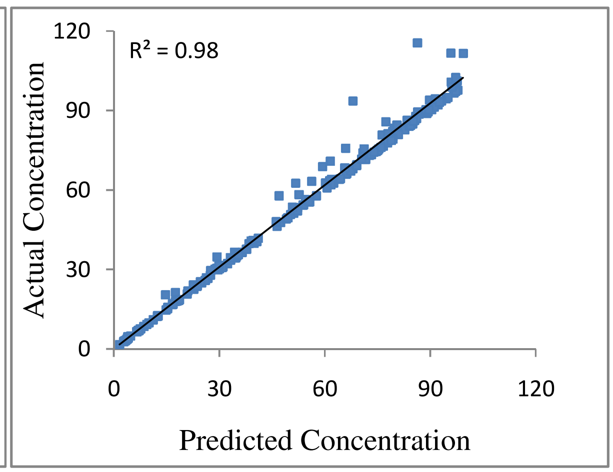
Output 61



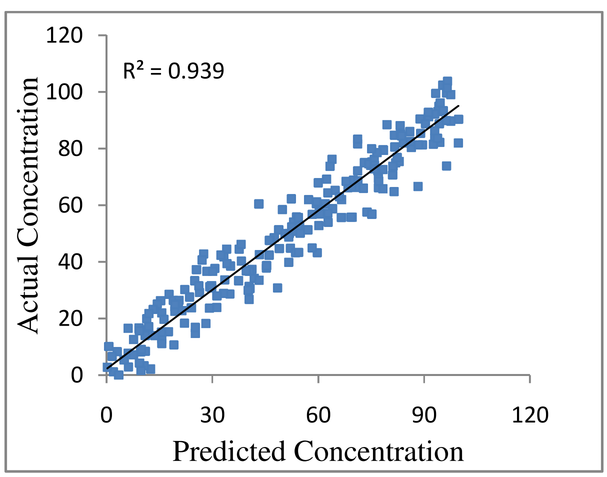
Output 62



Output 63

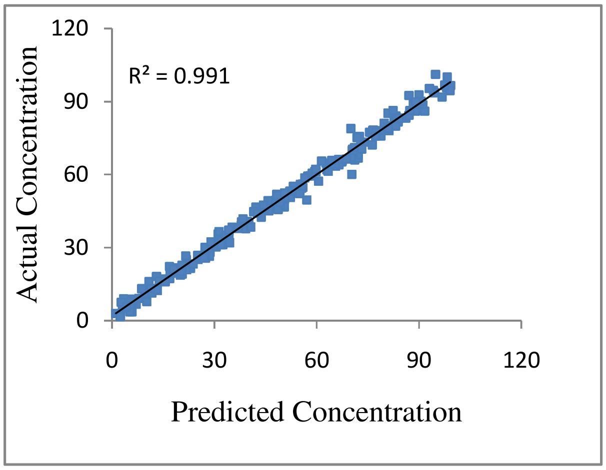


Output 64

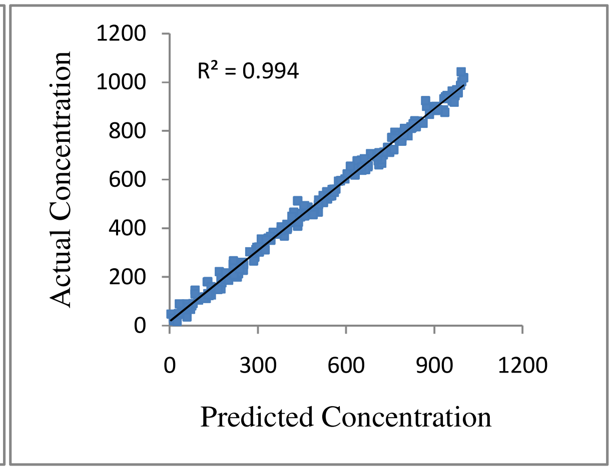


Output 65

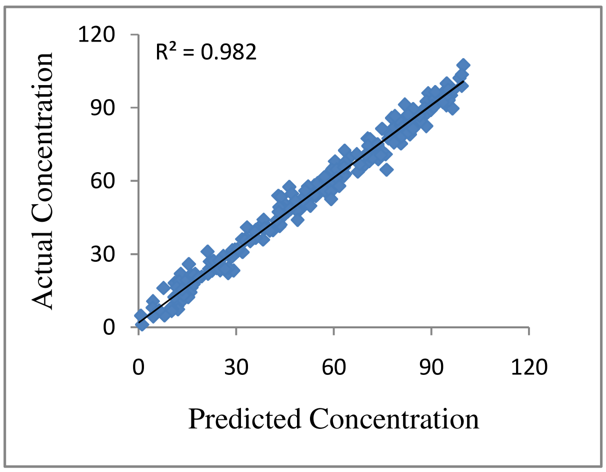
Fig. 4.23: Scatter Plot for Well 6 for second study area



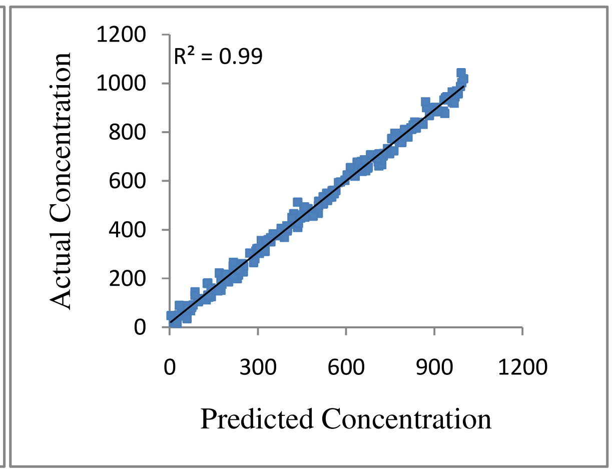
Output 71



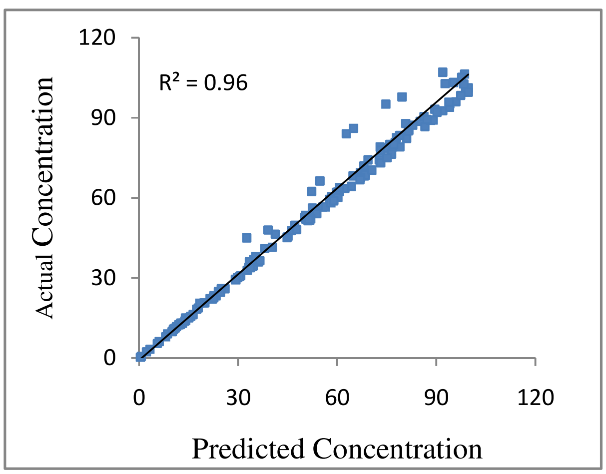
Output 72



Output 73

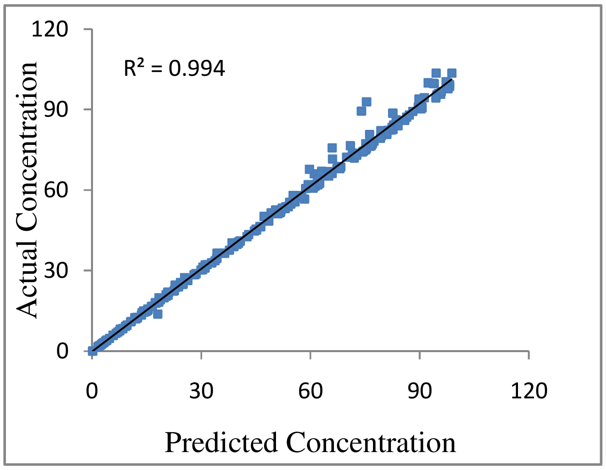


Output 74

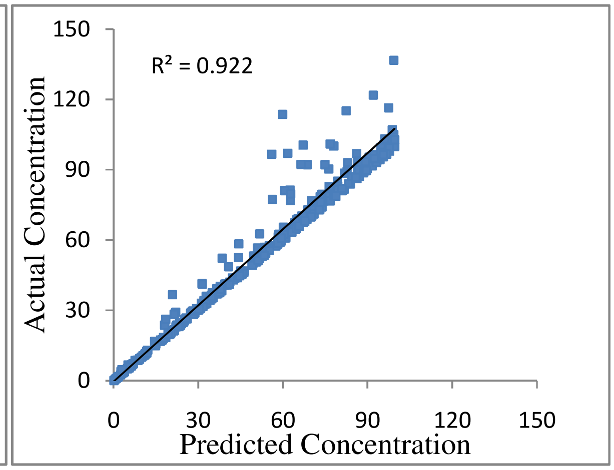


Output 75

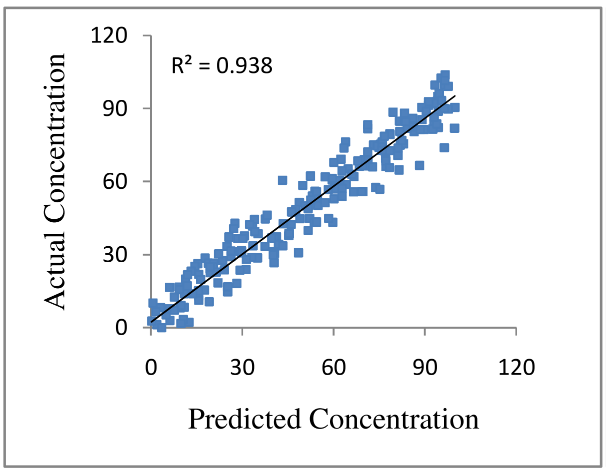
Fig. 4.24: Scatter Plot for Well 7 for second study area



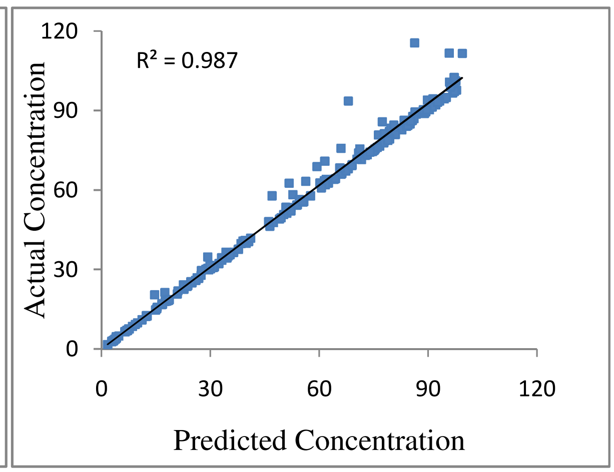
Output 81



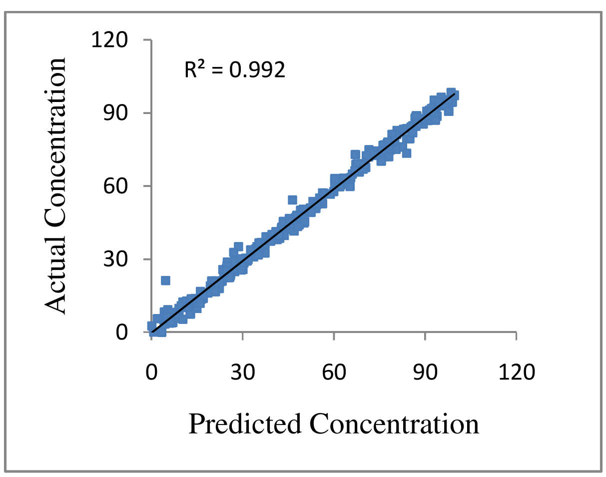
Output 82



Output 83



Output 84



Output 85

Fig. 4.25: Scatter Plot for Well 8 for second study area

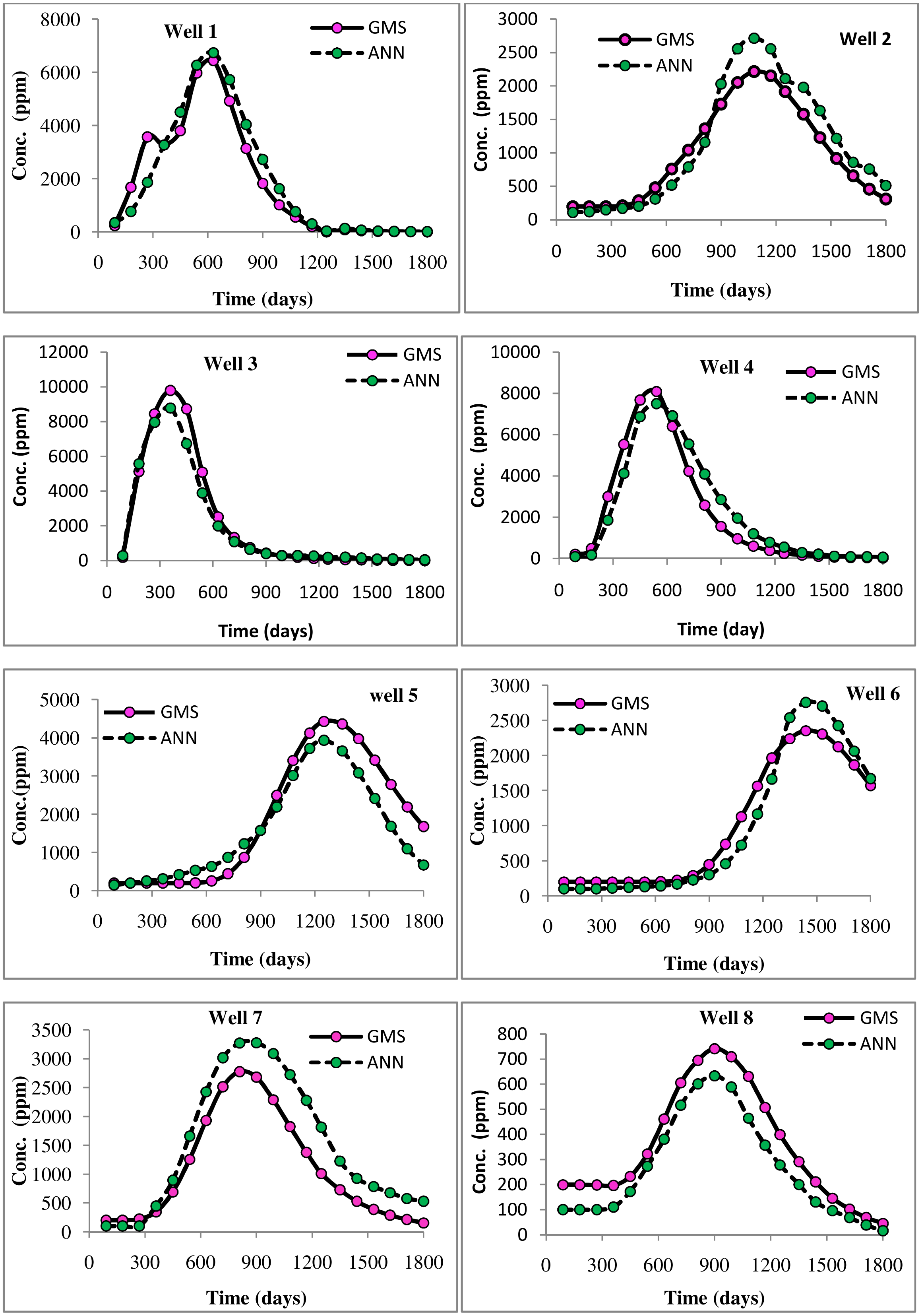


Fig. 4.26: Break through curve for second study area

less than 20%; 98.82% pattern are less than 30%; 99.54% pattern are less than 40% and 99.93% pattern are less than 50%. Table 4.9 shows all the values for 40 outputs. These evaluations show that the performance of the ANN model when trained with error free data is quite good and in acceptable range.

4.2.7 Performance of ANN Model with erroneous data:

We have also checked the performance of the model with erroneous data for second study area. Table 4.10, 4.11, 4.12, 4.13 show the performance of the ANN model when the model is trained with erroneous data which has been generated using SD value of 0.05, 0.10, 0.15 and 0.20. For SD 0.05, the highest and lowest AARE values are 8.34 and 1.87 with an average of 3.63. This value is small and acceptable range. The highest and lowest value of R is 0.987 and 0.929 with an average value of 0.9592. The AARE and R values for all the output are shown in Table 4.10. The performance of the model with erroneous data with SD 0.05 is also evaluated for TS5, TS10, TS20, TS30, TS40 and TS50. The average values of TS5, TS10, TS20, TS30, TS40 and TS50 are 86.02%, 92.53%, 95.55%, 97.57%, 98.79% and 99.56% respectively. It means ARE values for that 86.02% pattern are less than 5%; 92.53% pattern are less than 10%; 95.55% pattern are less than 20%; 97.57% pattern are less than 30%; 98.79% pattern are less than 40% and 99.56% pattern are less than 50%. Table 4.11, Table 4.12 and Table 4.13 show the AARE, R and TSx statistics for SD of 0.10, 0.15 and 0.20 respectively. From this result it can be concluded that the predicting ANN model can simulate the flow and transport processes with high degree of accuracy for the real world groundwater problem.

Fig. 4.27 shows the variation of AARE with respect to different level of noise. The AARE value is 2.99 when the model is trained with error free data. This error increases to 3.63, when the model is trained with SD 0.05 and mean zero. This error is further increases to 4.16, 4.81 and 5.28 when the model is trained with erroneous data of SD 0.10, 0.15 and 0.20. It shows that the performance of the model does not degrade much when trained with erroneous data. Fig.4.28 shows the variation of R, when the ANN model is trained with different level of noise. Initially the R value is 0.9831 when the model is trained with error free data. This value decreases to 0.9592, when the model is trained with SD 0.05 and mean zero. This value further decreases to 0.9368, 0.913 and 0.8802 when the model is trained with erroneous data of SD 0.10, 0.15 and 0.20. It is observed that the R value is not decreasing much when the training patterns are perturbed

Table 4.9: Performance of the ANN Model Trained with Error Free Data

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	2.081	0.995	95.05	97.40	100.00	100.00	100.00	100.00
O12	2.398	0.992	96.40	99.60	100.00	100.00	100.00	100.00
O13	5.992	0.966	79.60	90.40	92.40	96.40	98.80	100.00
O14	3.791	0.987	99.40	100.00	100.00	100.00	100.00	100.00
O15	1.61	0.998	92.40	94.40	97.20	99.60	100.00	100.00
O21	5.02	0.969	78.20	89.20	94.80	97.60	99.20	100.00
O22	1.792	0.998	78.20	98.00	100.00	100.00	100.00	100.00
O23	2.68	0.997	93.20	99.20	100.00	100.00	100.00	100.00
O24	7.45	0.942	67.20	88.20	91.60	95.00	97.20	98.80
O25	1.336	0.998	96.00	99.60	100.00	100.00	100.00	100.00
O31	4.11	0.966	81.20	89.00	93.80	97.20	99.00	100.00
O32	3.95	0.972	85.00	92.00	96.40	97.80	100.00	100.00
O33	1.822	0.996	90.00	98.60	100.00	100.00	100.00	100.00
O34	2.33	0.993	89.80	97.60	99.40	100.00	100.00	100.00
O35	6.412	0.977	68.80	84.20	94.60	97.40	98.60	100.00
O41	1.37	0.998	97.20	99.60	100.00	100.00	100.00	100.00
O42	4.109	0.977	75.00	92.40	94.20	98.20	99.40	100.00
O43	2.43	0.994	92.40	96.40	98.60	100.00	100.00	100.00
O44	1.24	0.997	93.20	99.20	100.00	100.00	100.00	100.00
O45	5.22	0.95	72.20	84.40	94.40	96.60	98.20	99.00
O51	4.87	0.97	83.60	93.40	94.80	97.60	99.40	100.00
O52	1.13	0.998	99.40	100.00	100.00	100.00	100.00	100.00
O53	1.34	0.998	98.40	99.60	100.00	100.00	100.00	100.00
O54	2.871	0.991	83.40	94.60	99.20	99.80	100.00	100.00
O55	1.972	0.994	90.40	97.80	99.80	100.00	100.00	100.00
O61	3.01	0.98	85.00	91.00	95.40	97.00	99.20	100.00
O62	2.911	0.961	81.00	89.40	95.00	99.80	100.00	100.00
O63	6.15	0.94	61.40	79.00	85.40	91.20	96.60	99.20
O64	1.11	0.99	95.05	97.40	100.00	100.00	100.00	100.00
O65	1.763	0.998	95.05	98.00	100.00	100.00	100.00	100.00
O71	3.87	0.969	86.80	89.20	91.20	96.00	98.40	100.00
O72	2.73	0.986	91.00	95.60	98.20	100.00	100.00	100.00
O73	1.35	0.979	89.20	93.40	97.80	100.00	100.00	100.00
O74	1.865	0.998	96.00	99.60	100.00	100.00	100.00	100.00
O75	3.088	0.993	88.00	94.00	98.00	99.60	98.60	100.00
O81	2.756	0.994	91.40	95.00	97.80	99.60	100.00	100.00
O82	1.75	0.99	95.05	97.40	100.00	100.00	100.00	100.00
O83	1.56	0.994	96.40	99.60	100.00	100.00	100.00	100.00
O84	4.90	0.943	79.60	90.40	92.40	96.40	98.80	100.00
O85	1.33	0.998	99.40	100.00	100.00	100.00	100.00	100.00
Average	2.99	0.9831	87.65	94.60	97.31	98.82	99.54	99.93

Table 4.10: Performance of the ANN Model Trained with SD=0.05

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	2.713	0.969	93.20	95.60	99.40	99.80	100.00	100.00
O12	1.89	0.972	94.60	97.20	99.20	100.00	100.00	100.00
O13	6.235	0.945	77.80	88.20	90.00	94.20	96.60	98.20
O14	4.11	0.951	97.40	99.20	100.00	100.00	100.00	100.00
O15	2.10	0.973	90.60	92.80	96.20	97.20	99.20	100.00
O21	5.94	0.951	75.80	87.20	92.20	95.80	97.60	99.20
O22	2.34	0.973	94.60	96.80	97.20	100.00	100.00	100.00
O23	3.015	0.978	91.40	97.40	99.20	100.00	100.00	100.00
O24	8.34	0.929	65.20	83.40	89.00	92.60	95.40	97.40
O25	1.96	0.977	95.20	96.60	99.20	100.00	100.00	100.00
O31	4.95	0.923	75.80	85.60	89.20	93.20	95.60	97.80
O32	4.112	0.914	80.40	87.20	92.20	93.20	96.60	98.40
O33	2.562	0.973	88.60	96.20	98.80	100.00	100.00	100.00
O34	3.21	0.97	88.20	95.80	97.20	99.80	100.00	100.00
O35	7.19	0.914	65.40	82.40	92.20	95.40	96.60	98.00
O41	2.05	0.976	95.80	97.00	99.20	100.00	100.00	100.00
O42	4.99	0.909	73.20	90.40	92.40	96.60	98.40	100.00
O43	2.97	0.974	90.80	94.20	96.60	98.40	100.00	100.00
O44	2.14	0.987	91.20	97.20	99.60	100.00	100.00	100.00
O45	5.87	0.94	71.20	82.40	92.80	94.20	96.20	98.20
O51	5.45	0.96	81.60	90.40	92.80	95.40	97.80	100.00
O52	1.97	0.978	97.80	99.20	100.00	100.00	100.00	100.00
O53	2.53	0.976	96.20	97.40	99.40	100.00	100.00	100.00
O54	3.712	0.972	81.20	92.40	97.20	99.20	100.00	100.00
O55	2.55	0.978	88.80	95.20	97.60	99.40	100.00	100.00
O61	3.96	0.967	83.40	89.80	93.20	95.40	98.20	100.00
O62	4.17	0.933	79.00	86.80	91.40	95.60	97.20	100.00
O63	6.89	0.923	59.80	77.20	83.40	89.20	94.40	97.20
O64	1.93	0.977	93.20	95.60	98.20	100.00	100.00	100.00
O65	2.133	0.968	91.20	97.20	99.60	100.00	100.00	100.00
O71	4.25	0.943	83.40	87.40	89.60	93.20	96.40	99.00
O72	3.054	0.961	89.80	93.20	96.40	98.60	100.00	100.00
O73	1.92	0.96	87.20	92.20	96.20	98.80	100.00	100.00
O74	2.43	0.971	93.00	97.80	99.20	100.00	100.00	100.00
O75	3.976	0.973	86.20	92.40	96.80	97.20	98.40	100.00
O81	3.42	0.983	89.80	93.20	95.60	97.00	99.40	100.00
O82	2.73	0.975	93.40	95.40	99.20	100.00	100.00	100.00
O83	2.05	0.976	94.20	97.60	99.20	100.00	100.00	100.00
O84	5.33	0.929	77.80	88.20	89.00	93.40	97.60	99.00
O85	1.87	0.975	97.20	99.80	100.00	100.00	100.00	100.00
Average	3.63	0.9592	86.02	92.53	95.55	97.57	98.79	99.56

Table 4.11: Performance of the ANN Model Trained with SD=0.10

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	3.331	0.953	91.80	93.20	97.20	98.00	99.80	100.00
O12	2.35	0.96	92.80	95.20	96.80	98.20	100.00	100.00
O13	6.992	0.923	73.20	86.20	88.80	91.20	94.20	96.40
O14	4.943	0.939	95.80	97.80	98.60	100.00	100.00	100.00
O15	2.98	0.951	88.20	89.60	93.80	95.80	98.00	100.00
O21	6.45	0.939	73.20	85.60	90.80	93.60	95.20	98.60
O22	2.89	0.953	92.20	94.60	95.20	98.80	99.20	100.00
O23	3.88	0.962	89.60	93.20	96.20	97.00	98.20	100.00
O24	8.87	0.893	63.44	80.34	87.89	90.12	93.40	95.20
O25	2.55	0.956	93.20	93.40	97.40	99.00	100.00	100.00
O31	5.36	0.904	73.22	83.20	87.40	91.60	93.60	96.00
O32	4.955	0.897	78.20	85.80	90.20	91.40	94.20	96.40
O33	3.35	0.956	85.60	93.20	95.20	97.80	99.00	100.00
O34	3.845	0.945	86.20	93.80	95.60	97.20	98.80	100.00
O35	7.85	0.872	62.40	80.40	79.60	89.40	90.20	91.60
O41	2.621	0.954	93.20	95.60	97.20	99.00	100.00	100.00
O42	5.24	0.887	71.20	88.60	89.00	93.20	96.60	99.00
O43	3.37	0.954	88.80	91.20	93.80	95.60	98.20	100.00
O44	2.64	0.962	89.20	95.60	97.60	99.00	100.00	100.00
O45	6.19	0.93	69.20	80.80	89.20	91.60	94.40	96.20
O51	5.89	0.94	79.20	88.60	89.20	93.20	95.20	97.60
O52	2.33	0.968	95.20	97.80	99.20	100.00	100.00	100.00
O53	2.87	0.955	94.20	96.40	97.80	99.20	100.00	100.00
O54	4.11	0.956	79.00	88.20	95.60	97.80	99.00	100.00
O55	2.91	0.967	85.20	93.40	96.20	97.80	99.20	100.00
O61	4.38	0.941	81.20	87.60	90.20	93.40	97.60	98.20
O62	4.79	0.915	77.20	83.40	89.60	93.40	95.60	98.80
O63	7.24	0.904	57.20	76.20	80.80	87.20	92.80	95.80
O64	2.18	0.956	91.20	94.20	97.00	98.80	100.00	100.00
O65	2.45	0.93	89.20	93.20	95.80	97.80	100.00	100.00
O71	4.92	0.912	81.20	85.60	87.20	91.00	93.20	95.80
O72	3.55	0.932	87.20	91.20	94.80	97.60	99.60	100.00
O73	2.12	0.952	86.20	90.80	93.80	97.20	99.20	100.00
O74	2.85	0.955	91.80	95.20	97.60	99.20	100.00	100.00
O75	4.97	0.914	82.80	87.40	90.80	92.20	95.60	97.20
O81	3.89	0.938	87.20	89.80	91.40	93.40	97.80	99.20
O82	3.09	0.941	91.80	93.60	97.80	99.20	100.00	100.00
O83	2.79	0.963	91.80	95.40	97.20	99.00	100.00	100.00
O84	6.01	0.892	75.80	85.40	87.20	91.80	95.00	96.80
O85	2.28	0.956	95.20	96.80	98.20	99.00	100.00	100.00
Average	4.16	0.9368	83.76	90.19	93.12	95.64	97.57	98.72

Table 4.12: Performance of the ANN Model Trained with SD=0.15

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	3.99	0.925	89.80	91.20	94.00	96.20	97.20	100.00
O12	2.97	0.935	90.20	91.80	95.20	96.00	97.20	100.00
O13	7.43	0.892	70.20	83.40	86.40	89.80	89.80	92.80
O14	5.66	0.89	93.40	95.80	96.20	97.80	98.00	100.00
O15	3.67	0.932	83.20	86.40	88.00	92.80	95.60	97.80
O21	7.28	0.893	69.20	75.80	85.60	91.20	93.00	95.80
O22	3.56	0.934	89.20	91.80	93.00	96.80	97.80	99.00
O23	4.34	0.929	86.40	90.20	93.80	95.80	96.00	98.80
O24	9.01	0.837	59.40	75.80	80.40	85.20	89.40	91.20
O25	2.85	0.925	89.20	90.20	95.80	96.20	97.20	100.00
O31	5.99	0.875	69.20	80.20	84.80	87.60	91.20	93.40
O32	5.39	0.855	75.20	81.20	87.60	89.40	91.20	93.00
O33	3.95	0.927	81.80	90.40	91.20	94.00	97.80	100.00
O34	4.67	0.907	84.20	89.40	92.20	95.20	96.00	97.80
O35	8.27	0.823	59.40	71.20	75.80	81.20	85.60	88.00
O41	3.45	0.923	89.20	91.60	94.40	95.60	97.00	100.00
O42	5.89	0.857	67.80	83.80	85.80	90.80	94.20	97.60
O43	4.18	0.933	85.20	89.60	90.60	93.20	95.60	98.60
O44	3.25	0.94	86.00	92.40	95.00	97.80	99.00	100.00
O45	6.92	0.91	65.20	75.80	85.20	89.00	91.20	93.40
O51	6.49	0.93	75.80	83.20	85.60	90.20	92.00	95.80
O52	2.87	0.957	93.20	95.80	97.00	98.20	99.00	100.00
O53	3.39	0.943	91.80	93.40	95.80	96.20	97.40	100.00
O54	4.79	0.941	75.80	83.40	88.60	92.40	97.60	98.20
O55	3.67	0.95	81.20	89.40	92.80	95.20	97.80	99.00
O61	4.82	0.944	77.80	82.40	87.60	89.20	95.60	97.80
O62	5.46	0.897	75.20	78.20	85.80	89.20	91.40	95.80
O63	7.99	0.887	55.20	71.20	75.40	82.40	89.80	93.00
O64	2.57	0.94	87.40	89.80	93.40	96.80	97.80	100.00
O65	2.89	0.933	82.40	89.60	91.20	94.60	97.80	100.00
O71	5.62	0.893	79.80	82.40	85.60	89.20	91.00	93.40
O72	4.28	0.92	85.60	89.40	91.20	94.60	97.60	99.20
O73	3.67	0.943	81.20	85.60	90.20	94.60	96.40	98.80
O74	3.55	0.938	89.40	92.20	94.60	96.40	98.80	100.00
O75	5.77	0.897	79.20	85.40	87.20	90.00	91.40	95.00
O81	4.62	0.916	85.40	87.20	89.00	91.40	95.80	97.80
O82	3.94	0.923	89.60	91.20	94.60	97.00	99.20	100.00
O83	3.55	0.932	88.60	91.20	94.40	95.20	97.80	100.00
O84	6.91	0.85	71.20	81.60	83.40	88.40	92.80	94.00
O85	2.98	0.946	92.80	93.80	96.20	98.00	99.20	100.00
Average	4.81	0.913	80.55	86.36	89.77	92.77	95.16	97.38

Table 4.13: Performance of the ANN Model Trained with SD=0.20

Output	AARE	R	TSx					
			5%	10%	20%	30%	40%	50%
O11	4.84	0.893	83.45	87.65	91.20	93.80	95.20	98.20
O12	3.63	0.915	87.60	89.20	91.40	93.20	95.60	97.80
O13	8.26	0.834	64.60	79.80	84.40	85.20	87.60	90.40
O14	6.55	0.865	91.40	92.40	93.40	95.40	95.40	97.80
O15	4.35	0.897	81.60	83.40	85.40	89.60	92.40	93.40
O21	8.05	0.876	65.80	73.40	81.20	89.60	91.00	92.80
O22	4.32	0.897	86.60	88.20	89.20	93.40	95.80	96.80
O23	5.28	0.88	83.20	85.60	91.20	93.40	94.60	96.20
O24	9.89	0.798	55.80	72.60	75.60	82.20	85.60	89.40
O25	3.54	0.904	87.80	89.20	93.40	95.60	97.80	99.00
O31	6.67	0.839	65.20	77.80	81.40	84.60	88.00	89.40
O32	5.82	0.843	71.34	78.40	85.20	85.40	87.40	90.00
O33	4.23	0.895	78.65	85.60	88.20	89.40	95.40	98.80
O34	4.89	0.852	81.20	84.40	88.40	93.60	94.40	96.60
O35	8.44	0.799	57.00	66.40	72.20	78.60	81.20	85.80
O41	3.82	0.903	87.60	89.20	92.40	91.60	94.80	98.40
O42	6.36	0.828	64.20	81.20	83.40	88.80	91.60	95.40
O43	4.56	0.903	81.60	86.40	87.60	91.40	92.60	96.60
O44	3.76	0.922	83.40	89.80	92.20	95.60	97.80	98.20
O45	7.13	0.85	62.20	71.40	82.40	86.60	89.40	91.40
O51	6.82	0.91	72.40	81.20	83.40	87.60	88.40	91.40
O52	3.11	0.922	91.40	92.80	94.60	96.60	97.20	98.40
O53	3.67	0.918	89.40	89.60	91.60	94.20	95.40	97.80
O54	5.02	0.911	71.40	79.40	85.60	89.20	95.60	96.20
O55	3.99	0.937	79.20	84.80	89.20	91.60	94.60	96.00
O61	5.27	0.89	74.20	78.80	82.40	85.60	89.60	95.80
O62	5.87	0.873	71.20	74.40	79.80	84.40	87.60	92.20
O63	8.342	0.838	52.40	67.60	69.60	79.20	84.40	91.80
O64	3.45	0.932	85.40	82.60	89.60	91.20	94.40	96.60
O65	3.22	0.905	78.60	85.40	88.20	91.20	96.40	98.80
O71	5.97	0.843	75.60	78.80	82.40	83.40	89.60	89.40
O72	4.87	0.893	81.40	85.60	89.40	91.40	95.80	97.60
O73	3.76	0.916	78.80	81.20	85.60	91.20	93.40	96.20
O74	3.91	0.909	86.40	89.20	91.40	93.80	95.60	98.20
O75	5.93	0.817	72.40	81.20	83.40	86.40	89.20	94.20
O81	4.89	0.887	81.80	83.60	86.60	89.40	91.40	95.60
O82	4.25	0.887	84.60	87.80	91.20	95.60	89.60	95.60
O83	3.89	0.892	83.40	87.40	89.60	91.20	95.40	97.80
O84	7.26	0.823	67.80	77.80	79.20	85.60	88.80	91.20
O85	3.31	0.908	89.20	91.20	94.60	94.60	96.80	98.80
Average	5.28	0.8802	77.18	82.81	86.43	89.76	92.32	95.05

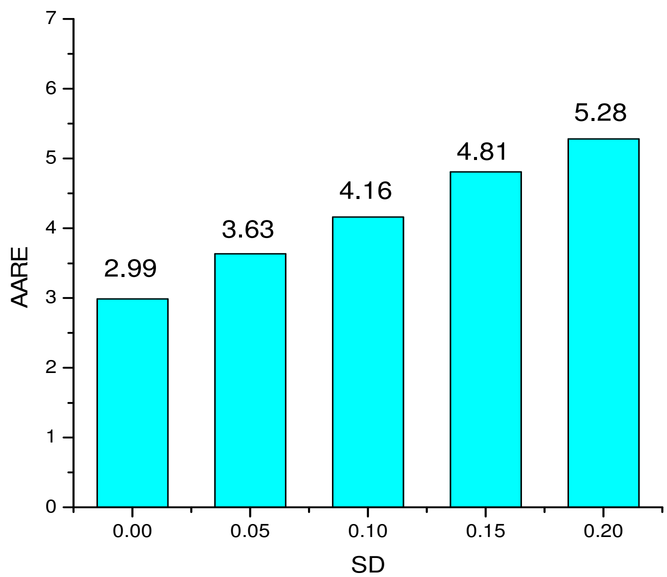


Fig. 4.27: Variation of AARE with increase in Noise level

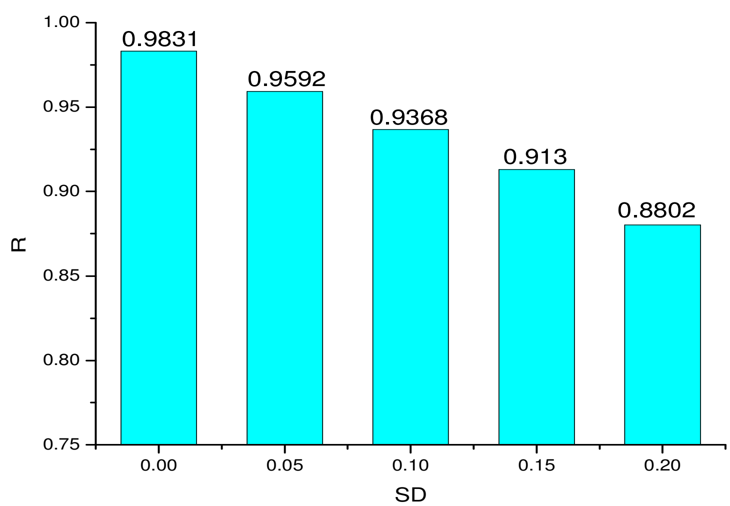


Fig. 4.28: Variation of R with increase in Noise level

with SD 0.05, 0.10, 0.15 and 0.20. It means that, for the second study area also, the model performance does not degrade much when trained with erroneous data.

Fig. 4.29 shows the performance of the ANN model in terms of TSx statistics for the second study area also, when the training patterns are perturbed with different level of noise. From the figure, it is clear that the model is very sensitive with respect to the perturbed error for TS5 statistics. The average value of TS5 is varying from 87.65% to 77.18%. It means that 87.65% of predicted concentration has an ARE value less than 5% when the model is trained with an error free data. The ARE is decreases to 86.02%, when the training patterns are perturbed with random error of SD of 0.05 and mean zero. This

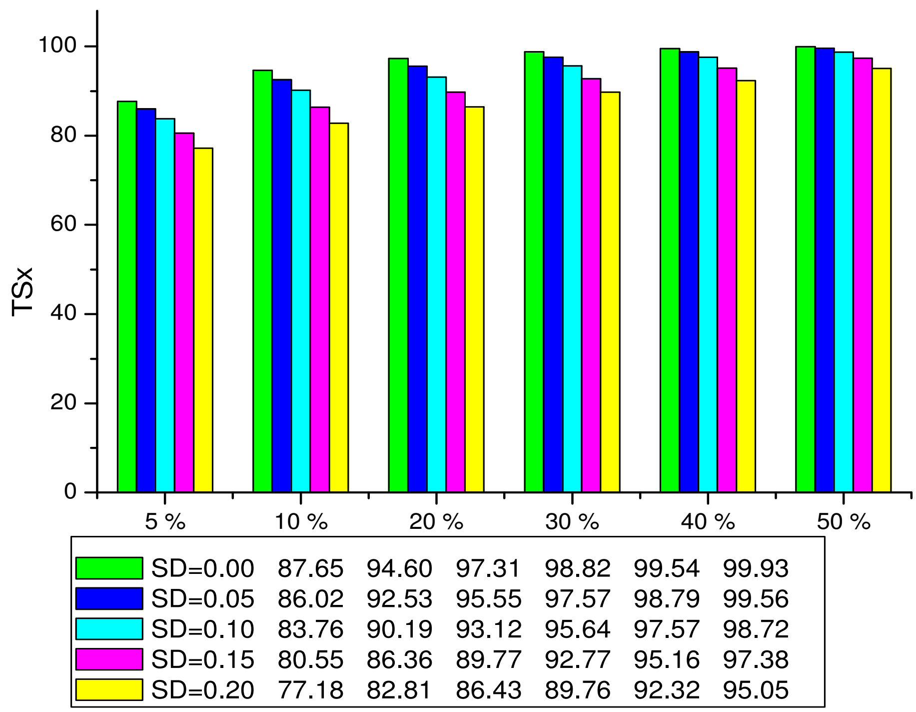


Fig. 4.29: Performance of the ANN model in Terms of TS statistics value with erroneous data

value further decreases to 83.76%, 80.55%, 77.18% when the model is trained erroneous data generated with SD of 0.10, 0.15, 0.20. It is observed from the figure that 94.60% of predicting concentration has an ARE value less than 10% when the model is trained with error free data. This value decreases to 82.81% when the model is trained with an erroneous data of SD of 0.20 and mean zero. The model is little bit sensitive for TS30 also. From the figure, it is clear that the value is varying from 98.82% to 89.76%. But the model is less sensitive for TS40 and TS50.

4.2.8 Transfer function and Optimization algorithms in Second Study area

A study is also conducted to select the best transfer function and optimization algorithm for the second study area. Table 4.14 shows the MSE, computational time and iteration needed for different transfer functions on input and output layers. The *tansig* transfer function for hidden layer and *purelin* transfer function for output layer give the lowest value of MSE, which is 0.00308. For this combination, computational time required for simulation as well as iteration needed are also low as compared to other transfer functions.

Another experiment is also conducted to find out the best optimization algorithm to

Table 4.14: Performance of different transfer function for the ANN model

Transfer function Hidden Layer	Transfer function Output Layer	MSE	Number of iteration
tansig	*tansig*	0.0345	17
tansig	*logsig*	0.1993	14
logsig	*tansig*	0.2477	33
logsig	*logsig*	0.2037	37
logsig	*purelin*	0.0696	12
purelin	*tansig*	0.0317	17
purelin	*logsig*	0.184	36
purelin	*purelin*	0.02897	18
tansig	*purelin*	0.00308	16

training the ANN model. Table 4.15 shows the MSE, computational time and number of iteration needed for different transfer functions. It can be observed that *trainlm* gives the best performance over the other methods.

4.2.9 Performance of ANN based simulation-optimization model for first study area

The proposed inverse models, *i.e.* ANN-DS and ANN-GA are utilized to obtain the unknown pollution sources of first study area. The results obtained by ANN-DS and ANN-GA models have been compared with the actual source concentration and the results obtained using GMS-GA model. Table 4.16 shows the comparison of ANN-DS, ANN-GA and GMS-GA methodologies. It can be observed from the table that

Table. 4.15: Performance of different optimization Algorithms for the ANN model

Training Algorithms	MSE	Computational Time (Sec)	Iteration
trainbfg	0.0349	2557	286
traincgp	0.0299	185	661
trainlm	0.0013	160	16
trainrp	0.0411	123	1000
traingdm	0.3981	117	1000
traingd	0.3829	115	1000
traincgb	0.0348	111	382
trainoss	0.0358	107	839
trainscg	0.0358	99	364
traincgf	0.0359	63	405
traingda	0.1804	17	141
traingdx	0.2088	14	103

concentration estimated by ANN-GA model is superior to ANN-DS model whereas the concentration estimated by GMS-GA methodology is closer to the actual concentration. The concentration estimated by ANN-DS and ANN-GA models for first time step is 41.48 gm/sec and 41.67 gm/sec respectively whereas concentration estimated by GMS-GA model is 45.29 gm/sec, which is more closed to the actual concentration. This shows that ANN-GA model is slightly better than ANN-DS model and GMS-GA model is significantly better than both ANN-GA and ANN-DS model. For the other time steps also same trend have been observed. Thus for this study area, it can be conclude that GMS-GA model is better than ANN-DS and ANN-GA model in predicting the unknown pollution sources of the aquifer. Fig. 4.30 shows the comparison of these methodologies using bar diagram.

The relative efficiency of these methodologies is also evaluated using relative error. Table 4.17 shows that relative error of prediction by ANN-DS and ANN-GA models and also by GMS-GA model. The relative error is calculated with respect to the actual concentration. It can be observed that the relative error of ANN-GA method is lesser than ANN-DS model. But the relative error prediction by GMS-GA model is lesser than ANN-GA model. For example, for the first time step, relative error achieved by ANN-DS

Table 4.16: Actual and Estimated concentration by three methodologies at three point sources

Time Step	Source Flux (gm/sec)											
	Actual Concentration			Estimated Concentration by GMS-GA			Estimated Concentration by ANN-DS			Estimated Concentration by ANN-GA		
	Source			Source			Source			Source		
	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
1	47.00	30.00	0	45.29	29.57	0.234	41.48	27.45	0.18	41.67	27.60	0.192
2	15.00	58.80	0	17.91	55.46	0	18.92	54.29	0	18.90	54.63	0
3	37.00	0	0	31.37	2.64	0	27.66	2.54	0	27.68	2.49	0
4	0	35.00	0	1.87	34.39	0.182	1.65	33.06	0.20	1.69	33.13	0.177

model for S1 source is 11.70% and that of achieved by ANN-GA methodology is 11.34%. The relative error achieved by GMS-GA model is 3.63% only. Similarly, for S1 source at second time step, relative error achieved by ANN-DS model is 26.13%, whereas that is achieved by ANN-GA and GMS-GA models are 26.03% and 19.40% respectively. This clearly indicates that GMS-GA is better than the other methodologies. Fig. 4.31 shows the comparison of relative errors achieved by different models as bar diagram.

4.2.10 Performance of ANN based simulation-optimization model for second study area

The ANN based simulation-optimization model is also applied to the second study area to evaluate the applicability of the method on larger aquifer system. The unknown source flux obtained by ANN-DS and ANN-GA models are compared with the actual source flux value and also with the results obtained using GMS-GA model. The actual concentration and the concentration values obtained by ANN-DS and ANN-GA models are shown in Table 4.18. It can be observed in the table that the concentration predicted by the ANN-DS and ANN-GA model is almost similar whereas GMS-GA model is closer to the actual concentration. The actual value at source S1 for first time step is 908.42 gm/sec. The concentration predicted by GMS-GA model is 990.72 gm/sec whereas the concentrations predicted by ANN-DS and ANN-GA models are 1027.06 gm/sec and 1027.40 gm/sec respectively. Similarly for the S2 source location, the actualconcentration is 644.02 gm/sec whereas concentration predicted by GMS-GA, ANN-DS and ANN-GA

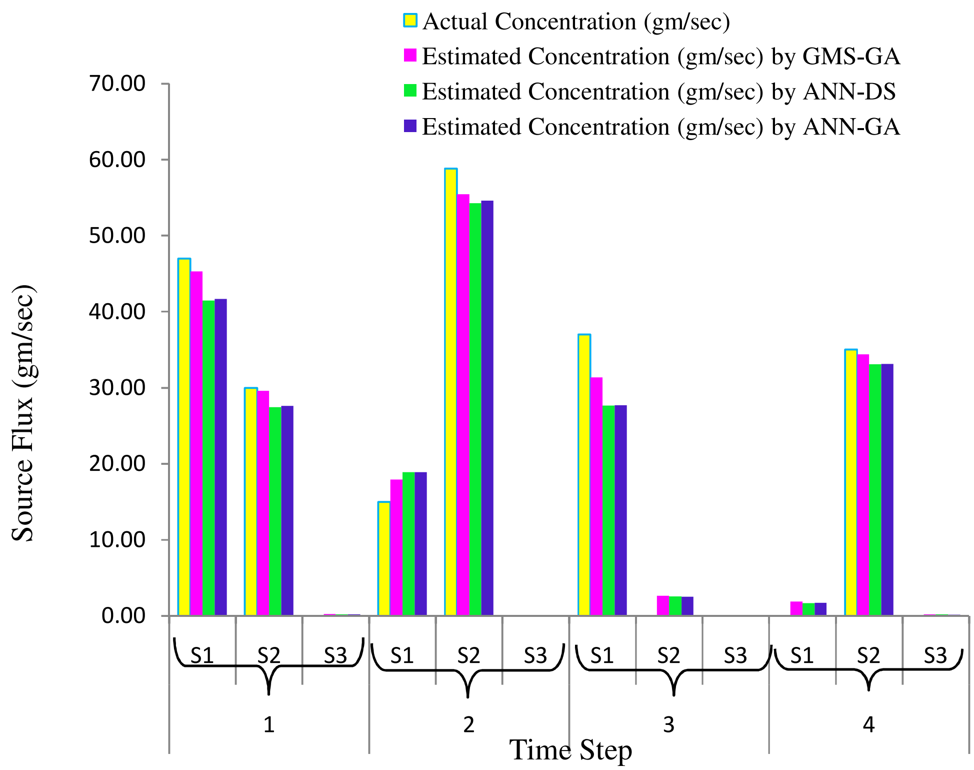


Fig. 4.30: Comparison of Actual and Estimated Concentration by GMS-GA, ANN-DS and ANN-GA models

are 771.30 gm/sec, 886.42 gm/sec and 818.54 gm/sec location, the actual concentration is 644.02 gm/sec whereas concentration predicted by GMS-GA, ANN-DS and ANN-GA are 771.30 gm/sec, 886.42 gm/sec and 818.54 gm/sec respectively. For the other sources and time steps also the concentration predicted by GMS-GA is better than the ANN-DS and ANN-GA model. But the concentration predicted by ANN-GA is slightly better than ANN-DS model. Fig. 4.32 shows the comparison of these methodologies using bar diagram.

Table 4.17: Relative error (%) for GMS-GA, ANN-DS and ANN-GA models

Time Step	Relative Error (%)								
	GMS-GA			ANN-DS			ANN-GA		
	Source			Source			Source		
	S1	S2	S3	S1	S2	S3	S1	S2	S3
1	3.63	1.43	-	11.74	8.49	-	11.34	7.97	-
2	19.40	5.68	-	26.13	7.66	-	26.03	7.09	-
3	15.22	-	-	25.25	-	-	25.18	-	-
4	-	1.74	-	-	5.54	-	-	5.33	-

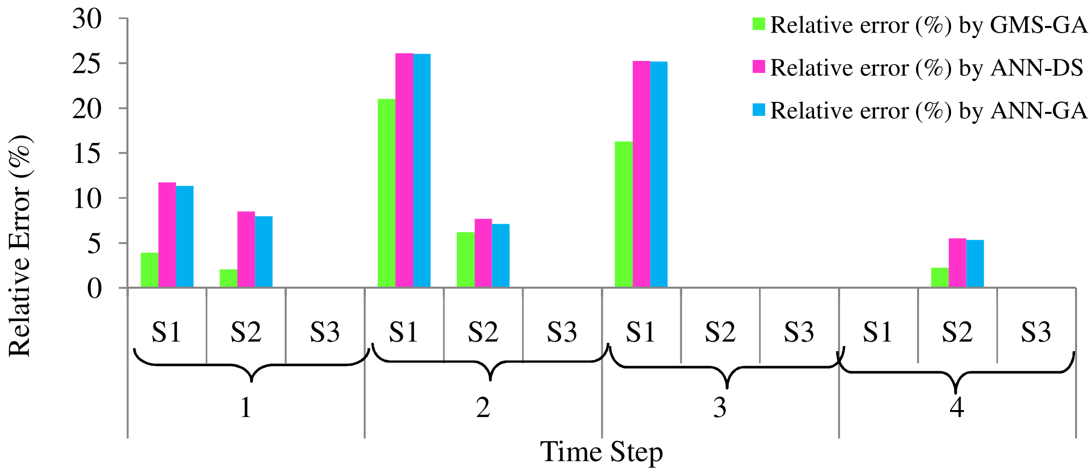


Fig. 4.31: Comparison of Relative Error of GMS-GA, ANN-DS and ANN-GA models

Table 4.18: Comparison of actual and predicted concentration for second study area

Time Step	Source Location	Actual Conc. (gm/sec)	Predicted Conc. (gm/sec) GMS-GA	ANN-DS	ANN-GA
1	S1	908.42	990.72	1027.40	1027.06
	S2	644.02	771.30	886.42	818.54
	S3	0	36.19	39.28	38.19
	S4	0	0	0	0
	S5	987.08	748.60	658.63	659.56
2	S1	1130.50	1010.78	886.41	888.34
	S2	1023.87	1149.70	1214.35	1210.01
	S3	0	0	0	0
	S4	1024.16	765.87	683.42	697.76
	S5	0	23.97	29.47	21.33
3	S1	653.35	730.83	793.02	791.46
	S2	1139.88	1133.27	1156.32	1124.95
	S3	0	66.74	69.63	64.39
	S4	652.05	733.36	768.26	766.29
	S5	0	97.07	102.37	99.23
4	S1	902.13	1068.66	1101.04	1099.69
	S2	781.09	807.35	852.84	851.15
	S3	0	0	0	0.00
	S4	1117.45	1083.70	1031.00	1017.21
	S5	1104.82	910.59	875.34	882.86
5	S1	721.25	656.26	628.93	630.00
	S2	889.77	775.26	735.70	744.20
	S3	0	70.64	75.29	70.75
	S4	889.77	797.95	747.57	747.77
	S5	639.93	690.61	711.73	711.15

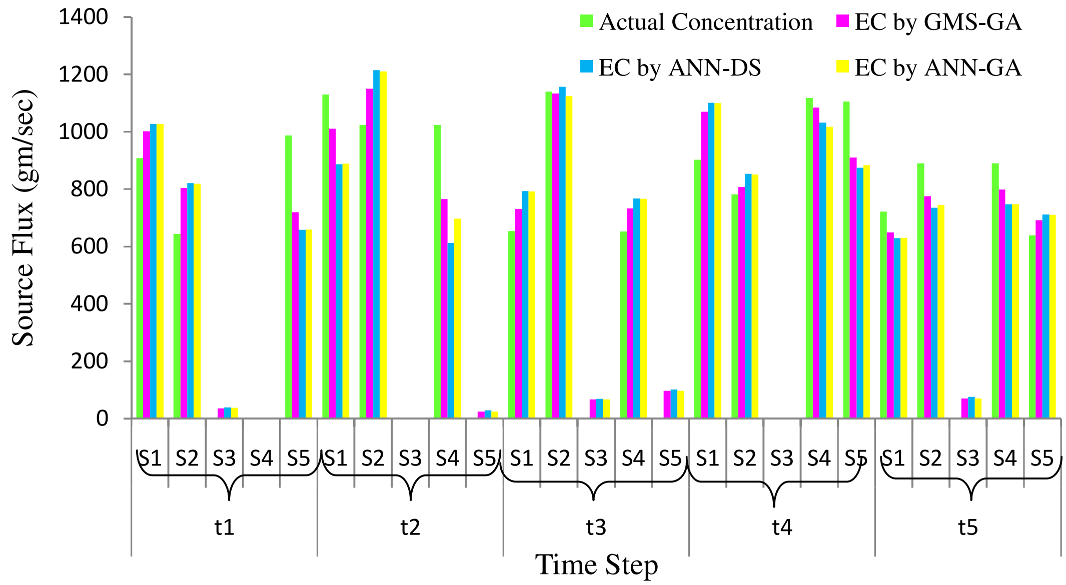


Fig. 4.32: Actual and predicted source fluxes obtained by different models for second study area

Table 4.19 shows the relative error achieved by GMS-GA, ANN-DS and ANN-GA methodologies. It can be observed that the relative error achieved by GMS-GA is lesser than the ANN-DS and ANN-GA models. For the first time step, relative error achieved for S1 source by the GMS-GA is 10.25% and that of achieved by ANN-DS and ANN-GA methodology are 13.09% and 13.06 % respectively. For S2 source location, relative error achieved by GMS-GA, ANN-DS and ANN-GA are 24.87%, 27.58% and 27.10% respectively. This clearly indicates that the GMS-GA model is better than the other two methodologies. Fig. 4.33 shows the comparison of the relative errors among GMS-GA, ANN-DS and ANN-GA as bar diagram.

Table. 4.19 Relative error(%) for GMS-GA, ANN-DS, ANN-GA models for second study area

	Relative Error (%)														
	GMS-GA					ANN-DS					ANN-GA				
	Time Step					Time Step					Time Step				
SL	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
S1	10.25	10.59	11.86	18.46	10.01	13.09	21.59	21.37	22.05	12.80	13.06	21.42	21.14	21.90	12.65
S2	24.87	12.29	0.58	3.36	12.87	27.58	18.61	1.44	9.19	17.31	27.10	18.18	1.31	8.97	16.36
S3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S4	-	25.22	12.47	3.02	10.32	-	32.30	17.80	9.31	15.98	-	31.87	17.52	8.97	15.96
S5	27.15	-	-	17.58	7.92	33.27	-	-	20.77	11.22	33.18	-	-	20.09	11.13

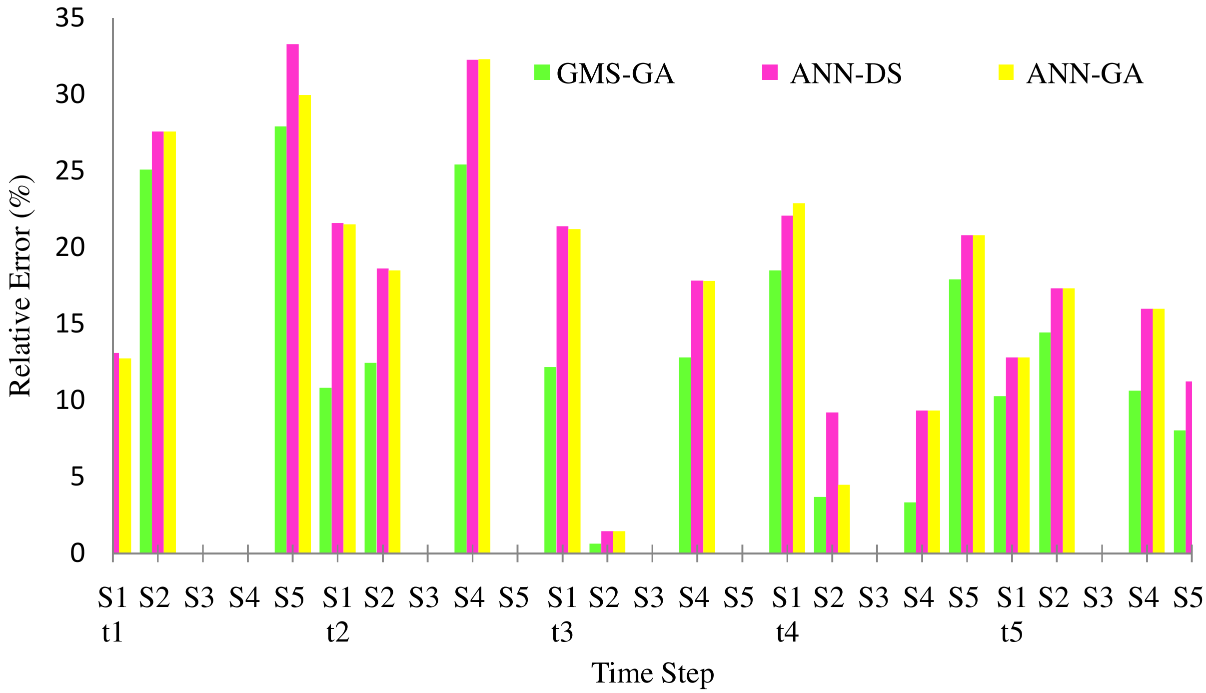


Fig. 4.33: Comparison of relative error achieved by different models for second study area

4.2.11 Computational efficiency of the ANN based simulation-optimization models

We have conducted an analysis to evaluate the efficiency of the models in terms of computational time requirement. Table 4.20 shows comparison of the objective function value and computational time requirement for GMS-GA, ANN-DS and ANN-GA methods for both the study areas. It may be observed from the table that for the first study area, the time required for solving the problem by GMS-GA model is 2 days 1 hour 15

Table 4.20: Comparative evaluation of the models in terms of computational time

Models	First Study area		Second Study area	
	Obj. funct. value	Time DD HH MM SS	Obj. funct. value	Time DD HH MM SS
GMS-GA	1.34E-06	02:01:15:24	1.340E-06	05:01:54:45
ANN-DS	2.33E-04	00:02:15:49	3.160E-04	00:04:00:56
ANN-GA	1.94E-04	00:02:48:43	1.031E-04	00:04:00:09

minutes and 24 seconds. On the other hand the times required by ANN-DS and ANN-GA models are 2 hours 15 minutes 49 seconds and 2 hours 48 minutes 43 seconds respectively. This clearly shows that GMS-GA model is computationally less efficient than ANN-DS and ANN-GA models. For the second study area, time required by GMS-GA model to solve the problem is 5 days 1 hours 54 minutes and 45 seconds. On the other hand the times required by ANN-DS and ANN-GA models are 4 hours 56 seconds and 4 hours 9 seconds respectively. This shows that GMS-GA model is computationally less efficient than ANN-DS and ANN-GA models. Further it can be observed from the table that ANN-DS and ANN-GA models are less efficient in predicting the unknown sources than GMS-GA model. Thus it can be concluded that ANN-DS and ANN-GA models are computationally more efficient than GMS-GA model but inferior in terms of prediction capabilities of unknown pollution sources.

In summary, it can be concluded that in terms of predicting capability GMS-GA model is more efficient than ANN-GA and ANN-DS model. However, in terms of computational time requirement, ANN-GA and ANN-DS is more efficient than GMS-GA model. As such there is a need to develop a new methodology to reduce the computational time of the simulation-optimization process while maintaining the predicting capability at par GMS-GA model.

4.3 Conclusions

This chapter presents two linked simulation-optimization models for optimal identification of unknown groundwater pollution sources. In the first model, the artificial neural network (ANN) is linked with the direct search (DS) based optimization model. In the second approach, ANN model is linked externally with GA based optimization model for solving source identification problem. The main advantage of ANN-DS and ANN-GA models is that it takes only few hours to solve a relatively medium scale source identification problem of medium scale aquifer system. This approach drastically reduces the computational time of the simulation-optimization model. The problem which was solved in few days using GMS-GA approach can now be solved in few hours. However, most of the time, it yields only the near optimal solution. The main drawback is that the quality of the solution is much inferior as compared to GMS-GA model. In order to improve computational efficiency and the predicting capability, the next chapter, proposes a hybrid optimization technique where, ANN-GA model is used to solve the pollution source identification problem. The solution obtained by ANN-GA model is then refined using GMS-DS approach.

Chapter 5

Development of Simulation-Optimization Models Using Hybrid-Optimization for Identification of Unknown Groundwater Pollution Sources

5. General

In the previous chapter, we have developed ANN based simulation optimization model for obtaining unknown pollution sources in a groundwater aquifer. In this method, the ANN model is linked externally with the optimization model. Using this approach, we can reduce the computational time of the simulation-optimization model drastically. However, the solution obtained using the methodology is not as accurate as obtained by GMS-GA model. To overcome the disadvantages of the approach, we have proposed a hybrid optimization technique to obtain the unknown groundwater pollution sources.

5.1 Methodology Development

5.1.1. Development of simulation-optimization methodology

For improving efficiency in both computational time and predicting capability, we have proposed a hybrid optimization approach for optimal identification of pollution sources in an aquifer. In this approach, initially the simulation-optimization model is solved by using ANN-GA approach. The solution obtained by ANN-GA model is then used as the initial solution for GMS-DS model. This has reduced the computational burden and at the same time it also improves the predicting efficiency of the model. The schematic representation of the methodology is presented in Fig. 5.1. The model is named as Hybrid-Optimization model. The efficiency and accuracy of the proposed approach are evaluated using first and second illustrative study areas.

5.1.2 Hybrid-Optimization model

A hybrid function is an optimization function that runs after one algorithm terminates. The final point/solution obtained from one optimization algorithm is used as the initial

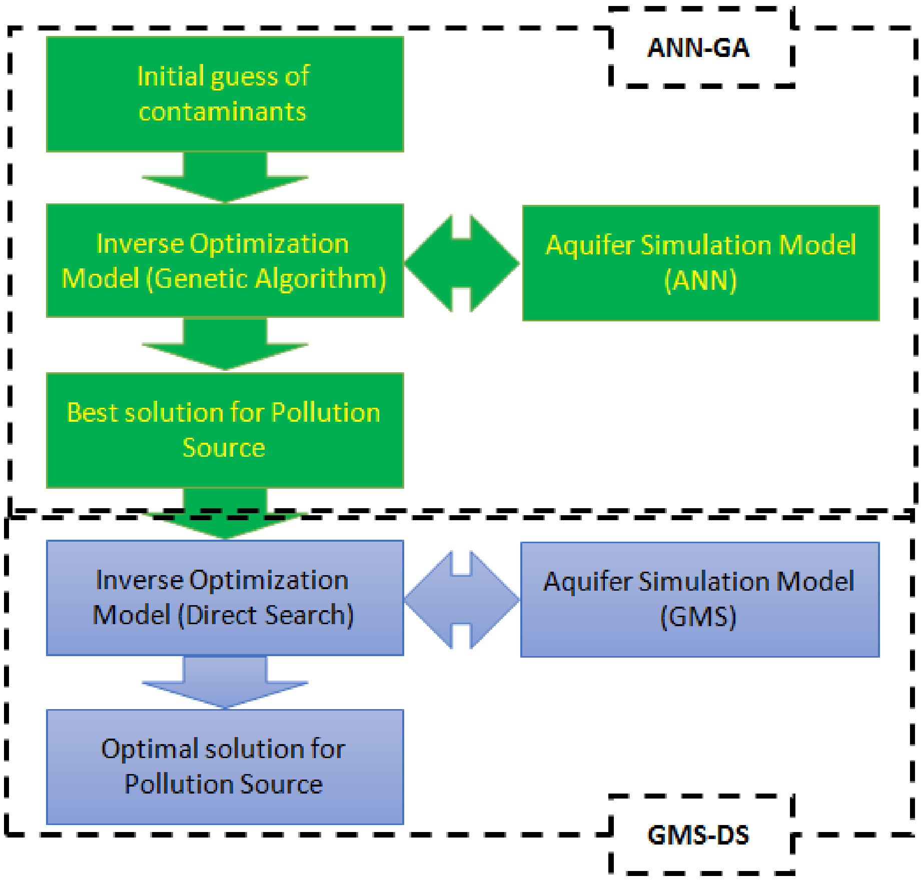


Fig.5.1: Schematic representation of Hybrid-Optimization Methodology

point of the next algorithm. As shown in the flowchart in Fig. 5.1, initially genetic algorithm is used to solve the ANN based simulation optimization model. As ANN model is used in place of GMS, less computational time is required by the simulation-optimization model to obtain the solution. However, as discussed earlier, the solution obtained by the ANN-GA model may not be the true optimal solution of the problem, rather it may be a near optimal solution. As such, the best solution obtained using the ANN-GA model is then used as the initial solution for GMS-DS model. As GMS-DS algorithm starts from a near optimal solution, it could predict the true optimal solution very quickly. Further, as on later stage, GMS is used in place of ANN, the solution obtained by the hybrid-optimization model is similar to the solution obtained by the GMS-GA or GMS-DS model.

5.2 Results and Discussion

5.2.1 Performance of Hybrid-Optimization models for first study area

The Hybrid-Optimization model is utilized to obtain the unknown pollution sources of first study area. The parameters and stopping criteria used in genetic algorithms and direct search optimization model are same with the previous models. The results obtained by the proposed model have been compared with the actual source concentration. In order to show the relative efficiency, the results obtained by the model are also compared with the results obtained by GMS-GA model and ANN-GA models. Table 5.1 shows the estimated concentration (EC) obtained by using GMS-GA, ANN-GA and Hybrid-Optimization methodologies along with the actual concentration (AC). This comparison shows that the results obtained by proposed Hybrid-Optimization method are comparable with GMS-GA model and are more closed to the actual concentration.

Table 5.1: Actual Concentration (AC) and Estimated Concentration (EC) obtained using GMS-GA, ANN-GA, Hybrid-Optimization

Time Step	Source Flux (gm/sec)											
	Actual Concentration			Estimated Concentration by GMS-GA			Estimated Concentration by ANN-GA			Estimated Concentration by Hybrid-Optimization		
	Source			Source			Source			Source		
	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
1	47.00	30.00	0	45.29	29.57	0.234	41.67	27.60	0.192	45.38	29.46	0.21
2	15.00	58.80	0	17.91	55.46	0	18.90	54.63	0	17.83	55.12	0
3	37.00	0	0	31.37	2.64	0	27.68	2.49	0	30.96	2.54	0
4	0	35.00	0	1.87	34.39	0.182	1.69	33.13	0.177	1.77	34.20	0.18

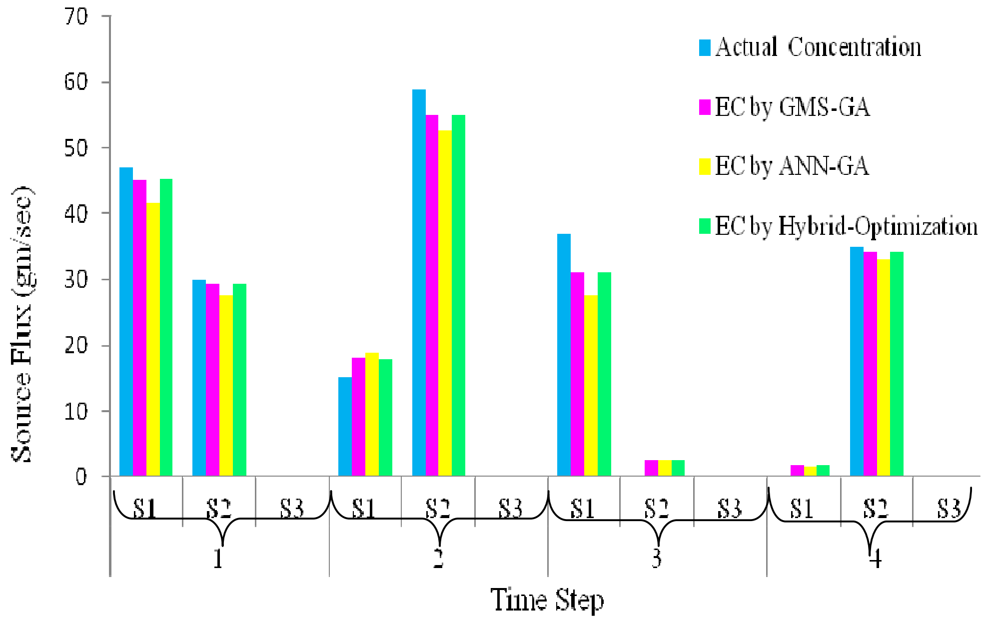


Fig. 5.2: Actual and Estimated Concentration by different models for first study area

For example, the actual concentration of source S1 at time step 1 is 47gm/sec. The concentration predicted by the GMS-GA model and Hybrid-Optimization model are 45.29 gm/sec and 45.38 gm/sec respectively. The concentration predicted by ANN-GA model is 41.67 gm/sec. It shows that the concentration predicted by the Hybrid-Optimization is slightly better GMS-GA methodology and superior than ANN-GA model. For the second time step of S1 source location, the actual source concentration is 15 gm/sec. The concentration predicted by ANN-GA, GMS-GA and Hybrid-Optimization models are 18.90 gm/sec, 17.91 gm/sec and 17.83 gm/sec respectively. In this case also, the concentration estimated by the hybrid optimization model is better than GMS-GA model and superior than ANN-GA model. Similar trend has also been observed in the other sources and other time steps. Thus it can be said that the concentration predicted by the Hybrid-Optimization model is much closed to the actual concentration. Fig. 5.2 shows the comparison of the concentrations as bar diagram. The figure also shows that the result obtained by the Hybrid-Optimization model is much closer to the actual concentration than GMS-GA model. We have also evaluated the relative efficiency of the hybrid optimization model over GMS-GA and ANN-GA models. Table 5.2 shows that relative error achieved by GMS-GA, ANN-GA and Hybrid-optimization models.

Table 5.2: Relative error (%) for GMS-GA, ANN-GA and Hybrid-optimization models

Time Step	Relative Error (%)								
	GMS-GA			ANN-GA			Hybrid-Optimization		
	Source			Source			Source		
	S1	S2	S3	S1	S2	S3	S1	S2	S3
1	3.64	1.43	-	11.34	7.97	-	3.43	1.80	-
2	19.40	5.68	-	26.03	7.09	-	18.90	4.26	-
3	15.22	-	-	25.18	-	-	12.30	-	-
4	-	1.74	-	-	5.33	-	-	1.29	-

The relative error is calculated with respect to the actual concentration. It can be observed from the table that the relative error of the Hybrid-Optimization model is lesser than ANN-GA model and GMS-GA model. For example, for the first time step, relative error achieved by Hybrid-Optimization model for S1 source is 3.43% and that is achieved by GMS-GA methodology is 3.64 %. The relative error achieved by ANN-GA is 11.34%. For the second time step, relative error achieved by Hybrid-Optimization, GMS-GA and ANN-GA models at S1 source are 18.90%, 19.40% and 26.03% respectively. Similarly for third and fourth time steps also, the relative errors achieved by the Hybrid-Optimization model are lesser than the errors achieved by GMS-GA and ANN-GA

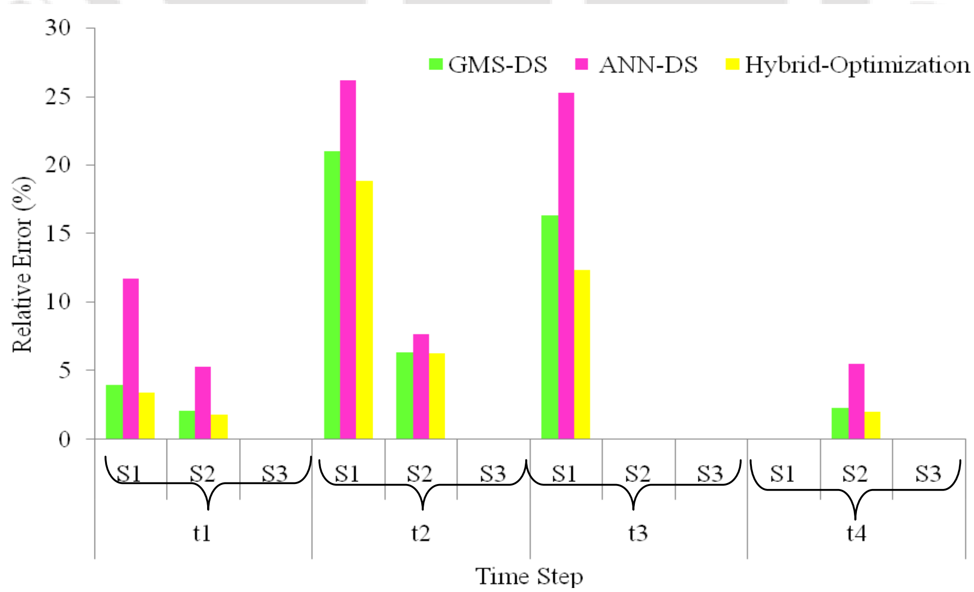


Fig. 5.3: Comparison of Relative Error of GMS-GA, ANN-GA and Hybrid-Optimization model

models. It clearly shows that the performance of Hybrid-Optimization is better than GMS-GA and ANN-GA models. Fig. 5.3 shows the comparison of relative errors achieved by three models as bar diagram. This evaluation of the results shows that for the first study area, the hybrid-optimization model yields better estimation of unknown groundwater pollution sources.

5.2.2 Performance of Hybrid-Optimization models for second study area

The hybrid optimization model is also applied to the second study area. The unknown source flux obtained by using Hybrid-Optimization model is compared with the actual source flux value and the source flux obtained by GMS-GA and ANN-GA models. The concentration values obtained by Hybrid-Optimization, GMS-GA and ANN-GA models are shown in Table 5.3. Similar to the first study area, in this case also, the concentration predicted by Hybrid-Optimization model is closer to the actual concentration.

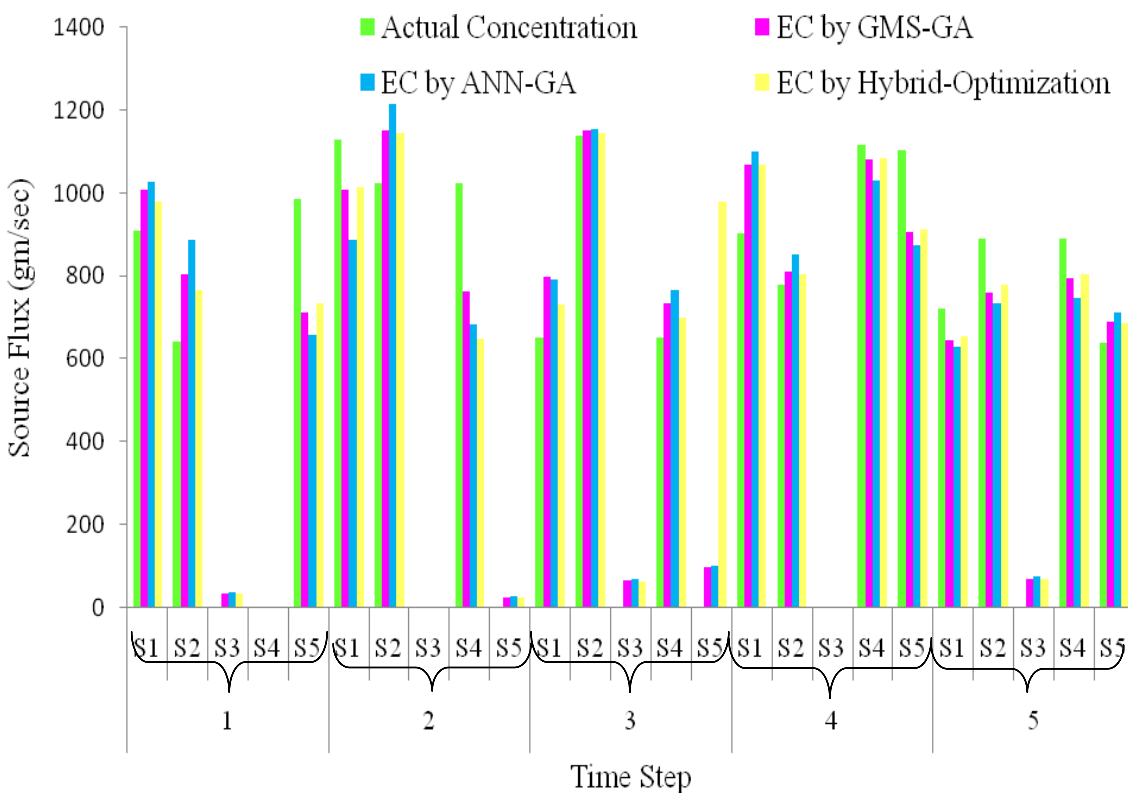


Fig. 5.4: Source Fluxes obtained by using GMS-DS, ANN-DS and Hybrid-Optimization for second study area

For example, the actual concentration at source S1 at first time step is 908.42 gm/sec. The concentration predicted by the Hybrid-Optimization model is 978.66 gm/sec. The concentration predicted by GMS-GA and ANN-GA models are 990.72 gm/sec and 1027.06 gm/sec respectively. For S2 source location at time step 1, the actual

Table 5.3: Comparison of actual and predicted concentration for second study area

Time Step	Source Location	Actual Conc. (gm/sec)	Predicted Conc. (gm/sec) GMS-GA	ANN-GA	Hybrid-Optimization
1	S1	908.42	990.72	1027.06	988.66
	S2	644.02	771.30	818.54	765.23
	S3	0	36.19	38.19	36.05
	S4	0	0	0	0
	S5	987.08	748.60	659.56	754.46
2	S1	1130.50	1010.78	888.34	1015.34
	S2	1023.87	1149.70	1210.01	1147.04
	S3	0	0	0	0
	S4	1024.16	765.87	697.76	777.92
	S5	0	23.97	21.33	24.04
3	S1	653.35	730.83	791.46	731.06
	S2	1139.88	1133.27	1124.95	1145.96
	S3	0	66.74	64.39	64.74
	S4	652.05	733.36	766.29	701.31
	S5	0	97.07	99.23	978.51
4	S1	902.13	1068.66	1099.69	1018.79
	S2	781.09	807.35	851.15	803.83
	S3	0	0	0.00	0
	S4	1117.45	1083.70	1017.21	1083.80
	S5	1104.82	910.59	882.86	913.09
5	S1	721.25	656.26	630.00	659.70
	S2	889.77	775.26	744.20	778.86
	S3	0	70.64	70.75	69.85
	S4	889.77	797.95	747.77	804.58
	S5	639.93	690.61	711.15	687.67

concentration is 644.02 gm/sec. The concentration predicted by the Hybrid-Optimization model is 765.23 gm/sec. The concentration predicted by GMS-GA and ANN-GA models are 771.30 gm/sec and 818.54gm/sec respectively. Thus the concentration predicted by the hybrid optimization model is better than GMS-GA model and much superior than the

ANN-GA model. For the other sources and time steps also, the concentration predicted by the Hybrid-Optimization is better than GMS-GA and ANN-GA model. Fig. 5.4 shows the comparison of these methodologies using a bar diagram.

The evaluation of relative error also shows that the performance of hybrid optimization model is better than GMS-GA model and ANN-GA model. Table 5.4 shows the relative error achieved by Hybrid-Optimization, GMS-GA and ANN-GA methodologies. It can be observed that the relative error achieved by Hybrid-Optimization model is lesser than GMS-GA and ANN-GA models. For the first time step, relative error achieved for S1 source by Hybrid-Optimization is 7.73% and that of achieved by GMS-GA methodology is 10.25 %. The relative error achieved by ANN-GA methodology is 13.06%. For the S2 source location at time step 1, the relative error achieved by Hybrid-Optimization, GMS-

Table 5.4: Relative error (%) for GMS-GA, ANN-GA, Hybrid-Optimization models for second study area

	Relative Error (%)														
	GMS-GA					ANN-GA					Hybrid-Optimization				
	Time Step					Time Step					Time Step				
SL	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
S1	9.06	10.59	11.86	18.46	9.01	13.06	21.42	21.14	21.90	12.65	8.83	10.19	11.89	12.93	8,53
S2	19.76	12.29	0.58	3.36	12.87	27.10	18.18	1.31	8.97	16.36	18.82	12.03	0.534	2.92	12.46
S3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S4	-	25.22	12.47	3.02	10.32	-	31.87	17.52	8.97	15.96	-	24.04	7.55	3.01	9.57
S5	24.16	-	-	17.58	7.92	33.18	-	-	20.09	11.13	23.57	-	-	17.3	6.94

GA and ANN-GA models are 18.82%, 24.87% and 27.10% respectively. This clearly indicates that the Hybrid-Optimization is better than other two methodologies. Fig. 5.5 shows the comparison of the relative errors of Hybrid-Optimization, GMS-DS, and ANN-DS models as a bar diagram.

5.2.3 Comparative evaluation of Hybrid-Optimization model with GMS-GA and ANN-GA models

The hybrid optimization model is also evaluated in terms of computation time and objective function value. Table 5.5 shows the comparison of the objective function value and computational time requirement for Hybrid-Optimization, GMS-GA and ANN-GA methods for both the study areas. It may be observed from the table that, for first study

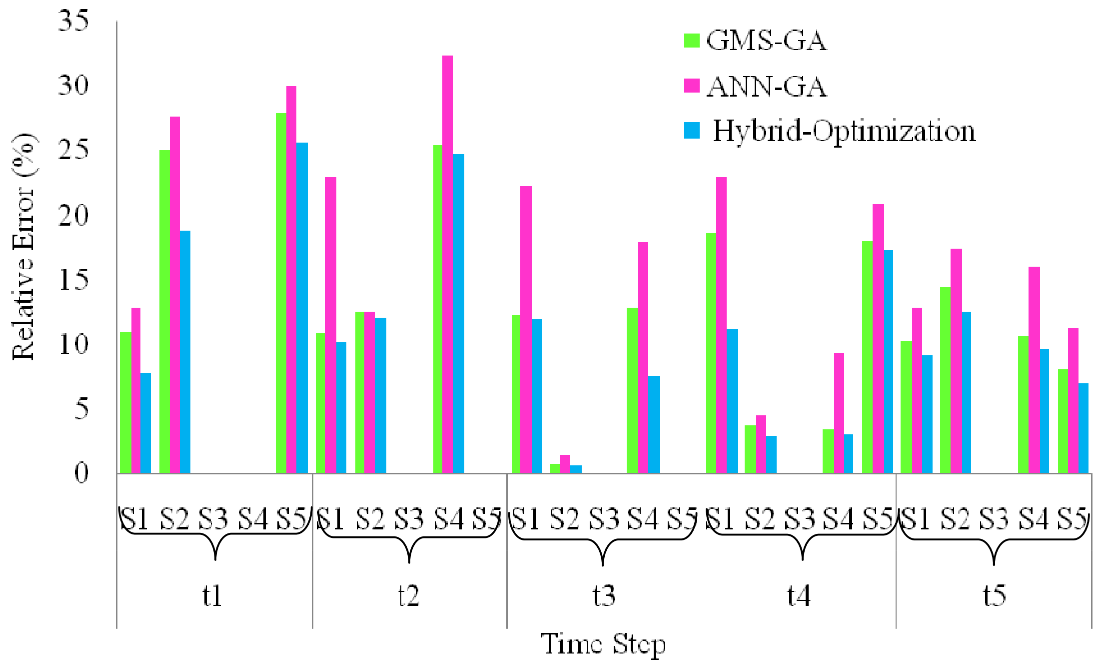


Fig. 5.5: Comparison of Relative Error of GMS-GA, ANN-GA and Hybrid-Optimization model for second illustrative study area

area, the time required for solving the problem by GMS-GA and ANN-GA models are 2 days 1 hour 15 minutes 24 seconds and 2 hour 48 minutes 43 seconds respectively. On the other hand, the time required by Hybrid-Optimization model is 1 day 2 hours and 23 seconds. This has shown that the hybrid optimization model is computationally more efficient than GMS-GA model. Further, from the objective function point of view, the Hybrid-Optimization model is better than ANN-GA model, though it is not computationally efficient than ANN-GA model. For the second study area, time required

Table 5.5: Comparative evaluation of GMS-GA, ANN-GA and Hybrid-Optimization model for second illustrative study area

Models	First Study area		Second Study area	
	Obj. funct.value	Time DD HH MM SS	Obj. funct. value	Time DD HH MM SS
GMS-GA	1.34E-06	02:01:15:24	1.34E-06	05:01:54:45
ANN-GA	1.94E-04	00:02:48:43	1.031E-04	00:04:00:09
Hybrid-Optimization	1.34E-06	01 02 00 23	3.43E-05	03 08 02 19

by GMS-GA and ANN-GA models to solve the problem are 5 days 1 hour 54 minutes 45 seconds and 4 hours 9 seconds respectively. The time taken by the Hybrid-Optimization model is 3 days 8 hours 2 minutes and 19 seconds. This shows that for the second study area also, the Hybrid-Optimization model is computationally more efficient than GMS-GA model. From the objective function point of view, the prediction capability of Hybrid-Optimization model is better than GMS-GA and ANN-GA models. The ANN-GA model is very fast, but it lacks on the prediction capability of unknown groundwater pollution sources.

5.3 Conclusions

This chapter presents a hybrid optimization model for optimal identification of unknown groundwater pollution sources. The performance of the source identification model is highly related to the aquifer simulation model. Incorporation of GMS model will give better performance, but the model will be more computationally expensive. On the other hand, the model will be less computationally expensive, if ANN model is incorporated with the optimization model. But the quality of the solution will be inferior. The proposed hybrid optimization model comes out as an attractive alternative solution procedure where efficiency in both computational time and predicting performance can be achieved. Application of the model on the illustrative study areas shows that GMS-GA model has better predicting capability. But the model is computationally extensive. The ANN-GA model is computationally efficient, but the predicting capability is inferior then GMS-GA model. The hybrid optimization model is more efficient in predicting the unknown pollution sources. The model is also computationally better than GMS-GA model. These limited evaluations show that the Hybrid-Optimization model may be a promising method for solving real world source identification problem. However, the model is not efficient as ANN-GA model. As such, in the next chapter we have proposed a new simulation-optimization methodology using GMS and ANN simulation models together.

Chapter 6

Development of Simulation-Optimization Models by Linking GMS and ANN Simulation Model with GA based Optimization Model

6. General

In the previous chapter, we have presented a hybrid-optimization model for identification of unknown groundwater pollution sources. The hybrid-optimization model is more computationally efficient than GMS-GA model. The predicting capability of the model is also better than ANN-GA model. However, the hybrid-optimization model is not computationally efficient as compared to ANN based simulation-optimization model. As such some improvement in computational time is necessary. In this chapter, we have presented another improved methodology which is computationally better than GMS-GA model while maintaining predicting capability at per GMS-GA model. The methodology incorporates both numerical and approximate simulation models simultaneously. In this methodology, the fitness values of the few best individual of the population are calculated using GMS model and fitness of the other individuals are calculated using the ANN model. This technique has drastically reduced the computational time of the model. The performance of the model is evaluated using the illustrative study areas.

6.1 Methodology Development

6.1.1. Development of simulation-optimization methodology

In order to achieve efficiency in both computational time and predicting capability, we have proposed a new simulation-optimization methodology using GMS and ANN simulation models together. In this algorithm, initially the fitness value of all the population are calculated using the ANN model. The population is then shorted in descending order. The shorted population is then divided in two groups. The fitness values of the better $x\%$ of the population are calculated using the GMS model. The fitness values of the remaining (100-x) % of population are calculated using ANN model. The population is then sent through the normal genetic algorithms processes. In genetic algorithm, the better individuals have greater probability to be selected to the next

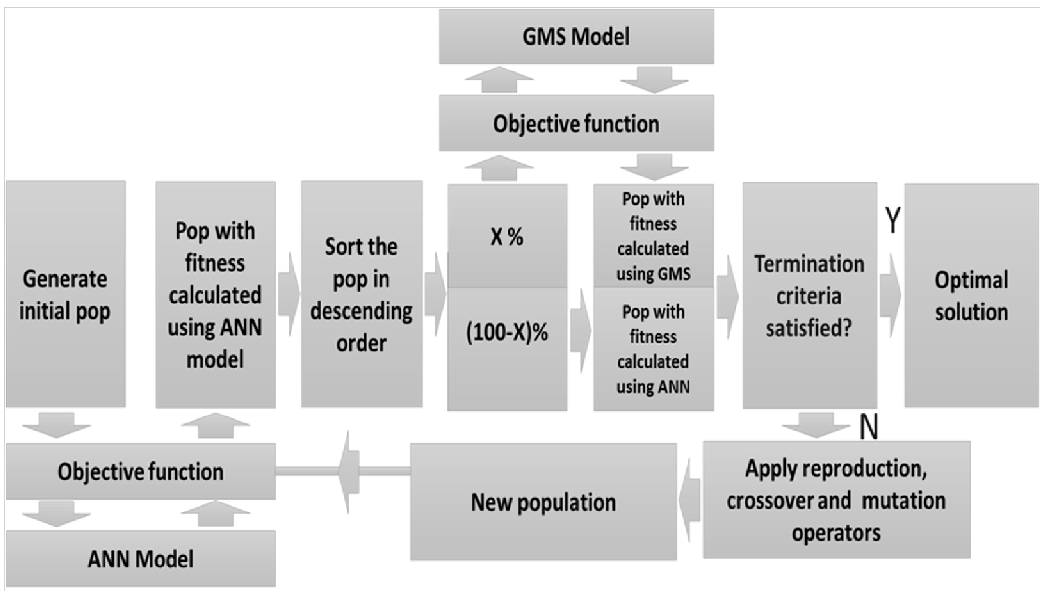


Fig. 6.1: Flow chart of ANN-GMS-GA algorithm

generation and the fitness values of these individuals have been calculated using the GMS model. As such, the performance of the model will be similar to GMS-GA model. The solutions are then checking for the termination criteria. If the termination criteria are satisfied, the optimal solution will be displayed. Otherwise, the solution will pass through the genetic operators to obtain new solutions. This iterative process will continue until termination criteria are not satisfied. As both ANN and GMS are used in this methodology and GA is used to solve the inverse model, the methodology can be named as ANN-GMS-GA model. Fig. 6.1 shows the flowchart of the algorithm.

As discussed earlier, in case of ANN-GMS-GA model, x is the percentage of the population whose fitness values have been calculated using GMS model. Fitness values of the rest population have been calculated using ANN model. It is worth to mention here that when x is equal to 0, the model is equivalent to ANN-GA model and when x is equal to 100, the model is equivalent to GMS-GA model. In this study, we have done two experiments. In the first experiment, the model is applied without the sorting operation. Thus the fitness of the top $x\%$ of the solution is calculated using GMS model. Initial value of x considered is 10 and the x value is the increased upto 90% at an interval of 10%, *i.e.* we have calculate the fitness value for x of 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80% and 90% of solutions by GMS simulator. In the second experiment, we have applied the sorting operator. Thus the fitness of the best $x\%$ of the solution is calculated using GMS model. In this experiment (with sorting), we have considered x value of 10, 20 and 30 percent only. This methodology is applied to both the study areas.

6.2. Results and Discussion

6.2.1 Performance evaluation of the simulation-optimization model without sorting

The performance of the proposed ANN GMS GA based source identification model is evaluated for different combination of GMS and ANN simulation for first and second study areas. As mentioned above, the performance of the model is evaluated with or without shorting operator. This is just to see the effect of sorting operator in the algorithm. Table 6.1 presents the source concentration obtained for different percentage of x when sorting algorithm was not used in the model. In this case, the fitness of the initial $x\%$ of the solutions of the population is calculated using GMS model and the fitness of $(100 - x)\%$ of the solutions is calculated using ANN model. The concentration obtained by ANN-GMS-GA model is compared with actual concentration and the concentration estimated using GMS-GA, ANN-GA and hybrid-optimization models. It can be observed from the table that quality of the solutions has been improved when more percentage of solutions are simulated by using GMS model. For example, the concentration estimated for the first time step by the ANN-GMS-GA model at S1 source for x of 10% is 43.32 gm/sec. The estimated value of concentration for x of 20% is 43.95gm/sec which is better than the value obtained for x of 10%. The source concentration for x equal to 30%, 40%, 50%, 60%, 70%, 80% and 90% are, 44.36gm/sec, 44.60 gm/sec, 44.70gm/sec, 44.97gm/sec, 44.99gm/sec, 45.03gm/sec and 45.06gm/sec respectively. The actual concentration is 47 gm/sec. Thus with the increase in x value, the predicting capability of the model is also improving. The concentration estimated by Hybrid-Optimization, GMS-GA and ANN-GA are 45.38 gm/sec, 45.29 gm/sec and 41.7 gm/sec respectively (from Table 5.1). Thus for x equal to 90%, the concentration predicted by the ANN-GMS-GA model is nearly equivalent to GMS-GA model which is also closed to the actual concentration. On the other hand, for x equal to 10%, the estimation of ANN-GMS-GA model is equivalent to ANN-GA model. Similarly, for the other time steps also similar trend have been observed. Thus, it can be summarized that the predicting capability of the model increases with the increase in $x\%$ value and the

Table 6.1: Comparison of actual concentration and predicted concentration by ANN-GMS-GA model for first study area (without sorting)

Time Step	Source (gm/sec)	Actual Conc.	ANN-GA	Concentration predicted by ANN-GMS-GA									GMS-GA
				Percentage of population ($x\%$)									
			$x = 0$	10	20	30	40	50	60	70	80	90	$x = 100$
1	S1	47	41.7	43.32	43.95	44.36	44.60	44.70	44.97	44.99	45.03	45.06	45.29
	S2	30	27.6	28.6	28.6	28.9	29.1	29.1	29.2	29.2	29.3	29.3	29.6
	S3	0	0.19	0.19	0.2	0.2	0.19	0.19	0.20	0.217	0.22	0.22	0.23
2	S1	15	18.9	18.5	18.5	18.4	18.3	18.3	18.2	18.2	18.2	18.2	17.9
	S2	58.8	54.6	54.4	54.5	54.5	54.6	54.6	54.8	54.9	55	55	55.5
	S3	0	0	0	0	0	0	0	0	0	0	0	0
3	S1	37	27.7	29.7	29.9	30.2	30.5	30.6	30.7	30.8	30.9	30.9	31.4
	S2	0	2.49	2.33	2.4	2.52	2.43	2.45	2.44	2.44	2.43	2.43	2.64
	S3	0	0	0	0	0	0	0	0	0	0	0	0
4	S1	0	1.69	1.77	1.79	1.81	1.87	1.85	1.82	1.8	1.73	1.73	1,87
	S2	35	33.1	33.4	33.4	33.5	33.5	33.5	33.9	33.8	33.9	34	34.4
	S3	0	0.17	0.17	0.17	0.18	0.18	0.18	0.18	0.18	0.18	0.18	0.18

model gives the best performance, when x is equal to 100%, *i.e.* GMS-GA model.

The performance of ANN-GMS-GA model is also evaluated on second study area. As done in the first study area, concentrations at different source locations are obtained for different values of x. Table 6.2 shows the concentration results obtained for different percentage of x. In this case also, the concentration obtained by ANN-GMS-GA model is compared with actual concentration, GMS-GA, ANN-GA and hybrid-optimization. Like the first study area, the predicting capability of the model has improved with the increase

Table. 6.2 Comparison of actual concentration and predicted concentration by ANN-GMS-GA model for second study area (without sorting)

Time Step	SL	Actual Conc. (gm/sec)	Concentration Predicted by ANN-GMS-GA								
			$x\%$								
			10	20	30	40	50	60	70	80	90
1	S1	908.42	1023.52	1020.15	1018.79	1014.8	1012.8	1011.16	1007.17	1004.35	1001.44
	S2	644.02	807.15	805.99	804.18	800.77	794.91	793.5	791.44	786.54	786.28
	S3	0	38.04	38.91	38.25	37.92	37.22	37.32	37.04	37.01	36.23
	S4	0	0	0	0	0	0	0	0	0	0
	S5	987.08	684.34	686.02	688.09	696.98	697.57	702.99	707.04	711.39	725.11
2	S1	1130.5	896.6	892.98	912.65	926.67	883.4	969.74	979.47	996.54	1007.5
	S2	1023.87	1191.89	1189.73	1183.18	1168.54	1164.34	1161.79	1158.1	1153.49	1152.37
	S3	0	0	0	0	0	0	0	0	0	0
	S4	1024.16	714.55	714.55	722.74	729.61	740.64	745.95	751.22	755.52	759.1
	S5	0	29.39	29.28	28.66	26.96	25.86	25.54	25.32	24.93	24.08
3	S1	653.35	784.87	783.36	779.9	768.6	754.94	748.41	743.25	735.48	732.4
	S2	1139.88	1126.99	1129.62	1130.65	1131.78	1132.58	1132.13	1132.36	1133.04	1133.04
	S3	0	69.57	62.23	69.46	69.84	70.17	69.17	69.02	68.18	67.49
	S4	652.05	759.89	759.9	752.4	748.29	743.99	742.55	740.47	737.27	734.99
	S5	0	99.19	98.99	98.79	98.66	98.52	98.19	97.99	97.92	97.53
4	S1	902.13	1092.57	1090.4	1085.26	1082.91	1080.39	1077.05	1074.61	1070.02	1069.57
	S2	781.09	850.45	845.06	844.12	838.81	831.47	827.01	820.92	815.85	810.38
	S3	0	0	0	0	0	0	0	0	0	0
	S4	1117.45	1035.99	1035.99	1047.72	1057.67	1065.82	1066.71	1069.96	1076.66	1081.8
	S5	1104.82	884.74	893.52	899.1	899.99	901.63	903.52	904.44	905.01	908.62
5	S1	721.25	639.97	641.55	642.85	647.03	648.33	649.7	651.65	653.16	654.25
	S2	889.77	747.23	750.43	753.01	759.7	767.6	768.58	772.05	775.07	781.04
	S3	0	70.64	71.79	72.63	72.94	70.65	70.69	70.67	70.51	70.72
	S4	889.77	763.16	766.18	768.49	773.12	779.79	784.15	788.43	792.07	796.97
	S5	639.93	704.5	702.45	700.27	699.5	698.23	696.69	695.41	693.55	691.76

in x value. For example, the concentration estimated for the first time step by ANN-GMS-GA model for source location S1 for x of 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80% and 90% are 1023.52 gm/sec, 1020.15 gm/sec, 1018.79 gm/sec, 1014.8 gm/sec, 1012.8 gm/sec, 1011.16 gm/sec, 1007.17 gm/sec, 1004.35 gm/sec and 1001.44 gm/sec respectively. Thus in this case also it has been observed that, with the increase of x value, the predicting capability of the model has also increased. From Table 5.3, it is observed that concentration achieved by GMS-GA, ANN-GA and Hybrid-Optimization are 990.72gm/sec, 1027.06 gm/sec and 978.66 gm/sec respectivrly and the actual concentration is 908.42gm/sec. It may be noted that when 90% of the population is simulated by GMS simulator, the concentration estimated by the ANN-GMS-GA model is 1001.44 gm/sec which is near to the solution obtained by GMS-GA model and also closed to the actual concentration. It can be seen that for without sorting cases, the model gives the best performance, equivalent to the GMS-GA model when x equal to 100%.

The relative efficiency of ANN-GMS-GA model is also evaluated using relative error. Table 6.3 shows that relative error achieved by ANN-GMS-GA model in predicting source concentration of first study area when applied without sorting operator. As per the relative error criteria also, the model shows improved performance when the value of x increases, and best results has been achieved for x equal to 100%. For the first time step, relative error achieved by ANN-GA model for S1 source is 11.34% and that is achieved by ANN-GMS-GA (x=90%) methodology is 3.98%. The relative error achieved by GMS-GA model is 3.63% only. Similarly, for S2 source, at first time step, relative error achieved by ANN-GMS-GA model is 2.11% (x=90%), whereas that is achieved by ANN-

Table. 6.3 Relative error (%) achieved by ANN-GMS-GA for first study area (without sorting)

Time Step	SL	Relative error (%) of ANN-GMS-GA model										
		ANN-GA	x%									GMS-GA
		$x = 0$	10	20	30	40	50	60	70	80	90	$x = 100$
1	S1	11.34	7.83	6.49	5.62	5.11	4.89	4.32	4.11	4.02	3.98	3.63
	S2	7.97	4.76	4.56	3.56	3.15	2.93	2.79	2.53	2.17	2.11	1.43
	S3	-	-	-	-	-	-	-	-	-	-	-
2	S1	26.13	23.5	23.2	22.8	22.2	22	21.4	21.3	21.1	21.03	19.4
	S2	7.09	7.45	7.39	7.33	7.23	7.07	6.89	6.71	6.32	6.24	5.68
	S3	-	-	-	-	-	-	-	-	-	-	-
3	S1	25.2	19.7	19.3	18.3	17.5	17.2	17.1	16.8	16.5	16.3	15.2
	S2	-	-	-	-	-	-	-	-	-	-	-
	S3	-	-	-	-	-	-	-	-	-	-	-
4	S1	-	-	-	-	-	-	-	-	-	-	-
	S2	5.33	4.89	4.65	4.52	4.37	4.19	4.13	3.46	3.19	2.86	1.74
	S3	-	-	-	-	-	-	-	-	-	-	-

-GA and GMS-GA models are 7.97% and 1.43% respectively. Thus it can be concluded GMS-GA model gives the best results while ANN-GMS-GA model gives better performance when 100% of solutions of the population are evaluated using GMS model. Similar results have also been achieved in case of the second study area. Table 6.4 shows the relative error achieved by the model for different values of x. In this study area also, the relative error achieved by ANN-GMS-GA model is very low when fitness value of more solutions are calculated using GMS model. The relative error achieved by ANN-GA model is very high as compared to GMS-GA and ANN-GMS-GA (x=90%) model. For

Table 6.4: Relative error (%) achieved by ANN-GMS-GA for second study area (without sorting)

Time Step	SL	Relative Error (%) for ANN-GMS-GA model										
		ANN -GA	x%									GMS-GA
		x=0	10	20	30	40	50	60	70	80	90	x=100
t1	S1	13.06	12.67	12.3	12.15	11.71	11.49	11.31	10.87	10.56	10.24	9.06
	S2	27.09	25.33	25.15	24.87	24.34	23.43	23.21	22.89	22.13	22.09	19.76
	S3	-	-	-	-	-	-	-	-	-	-	-
	S4	-	-	-	-	-	-	-	-	-	-	-
	S5	33.18	30.67	30.23	30.11	29.39	29.33	28.78	28.37	27.93	26.54	24.16
t2	S1	21.42	21.19	21.01	19.27	18.03	15.34	14.22	13.36	11.85	10.88	10.59
	S2	18.18	16.41	16.2	15.56	14.13	13.72	13.47	13.11	12.66	12.55	12.29
	S3	-	-	-	-	-	-	-	-	-	-	-
	S4	31.87	30.23	30.1	29.43	28.76	28.17	27.16	26.65	26.23	25.88	25.22
	S5	-	-	-	-	-	-	-	-	-	-	-
t3	S1	21.13	20.13	19.9	19.37	17.64	15.55	14.55	13.76	12.57	12.1	11.86
	S2	1.309	1.13	0.9	0.81	0.71	0.69	0.68	0.66	0.61	0.6	0.58
	S3	-	-	-	-	-	-	-	-	-	-	-
	S4	17.52	16.54	16.32	15.39	14.76	14.1	13.88	13.56	13.07	12.72	12.47
	S5	-	-	-	-	-	-	-	-	-	-	-
t4	S1	21.89	21.11	20.87	20.3	20.04	19.76	19.39	19.12	18.61	18.56	18.46
	S2	8.96	8.88	8.19	8.07	7.39	6.45	5.88	5.1	4.45	3.75	3.36
	S3	-	-	-	-	-	-	-	-	-	-	-
	S4	8.97	7.73	7.29	6.24	5.35	4.62	4.54	4.25	3.65	3.19	3.02
	S5	20.09	19.84	19.55	19.43	19.14	18.77	18.42	18.13	18.1	17.92	17.58
t5	S1	12.65	11.27	11.05	10.87	10.29	10.11	9.92	9.65	9.44	9.29	9.01
	S2	16.36	16.02	15.66	15.37	14.61	13.73	13.62	13.23	12.99	12.91	12.87
	S3	-	-	-	-	-	-	-	-	-	-	-
	S4	15.95	14.29	13.89	13.63	13.11	12.36	11.87	11.39	10.98	10.43	10.32
	S5	11.12	10.09	9.77	9.43	9.31	9.11	8.87	8.67	8.38	8.1	7.92

example, for the first time step, relative error achieved by ANN-GA model for S1 source is 13.06% and that of achieved by ANN-GMS-GA (x=90%) methodology is 10.24%. The relative error achieved by GMS-GA model is 9.06% only. Similarly, at S2 source location for first time step, relative error achieved by ANN-GMS-GA (x=90%) model is 22.09%, whereas that is achieved by ANN-GA, GMS-GA and Hybrid-Optimization models are 27.09 gm/sec, 19.76% and 18.82 % respectively. Thus it can be conclude that ANN-GMS-GA model without sorting is not as efficient as hybrid-optimization model or GMS-GA model in identifying the unknown groundwater pollution sources.

6.2.2 Performance evaluation of the simulation-optimization model with sorting

It is already mention that, in the second experiment, ANN-GMS-GA model is applied with sorting operation. Thus the fitness of the best x% of the solutions is calculated using GMS model. Table 6.5 shows the concentration predicted by ANN-GMS-GA model for the first study area. It may be noted that the concentration estimated by ANN-GMS-GA model for first time step at x of 30% is 45.57 gm/sec. The concentration estimated by

Table 6.5: Comparison of actual concentration and predicted concentration by ANN-GMS-GA model for first study area (with sorting)

Time Step	Source Location	Actual Conc. (gm/sec)	Conc. Predicted by ANN-GMS-GA Model			Relative error (%)		
			x%			x%		
			10	20	30	10	20	30
1	S1	47.00	44.69	44.97	45.57	4.91	4.32	3.04
	S2	30.00	28.82	29.09	29.77	3.93	3.03	0.77
	S3	0	0.21	0.22	0.235	-	-	-
2	S1	15.00	18.28	18.19	18.11	21.87	21.27	20.73
	S2	58.8	54.66	54.75	55.14	7.02	6.75	6.19
	S3	0	0	0	0	-	-	-
3	S1	37.00	30.61	30.79	31.53	17.20	16.78	14.78
	S2	0	2.37	2.51	2.54	-	-	-
	S3	0	0	0	0	-	-	-
4	S1	0	1.84	1.91	1.95	-	-	-
	S2	35.00	33.53	33.81	34.25	4.21	3.39	2.14
	S3	0	0.169	0.171	0.182	-	-	-

GMS-GA and Hybrid-Optimization models are 45.29 gm/sec and 45.38 gm/sec respectively. The actual concentration at this time step with respect to S1 source location is 47gm/sec. This shows that performance of ANN-GMS-GA model with sorting is better as compared to other models. For the other time steps also same trend have been observed. Table 6.5 also shows the relative error achieved for x equal to 10%, 20% and 30 % at different time steps. The relative error (%) achieved by sorting ANN-GMS-GA model is less than the without sorting algorithm. Consider the source S1 at time step 1, it can be observed that for x equal to 30%, the relative error achieved without sorting (WOS) is 5.62% and that is achieved with sorting (WS) is 3.04% (Table 6.3 and Table 6.5). For the other time steps and source locations also we have found lesser values of relative error with sorting operator. Based on this experiment, it can be concluded that in terms of predicting capability, ANN-GMS-GA model with sorting gives the best performance for x equal to 30% and no improvement has been seen with the increase in value of x after 30%.

For the case of second study area, Table 6.6 shows the concentration predicted by ANN-GMS-GA model for different values of x when sorting operation is applied. In this case also, it is observed that the model gives the best prediction for x equal to 30%. The concentration estimated by ANN-GMS-GA model for first time step for x equal to 30% is 981.18 gm/sec whereas the concentration estimated by GMS-GA and Hybrid-Optimization models are 990.72 gm/sec and 988.66 gm/sec respectively. The actual concentration is 908.42 gm/sec. This shows that the concentration predicted by ANN-GMS-GA model with sorting is better than the other models. For the other time steps also same trend have been observed. Table 6.6 shows the relative error achieved by the model for x values of 10%, 20% and 30%. It is observed that, relative error (%) achieved by-ANN-GMS-GA model with sorting is far better than the without sorting. For example, relative error achieved by ANN-GMS-GA (without sorting) for the first time step at S1 source location is 10.24% (when x=90%) and that is obtained by the algorithm with sorting is 8.01% (when x=30%) only. For the other time steps and source locations also, the model with sorting operator achieved less relative error than without sorting algorithm. From these results, it can be concluded that ANN-GMS-GA with sorting model yields the best performance which is at par GMS-GA and Hybrid-Optimization models.

Table 6.6: Comparison of actual concentration and predicted concentration by ANN-GMS-GA model for second study area (with sorting)

Time Step	SL	Actual Conc. (gm/sec)	Concentration Predicted by ANN-GMS-GA			Relative error (%)		
			x %			x %		
			10	20	30	10	20	30
1	S1	908.42	1013.98	1004.17	981.18	11.62	10.54	8.01
	S2	644.02	794.91	786.99	770.05	23.43	22.2	19.57
	S3	0	37.29	37.14	36.22	-	-	-
	S4	0	0	0	0	-	-	-
	S5	987.08	698.16	712.77	740.61	29.27	27.79	24.97
2	S1	1130.5	955.84	998.8	1004.45	15.45	11.65	11.15
	S2	1023.87	1164.86	1153.18	1151.34	13.77	12.63	12.45
	S3	0	0	0	0	-	-	-
	S4	1024.16	734.22	754.91	764.95	28.31	26.29	25.31
	S5	0	25.56	24.98	23.99	-	-	-
3	S1	653.35	753.38	736.98	726.59	15.31	12.8	11.21
	S2	1139.88	1131.79	1132.81	1133.15	0.71	0.62	0.59
	S3	0	70.26	68.54	66.79	-	-	-
	S4	652.05	743.73	737.53	731.99	14.06	13.11	12.26
	S5	0	98.5	97.99	97.1	-	-	-
4	S1	902.13	1077.77	1070.65	1067.67	19.47	18.68	18.35
	S2	781.09	829.28	816	806.63	6.17	4.47	3.27
	S3	0	0	0	0	-	-	-
	S4	1117.45	1073.76	1073.76	1082.14	4.56	3.91	3.16
	S5	1104.82	902.75	902.75	912.91	18.57	18.29	17.37
5	S1	721.25	652.88	652.88	654.53	10.08	9.48	9.25
	S2	889.77	773.74	773.74	788.25	13.67	13.04	11.41
	S3	0	70.69	70.54	70.69	-	-	-
	S4	889.77	793.05	793.05	798.21	12.2	10.87	10.29
	S5	639.93	698.8	694.58	690.61	9.2	8.54	7.92

6.2.3 Comparative evaluation of GMS-ANN-GA model without and with sorting

The performance of the ANN-GMS-GA model is also evaluated in terms of computational time and objective function values achieved at the optimal solution. In the first experiment, the model is applied without the sorting operation. Table 6.7 shows objective function values and computational time required for different values of x in case of first study area for without sorting. It can be observed that, the computational time requirement increases with the increase in x value. As per the objective

Table 6.7: Objective function values and computational time required for first study area for ANN-GMS-GA (without sorting)

x (%)	Objective function	Comp. Time (DD:HH:MM:SS)
0(ANN-GA)	7.641-04	00:02: 48:43
10	6.08E-04	00:06: 28:12
20	3.20E-04	00:09:59:36
30	1.34E-04	00:13:23:15
40	8.46E-05	00:17:48:52
50	7.63E-05	00:21: 53:00
60	4.64E-05	01:01:21:12
70	3.29E-05	01:07:39:21
80	8.20E-06	01:12:42:54
90	6.11E-06	01:17:23:48
100(GMS-GA)	1.04E-06	02:01:15:24

function value, it is observed that the quality of the solution has also improved with the increase in x value. Fig. 6.2 shows that, best performance can be achieved when x is equal to 100%, *i.e.* nothing but the GMS-GA model. The performance of the model inferior when x is equal to 0%, *i.e.* the ANN-GA model. Table 6.8 shows the variation of objective function value and computational time required for x equal to 10 %, 20% and 30% respectively when the model is used with sorting operator. The table shows that, objective function value reduces when x value increases. It can be seen that the model

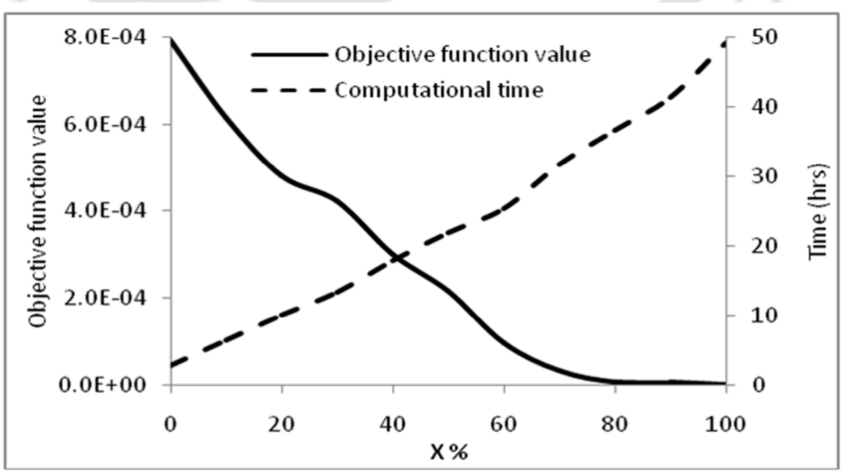


Fig. 6.2: Variation of objective function value and computational time with x for first study area (ANN-GMS-GA-WOS)

Table 6.8: Objective function values and computational time required for first illustrative study area (with sorting)

x(%)	Objective function value	Comp. Time (DD:HH:MM:SS)	Relative comp. advantage over GMS-GA model (%)
0 (ANN-GA)	7.04E-04	00:02:48:43	94.29
10	1.35E-04	00:06:23:24	87.03
20	6.45E-05	00:09:27:10	80.81
30	1.79E-06	00:13:18:50	72.97
100 (GMS-GA)	1.72E-06	02:01:15:24	00.00

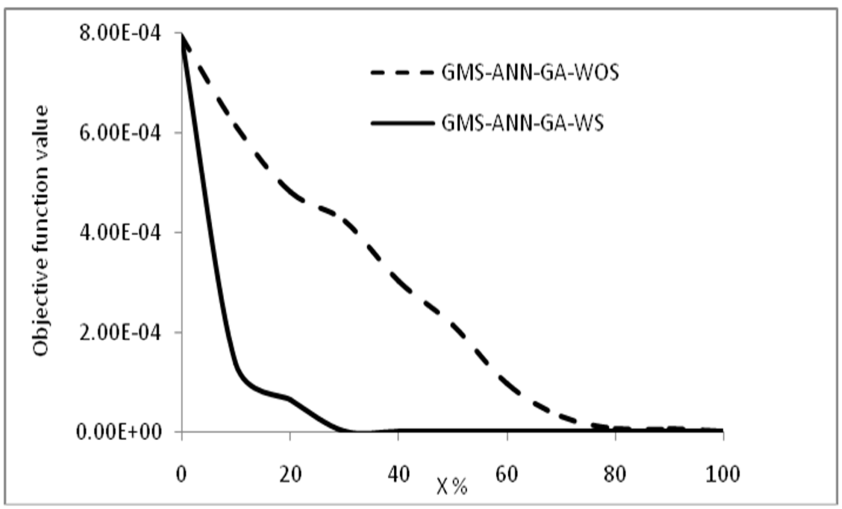


Fig.6.3: Comparison of objective function value of ANN-GMS-GA model without sorting and with sorting for first study area

gives the best performance which equivalent to GMS-GA model for x equal to 30%. As only 30% of the solutions of the population are evaluated using GMS model, the model is computationally more efficient than GMS-GA model and the Hybrid-Optimization model. Thus significant amount of computational time can be reduced in obtaining the unknown sources of a groundwater aquifer by using ANN-GMS-GA model with sorting. It can be observed that around 72.97% of computational efficiency can be achieved by the ANN-GMS-GA model over the GMS-GA model. For example, GMS-GA took 2 days 1 hour 15 minutes and 24 second to solve the problem. On the other hand, ANN-GMS-GA with sorting (GMS-ANN-GA-WS) took only 13 hours 18 minutes and 50 seconds.

Table 6.9: Objective function values and computational time required for different values of x in case of second study area (without sorting)

x(%)	Objective function	Comp. Time (DD:HH:MM:SS)
0 (ANN-GA)	9.0314E-04	00:04:00:09
10	8.0343E-04	00:12:45:54
20	5.9174E-04	01:00:27:34
30	4.4013E-04	01:06:56:20
40	3.134E-04	01: 23:57:11
50	2.5055E-04	02:07:34:23
60	1.3402E-05	02:17:36:25
70	9.6645E-05	03:01:07:45
80	6.2087E-05	03:12:42:54
90	9.7765E-06	04:05:35:30
100 (GMS-GA)	1.3402E-06	05:01:54:45

Table 6.9 shows the objective function values and computational time required by the ANN-GMS-GA model without sorting for the second study area. In this case also, the computational time increases with the increase in x value. Fig. 6.4 shows the variation of objective function value and computational time requirement with x value. For the model without sorting operator, the best performance can be achieved when x is equal to 100%. This experiment shows that model will be efficient in capturing the unknown pollution sources, when x is equal to 100% and it is computationally inefficient when x is equal to

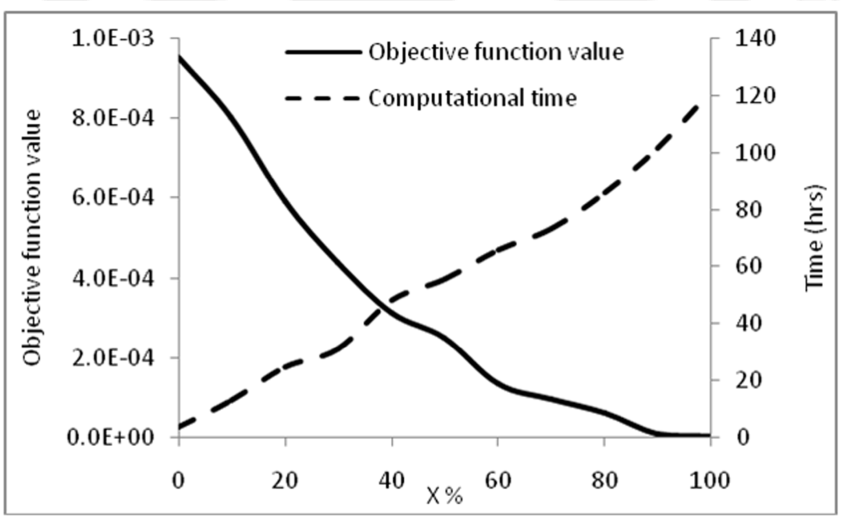


Fig. 6.4: Variation of objective function value and computational time for second study area (ANN-GMS-GA-WOS)

Table 6.10: Objective function values and computational time required for second illustrative study area (with sorting)

x(%)	Objective function value	Comp. Time (DD:HH:MM:SS)	Relative comp. advantage over GMS-GA model (%)
0 (ANN-GA)	9.03E-04	00:04:12:09	96.55
10	2.34E-04	00:12:45:54	89.53
20	7.88E-05	01:00:27:34	79.94
30	1.55E-06	01:06:56:20	74.62
100 (GMS-GA)	1.34E-06	05:01:54:45	00.00

zero. In the second experiment, we have applied the sorting operator for the second study area. Table 6.10 shows the objective function values and computational time requirement by the ANN-GMS-GA model for x value of 10%, 20% and 30% respectively. It can be seen that the model with sorting operator could achieved the results of GMS-GA model when x value is equal to just 30%. This can also be visualized in Fig. 6.5. This shows that significant amount of computational time can be reduced in obtaining the unknown pollution sources of a groundwater aquifer. Around 74.62% of computational efficiency can be achieved by ANN-GMS-GA model over GMS-GA model while maintain the quality of the solution at per GMS-GA model. With this evaluation, it can be concluded that the proposed ANN-GMS-GA model is one of the most potential methods to identify the unknown pollution sources in an aquifer.

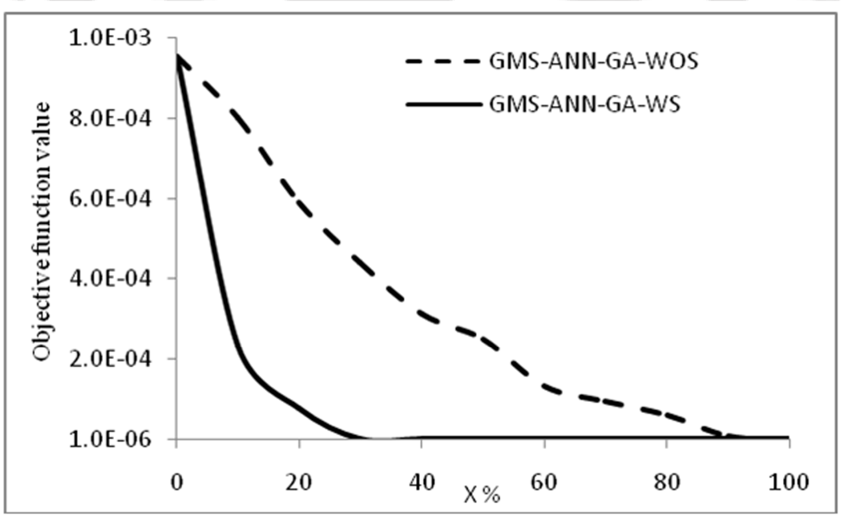


Fig. 6.5: Comparison of objective function value of ANN-GMS-GA model without sorting and with sorting for second study area

6.3 Conclusions

This chapter presents a new simulation-optimization methodology for optimal identification of unknown pollution sources in a groundwater aquifer. We have incorporated both numerical and approximate aquifer simulation models with the GA based optimization model. In order to achieve efficiency both in computational time and predicting performance, we have proposed this methodology using ANN, GMS and GA where fitness of few best individuals of the population have been calculated using GMS and finesses of the other individuals of the population have been calculated using ANN model. The model is as efficient as GMS-GA model and also computationally much more efficient than GMS-GA model. Application of the model on the illustrative study areas shows that ANN-GMS-GA model has better predicting capability and also computationally efficient. The model also applied without sorting of population. However, performance of the model is not better without the sorting operator. The limited evaluations show that up to 73% of computational efficiency can be achieved by the model over GMS-GA model. This shows that the proposed model is efficient and has the potential for field application.

Chapter 7

Summery and Conclusions

7.1 Summery

Identification of pollution sources is an important task for the engineers working in management of groundwater aquifer system. Prior knowledge of pollution sources and its movement is necessary for planning of water resources of a region. The pollution sources can be identified using inverse optimization technique. In this technique, the aquifer simulation model needs to incorporate with the optimization model. Researchers have developed various methods for incorporating simulation model with the optimization model. Out of which, linked simulation-optimization model has emerged as one of the most promising techniques for solving the problem. The advantage of this technique is that sophisticated GIS based aquifer simulation model can be used and the optimization model has fewer decision variables. An optimization algorithm is necessary to solve the optimization model. Various classical and non-classical methods have been used for solving the problem. However, the gradient less non-classical methods, such as genetic algorithms (GA), simulated annealing (SA), evolutionary strategies (ES), direct search method (DS), differential evolution (DE), *etc.* have been appeared as most efficient algorithms for solving the source identification model.

This study presented four improved methodologies for optimal identification of unknown location and magnitude of groundwater pollution sources.We have used linked simulation-optimization algorithm. The simulation model is developed using Groundwater Modeling

System (GMS). GMS has a GIS based interface which is helpful for simulating real world groundwater aquifer system. In the first model, GMS is linked with the optimization model in Matlab environment. The MODFLOW and MT3DMS modules available in GMS are used to simulation flow and transport processes in aquifer. The inverse model is solved using genetic algorithms and direct search method. The performance of the proposed model is evaluated using two illustrative study areas. The performance evaluation reveals that the approach is better than embedded optimization approach. However, this is a very time consuming approach and model takes around few days to solve a problem of medium scaled aquifer around one square kilometer aquifer system on Pentium (R) Dual-Core CPU, E 5700@ 3.00 GHz, 1.96 GB of RAM. In order to reduce the computational time, the second approach uses Artificial Neural Networks model in place of GMS to simulate the flow and transport processes. The model is computationally efficient now, but qualitative point of view, it is less efficient that the use of GMS model. Both computational efficient and quality of prediction can be achieved using hybrid optimization method, i.e. the third approach. In this method, near optimal solution is used using ANN based simulation optimization model. The solution obtained using the ANN based model is then refined using GMS based simulation-optimization model. In the fourth approach, another new algorithm is presented by using ANN and GMS simulation models with the optimization model. This approach is more efficient both computationally and qualitatively than the other approaches.

7.2 General Conclusions

The following conclusion can be made based on the evaluation study conducted in two illustrative study areas.

- ✓ The integration of GMS software with GA based optimization model enables us to solve very complicated real world source identification problem as GMS has the capability to simulate more complex groundwater aquifer system. Further, the GIS based graphical interface is quite helpful in incorporating irregular boundaries of an aquifer and irregular shape of strata along with other spatial and temporal data.
- ✓ The performance evaluation of GMS-GA model on two illustrative study areas shows that the model is capable in identifying the unknown pollution sources and the model performance is slightly better than the embedded optimization approach used by Mahar and Datta (2000).

- ✓ The study also shows that GMS-GA model is computationally very extensive. It takes around five days of computational time to solve a source identification problem of an aquifer having one square kilometer area in a computer system with Pentium (R) Dual-Core CPU, E 5700@ 3.00 GHz, 1.96 GB of RAM..
- ✓ The computational time required to solve the problem is directly related to the computational time required by the aquifer simulation model. Thus the computational time can be reduced using simplified aquifer simulation models.
- ✓ Artificial neural network can be used as an approximate simulator of the flow and transport processes in aquifer. Evaluation of the model with erroneous data shows that model performance does not degrade much with the introduction noise in the training patterns.
- ✓ The use of artificial neural network model as approximate simulator in place of GMS has drastically reduced the computational time of the simulation-optimization model. The problem which took around five days of computational time can now be solved within few hours.
- ✓ The main drawback of the ANN based simulation-optimization model is that the quality of the solution is much inferior as compared to GMS-GA model. It can only obtain the near optimal solution.
- ✓ The efficiency both in predicting capability and in computational time can be achieved by using hybrid optimization method. In this technique, initially ANN-GA model is used to solve the pollution source identification problem. The solution obtained by ANN-GA model is then refined by using GMS-DS approach. The method is computationally more efficient than GMS-GA model.
- ✓ The proposed hybrid optimization model comes out as an attractive alternative solution procedure where efficiency in both computational time and predicting performance can be achieved. But still it takes few days to solve the problem.
- ✓ Some drastic improvement in computational time can be achieved using ANN-GMS-GA model. This methodology is better than the GMS-GA model computationally while maintaining predicting capability at per GMS-GA model.
- ✓ The performance evaluation of the models show that GA based simulation-optimization is superior than DS based simulation-optimization model.

- ✓ Application of the model to the illustrative study areas shows that the model has the potential for field application.

7.3 Future scope of research

It is felt that detailed studies are required to address the following issues.

- ✓ The developed methodology can be extended to the source identification problem of unconfined aquifer or aquifer with fracture.
- ✓ This study has used genetic algorithms and direct search method to solve the inverse optimization problem. However, performance of other non-classical algorithms, such as simulated annealing (SA), differential evolution (DE), evolutionary strategies (ES), particle swarm algorithm (PSA), *etc.* can also be evaluated.
- ✓ ANN model is used as an approximate simulator of the flow and transport processes in groundwater aquifer. Performance of other approximate simulation techniques, such as genetic programming, linear and non-linear regression models, *etc.* can also be evaluated.
- ✓ One of the possible extensions of this study is the identification of multiple reactive and radioactive pollutant sources in real world groundwater aquifer.
- ✓ Finally, this methodology used static monitoring network available in the aquifer. The methodology can be extended incorporating optimal monitoring network design algorithm.

References

1. Aguado, E., and Remson, I. (1980), 'Groundwater management with fixed charges.' *J. Water Resour. Plag. Mngt Div., ASCE, 100(WR2), 375-382.*
2. Aguado, E., and Remson, I. (1974), 'Groundwater hydraulics in aquifer management.' *J. Hydraulic Division, ASCE, 100(HY1), 103-118.*
3. Aguado, E., Remson, I., Pikul, M.F., and Thomas, W.A. (1974), 'Optimal pumping for aquifer dewatering.' *J. Hydraulic Division, ASCE, 100(HY7), 860-877.*
4. Aguado, E., Sitar, N. and Remson, I. (1977), ' Sensitivity analysis in aquifer studies.' *Water Resour. Res., 13(4), 733-737.*
5. Ahlfeld, D.P., and Baro- Montes, G. (2008), 'Solving Unconfined Groundwater Flow Management Problems with Successive Linear Programming.' *J. Water Resour. Plng. Mgmt. 134 (5), 404 -412.*
6. Ahlfeld, D.P.and Heidari, M. (1994), 'Applications of optimal hydraulic control to groundwater systems.' *J. Water Resour. Plng. Mgmt., ASCE 120, 350–365.*
7. Ahlfeld, D.P. (1990), 'Two-stage groundwater remediation design.*' J. Water Resour. Plng. Mgmt., ASCE, 116(4), 517-529.*
8. Ahlfeld, D.P. and Sawyer, C.S. (1990), ' Two-stage groundwater remediation design.' *J. Water Resour. Plng. Mgmt., ASCE, 116(4), 517-529.*
9. Ahlfeld, D.P., Mulvey, J.M., and Pinder, G.F. (1988b), 'Contaminated groundwater remediation design using simulation, optimization, and sensitivity theory 2. Analysis of a field site.' *Water Resour.Res., 24(3), 443-452.*
10. Ahlfeld, D.P., Mulvey, J.M., and Pinder, G.F. (1988a), 'Contaminated groundwater remediation design using simulation, optimization, and sensitivity theory 1. Analysis of a field site.' *Water Resour. Res., 24(3), 431-441.*
11. Ahlfeld, D. P., and Mulvey, J.M. (1986), 'Designing optimal strategies for contaminated groundwater remediation.' *Adv. in Water Resources, 9 (2), 77-84.*
12. Aly, A.H., and Peralta, R.C. (1999a), 'Comparison of a Genetic Algorithm and Mathematical Programming to the Groundwater Cleanup Systems.' *Water Resour. Res., 35(8), 2415-2425.*
13. Aly, A.H., and Peralta, R.C. (1999b), 'Optimal Design of Aquifer Cleanup Systems Under Uncertainty Using a Neural Network and Genetic Algorithms.' *Water Resour. Res., 35(8), 2523-2532.*

14. Alley, W.M., Aguado, E., and Remson, I. (1976). 'Aquifer management under transient and steady state conditions.' *Water Resour. bull., 12(5), 963-972.*

15. Alley, W.M. (1986), 'Regression Approximations for Transport Model Constraint Sets in Combined Aquifer Simulation-Optimization Studies.' *Water Resour. Res., 22(4), 581-586.*

16. Amirabdollahian, M. and Datta, B. (2013), 'Identification of Contaminant Source Characteristics and Monitoring Network Design in Groundwater Aquifers: An Overview.' *J. Environ. Pro., 4, 26-41.*

17. Aral, M.M, Guan, J., and Maslia, M.L. (2001), 'Identification of Contaminant Source Location and Release History in Aquifers.' *J. Hydrol. Engg., ASCE, 6(3), 225-234.*

18. Aral, M.M. and Guan, J. (1996), 'Genetic algorithms in search of groundwater pollution sources, in Advances in groundwater pollution control and remediation.' *Kluwer Academic Publishers, Netherlands, 347-369.*

19. Atmadja, J., and Bagtzoglou, A. C. (2001), ' Pollution source identification in heterogeneous porous media.' *Water Resour. Res. 37,2113–2125.*

20. Ayaz, Md., Srivastava, R. and Jain, A. (2014), 'Groundwater Pollution Source Identification using Linked ANN-Optimization Model.' *Geophysical Research, Vol. 16, EGU2014-830.*

21. Ayvaz, M.T. (2010), 'A linked simulation-optimization model for solving the unknown groundwater pollution source identification problems.' *J. Contamt. Hydro. 117, 46-59.*

22. Azaiez, M.N., Hariga, M. (2001), 'A single-period model for conjunctive use of ground and surface water under severe overdrafts and water deficit.' *Eur. J. Oper. Res.1333, 653–666.*

23. Azghadi, S. and Kerachian, R. (2010), 'Locating monitoring wells in groundwater systems using embedded optimization and simulation models.' *Science of The Total Environ., 408(10), 2189-2198.*

24. Aziz, A.R.A, and Wong, K.F.V. (1992), 'Neural network approach to the determination of aquifer parameters.' *Groundwater, 30(2), 164-166.*

25. Bagtzoglou A.C., Dougherty D. E. and Tompson A. F. B. (1992) 'Application of particle methods to reliable identification of groundwater pollution sources.' *Water Resour. Mgmt. 6(1), 15-23.*

26. Becker, L., and Yeh, W. W. G. (1972), Identification of parameters in unsteady open channel flows.' *Water Resour. Res., 8(4), 956-965.*

27. Beale, R., and Jackson, T. (1991), 'Neural computing: An introduction, Adam Hilger, Techno House, Bristol.'
28. Bhattacharjya, R. K., Datta, B., and Satish, M. G. (2009), 'Performance of an Artificial Neural Network Model for Simulating Saltwater Intrusion Process in Coastal Aquifer when Training with Noisy Data.' *KSCE J. Civ. Engg., 13(3), 205-215.*
29. Bhattacharja, R. and Datta, B. (2009), 'ANN-GA-based model for multiple objective management of coastal aquifers.' *J. Water Resour. Plng. and Mgmt., 135 (5),314-322.*
30. Bhattacharjya, R.K., Datta, B., and Satish, M., G. (2007), 'Artificial neural networks approximation of density dependent saltwater intrusion processes in coastal aquifers.' *J. Hydrol. Engg., 12(3), 273-282.*
31. Bhattacharjya, R.K., Datta, B., and Satish, M., G. (2005), 'Optimal management of coastal aquifer using linked simulation optimization approach.' *Water Resour. Mgmt., 19(3), 295-320.*
32. By the ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. *J. Hydrol. Engg., 5 (2), 124-137.*
33. Chadalavada, S., and Datta, B. (2008), 'Dynamic Optimal Monitoring Network Design for Transient Transport of Pollutants in Groundwater Aquifers.' *Water Resour. Management, 22(6), 651-670.*
34. Chadalavada, S., Datta, B. and Naidu, R. (2011), 'Uncertainty based optimal monitoring network design for a chlorinated hydrocarbon contaminated site.' *Environ. Monit. Assess., 173, 929-940.*
35. Cieniawski, S.E., Eheart, W., and Ranjithan, S. (1995), 'Using Genetic Algorithms to Solve Multiobjective Groundwater Monitoring Problem', *Water Resour. Res., 31(2), 399-409.*
36. Colarullo, S.J., Heidari, M., and Maddock, T. III (1984), 'Identification of an optimal groundwater strategy in contaminated aquifer.' *Water Resour. Bull., 20(5), 747-760.*
37. Coope, D.I. and Price, C.J. (2000), 'A direct search conjugate directions algorithm for unconstrained minimization.' ANZIAM J., 42, C478–498.
38. Datta, B., Prakash, O. and Campbell, S. (2013), 'Efficient Identification of Unknown Groundwater Pollution Sources Using Linked Simulation-Optimization Incorporating Monitoring Location Impact Factor and Frequency Factor.' *Water Resour.Mgmt. 27,4959–4976, DOI 10.1007/s11269-013-0451-8*

39. Datta, Bithin, Chakrabarty, D., & Dhar, A. (2011), 'Identification of unknown groundwater pollution sources using classical optimization with linked simulation.' *J. Hydro-Environ. Res., 5(1), 25-36*

40. Datta, B., Chakrabarti, D., and Dhar, A. (2009), 'Optimal Dynamic Monitoring Network Design and Identification of Unknown Groundwater Pollution Sources.' *Water Resour. Mgmt.,23(10), 2031-2049.*

41. Datta B., Chakrabarty D. (2003), 'Optimal identification of unknown pollution sources using linked optimization simulation methodology.' *Proc. Of Symposium on Advances in Geotechnical Engineering (SAGE 2003). I.I.T. Kanpur, India, pp. 368-379.*

42. Datta, B., and Dhiman, S.D. (1996), 'Chance-Constrained Optimal Monitoring Network Design for Pollutants in Ground Water.' *J. Water Resour. Plng. and Mgmt., ASCE, 22(3), 180-188.*

43. Datta, B., Beegle, J.E., Kavvas, M.L., and Orlob, G.T. (1989), 'Development of an expert system embedding pattern recognition techniques for pollution source identification.' *Completion Report for U.S.G.S. grant no. 14-08-0001-G1500, Dept. Of Civ. Engrg., Univ. Of California, Davis, Calf., U.S.A.*

44. Datta, B., and Peralta, R.C. (1986), 'Interactive computer graphics-based multi-objective decision-making for regional groundwater management.' *Agric. Water Manage, 11 91-116.*

45. Das, A., and Datta, B. (2000), 'Optimization Based Solution of Density Dependent Seawater Intrusion in Coastal Aquifers.' *J. Hydrol. Engg., ASCE, 5(1), 82-89.*

46. Deb, K. (2000), 'An efficient constraint handling method for genetic algorithms.' *J. Comput Methods Appl. Mech. Engrg, 186(2000), 311-338.*

47. Deb, K. (2001), 'Multi-Objective Optimization Using Evolutionary Algorithms', John Wiley & Sons, Ltd, UK.

48. Deb, K., Agarwal, S., Pratap, A., and Meyarivam, T. (2000), 'A fast Elitist Non-dominating sorting Genetic Algorithm for multi-objective optimization: NSGA II.' *In the proceedings of the parallel problem solving from nature VI conference, 849-858.*

49. Deninger, R.A. (1970), 'System analysis of water supply systems.' *Water Resour. Bull., 6(4), 573-579.*

50. Dhar, A.,and Datta, B. (2009), 'Logic based design of groundwater monitoring network for redundancy reduction.' *J. Water Resour. Plng. and Mgmt. (ASCE), Vol 136 (1), 88-94.*

51. Dhar, A., and Datta, B. (2009), 'Saltwater intrusion management of coastal aquifers-I: linked simulation-optimization.' *J. Hydrol. Engg., ASCE, 14 (12), 1263-1272.*

52. Dhar, A. and B. Datta, B. (2007), 'Multiobjective Design of Dynamic Monitoring Networks for Detection of Groundwater Pollution.' *Water Resour. Plng. and Mgmt., 133(4), 329-338.*

53. DibiKe, Y.B., and Solomatine D.P.(1999), "River Flow Forecasting using Artificial Neural Networks", ESG Journal of Physics and Chemistry of the Earth19-23 April..

54. Ejaz, M.S., and Peralta, R.C.: 1985, 'Modeling for optimal management of agricultural and domestic wastewater loading to streams.' *Water Resour. Res., 31(4), 1087-1096.*

55. Elongo, K., and Rouve, G. (1980), 'Finite-element linear programming model.' *J. Hydraulic Div., ASCE, 106 (HY10), 1641-1658.*

56. El Magnouni, S., and Treichel, W. (1994), 'A multicriterion approach to groundwater management.' *Water Resour. Res., 30(6), 1881-1895.*

57. Emch, P.G., and Yeh, W. W-G.: 1998, 'Management model for conjunctive use of coastal surface water and ground water.' *J. Water Resour. Plng. and Mgmt. ASCE, 124(3), 129-139.*

58. Emsellem, Y. and de Marsily,G. (1971), 'An automatic solution for the inverse problem.' *Water Resour. Res., 7(5), 1264-1283.*

59. Fausett, L. (1994), 'Fundamentals of neural networks.' Prentice Hall, Englewood Cliffs, N.J.

60. Finney, B.A., Samsuhadi, and Willis, R. (1992), 'Quasi-Three dimensional model of Jakarta basin.' *J. Water Resour. Plng. Mgmt., ASCE, 118(1), 18-31.*

61. Fonseca, C.M., and Flemming, P.J. (1993), 'Genetic Algorithms for multi-objective optimization: Formulation, discussion, and generalization.' In the proceeding of the fifth international conference on Genetic Algorithms, 419-423.

62. Frind, E.O. and Pinder G. F. (1973), 'Galerkin solution of the inverse problem for aquifer transmissivity.' *Water Resour. Res., 9(5), 1397-1410.*

63. Futagami, T., Tamai, N., and Yatsuzuka, M. (1976), 'FEM coupled with LP for water pollution control.' *J. Hydraul. Div., ASCE, 102 (HY7), 881-897.*

64. Galeati, G., and Gambolati, G. (1988), 'Optimal dewatering schemes in the foundation design of an electronuclear plant.' *Water Resour. Res., 24(2), 541-552.*

65. Gailey, R.M. and Gorelick, S.M. (1993), 'Design of optimal, Reliable Plume Capture Schemes: Application to the Gloucester Landfill Groundwater contamination problem.' *Groundwater, 31(1), 107-114.*

66. Gharbi, A. And Peralta, R. C. (1994), 'Integrated embedding optimization applied to salt lake valley aquifer.' Water *Resour. Res., 30(3), 817-832.*

67. Gill, P.E., Murray, W., Saunders, M.A., and Wright, M.H. (1986), 'User's guide for LSSOL, Report SOL 86-1.' Dept. Of Operations Research, Stanford University, Standford, CA.

68. Goldberg, D.E. (1989), 'Genetic algorithms in search, optimization and machine learning, Addison-wesley, Readind, Mass.'

69. Goldberg, D.E. (1989), 'Genetic Algorithms in Search, Optimization, and in Machine Learning', Addison Wiley, Bangalore, India.

70. Gorelick, S.M., Voss, C.I., Gill, P.E., Murray, W., Saunders, M.A., and Wright, M.H. (1984), 'Aquifer reclamation design: The use of contaminant transport simulation combined with nonlinear programming.' *Water Resour. Res., 20(4), 415-427.*

71. Gorelick, S.M.: 1983, 'A review of distributed parameter groundwater management modeling methods.' *Water Resour. Res., 19(2), 305-319.*

72. Gorelick , S.M, Evans, B., and Remson, I. (1983), 'Identifying sources of groundwater pollution: An optimization approach', *Water Resour. Res., 19(3), 779-790.*

73. Gorelick, S.M. (1982), 'A model for managing sources of groundwater pollution.' *Water Resour. Res.,18(4), 773-781.*

74. Gorelick, S.M., and Remson, I. (1982a), 'Optimal location and management of waste disposal facilities affecting groundwater quality.' *Water Resour. Bull., 18(1), 43-51.*

75. Gorelick, S.M., and Remson, I. (1982b), 'Optimal dynamic management of groundwater pollutant sources.' *Water Resour. Res., 18(1), 71-76.*

76. Gorelick, S.M., Remson, I., Cottle, R.W. (1979), 'Management model of a groundwater system with a transient pollutant source.' *Water Resour. Res. 5, 1243–1249.*

77. Guan, J.,Elcin, K., Aral,M.(2008), 'Genetic Algorithm for Constrained Optimization Models and its Application in Groundwater Resources Management.' *J. Water Resour. Plng. and Mgmt. 134(1), 64-72.*

78. Guan, J. and Aral, M. M. (1999), 'Optimal Remediation with Well Locations and Pumping Rates Selected as Continuous Decision Variables.' *J. Hydrol., Vol. 221, 20-42.*

79. Guo, S., Yang, R., Zhang, H., Weng, W., and Fan, W. (2009), 'Source identification for unsteady atmospheric disperision of hazardous materials using MCMC method.' *Int J. Heat Mass Transf 52, 3955-3962.*

80. Hassoun, M.H. (1999), 'Fundamentals of Artificial Neural Networks,' Prentice Hall of India Pvt. Ltd., New Delhi, pp. 226-230.

81. Hazart, A., Giovannelli, J.F., Dubost, S., and Chatellier, L.(2007), 'Contaminant source estimation in a two-layers porous environmental using a Bayesian approach.' *In: IEEE international geosciences and remote sensing symposium.*

82. Haykin, S., 'Neural Networks, A Comprehensive Foundation.' Second Edition, Pearson.

83. Hefez, E.,Shimar V., Bear J. (1975), 'Identifying the parameters of an aquifer cell model.' *Water Resour. Res., 11(6), 993-1004.*

84. Heidari, M. (1982), 'Application of linear systems theory and linear programming to groundwater management in Kansas.' *Water Resour. Bull., 18(4), 1003-1012.*

85. Hilton, A. and Culver, T. (2005), 'Groundwater remediations design under uncertainty using genetic algorithms.' *J. Water Resour. Plng. and Mgmt., ASCE,131(1),25-34.*

86. Holland, J.H.: 1975. 'Adaptation in Natural and Artificial Systems.' University of Michigan press, Michigan.

87. Horn, J. Nafploits, N., and Goldberg, D.E. (1994), 'A niched Pareto Genetic algorithm for multi-objective optimization.' In the proceedings of the first IEEE conference on evolutionary computation, Z. Michalewicz ed. Piscataway, New Jersey, IEEE service centre, 82-87.

88. Hsiao, C-T., and Chang, L-C. (2002), 'Dynamic optimal groundwater management with inclusion of fixed costs.' *J. Water Resour. Plng. and Mgmt., ASCE, 128(1), 57-65.*

89. Huang, C., Hu, B., Li, X., and Ye, M. (2011), 'Using data assimilation method to calibrate a heterogeneous conductivity field and improve solute transport prediction with an unknown contamination source.' *Stoch. Env. Res. Risk Assess., 23(8), 1155-1167.*

90. Huyakorn, P.S. and Pinder, G.F. (1983), 'Computational methods in subsurface flow.' Academic Press, San Diego, Calf.
91. James A. Freeman/ David M. Skapura, 'Neural Networks Algorithms, Applications, and Programming Techniques.' (Publisher: Addison Wesley, 1 August 1991).
92. Jain, A. and Kumar, A. (2006), 'An evaluation of artificial neural network technique for the determination of infiltration model parameters.' *J. Soft Computing. 6(3),272-282.*
93. Jain, S.K, Singh, V.P., Asce, F., Genuchten, M.T.V.(2004), 'Analysis of Soil Water Retention Data Using Artificial Neural Networks.' *J. Hydrol. Engg., 9 (5), 415-420.*
94. Jain, A., Varshney, A.K., Joshi, U.C.(2001), 'Short-term water demand forecasting modelling at IIT Kanpur using artificial neural network.' *Water Resour. Mgmt. 15 (5), 299-321.*
95. Jin, X., Mahinthakumar, G., Zechman, E., and Ranjithan, R.S. (2009), 'A genetic algorithms-based procedure for 3D source identification at Borden emplacement site.' *J Hydroinfo. 11 (1), 51-64.*
96. Johnson, V.M., and Rogers, L.L (2000), 'Accuracy of Neural Network Approximations in Simulation-Optimization.' *J. of Water Resour. Plng. and Mgmt., ASCE, 126(2), 48-56.*
97. Karatzas, G.P., and Pinder, G.F. (1993), 'Groundwater management using numerical simulation and the outer approximation method for global optimization.' *Water Resour. Res., 29(10), 3371-3378.*
98. Karterakis, S.M., Karatzas, G.P., Nikolos, I.K., Papadopoulou, M.P. (2007), 'Application of linear programming and differential evolutionary optimization methodologies for the solution of coastal subsurface water management problems subject to environmental criteria.' *J. Hydrol. 342, 270–282.*
99. Kisi,O.,Asce,M.(2004), 'River Flow Modeling Using Artificial Neural Networks", *J.Hydrol. Engg.' 9 (1), 60-63.*
100. Kleinecke, D. (1971), 'Use of linear programming for estimating geohydrologic parameters of groundwater basins.' *Water Resour. Res., 7(2), 367-374.*
101. Kollat, J. B., Reed, P. M., & Maxwell, R. (2011), 'Many-objective groundwater monitoring network design using bias-aware ensemble kalman filtering, evolutionary optimization, and visual analytics.' *Water Resour. Res., 47, W02529.*

102. Konikow, L.F. and Bredehoeft, J.D. (1978), 'Computer model of two-dimensional solute transport and dispersion in groundwater.' U.S. Geol. Surv. Tech.' Water Resour. Invest. Book 7, U.S.G.S., Reston, Va.

103. Kourakos, G., and Mantoglou, A. (2009), ' Pumping optimization of coastal aquifers based on evolutionary algorithms and surrogate modular neural network models.' *Adv. Water Resour. 324, 507–521.*

104. Lee, A.S., Aronafsky, J.S. (1958), 'A linear programming model for scheduling crude oil production.' *J. Petrol. Technol. 213, 51–54.*

105. Lefkoff, L.J., and Gorelick, S.M. (1990), 'Simulating physical processes and economic behavior in saline, irrigated agriculture: Model development.' *Water Resour. Res.., 26(7), 1359-1369.*

106. Lefkoff, L.J., and Gorelick, S.M. (1986), 'Design and cost analysis of rapid aquifer restoration systems using flow simulation and quadratic programming.' *Groundwater., 24 (6), 777-790.*

107. Liu, X., Cardiff, M.A., and Kitanidis, P.K. (2010), 'Parameter estimation in nonlinear environmental problems.' *Stoch Environ Res Risk Assess 24, 1003-1022.*

108. Liu, C. and Ball, W. P. (1999), 'Application of inverse methods to contaminant source identification from aquitard diffusion profiles at Dover AFB, Delaware.' *Water Resour. Res. 35,1975–1985.*

109. Maddock III, T. (1972), 'Algebraic technological function from a simulation model.' *Water Resour. Res., 8(1), 129-134.*

110. Mahar, P.S., and Datta, B. (2001), 'Optimal Identification of Ground-Water Pollution Sources and Parameter Estimation.' *J. Water Resour. Plng. And Mgmt., ASCE, 127(1), 20-29.*

111. Mahar, P.S., and Datta, B. (2000), 'Identification of Pollution Sources in Transient Groundwater Systems.' *Water Resources Management, 14(3), 209-227.*

112. Mahar, P.S., and Datta, B. (1997), 'Optimal Monitoring Network and Ground-Water-Pollution Source Identification.' *J. Water Resour. Plng. and Mgmt., ASCE, 123(4), 199-207.*

113. Mahon, G.L., Terry, J.E., and Peralta, R.C. (1989), 'Water resources development alternatives for the Mississipi alluvial plain in eastern Arkansan.' In symposium proceedings water laws and management, sept., 1987 11B-29-11B-43.

114. Mahinthakumar, G. and Sayeed, M. (2005), 'Hybrid genetic algorithms-local search methods for solving groundwater source identification inverse problem.' *J. water Res. Plng. and Mgmt., 131(1), 45-57.*

115. Mantoglou, A., Papantoniou, M., Giannoulopoulos, P. (2004), 'Management of coastal aquifers based on nonlinear optimization and evolutionary algorithms.' *J.Hydrol. 297, 209–228.*

116. Mantoglou, A. (2003), 'Pumping management of coastal aquifers using analytical models of saltwater intrusion.' *Water Resour. Res. 39(12), 1–12.*

117. Marino, M.A., and Yeh, W. W. G. (1973), 'Identification of parameters in finite leaky aquifer system.' *J. Hydraul. Div. 99(2), 319-336.*

118. Mayer, D. G., Belward, J. A., Widell, H. and Burrage, K. (1999), 'Survival of the fittest – genetic algorithms versus evolution strategies in the optimization of systems models.' *Agricultural Systems, 60, 113-122.*

119. McKinney, D.C., and Lin, M-D. (1994), 'Genetic Algorithms Solution of Groundwater Management Models,' *Water Resour. Res., 30(4), 1897-1906.*

120. McPhee, J., Yeh, W. W-G. (2008), 'Groundwater Management using Model Reduction via Empirical Orthogonal Functions.' *J. Water Resour. Plng.and Mgmt., ASCE, 134(2), 161-170.*

121. Meyer, P.D and Brill, E. D. Jr. (1988), 'A method for locating wells in a groundwater pollution monitoring network under conditions of uncertainity.' *Water Resour Res. 24(8), 1277-1282.*

122. Michalak A. M. And Kitanidis P.K. (2004), 'Estimation of historical groundwater contaminant distribution using the adjoint statemethod applied to geostatistical inverse modelling.' *Water Resour Res., 40:W08302.doi:10.1029/2004WR003214.*

123. Michalak, A.M., and Kitanidis, P.K. (2003), 'A Method for Enforcing Parameter Nonnegativity in Bayesian Inverse Problems with an Application to Contaminant Source Identification.' *Water Resour. Res., 39(2), 1033, doi:10.1029/2002WR001480.*

124. Michalak, A.M., and Kitanidis, P.K. (2002), 'Application of Bayesian Inference Methods to Inverse Modeling for Contaminant Source Identification at Gloucester Landfill, Canada.' *In Computational Methods in Water Resources XIV, 2, 1259-1266, edited by S.M. Hassanizadeh, R.J. Schotting, W.G. Gray and G.F. Pinder, Elsevier, Amsterdam, The Netherlands.*

125. Milnes. E. and Perroochchet, P. (2007), ‘ Simultaneous identification of a single pollution point-source location and contamination time under known flow-field conditions.’ *Advances in Water Resources, 30, 2439-2446.*

126. Molx, F.J., and Bell, L.C. (1977), ‘Head gradient control in aquifers used for fluid storage.’ *Water Resour. Res., 13(4), 795-798.*

127. Momtahen, Sh. And Dariane, A.B (2007), “Direct Search Approaches Using Genetic Algorithms for Optimization of Water Reservoir Operating Politicies. ” *J. Water Resour. Plng.and Mgmt., ASCE, 13(3), 202-209.*

128. Montoglou, A. (2003), ‘Pumping management of coastal aquifer using analytical models of saltwater intrusion.’ *Water Resour. Res. 39(12),1335. doi:10.1029/2002WR001891.*

129. Montas, H.J., Mohtar, R.H., Hassan, A.E. and AlKhal, F.A. (2000), ‘Heuristic Space-Time Design of Monitoring Wells for Contaminant Plume Characterization in Stochastic Flow Fields.’ *J. Cont. Hydrol., 43(3-4), 271-301.*

130. Morshed, J., and Kaluarachchi, J.J. (1998), ‘Application of artificial neural networks and genetic algorithms in flow and transport simulation.*’ Adv. in Water Resour., 22(2), 145-158*

131. Morshed, J., and Kaluarachchi, J. (2000), ‘Enhancements to Genetic Algorithm for Optimal Ground-Water Management.*’ J. Hydrol. Eng., ASCE, 5(1), 67-73.*

132. Mugunthan, P. and Shoemaker, C.A. (2004), ‘Time Varying Optimization for Monitoring Multiple Contaminants under Uncertain Hydrogeology,’ *Bioremediation J., 8(3-4), 129-146.*

133. Murtagh, B.A., and Saunders, M.A. (1993), ‘MINOS 5.4 user’s guide, technical report, SOL 83-20R’, Syspems Optimization Laboratory, Department of Operation Research, Stanford University, Stanford, California, 94305-4022.

134. Murty V. V. N., and Scott, V. H. (1977), ‘Determination of transport model parameters in groundwater aquifers.’ *Water Resour. Res., 13(6), 941-947.*

135. Navarro, A. (1977),‘A modified optimization method of estimating aquifer parameters. ’ *Water Resour. Res., 13(6), 935-939.*

136. Neuman, S. P., and Yakowitz S. (1979), ‘A statistical approach to the inverse problem of aquifer hydrology: 1. Theory’, *Water Resour. Res., 15(4), 845-860.*

137. Neuman, S.P. (1973), ‘Calibration of distributed parameter flow models viewed as a multiple-objective decision process under uncertainty.’ *Water Resour. Res. ,9(4), 1006-1021.*

138. Neupauer, R.M., Borchers, B., and Wilson, J.L. (2000), 'Comparison of inverse methods for reconstructing the release history of groundwater contaminant source.' *Water Resour. Res., 36(9), 2469-2475.*

139. Nicklow, J., Reed, P., Savic, D., Dessalegne, T., Harrell, L., Chan-Hilton, A., Karamouz,M., Minsker, B., Ostfeld, A., Singh, A., Zechman, E. (2010), 'State of the art for genetic algorithm and beyond in water resources planning and management.' *J. Water Resour. Plng. and Mgmt. 1364, 412–432.*

140. Nishikawa, T. (1998), 'Water resources optimization model for Santa Barbara, California.' *J. Water Resour. Plng. and Mgmt., ASCE 1245, 252–263.*

141. Nunes, L.M., Paralta, E., Cunha, M.C. and Ribeiro, L. (2004) 'Groundwater Nitrate Monitoring Network Optimization with Missing Data,' *Water Resour. Res., 40(2), 1-18.*

142. Nutbrown, D.A. (1971), 'Normal mode analysis of the linear equation of groundwater flow.' *Water Resour. Res., 11(6),979-987.*

143. Peralta, R.C., Azarmnia, H., and Takahashi, S. (1991), 'Embedding and Response matrix techniques for maximizing steady-state groundwater extraction computational comparison.' *Ground Water 25(3) 357-364.*

144. Peralta, R. C. and Datta, B. (1990), 'Reconnaissance - level alternative optimal groundwater use strategies.' *J. Water Resour. Plng. and Mgmt. , 116 (5), 676-692.*

145. Peralta, R.C., and Kowalski, K. (1986), 'Optimizing the rapid evolution of target groundwater potentiometric surface attainment.' Transactions of the ASAE, 29, 940-947.

146. Prakash, O. and Datta, B. 'Sequential Optimal Monitoring Network Design and Iterative Spatial Estimation of Pollutant Concentration for Identification of Unknown Groundwater Pollution Source Locations,' *Environ. Monit. Assess., 2012, 1-16.*

147. Ranjithan, S., Eheart, J.W., and Garrett, J.H.Jr. (1993), 'Neural Network Based Screening for Groundwater Reclamation Under Uncertainty.' *Water Resour. Res., 29(3), 563-574.*

148. Rao, S.V.N., Murthy, S.B., Thandaveswara, B.S., Mishra, G.C. (2004), 'Conjunctive use of surface and groundwater for Coastal and Deltic systems.' *Water Resour. Plng and. Mgmt., ASCE 1303, 255–267.*

149. Reed, P., Minesker B., and Goldberg, D.E., (2002), 'Designing a competent genetic algorithm, for search and optimization.' *Water Resour.Res. 36(12), 3757-3761.*

150. Remson, I., and Gorelick, G.M. (1980), 'Management models incorporating groundwater variables, in Operations Research in Agriculture and Water Resourcese.' edited by D. Yaron and C. S. Tapiero, North Holland Amstedam.

151. Remson, I., Hornberger, G.M., and Molz, F. J. (1971), 'Numerical methods in subsurface Hydrology.' Wiley-Interscience, New York.

152. Reilly, T.E., Franke, O.L., and Bennett, G.D. (1987), 'The principle of superposition and its application in groundwater hydraulics.' U.S. Govt. Printing Office, Washington, D.C., Techniques of water Res. Investigations of the U.S.G.S., Book 3, Chapter B6.

153. Ritzel, B.J., Eheart, J.W, and Ranjithan, S. (1994), 'Using Genetic Algorithms to Solve a Multiple Objective Groundwater Pollution Containment Problem.' *Water Resour. Res., 30(5), 1589-1603.*

154. Rizzo, D.M., and Dougherty, D.E. (1994), 'Characteristic of aquifer properties using artificial neural networks: neural kriging.' *Water Resour. Res., 30(2), 483-497.*

155. Rogers, L.L, and Dowla, F.U. (1994), 'Optimization of Groundwater Remediation Using Artificial Neural Networks With Parallel Solute Transport Modeling', *Water Resour. Res., 30(2), 457-481.*

156. Rosenwald, G.W., and Green, D.W. (1974), 'A method for determining the optimal location of wells in a reservoir using mixed-integer programming', *Soc. Res. Eng. J., 14, 44-54.*

157. Sawer, C.S., and Lin,Y.F. (1998), 'Mixed-integer chance-constrained models for groundwater remediation.' *J. Water Resour. Plng. And Mgmt., ASCE, 124(5), 285-294.*

158. Segol, G., Pinder, G.F., and Gray, W.G. (1975), 'A Galerkin finite element technique for calculating the transport position of the saltwater front.' *Water Resour. Res., 11(2), 343-347.*

159. Sethi, L.N., Kumar, D.N., Panda, S.N., Mal, B.C. (2002), 'Optimal crop planning and conjunctive use of water resources in a coastal river basin.' *Water Resour. Mgmt. 16, 145–169.*

160. Sheta, A.F., Mahmoud, A. (2001), 'Forecasting using genetic programming.' In: Proceeding of the 33rd Southeastern Symposium on System Theory, 343-347.

161. Sidaruk, P., Cheng, A.H.D. and Quazar, D. (1998), 'Groundwater contaminant source and transport parameter identification by correlations coefficient optimization.' *Groundwater, 36(2), 208-214.*

162. Singh, R.M, and Datta, B (2007), 'Artificial neural network modeling for identification of unknown pollution sources in groundwater with partially missing concentration observation data.' *Water Resour. Mgmt., 21(3), 557-572.*

163. Singh, R.M, and Datta, B (2006), 'Identification of Groundwater Pollution Sources (a)Using GA-based Linked Simulation Optimization Model.' *J. Hydrol. Engg, 11(2),101-109.*

164. Singh, R. M. and Datta, B.(2004), "Groundwater pollution source Identification and Simultaneous Parameter Estimation Using Pattern Matching By Artificial Neural Network of unknown groundwater pollution sources using artificial neural network.' *J. Environ. Foren., 5:143-153.*

165. Singh, R. M., Datta, B., Jain, A. (2004), 'Identification of unknown groundwater pollution sources using artificial neural network.' *J. of Water Resour. Plng. and Mgmt., 130 (6), 506-514.*

166. Singh, R. M., Datta, B., and Jain, A. (2002), 'Identification of unknown groundwater pollution sources using artificial neural network. Proceedings of the International Conference on Advances in Civil Engineering (ACE-2002), Kharagpur, India: Indian Institute of Technology, pp. 83–93.

167. Skaggs, T.H. and Kabala, Z.H. (1995), 'Recovering the release history of a groundwater contaminant plume: Method of quasi-reversibility.' *Water Resour. Res., 31(11), 2669-2673.*

168. Skaggs, T. H. and Kabala, Z. H. (1994), 'Recovering the release history of a groundwater contaminant.' *Water Resour. Res., 30(1), 71-79.*

169. Snodgrass, M.F., and Kitanidis, P.K. (1997), 'A geostatical approach to contaminant source identification.' *Water Resour. Res., 33(4), 537-546.*

170. Sreekanth, J., and Datta, B. (2011), 'Comparative Evaluation of Genetic Programming and Neural Network as Potential Surrogate Models for Coastal Aquifer Management.' *Water Resour. Mgmt., 25(13), 3201-3218.*

171. Srivastava, D. and Singh, R.M. (2014), 'Breakthrough Curves Characterization and Identification of an Unknown Pollution Source in Groundwater System Using an Artificial Neural Network (ANN),' *Environ. Foren., 15:175–189.*

172. Srinivas, N., and Deb, K. (1994), 'Multi-objective function optimization using non-dominating sorting Genetic Algorithms.' *Evol. Comp., 2(3), 221-248.*

173. Srinivas, N., and Deb, K. (1994), 'A robust maximum likelihood approach to contaminant source identification.' *Water Resour.Res., 43(2),123-137.*

174. Sun, A.Y., Painter, S.L, Wittmeyer, G.W. (2006a), 'A constrained robust least squares approach for contaminant source release history identification.', *Water Resour. Res., 42(4): W04414. Doi: 10.1029/2005WR004312.*

175. Sun, A.Y., Painter, S.L, Wittmeyer, G.W. (2006b), 'A robust approach for contaminant source location and release history recovery.' *J. Contam. Hydrol., 88(3-4): 29-44. Doi: 10.1016/j.jconhyd. 2006.06.006.*

176. Tracy, N. (1998), 'Water resources optimization model for Santa Barbara, California.' *J. Water Resour. Plng. and Mgmt., ASCE 124, 252–263.*

177. Todd, D.K. (1980), 'Groundwater hydrology', John Wiley, New York, 1980.

178. Tokgoz, M., Yilmax, K.K., and Yazicigil, H. (2002), 'Optimal aquifer dewatering schemes for excavation of collector line.' *J. Water Resour. Plng. and Mgmt., ASCE, 128(4), 248-261.*

179. Tsai, F., Sun, N., Yeh, W. (2003), 'Global-local optimization for parameter structure identification in three-dimensional groundwater modelling.' *Water Resour. Res., 39(2), 1043-1056*

180. Tung, Y.k., and Kolterman, C.E. (1985), 'Some computational experiences using embedding technique for groundwater management.' *Groundwater, 23(4), 455-464.*

181. Voss, C.I. (1984), 'SUTRA: a finite element simulation model for saturated-unsaturated, fluid density dependent groundwater flow with energy transport of chemically reactive sample species solute transport', US Geol. Surv. *Water Resour. Invest., 84-4369, 409pp.*

182. Wang, M. and Zheng, C. (1998), 'Groundwater management optimization using genetic algorithms and simulated annealing: formulation and comparisons.' *J. Water Resour. Assoc. 34(3), 519-530.*

183. Wang, W., and Ahlfeld, D.P. (1994), 'Optimal groundwater remediation with well location as decision variables: Model development.' *Water Resour. Res., 30(5), 1605-1618.*

184. Wang, J.F., and Anderson, M.P. (1982), ' Introduction to groundwater modelling', Academic Press, San Diego.

185. Wagner, J. M., Shamir, U., and Nemati, H.R. (1992), 'Groundwater quality management under uncertainity: Stochastic programming approaches and the values of information.' *Water Resour. Res., 28(5), 1233-1246.*

186. Wagner, B. J. (1992), 'Simultaneous parameter estimation and contaminant source characterization for coupled groundwater flow and contaminant transport modelling.' *J. Hydrol. 135, 275-303.*

187. Wagner, B.J., and Gorelick, S.M. (1987), 'Optimal Groundwater quality management under parameter uncertainty.' *Water Resources Research, 23(7), 1162-1174.*

188. Wagner, B. J. And Gorelick, S.M. (1986), 'A statistical methodology for estimating transport parameters: Theory and application to one-dimensional advective dispersive systems.' *Water Resour. Res., 22(8), 1303-1315.*

189. Wanakule, N., Mays, L.W., Lasdon, L.S. (1986), 'Optimal management of large-scale aquifers, methodology and applications.' *Water Resour. Res. 22, 447–465.*

190. Wang, J.F., and Anderson, M.P. (1982), 'Introduction to Groundwater Modelling: Finite Difference and Finite Element Methods.' W.H. Freeman and Company, San Francisco.

191. Willis, R., and Finney, B.A. (1985), 'Optimal control of nonlinear groundwater hydraulics: Theoretical development and numerical experiments.' *Water Resour. Res., 21(10), 1476-1482.*

192. Willis, R. (1983), 'A unified approach to regional groundwater management in groundwater hydraulics.' Water Resour. Monogr. Ser. Edited by J.S. Rosenshein and G.D. Bennett, AGU, Washington, D.C.

193. Willis, R., and Liu, P. (1984), 'Optimization model for groundwater planning.' *J. Water Resour. Plng. and Mgmt., ASCE, 110(3), 333-347.*

194. Willis, R. (1979), 'A planning model for the management of groundwater quality.' *Water Resour. Res., 15(6), 1305-1312.*

195. Willis, R. (1977), 'Optimal groundwater resource management using the response equation method.' in Finite Elements in Water Resour. edited by W.G. Gray and G.F. Finder, Pentech, London

196. Willis, R., and Newman, B.A. (1977), 'Management model for groundwater development.' *J. of Water Resour. Plng. and Mgmt., ASCE, 103(WR1), 159-171.*

197. Willis, R. (1976), 'Optimal groundwater quality management: well injection of waste water.' *Water Resour. Res., 12(1), 47-53.*

198. Yan, S.,and Minsker, B., (2011), 'Applying Dynamic Surrogate Models in Noisy Genetic Algorithms to Optimize Groundwater Remediation Designs.' *J. Water Resour. Plng. Manage. 137(3), 276-292.*

199. Yan, S.,and Minsker, B., (2006), 'Optimal groundwater remediation design using an adaptive neural network genetic algorithm.' *Water Resour. Res. 42(5): WO407. doi: 10.1029/2005WR004303.*

200. Yazicigil, H., and Rasheeduddin, M. (1987), 'Optimization model for groundwater management in multi-aquifer systems.' *J. of Water Resour. Plng and Mgmt., ASCE, 113(2), 257-273.*

201. Yeh, H.D., Chang, T.H. and Lin, Y.C. (2007), 'Groundwater Contaminant Source Identification by a Hybrid Heuristic Approach.' *Water Resour. Res., 43(9), 1-16.*

202. Yeh, M. S., Lin, Y. P., & Chang, L. C. (2006), 'Designing an optimal multivariate Geostatistical groundwater quality monitoring network using factorial Kriging and genetic algorithm.' *J. Environ. Geo., 50, 101–121. 43:1–16.*

203. Yeh, W. W. G. (1986), 'Review of parameter identification procedure in groundwater hydrology: The inverse problem.' *Water Resour. Res., 22(2), 95-108.*

204. Yeh, W. W. G., Yoon, Y.S.. and Lee, K.S. (1983), 'Aquifer parameter identification with kriging and optimum parameterization.' *Water Resour. Res., 19(1), 225-233.*

205. Yeh, W.W.G. (1975), 'Optimal identification of parameters in a inhomogeneous medium with quardratic programming.' *Soc. Pet. Eng. J. 15(5), 371-375.*

206. Zheng, C., and Wang, P.P (1999), 'An integrated global and local optimization approach for remediation system design.' Water Resour Res, 20(12), 1837-1847.

207. Zurada, J.M. (1999), 'Introduction to artificial neural systems', Jaico publishing house, India.

208. By the ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. *J. Hydrol. Engg., 5(2) , April , 2000, 124-137.*

List of publications

Journals Referred Publications

1. Borah Triptimoni, and Bhattacharjya Rajib Kumar (2014), "Development of Unknown Pollution Source Identification Models Using GMS ANN Based Simulation-Optimization Methodology." *Journal of Hazardous, Toxic, and Radioactive Waste, ASCE, (in press).*

2. Borah Triptimoni, and Bhattacharjya Rajib Kumar (2013), "Solution of Source Identification Problem By Using GMS And MATLAB", *ISH Journal of Hydraulic Engineering*, 19(3), 297-304.

3. Borah Triptimoni, and Bhattacharjya Rajib Kumar (2014),"Development of Source Identification Model by Linking combined GMS-ANN Simulation Model with GA based Optimization Model." (Under Review).

4. Borah Triptimoni, and Bhattacharjya Rajib Kumar (2014), "Review Paper, Inverse optimization techniques for identification of unknown groundwater pollution sources." (Paper under preparation).

Conference Publications:

1. Borah Triptimoni, and Bhattacharjya Rajib Kumar (2013), "Solution of Source Identification Problem By Using GMS And MATLAB", HYDRO-2012, ISH, IIT Bombay, Dec 7-8, India.

2. Borah Triptimoni, and Bhattacharjya, Rajib Kumar, (2013), "Matlab-WMS based pollution source identification model for groundwater aquifer", IAH 2013, held at Perth Convention Centre, Western Australia from 15-20 September, 2013.

3. Borah Triptimoni, and Bhattacharjya, Rajib Kumar, (2012), "Coastal aquifer management models: A comprehensive review on model development", International

conference on Environmentally Sustainable Urban Ecosystem (ENSURE 2012), held in IIT Guwhati, Guwahati, India from 24-26 February 2012.

4. Borah Triptimoni, and Bhattacharjya, Rajib Kumar (2011), "Simple simulation procedure of transient groundwater flow process using spreadsheet solver", 4th International Perspective on Water Resources and Environment: IPWE 2011, held at National University of Singapore, Singapore from 4-6 of January, 2011.