PARAMETER ESTIMATION OF GROUND THERMAL PROPERTIES

by

NAGENDRA KUMAR JAIN

Bachelor of Science

Indian Institute of Technology

Bombay, India

1997

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the degree of
MASTER of SCIENCE
May, 1999

PARAMETER ESTIMATION OF GROUND

THERMAL PROPERTIES

Thesis Approved:

_____

Thesis Advisor

_____

_____

_____

Dean of the Graduate College

ACKNOWLEDGEMENTS

I would like to thank my parents and my brother for their constant support during my graduate studies. Without their support, this work would not have been possible.

I would like to extend my reverence to Dr. Jeffrey D. Spitler for his constant guidance and valuable input during the course of this project. You have shown me a way to my future and I hope to adopt your principles in my future life.

I would like to thank Dr. Rees Simon for his expert guidance and comments during this project. I learned a lot from you and I respect you for all your support.

I would like to extend my thanks and appreciation to the following people:

The members of my advisory committee for their willingness to provide me input and carve this work in a better way.

Cenk Yavuzturk for providing me his valuable support, input and working late hours with me together in the lab.

Andrew Chaisson for working together on the project which was useful to me in my thesis.

Ravikant Sharma for his support during pessimistic days.

Rajesh Chakravarthy for being supportive during strange situations and saving my time by making a way through them.

Naren for being quite helpful out of the way.

TABLE OF CONTENTS

Chapter *Page*

LIST OF FIGURES

Figure *Page*

LIST OF TABLES

Table *Page*

1. Introduction

1.1. Overview

Ground source heat pump systems (GSHP) offer a great advantage over conventional heat pump systems. GSHPs have high efficiency, low maintenance costs and low overall operating costs. The heat is absorbed and rejected through water, which is a more desirable heat transfer medium than air as used in conventional systems. The heat is absorbed from the ground during heating operation and rejected to the ground during cooling operation. Since the ground is nearly at a constant temperature, the heat is absorbed and rejected at a more constant temperature than in air source heat pump systems, which leads to high coefficient of performance (COP) of these systems.

In GSHPs, there are three loops in the system. The first loop is the water/air loop depending on whether the heated/cooled space requires water or air to heat/cool the space. The second loop is the refrigerant loop which exchanges heat with both source and sink loops. The third loop is the ground loop in which water exchanges heat with the ground and refrigerant. This loop might be closed loop or open loop depending on the GSHP system.

For commercial buildings the ground loop typically consists of a series of vertical heat exchangers, made up of three main components, the high-density polyethylene pipe (HDPE), material around the pipe called grout and the soil around grout. The pipe, which has a shape of U-tube, is inserted in a vertical borehole and the borehole is filled with grout around the pipe. The length of the borehole varies depending on the in-situ test.

Designing of GSHP systems requires accurate estimation of the thermal properties of the ground. Hence, to accurately estimate the ground thermal properties, a trailer was designed by Austin (1998, 2000) to experimentally measure power and temperature response. The power and temperature response can be used to inversely find the ground thermal conductivity.

Using power measurements, borehole geometry and a guess of ground thermal properties, a two-dimensional numerical model developed by Yavuzturk et al (1999) simulates the borehole by finite difference methods and temperatures are computed at the appropriate location of the borehole. The difference between the experimental and numerical temperature profiles, at each time step is calculated, squared and added. The thermal properties of the ground are adjusted such that this sum of the square of the error (SSQERR) is minimized. An optimization algorithm is required to systematically vary the properties to reach the minimum of SSQERR. This inverse problem of finding the parameters using an experimental test and a numerical model is called a *parameter estimation* problem. If the parameter estimation is performed after all the experimental data has been collected, it is called off-line parameter estimation.

To minimize SSQERR, several optimization methods have been applied so as to reduce the time taken in estimating the thermal properties of the ground. Since, it takes about three minutes of computer CPU time on a 233 MHz Pentium II processor to get the temperature profile at each set of ground thermal properties, it is extremely critical to reduce the number of times the model computes the temperature profile. Hence, the main aim of the project is to reduce the number of objective function evaluations (getting the SSQERR value from the model).

Only two parameters, soil and grout thermal conductivity, are considered for the parameter estimation results. Additional parameters are probably redundant and add unwarranted complexity to the problem. For more details on choosing the parameters, see the thesis by Austin (1998) and research paper Austin et al (2000).

It is also desirable to be able to estimate the ground thermal properties while the in-situ test is performed. This reduces the overhead of taking the experimental data and performing the off-line estimation of parameters. By performing online parameter estimation, we can obtain a plot of parameters as test is performed which helps in deciding the questions like the length of the test and final parameter values.

This thesis focuses mainly on the application of different optimization algorithms and finding the best algorithm to expedite the process of estimating the ground thermal properties. Online parameter estimation is also investigated.

1.2. Literature Review

Since the parameters for this problem are non-linearly related to the objective function and are unbounded, unconstrained non-linear optimization methods are considered. Optimization methods are divided into two major classes:

- Deterministic Methods.
  - Direct Search Methods.
  - Indirect Search (Gradient) Methods.
- Stochastic Methods.

Deterministic methods find the minimum by searching the minimum in a particular direction with finite number of steps. Stochastic methods use random search to find the minimum. Hence, the minimum found with stochastic methods is a minimum only in probabilistic sense. Most of the methods considered in this thesis are deterministic methods.

Deterministic methods are further divided into two classes, direct search methods and indirect search methods. Direct search methods require only the value of objective function (SSQERR value) to be calculated at a particular set of parameters. Indirect search methods use gradient of objective function to minimize.

The optimization methods used in this thesis are as follows:

- Exhaustive Search.
- Nelder Mead Simplex Algorithm, Nelder and Mead(1965)
- O’Neill’s implementation of Nelder Mead Simplex, R. O’Neill (1971)
- Box's Complex method, M. J. Box (1972)

- Hooke and Jeeves' method, Hooke and Jeeves (1972)
- Powell's method, M. J. D. Powell (1964)
- BFGS method, S. S. Rao (1996)
- Genetic Algorithms, D. E. Goldberg (1989)
- Quadratic fit method.
- O'Neill's implementation of Nelder Mead Simplex with exploratory search.

Exhaustive search is just used to see how the optimization domain looks like. This method of optimization is not feasible since the time taken to estimate the thermal properties is enormous.

Nelder Mead Simplex and O'Neill's Simplex are methods that use only the evaluation of objective function value at points that are obtained using reflection, expansion and contraction of a simplex (a n+1 dimensional figure in n dimensional space). Box's Complex method is similar to Nelder Mead Simplex except that it uses a 2*n+1 dimensional figure called a "Complex". Powell's method is generates conjugate directions and searches for the minimum along those directions. BFGS method is another gradient-based method. Genetic algorithms are stochastic method, which use random number generators to find the minimum in a domain. The Quadratic fit method is based on line minimization and fitting a quadratic polynomial. Exploratory search is useful in getting a better starting guess of parameters.

1.3. Experimental Apparatus

The experimental apparatus is contained within an enclosed trailer, constructed by Austin (1998), that contains all the components required to perform the in-situ test. The main components in the trailer are water heating elements, water supply/purge tank, pumps, valves, SCR power controller and two 7000 watts power generators. The instrumentation and data acquisition equipment are flow meter, two thermistor probes, watt transducer, thermocouple and a data logger. Since this trailer must be capable of testing even in undeveloped areas, it contains generator and water tank to supply power

and water. The trailer is divided into subsystems namely water supply, power supply, water heating, flow sensing/control equipment, pipe insulation, temperature measurement and data acquisition. Power Supply System, Data Acquisition System and Temperature measurement are described here. Figure 1-1 shows the schematic diagram of the trailer.

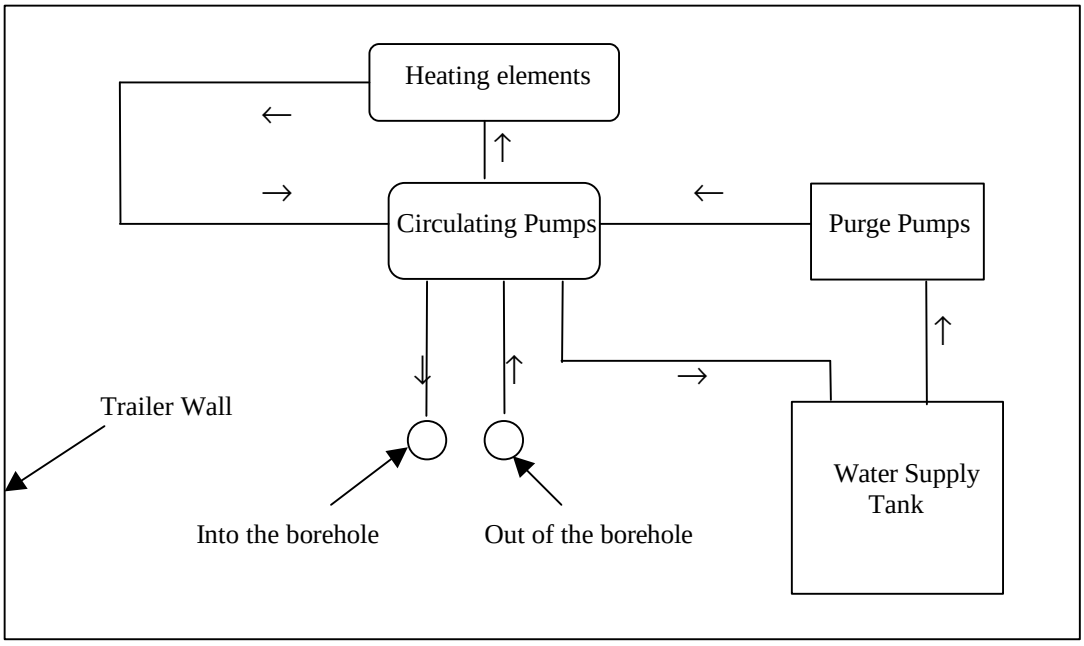

Figure 1-1. A schematic diagram of trailer

1.3.1. Power Supply System

Three heater elements are used to heat the circulating water. The heat input can be varied in the range of 0 kW to 4.5 kW using SCR power controller which can vary the power between 0 kW to 2 kW with other two heater elements of 1 kW and 1.5 kW on or off. The total heat input is the energy input from the three heater elements and power input to the circulating pumps. This energy input is measured with watt transducer.

1.3.2. Data Acquisition System

The output of watt transducer and digital display is measured by a Fluke Hydra Data Logger. Digital display displays the DC voltage on the scale of 0-10 Volts for each measurement. Three readings are recorded by the data logger is:

- Outlet temperature of water (leaving the trailer)
- Inlet temperature of water (returning the trailer)
- Flow rate

All of the measurements are in DC Volts. These measurements are converted from DC volts to the actual units by taking the raw voltage and fitting a linear curve. Inside and outside temperatures of the trailer are also recorded using thermocouples. These data are stored in the data logger's memory. If the data logger is connected to a computer, the data can be loaded on to the computer at any time by scanning the data logger's memory.

1.3.3. Temperature measurement

The experimental apparatus uses three thermistor probes. Two probes measure the temperature of water at the inlet and the outlet of the borehole. Third probe is used to measure the temperature of the pipe wall.

1.4. The Numerical Model

Yavuzturk et al (1999) developed the numerical model, which simulates the borehole with the U-tube, grout and soil. A sketch of the numerical domain is shown in Figure 1-2. The geometry of U-tube has been approximated by a "pie-sector". Only one half of the numerical domain has been simulated due to symmetry of the domain. The power is superimposed at each time step and is considered constant during a single time step. Radial conduction heat transfer equations have been solved over the geometry and temperatures have been calculated. The initial condition is that the temperature is constant and equal to far field temperature everywhere. The boundary conditions are far field temperature at the outer boundary and a constant heat flux at the inner boundary during a small time step.

Input parameters to the numerical model are borehole geometry (borehole radius, U-tube length, pipe diameter, distance between the legs of U-tube, thickness of pipe, heat capacity of pipe, heat capacity of grout and soil, number of hours for which the

experimental data has been collected, time step, far field temperature). All of these parameters are fixed due to the experimental set-up. Other input parameters are initial guesses of soil and grout conductivity. Experimental temperatures are calculated by averaging the inlet and the outlet water temperature to the borehole. The subroutine *ConductSingleSim* gives the temperature values for the whole length of the test at a particular K_{soil} and K_{grout}.

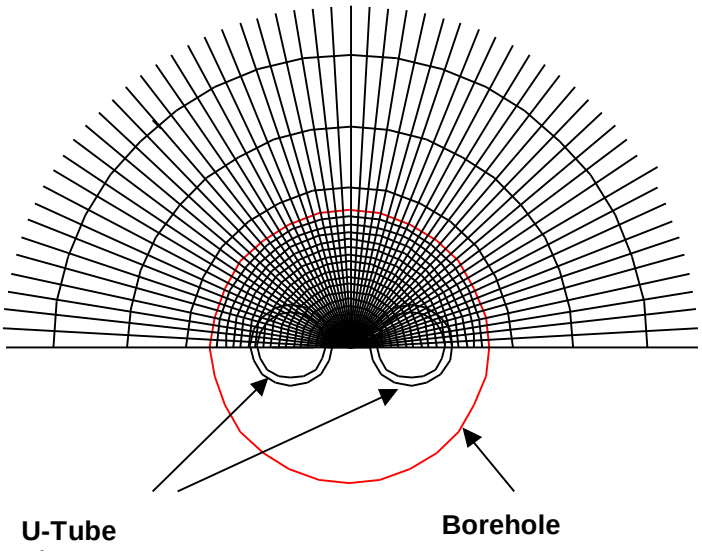

Figure 1-2. Domain for numerical model

1.5. Objective function

The objective function is the sum of the square of the errors between experimental temperatures and numerically estimated temperatures, specifically:

$$SSQERR = \sum_{n=1}^{N} \left(T_{experimental,n} - T_{numerical,n}\right)^2 \qquad \textbf{(1-1)}$$

Where, N is the total number of temperature measurements,

$T_{experimental,n}$ is the experimental temperature at the n_{th} time step,

$T_{numerical,n}$ is the average fluid temperature obtained by numerical model at the n_{th} time step.

A typical plot of experimentally obtained temperatures and numerically estimated temperatures is shown in Figure 1-3.

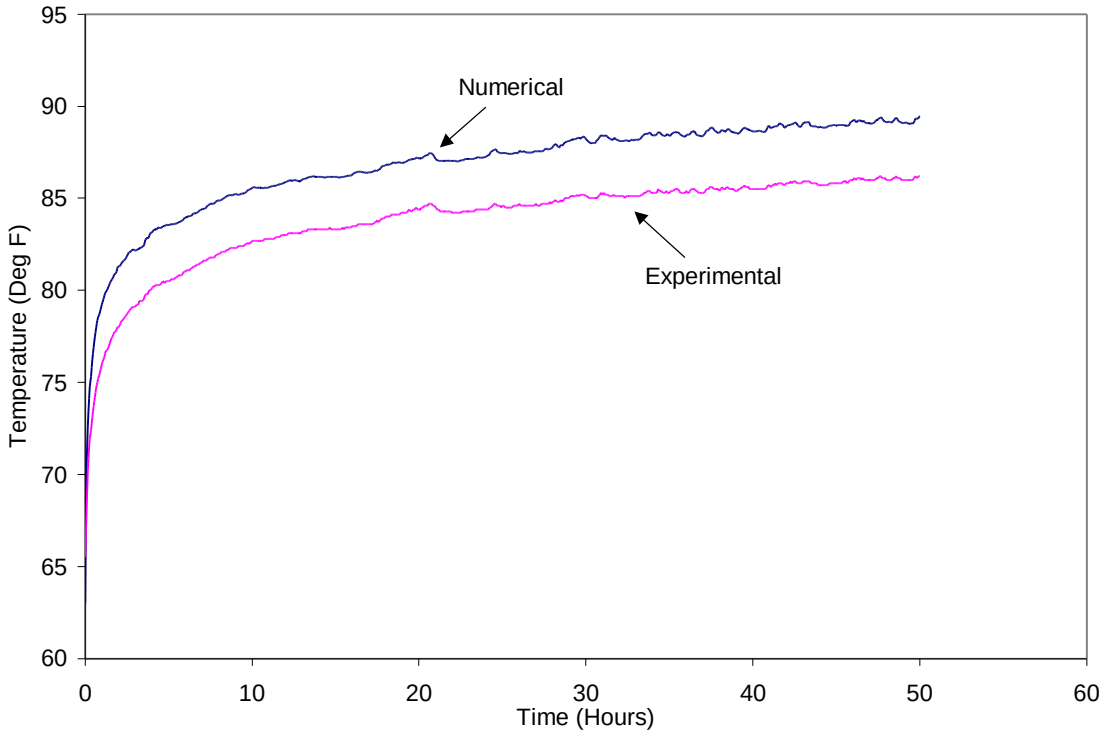

Figure 1-3. A typical plot of comparison between Experimental and Estimated temperatures by numerical model

2. Optimization Domain

2.1. Exhaustive Search

Exhaustive search has been performed just to view how the objective function surface looks like. The main aim of this search is to find out whether there are any ripples and local minima exist in the valley or the valley is smooth with no ups and downs. This is helpful because the minimum found by different optimization methods will be the global minimum if the method converges.

To apply exhaustive search to the optimization domain, we divide the domain of ground thermal properties (namely, K_{soil} and K_{grout}) into a grid of small step sizes in both the directions. At each grid node, a value of objective function is calculated from the SSQERR function. In the current application of this method, we have applied this to our SiteA1 in-situ test. The domain is divided into 100 X 80 grid in the direction of K_{soil} and K_{grout} respectively. The range of K_{soil} is 1.0 Btu/hr-ft-F to 1.8 Btu/hr-ft-F. The range of K_{grout} is 0.2 Btu/hr-ft-F to 1.0 Btu/hr-ft-F. These ranges are chosen based on some previous experience about the properties. This leads to 8000 objective function evaluations, which takes around 20 days to run on a Pentium II 233 MHz computer.

After computing all the objective function values, we find the minimum of all the values, which is the true global minimum. This minimum is not quite accurate because of the finite size of the grid. This method has a drawback that it requires a lot of objective function evaluations and uses an enormous amount of computer time. Hence, other optimization methods have been sought to expedite the process of estimating the ground thermal properties.

Figure 2-1 shows that the objective function surface is a deep turning valley. From the figure it can be observed that there are no local minimum in the valley and the floor of the valley is quite flat near the minimum. It is not possible to see the absolute minimum point with naked eyes.

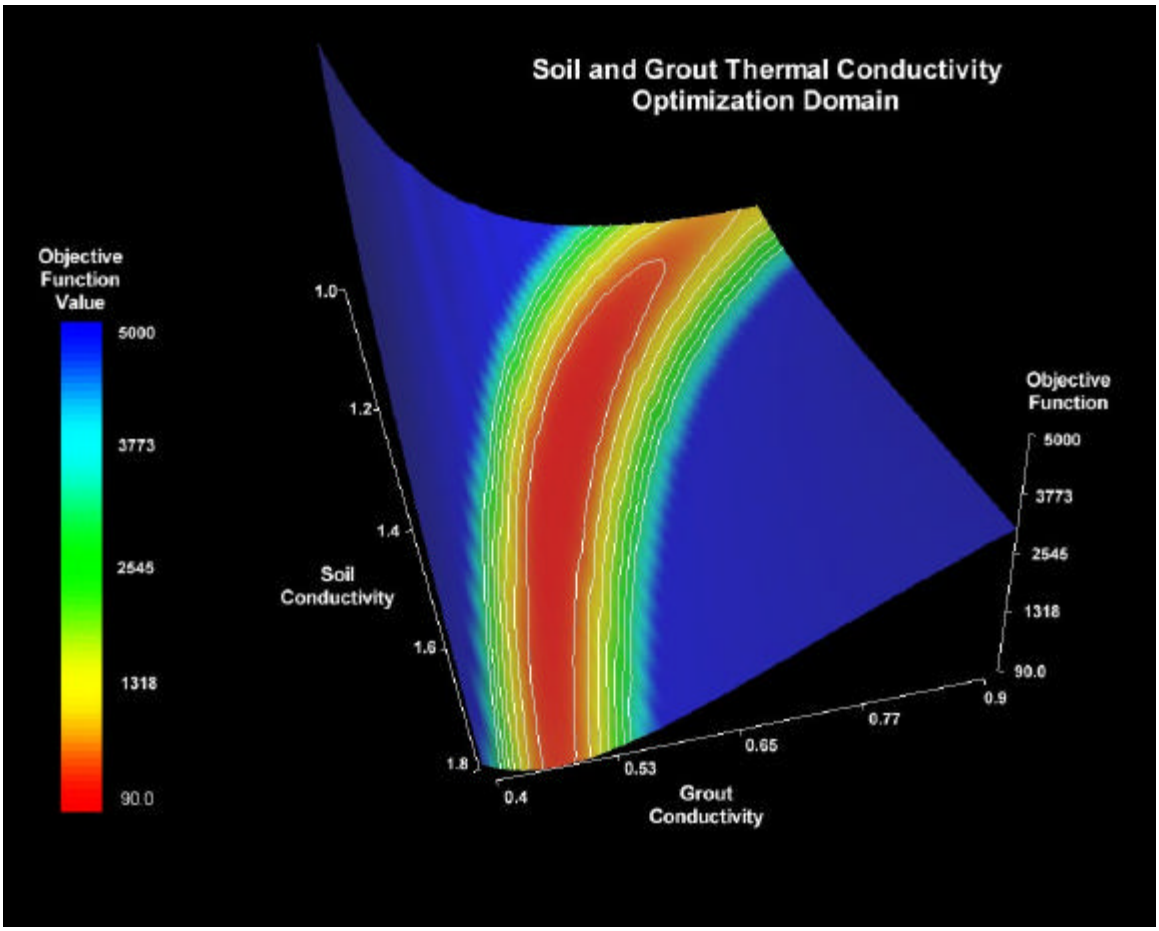

Figure 2-1. Domain of objective function

3. Optimization Methods

3.1. Nelder Mead Simplex

This algorithm, first proposed by Spendley et all (1962) and later developed by Nelder and Mead (1965), is a multidimensional minimization method, that is, finds the minimum of a function of more than one independent variable. This method requires only objective function evaluations, not derivatives. This is useful since it is quite costly to calculate derivatives for our problem. By costly, we mean that each objective function evaluation takes a lot of time, approximately three minutes on a Pentium II 233 MHz computer.

This algorithm requires a geometric figure of n+1 vertices in n dimensional space, called a "Simplex". In two dimensions, this figure becomes a triangle. The process of minimization requires different steps called "reflection", "expansion" and "contraction". For our case of two-dimensional minimization (K_{soil} and K_{grout}), the code starts with a starting simplex and objective function values at each vertex of the simplex. The simplex has to be formed in such a way as to ensure that all the three points are not collinear (actually a right angle triangle is constructed). Then the actual process of moving the simplex towards the minimum starts. This process uses reflection, expansion and contraction of the simplex. The amoeba code, implemented for this method, is available in Numerical Recipes (Press et. al, 1986).

3.1.1. Reflection

If $\mathbf{Y}_h$ is the vertex at which the objective function value is largest, then a point $\mathbf{Y}_r$ is obtained by reflecting this vertex across the centroid of the remaining vertices ($\mathbf{Y}_0$) and a lesser function value is expected at this vertex. Mathematically the reflection process is given by:

$$\mathbf{Y}_r = \mathbf{Y}_0 + \alpha (\mathbf{Y}_0 - \mathbf{Y}_h) \qquad \textbf{(3-1)}$$

Where α is the reflection coefficient and $\mathbf{Y}_h$ is the vertex corresponding to the maximum objective function value:

$$f(\mathbf{Y}_h) = \max_{j=1,n+1} f(\mathbf{Y}_j) \qquad \textbf{(3-2)}$$

and $\mathbf{Y}_0$ is the centroid of all vertices except j = h:

$$\mathbf{Y}_0 = \frac{1}{n} \sum_{\substack{j=1 \\ j \neq h}}^{n+1} \mathbf{Y}_j \qquad \textbf{(3-3)}$$

and α is defined as

$$a = \frac{\text{Distance between } \mathbf{Y}_r \text{ and } \mathbf{Y}_0}{\text{Distance between } \mathbf{Y}_h \text{ and } \mathbf{Y}_0} \qquad \textbf{(3-4)}$$

It is not entirely possible to reach a minimum by only reflection process due to certain difficulties. If the objective function value at reflected vertex turns out to be greater than or equal to the value at reflecting vertex, then we have to stop since this does not lead to a new simplex or we will have to reflect the second worst vertex. Using this, further improvements towards the minimum are made. To absolutely find the minimum we have to consider other operations like expansion and contraction.

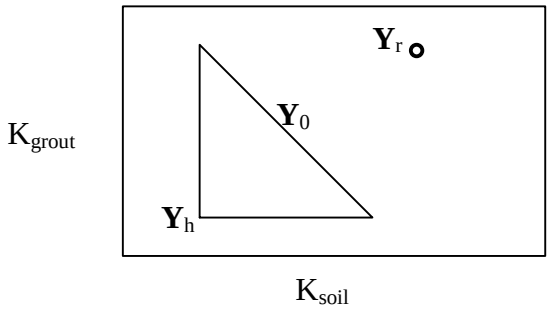

Figure 3-1. Reflection Process

3.1.2 Expansion

If the objective function value at the reflected vertex is minimum of all the vertices in the simplex, it is expected that a further improvement is possible in the same direction. Hence, expansion process is introduced. The expansion can be shown as:

$$\mathbf{Y}_e = \mathbf{Y}_0 + \gamma\,(\mathbf{Y}_r - \mathbf{Y}_0) \qquad \textbf{(3-5)}$$

Where γ is the expansion coefficient defined as:

$$? = \frac{\text{Distance between } \mathbf{Y}_e \text{ and } \mathbf{Y}_0}{\text{Distance between } \mathbf{Y}_r \text{ and } \mathbf{Y}_0} \qquad \textbf{(3-6)}$$

The value of this coefficient is greater than 1. If $f(\mathbf{Y}_e) < f(\mathbf{Y}_r)$, then $\mathbf{Y}_r$ is replaced by $\mathbf{Y}_e$ and the reflection process is restarted. If $f(\mathbf{Y}_e) > f(\mathbf{Y}_r)$, which means expansion process is not successful, $\mathbf{Y}_e$ is rejected and $\mathbf{Y}_h$ is replaced by $\mathbf{Y}_r$ and reflection process is started again.

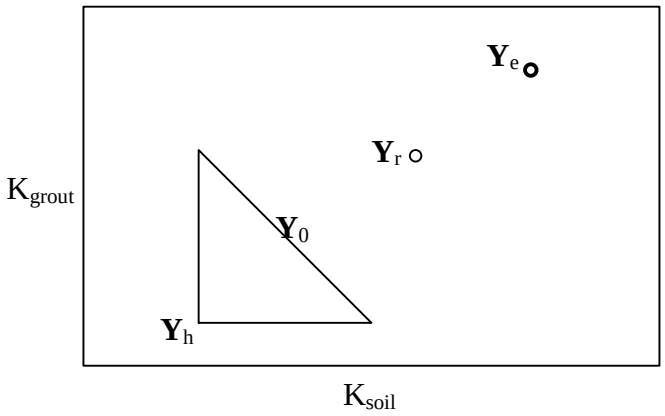

Figure 3-2. Expansion Process

3.1.3 Contraction

If the reflection produces a vertex with the highest objective function value, a contraction is tried. First $\mathbf{Y}_h$ is made equal to $\mathbf{Y}_h$ or $\mathbf{Y}_r$ (the reflected point) depending upon the lower objective function value at $\mathbf{Y}_h$ and $\mathbf{Y}_r$, or $\mathbf{Y}_h = \min(\mathbf{Y}_h, \mathbf{Y}_r)$. Now, a process of contraction is tried using $\mathbf{Y}_h$ and $\mathbf{Y}_0$ (the centroid) according to:

$$\mathbf{Y}_c = \mathbf{Y}_0 + \beta\,(\mathbf{Y}_h - \mathbf{Y}_0) \qquad \textbf{(3-7)}$$

Where β is contraction ratio defined as

$$ß = \frac{\text{Distance between } \mathbf{Y}_c \text{ and } \mathbf{Y}_0}{\text{Distance between } \mathbf{Y}_h \text{ and } \mathbf{Y}_0} \qquad \textbf{(3-8)}$$

This contraction ratio is always less than 1. If the objective function value at $\mathbf{Y}_c$ is greater than the value at $\mathbf{Y}_h$ then a point with lower objective function value could not be produced using reflection and contraction. Hence, a contraction about the vertex of minimum objective function value is performed as:

$$\mathbf{Y}_i = (\mathbf{Y}_i + \mathbf{Y}_{low})/2 \quad \text{for } i = 1\ldots ndim+1 \qquad \textbf{(3-9)}$$

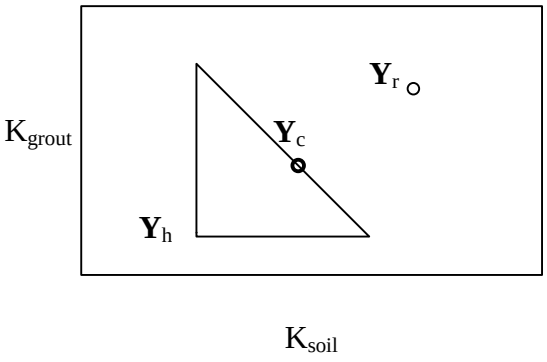

Figure 3-3. Contraction Process

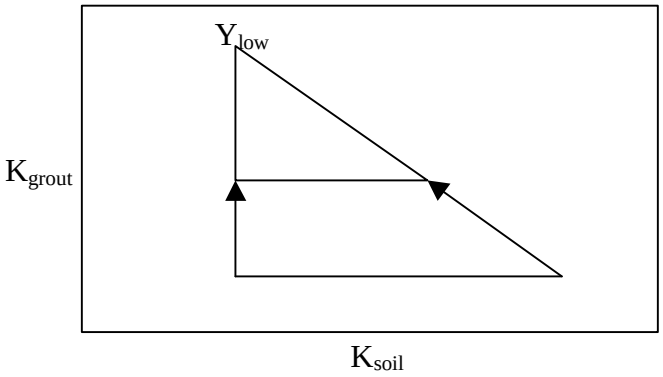

Figure 3-4. Contraction towards minimum

The reflection process is started again using this new simplex.

The convergence criteria to stop is:

$$\frac{2 \cdot (f(\mathbf{Y}_h) - f(\mathbf{Y}_{low}))}{(f(\mathbf{Y}_h) + f(\mathbf{Y}_{low}))} < e \qquad \textbf{(3-10)}$$

Where, f is the function value at the corresponding point, ε is the specified tolerance and $f(\mathbf{Y}_{low})$is defined as:

$$f(\mathbf{Y}_{low}) = \min_{j=1,n+1} f(\mathbf{Y}_j) \qquad \textbf{(3-11)}$$

3.2. O'Neill's implementation of Nelder Mead Simplex

R. O'Neill (1971) modified Nelder Mead Simplex algorithm and presented it in a different form. The advanced features of this algorithm are:

- Instead of passing an already constructed simplex to the optimization subroutine, this implementation requires that a starting vertex and step size in all the coordinate directions be provided. Then, the subroutine itself constructs a feasible simplex in such a manner that all the vertices of the simplex are not collinear. It computes the objective function values also at the vertices of the simplex.

- The convergence criterion has been changed. In this implementation, standard deviation is calculated using the centroid of the simplex. If the standard deviation is less than ε (a user-specified tolerance) then the process is stopped otherwise continued until the convergence criterion is met.

$$\text{Standard deviation} = \left\{ \sum_{i=1}^{n+1} \frac{[f(\mathbf{Y}_i) - \text{Centroid}]^2}{n+1} \right\}^{1/2} \leq e \qquad \textbf{(3-12)}$$

Where Centroid is calculated as:

$$\text{Centroid} = \frac{1}{n+1} \sum_{i=1}^{n+1} f(Y_i) \qquad \textbf{(3-13)}$$

- To ensure that the algorithm did not stop at the local minimum, a small number equal to δ*step size is added/subtracted to the current minimum in all the coordinate directions and objective function value is calculated at those vertices. If the objective

function values obtained are greater than the current minimum, then the global minimum has presumably been reached. If not, then the algorithm takes that vertex as a starting vertex and restarts itself with the step sizes set to original step size times a factor δ which is decided by the user.

The code for this method is available as Algorithm AS 47 in Applied Statistics (1971).

3.3. Box's Complex method

Box's complex method of optimization is a constraint optimization method and our problem domain is unbounded. But, fortunately we can restrict our attention to a range of $h <= K_{soil}$, $K_{grout} <= g$, where h=0.1 Btu/hr-ft-F and g=2.0 Btu/hr-ft-F (from experience) and eliminate the rest of the problem domain.

This method starts with a geometric figure of $2*n+1$ points (or vertices) called "Complex" rather than the n+1 points in simplex method. A starting vertex is provided which satisfies the constraints and other vertices ($2*n$) are generated randomly using the random number generators in the range of (0,1) and the range provided as follows:

$$Y_i = h + r * (g - h) \qquad \textbf{(3-14)}$$

Where, r is pseudo-random number in the range (0,1),

g is higher limit on the variable,

h is lower limit on the variable.

At each vertex, the value of the objective function is calculated. The vertex corresponding to the highest objective function value is reflected across the centroid of the vertices other than the vertex with highest objective function value. The amount of reflection can be varied using a parameter α, called reflection coefficient. The definition of reflection coefficient α is given by equation (3-4) in which the number n, is replaced with $2*n$.

If the reflection process produces a vertex with objective function value lower than the highest objective function value of the complex, the vertex is accepted and the reflection process is started again using the vertex having the highest objective function value in the new complex. Otherwise, the value of reflection coefficient, α, is reduced by half and the point is reflected again. This makes the reflected point come closer to the centroid of the complex. This process is repeated until the value of α becomes less than a minimum called $\alpha_{low.}$ If this still does not produce an objective function value less than the highest, the vertex corresponding to the second highest objective function value is reflected. This process is repeated until all the vertices are converged within some user-specified tolerance. Figure 3-5 shows the complex with $2*n+1$ vertices and the point with highest objective function value is reflected. The algorithm was available from Box M. J. (1965) and we developed the code for it.

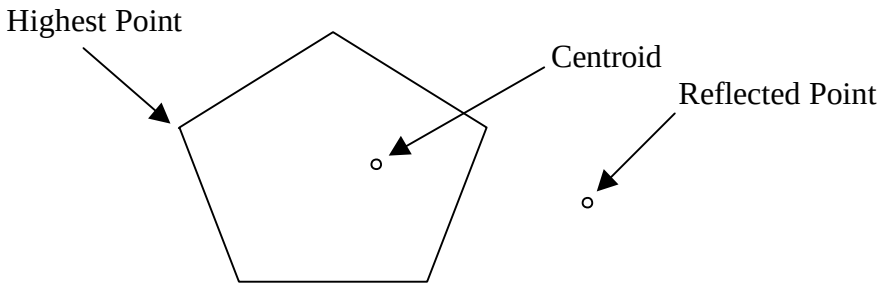

Figure 3-5. Complex in 2-D space

3.4. Hooke and Jeeves' method

Hooke and Jeeves (1972) devised a method for staying on the crest of the valley, while searching for an optimum. This pattern search method is based on the conjecture that the coordinate directions are worth trying again and again to search for an optimum along a ridge. This method starts with a small perturbation in each coordinate direction from the starting point and the step size increases as the success is achieved in the subsequent direction. A failure in all the directions would mean that the step size should be decreased.

There are two types of moves that are the main feature of this algorithm:

- Exploratory moves.
- Pattern moves.

In an *Exploratory move*, a small step in one coordinate direction is taken from the base point and a point corresponding to lower value of objective function is considered a temporary vertex. Then, next exploratory move is taken in the next coordinate direction (depending on the number of independent parameters) from this temporary vertex. This is repeated until all the coordinate directions are searched like this. The final point is designated as second base point.

In *Pattern moves*, a pattern direction is set with the two base points determined in the exploratory move. It is assumed that the same direction will result in a better point and the step size is doubled. This point is designated the temporary base point. Now, exploratory moves are carried out about this point and the point that gives a favorable objective function value is compared with the second base point. If this point turns out to be better than the second base point, another pattern direction is obtained by joining second base point and the current temporary vertex. This is repeated until no further base points could be obtained and exploratory steps are started again.

Step size is decreased when no further favorable steps can be taken in all the coordinate directions. This method has the advantage that it takes larger and larger steps in the favorable direction. The pattern direction turns as the valley turns and stays on the crest of the valley.

The algorithm was available from Rao S. S. (1996) and we developed the code for it.

3.4.1. Algorithm

- The algorithm starts with an initial guess $\mathbf{X}_1 = (x_1, x_2, x_3 \ldots x_n)^T$. This is termed as starting base point.

- A step size Δx_i is selected by the user in each independent unit coordinate direction $\mathbf{u}_i$ (i=1, 2…. n), where n is the number of parameters and $\mathbf{u}_i$ has i_{th} element as one.

- Evaluate objective function value at the starting point $\mathbf{X}_k$, f ($\mathbf{X}_k$), where k =1.

- Set $\mathbf{y}_{k,0} = \mathbf{X}_k$. Start the Exploratory step.

- Exploratory step: This step is performed to get a new temporary base point. Evaluate the objective function value at $\mathbf{y}_{k,i-1} + \Delta x_i * \mathbf{u}_i$. If this new point is better than $\mathbf{y}_{k,i-1}$ then, call $\mathbf{y}_{k,i-1} + \Delta x_i * \mathbf{u}_i$ the temporary base point $\mathbf{y}_{ki}$. If $\mathbf{y}_{k,i-1} + \Delta x_i * \mathbf{u}_i$ is not better than $\mathbf{y}_{k,i-1}$ then $\mathbf{y}_{k,i-1} - \Delta x_i * \mathbf{u}_i$ is tried. If this is better than $\mathbf{y}_{k,i-1}$ then, this is made temporary head otherwise $\mathbf{y}_{k,i-1}$ itself is made temporary base point. In summary, for minimization,

$$\mathbf{y}_{k,i} = \begin{cases} \mathbf{y}_{k,i-1} + \Delta x_i * \mathbf{u}_i & \text{if } f(\mathbf{y}_{k,i-1} + \Delta x_i * \mathbf{u}_i) < f(\mathbf{x}_1) \\ \mathbf{y}_{k,i-1} & \text{if } f(\mathbf{y}_{k,i-1}) < \min\left(f(\mathbf{y}_{k,i-1} + \Delta x_i * \mathbf{u}_i),\ f(\mathbf{y}_{k,i-1} - \Delta x_i * \mathbf{u}_i)\right) \\ \mathbf{y}_{k,i-1} - \Delta x_i * \mathbf{u}_i & \text{if } f(\mathbf{y}_{k,i-1} - \Delta x_i * \mathbf{u}_i) < f(\mathbf{y}_{k,i-1}) \end{cases} \quad \textbf{(3-15)}$$

This process is repeated with respect to the new base point found until all the coordinate directions are perturbed and final base point is designated as $\mathbf{y}_{k,n}$. If the new base point remains same as $\mathbf{X}_k$, then the step sizes Δx_i are reduced (by factor of 2) and the exploratory step is repeated again, otherwise, this new base point is

$$\mathbf{X}_{k+1} = \mathbf{y}_{k,n} \quad \textbf{(3-16)}$$

- With $\mathbf{X}_k$ and $\mathbf{X}_{k+1}$, a pattern direction **S** is established,

$$\mathbf{S} = \mathbf{X}_{k+1} - \mathbf{X}_k \quad \textbf{(3-17)}$$

and a new point is found as

$$\mathbf{y}_{k+1,0} = \mathbf{X}_{k+1} + \lambda \mathbf{S} \quad \textbf{(3-18)}$$

λ is assumed to be one in our optimization algorithm. Alternately, it can be found using one-dimensional minimization method in the pattern direction **S**. In that case, λ will be replaced with λ^*, the optimum step length.

- Increment to the next step and set k = k+1. Get the objective function value at $\mathbf{y}_{k,0}$, and repeat the exploratory move about this base point. If at the end of exploratory move, objective function value at the temporary base point is better than $\mathbf{X}_k$, then this point is assigned as new base point $\mathbf{X}_{k+1} = \mathbf{y}_{k,n}$ and new pattern direction is established again using this new base point and the old base point obtained in previous step. If the function value at $\mathbf{y}_{k,0}$ is greater than the function value at the old base point, then reduce the step size Δx_i, set k = k+1 and repeat the Exploratory step about this current base point.

- The process is assumed to have converged whenever the maximum step length falls below a user defined tolerance $\in$. Thus, the process is terminated when

$$\max_{i = 1,n} (\Delta x_i) < \in \qquad \textbf{(3-19)}$$

3.5. Powell's method

This method is a well-known pattern search method. In multi-dimensions this method consists of a sequence of line minimization. The main feature of these types of algorithms is to compute the next search direction starting with the coordinate directions. The method could be simpler if we just have to minimize in coordinate directions only. When the minimization is performed only in coordinate directions, the function does reach the minimum but it is extremely inefficient in the cases where a long turning valley exists at an angle to the coordinate directions. This is because the method has to take small steps in the coordinate directions to reach the minimum. Hence, for a long, narrow and twisted valley, like in our case, this method is extremely inefficient.

Hence, an algorithm, which computes a better set of directions, is needed than just using the coordinate directions. Some of these directions can reach very far in the narrow valley near the minimum. The concept of "non-interfering" directions (also called conjugate directions) helps the method reaching the minimum faster than with just the set of coordinate directions.

Two direction vectors, **u** and **v** are said to be conjugate with respect to function f if they satisfy the following criteria:

$$\mathbf{u} \bullet \mathbf{A} \bullet \mathbf{v} = 0 \qquad \textbf{(3-20)}$$

Where the components of **A** matrix are second partial derivative of function f and **A** is called Hessian matrix.

$$[\mathbf{A}]_{ij} \equiv \left. \frac{\partial^2 f}{\partial x_i \partial x_j} \right|_{\mathbf{P}} \qquad \textbf{(3-21)}$$

For two-dimensional case, the matrix becomes:

$$\mathbf{A}_{22} = \left[\begin{array}{cc} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1 x_2} \\ \dfrac{\partial^2 f}{\partial x_2 x_1} & \dfrac{\partial^2 f}{\partial x_2^2} \end{array} \right]_{\mathbf{P}} \qquad \textbf{(3-22)}$$

According to Powell, a quadratic function will converge in one cycle of line minimization if all the directions are mutually conjugate directions. Hence, Powell's method tries to generate N (number of dimensions) mutually conjugate directions. Powell states that for a quadratic function it will take N cycle of minimization to generate N mutually conjugate directions. Hence, a total of N*(N+1) number of line minimization will take us to the correct minimum. But this is only true when:

- The function is exactly quadratic.
- Each line minimization produces absolute minimum in that direction.

In reality, for any objective function like our problem this may not be true. Hence, it will take more number of line minimization than N*(N+1).

More heuristic schemes were suggested to Powell's basic method of quadratic convergence. The idea behind the modified Powell's method which is implemented here is to take $P_N - P_0$ (last point after N line minimization – starting point) as the new search direction for the next cycle of minimization and discard the direction in which there was a maximum decrease in the objective function value. This is because this was the best direction in the previous cycle of minimization and can play a major role in the new direction ($P_N - P_0$). Hence, by discarding this direction, the problem of linear dependence of directions can be reduced and this helps in generating the conjugate directions that are "linearly independent".

Both the algorithm and the code are available in Numerical Recipes (Press et. al, 1986).

3.5.1. Methodology

This method starts with coordinate directions as the starting search directions for the minimum starting with the first coordinate direction as shown in Figure 3-6. Then, it generates a pattern direction by taking the direction that is obtained after N univariate steps from the starting point. A univariate step is one in which the objective function is minimized along a particular direction. In each direction, the objective function is minimized using a one dimensional minimization routine. Our implementation of the method uses Brent's method to minimize in a particular direction. After this cycle of minimization, the new search direction is accepted or rejected based on the maximum decrease criterion and one of the old search directions is replaced with this new direction. Now, we have only N search directions. The new cycle of minimization is started again with these N search directions. This procedure is repeated until the desired minimum is reached.

The convergence criteria used is:

$$\frac{2 \cdot (f(X_0) - f(X_N))}{(f(X_0) + f(X_N))} < e \qquad \textbf{(3-23)}$$

Where $f(X_0)$ is the objective function value at the beginning of a cycle of line minimization and $f(X_N)$ is the objective function value at the end of a whole cycle of minimization consisting of N line minimization. ε is the tolerance value chosen by the user.

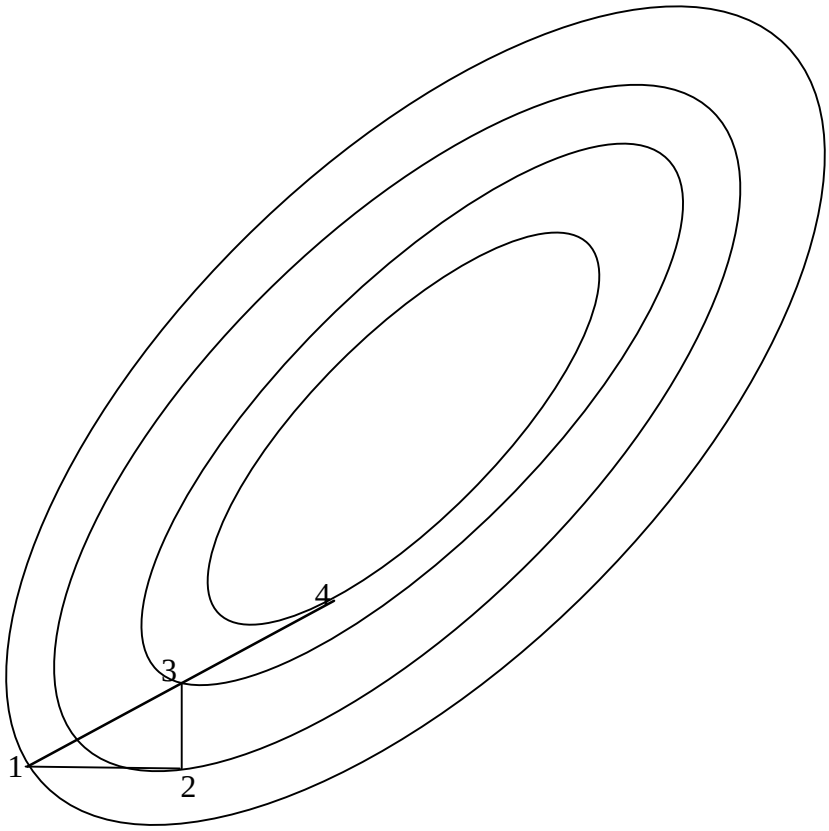

Figure 3-6. Steps taken in Powell's method.

3.6. Broyden-Fletcher-Goldfarb-Shanno (BFGS) method

3.6.1. Background

This method is one of the indirect search (gradient based) methods. Sometimes it is also called Quasi-Newton method. Different methods under this category differ only in their updating of inverse of Hessian matrix. Instead of directly taking the inverse of Hessian matrix, an iterative updating technique is used. The basic equation describing Newton's method is:

$$\mathbf{X}_{i+1} = \mathbf{X}_i - [\mathbf{J}_i]^{-1} \tilde{\mathbf{N}} f(\mathbf{X}_i) \quad \textbf{(3-24)}$$

Where, the Hessian matrix $[J_i]$ is composed of second partial derivatives of f and $\mathbf{X}_i$ is the vector in previous iteration. In quasi-Newton methods the idea is to approximate the inverse of Hessian matrix $[\mathbf{J}_i]^{-1}$ by some other matrix $[B_i]$. Hence, the approximated equation becomes:

$$\mathbf{X}_{i+1} = \mathbf{X}_i - \lambda_i^* [B_i] \tilde{\mathbf{N}} f(\mathbf{X}_i) \quad \textbf{(3-25)}$$

Where, λ_i^* is the optimal step length in the direction

$$\mathbf{S}_i = -[B_i] \tilde{\mathbf{N}} f(\mathbf{X}_i) \quad \textbf{(3-26)}$$

The matrix $[B_i]$ is approximated as:

$$[B_{i+1}] = [B_i] + c_1 \mathbf{z}_1 \mathbf{z}_1^T + c_2 \mathbf{z}_2 \mathbf{z}_2^T \quad \textbf{(3-27)}$$

Where, $\mathbf{z}_1$ and $\mathbf{z}_2$ are two independent n-component vectors and c_1, c_2 are constants. n is the number of dimensions. This is also called a rank 2 update of inverse of Hessian matrix. Finally, the Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula is given by

$$[B_{i+1}] = [B_i] + \frac{\mathbf{d}_i \mathbf{d}_i^T}{\mathbf{d}_i^T \mathbf{g}_i} \left(1 + \frac{\mathbf{g}_i^T [B_i] \mathbf{g}_i}{\mathbf{d}_i^T \mathbf{g}_i} \right) - \frac{[B_i] \mathbf{g}_i \mathbf{d}_i^T}{\mathbf{d}_i^T \mathbf{g}_i} - \frac{\mathbf{d}_i \mathbf{g}_i^T [B_i]}{\mathbf{d}_i^T \mathbf{g}_i} \quad \textbf{(3-28)}$$

The algorithm is available from Rao S. S. (1996) and we developed the code for it.

3.6.2. Algorithm

1. Algorithm starts with an initial guess $\mathbf{X}_i$.

2. Initialize the inverse of Hessian matrix $[B_i]$ with a positive definite symmetric matrix called identity matrix $[I]$.

3. Calculate the gradient of objective function ($\tilde{\nabla} f$) at the point $\mathbf{X}_i$ using forward finite difference formula.

$$\nabla f(X_i) = \left\{ \frac{f(X_{i+1}) - f(X_i)}{(X_{i+1} - X_i)} \right\} \qquad \textbf{(3-29)}$$

4. Compute the direction vector $\mathbf{S}_i = -[B_i]\, \tilde{\nabla} f(\mathbf{X}_i)$.

5. Minimize the function in the direction $\mathbf{S}_i$ and find the optimum length λ_i^*. In our implementation, minimization is done using Golden Section Search method.

6. Compute the new vector $\mathbf{X}_{i+1} = \mathbf{X}_i + \lambda_i^* \mathbf{S}_i$.

7. Calculate the gradient of objective function ($\tilde{\nabla} f_{i+1}$) at the point $\mathbf{X}_{i+1}$.

8. If $\|\nabla f_{i+1}\| \le \boldsymbol{e}$, where $\boldsymbol{e}$ is a user defined tolerance, then we assume that the minimum of the objective function is reached and stop the process. Otherwise we update the inverse of Hessian matrix.

9. Update of inverse of Hessian is done by

$$[B_{i+1}] = [B_i] + \frac{\mathbf{d}_i \mathbf{d}_i^T}{\mathbf{d}_i^T \mathbf{g}_i} \left(1 + \frac{\mathbf{g}_i^T [B_i] \mathbf{g}_i}{\mathbf{d}_i^T \mathbf{g}_i} \right) - \frac{[B_i] \mathbf{g}_i \mathbf{d}_i^T}{\mathbf{d}_i^T \mathbf{g}_i} - \frac{\mathbf{d}_i \mathbf{g}_i^T [B_i]}{\mathbf{d}_i^T \mathbf{g}_i} \qquad \textbf{(3-30)}$$

Where

$$\mathbf{d}_i = \mathbf{X}_{i+1} - \mathbf{X}_i = \lambda_i^* \mathbf{S}_i \qquad \textbf{(3-31)}$$

$$\mathbf{g}_i = \tilde{\nabla} f_{i+1} - \tilde{\nabla} f_i = \tilde{\nabla} f(\mathbf{X}_{i+1}) - \tilde{\nabla} f(\mathbf{X}_i) \qquad \textbf{(3-32)}$$

10. After updating the inverse of Hessian, we restart the whole process from step 2 until the condition in step 8 is satisfied.

3.7. Genetic Algorithms

Genetic algorithms (GA's) refer to a class of stochastic-based optimization/search techniques that rely on the theories of natural selection (i.e. "survival of the fittest"). As such, the purpose of a GA is to optimize a particular problem by simulating or imitating the evolution of life in the natural world.

According to Goldberg (1989), GA's owe their robustness over other optimization and search procedures to the following four attributes:

1. GA's work with a binary coding of the parameter set not the parameters themselves.
2. GA's search from a population of points, not a single point.
3. GA's use objective function information, not derivatives or other auxiliary knowledge.
4. GA's use probabilistic transition rules, not deterministic rules.

As an example of how GA's work, consider a problem where n parameters are to be optimized. The user selects a population size of 5, for example, meaning that at each generation, there are 5 points at which the objective function will be evaluated. At the start of the problem, all 5 parameter sets are generated randomly within a given range. For each parameter set, a binary string is formed which is made up of the binary equivalents of each individual parameter in the set. The result is that 5 binary strings (or 5 "parents") are formed. Each bit making up the parent is called a "gene". Parents are then selected for "reproduction" based on their function evaluation or "fitness". Parents with higher fitness are allowed to reproduce more times than those with a lower fitness. Reproduction is accomplished in GA's by "crossovers", which refer to a swapping of binary digits. Crossovers by "genetic mutations" can also be simulated by specifying a

probability at which a mutation may occur. Crossovers resulting from simulated biological "niching" can also occur. The results of binary digit swapping are the "children" which form the next generation of parents.

Computer Algorithm:

- Select number of parameters (nparam) to be optimized, the population size to work with (npopsize), and the number of generations to evaluate (maxgen).

- Initialize a random population of individuals at time = 0

```
    For i = 1 to nparam
        For j =1 to npopsize
            Parent (i,j) = random(parameter)
        Next j
    Next i
```

- Evaluate the objective function for each set of parameters (i.e. determine fitness)

For k = 1 to maxgen

- Test for termination criteria (no. of generations, etc.)
- Select a sub-population of parents for re-production based on fitness
- Code parameters of P(row,j) into one binary string and store in P'(row), where row is the selected parameter set
- Recombine "genes" of selected parents to form "children"
- Perturb the mated population stochastically
- Evaluate new fitness

Next k

End

3.8. Quadratic fit method

This method, developed for this project, is based on some experience from the exhaustive search method. From the exhaustive search we found that the SSQERR

surface in the domain of K_{soil} and K_{grout} is nearly quadratic and there exists a steep valley which is turning. Figure 3-7 shows the turning valley in two-dimensional domain. First optimization in the K_{grout} direction is performed using Brent's Quadratic Optimization method at three guessed K_{soil} values. After the three vertices are obtained along the floor of the valley, a quadratic curve is fit through these three vertices and minimization is performed along the quadratic by one dimensional line minimization routine called "Golden Section Search". Brent's method was used in K_{grout} direction since this method is based on fitting a quadratic curve that reduces the number of objective function evaluations in reaching the floor of the valley. This procedure leads the optimum point very close to the true global minimum of the valley in some cases. The idea of fitting the quadratic curve and optimizing along the quadratic comes from the fact that the valley of SSQERR surface is nearly quadratic and turning. We developed the code for this method.

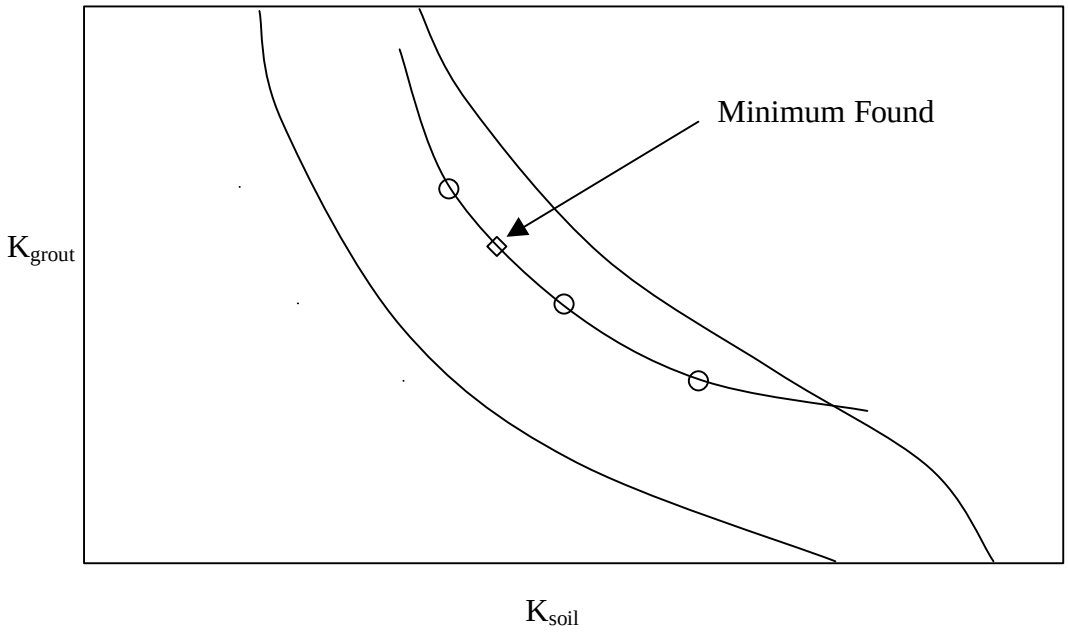

Figure 3-7. Quadratic Curve fitting in a valley

3.9. O'Neill's implementation of Nelder Mead Simplex with exploratory search

To apply O'Neill's Simplex or any other search method for parameter estimation, an initial guess of parameters is required to start with. If the initial guess of parameters is

close to the actual minimum then the search algorithms will find the minimum in less number of objective function evaluations than the initial guess being far away. Hence, a method that quickly gives an initial estimate of parameter values close to the minimum is desired.

Kelvin's line source theory and the cylinder source method presented by Ingersoll (1948, 1954) are the two well-known analytical methods to solve radial earth-coupled heat transfer problems using a series solution. Figure 3-8 shows the line source model. According to Hart and Couvillion (1986), line source theory is applicable for pipes also with modified interpretation of terms in equations (3-34) and (3-35). The line source method is modified to include a film of constant thickness around the equivalent diameter of the pipe and the resistance of this film is made equal to the grout resistance. The ratio of equivalent diameter to one pipe diameter, which is based on empirical results, can be changed based on our own experience. Our implementation uses the value of ratio to be 1.1 as opposed to $\sqrt{2}$ because the estimates of grout conductivity are better with this ratio.

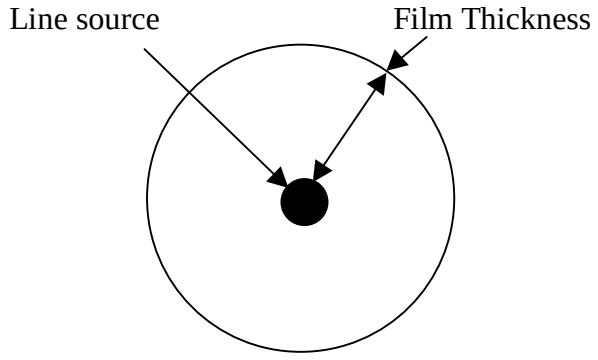

Figure 3-8. Line source model

Line source model is applicable for both constant heat flux per unit length of the source and constant temperature of the pipe. Since the power input varies with time during the experiment, the time-varying heat flux is decomposed into a series of constant heat flux during a time step and superposition principle is applied to get the temperature

response at any time. The expression for transient temperature distribution around the heat source/sink may be written as follows for our implementation:

$$T(r) = T_{\infty} + (\Delta T_{soil}) + (\Delta T_{grout}) \quad \textbf{(3-33)}$$

Where T_{∞} is the far-field temperature, ΔT_{soil} is the temperature difference due to soil and ΔT_{grout} is the temperature difference due to grout. Temperature difference due to soil is given by:

$$\Delta T_{soil} = T_{soil} - T_{\infty} = \frac{q}{2\,p\,k_{soil}}\left[\ln\frac{r_{\infty}}{r} - 0.9837 + \frac{4r^2}{2r_{\infty}^2} \ldots\ldots\ldots \frac{(-1)^{N+1}}{2N \cdot N!}\left(\frac{4r^2}{r_{\infty}^2}\right)^N\right] \quad \textbf{(3-34)}$$

Where,

$$r_{\infty} = 4\sqrt{at} \quad \textbf{(3-35)}$$

Where α is thermal diffusivity of ground, k_{soil} is thermal conductivity of soil, r_{∞} is the radius in which the rejected heat is completely absorbed in the ground, t is the time, N is the number of terms chosen by the user.

For equations (3-34) and (3-35) to be applicable for pipes, $\frac{r_{\infty}}{r_o} \geq 15$ has to be satisfied and the line source heat flux q has to be same as the heat flux at the inner radius of the pipe. This condition is achieved approximately after 45 minutes of experimental data for our test cases. Generally, two or three terms are sufficient in the equation (3-34). We have taken six terms in our implementation.

To estimate the temperature difference due to grout, the resistance of the film thickness around the equivalent diameter of the pipe is calculated as:

$$R_{grout} = \frac{\log\left(\frac{RAD\_BH}{DIS}\right)}{2 \cdot p \cdot k_{grout}} \quad \textbf{(3-36)}$$

Where RAD_BH is the radius of the borehole, DIS is the thickness of the grout around the pipe and k_{grout} is the thermal conductivity of the grout material. Temperature difference due to grout is then calculated as:

$$\Delta T_{grout} = T_{grout} - T_{\infty} = \frac{q}{L_{loop}} \cdot R_{grout} \quad \textbf{(3-37)}$$

Where L_{loop} is the length of the borehole and q is the power imposed.

O'Neill's Simplex algorithm is applied after the starting guess is obtained by this method which we hope is closer to the actual minimum. Step sizes of 0.2, smaller than the O'Neill step sizes of 0.5, are chosen in both the directions. The vertex obtained from exploratory search and the step sizes in both the directions are passed to the O'Neill Simplex method and optimum parameter values are obtained. We developed the code for exploratory search method.

3.10. Summary of all the methods

A summary of all the methods and their characteristics is shown in Table 3-1. Most of the methods are deterministic and non-gradient based.

Table 3-1. A summary of characteristics of all the methods

Method	Deterministic/ Stochastic	Gradient/Non-Gradient	Pattern Search
Nelder Mead Simplex	Deterministic	Non-Gradient	No
O'Neill Simplex	Deterministic	Non-Gradient	No
Box Complex	Deterministic	Non-Gradient	No
Powell	Deterministic	Non-Gradient	Yes
Hooke and Jeeves'	Deterministic	Non-Gradient	Yes
BFGS	Deterministic	Gradient	No
Genetic Algorithm	Stochastic	Non-Gradient	No
Quadratic fit	Deterministic	Non-Gradient	No
O'Neill's implementation with exploratory search	Deterministic	Non-Gradient	No

4. Off-line Estimation Results and Discussion

The results presented here are for 7 test sites. Table 4-1 shows the summary of experimental tests used for parameter estimation results. In-situ experiments were conducted using the trailer designed by Austin (1998) and temperature and power consumption data were collected. The two-dimensional numerical model that simulates the borehole with U-tube and grout uses this power data and the borehole parameters including the radius of the borehole, pipe conductivity, soil volumetric specific heat, grout volumetric specific heat, fluid viscosity, fluid density and fluid conductivity. This model generates the finite difference grid around the borehole and calculates the average fluid temperature, which is then compared with experimental data and sum of the square of the difference is calculated. This difference is minimized using the search methods.

It has already been established that 50 hours of experimental data are required to correctly estimate the soil and grout thermal conductivity with two-parameter estimation procedure. Hence, most of the parameter estimation results here are presented for 50 hours of data unless otherwise specified. For more details on the duration of experimental test, see Austin (1998).

Results obtained from Nelder Mead Simplex/O'Neill Simplex algorithms are considered to be "true minimum" because they provide the lowest objective function value (i.e. sum of the square of the error value) when compared to the other methods. Other method results are considered converged to true minimum if the percentage difference in K_{soil} is within $\pm 2.5\%$.

Results from all the methods are characterized for goodness of fit by the Estimated Standard Deviation (Scheaffer et al.1995). The Estimated Standard Deviation value is calculated as:

$$\text{Estimated Standard Deviation} = \sqrt{\frac{\sum_{i=1}^{N} (\text{Error})_i^2}{N - 2}} \qquad \textbf{(4-1)}$$

This provides a uniform comparison, regardless of the length of the experimental test.

Table 4-1. Summary of Experimental tests used for parameter estimation

Date	Location	Description	Code Name	Duration(hrs)
6/2/97	Stillwater, OK Site A	#1 - 3 1/2" Borehole, 244' deep, grouted with 30% solids Bentonite. Powered by electric line.	**SiteA1_98**	98
5/28/97	Stillwater, OK Site A	#2 -3 1/2" borehole, 252' deep, grouted with Thermal Grout 85. Powered by electric line.	**SiteA2_114**	114
1/9/97	Stillwater, OK Site A	#2 -3 1/2" borehole, 252' deep, grouted with Thermal Grout 85. Powered by electric line.	**SiteA2_170**	170
4/21/97	Stillwater, OK Site A	#5 -3 1/2" borehole, 252' deep, grouted with Benseal. Powered by electric line.	**SiteA5_93**	93
11/19/98	Stillwater, OK Site A	#6 -4.88" borehole, 258' deep, WyoBen-Groutwell. Powered by electric line.	**SiteA6_240**	240
9/26/97	Chickasha	3 1/2" borehole, 250' deep, grouted with 30% solids Bentonite. Powered by generators.	**Chickasha**	99
3/7/98	Weatherford	4 3/4" borehole, 248' deep, grouted with Benseal EZ-MUD. Powered by generators.	**Weatherford**	50

4.1. Results from Nelder Mead Simplex

The initial guess for the parameters (K_{soil}, K_{grout}) is taken to be (1.5 Btu/hr-ft-F, 0.575 Btu/hr-ft-F) which is assumed to be close to minimum. A simplex, which is a right angle triangle in this case, is constructed using 0.5 Btu/hr-ft-F as the step size. The triangle is shown in Figure 4-1.

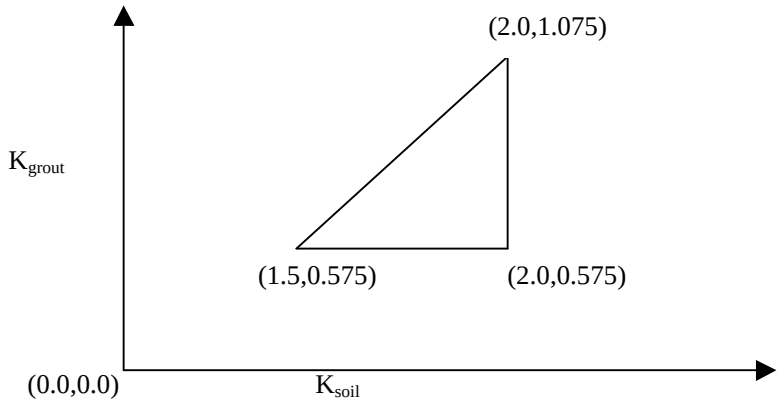

Figure 4-1. Simplex in 2-D domain

The value of the objective function is calculated at these three vertices and this information is passed to the subroutine “amoeba” since the subroutine requires an already constructed simplex. This algorithm then does all the processes of reflection, expansion and contraction of the simplex to give the optimum values of the two parameters, for which estimated temperature profile fits the best to the experimental temperature profile. The algorithm gives back the simplex whose first vertex contains the parameters corresponding to the minimum objective function value.

We do not have a local minimum in the valley as established by the exhaustive search. The nature of Nelder Mead Simplex search algorithm is such that it searches the objective function only at discrete points or computes the objective function value at certain points. When the three vertices of the simplex have their objective function values within the specified tolerance, 1-% in our case, the algorithm stops. If the algorithm is started with a different starting guess, it is likely that the simplex will roll down the valley taking a different path. The chances are high that another simplex might satisfy the same tolerance criteria and the minimum vertex, thus found, has a lower objective function value than the previous one. Making tolerance smaller than 1%, which is used currently, will also not cure the problem because we can still find a vertex, which will have slightly lesser objective function value and will lie within the final small simplex. This will increase the number of objective function evaluations only.

Hence, it is expected that a restart of the method sometimes give a lower objective function value than that obtained from the first run but certainly not worse than the previous one. Because the method is restarted with one vertex to be the same as the minimum found and the rest are constructed again with lesser step sizes since we are close to the minimum now. This gives us more confidence in the optimum parameter values. Sometimes a change of 1-1.5% in the value of soil conductivity is obtained. Table 4-2 gives the values of estimated K_{soil}, K_{grout}, Estimated Standard Deviation in the temperature and the number of objective function evaluations.

Table 4-2. A summary of results from Nelder Mead Simplex algorithm for 50 hours of data

Site	Ksoil	Kgrout	Estimated Standard Deviation	Function Evaluations
SiteA1_98	1.367	0.560	0.25762	73
SiteA2_114	1.460	0.858	0.22395	64
SiteA2_170	1.427	1.157	0.17603	72
SiteA5_93	1.435	0.581	0.15723	72
SiteA6_240	1.319	0.783	0.09842	83
Chickasha	1.490	0.758	0.15566	76
Weatherford	1.638	1.187	0.12046	109

Figure 4-2 shows the movement of simplex towards the minimum in Nelder Mead Simplex algorithm for SiteA5_93. The initial simplex is constructed with three (1.5,0.575), (2.0,0.575) and (1.5,1.075) non-collinear vertices forming a right angle triangle. The highest objective function value is at (1.5,0.575), not apparent from the figure. This vertex is reflected and then contracted to find a favorable vertex (1.625, 0.825) in terms of objective function value. The new simplex now has vertices (1.625, 0.825), (2.0,0.575) and (1.5,1.075). The highest objective function value in the new simplex is at vertex (1.625, 0.825) which is reflected against the centroid of remaining vertices and then contracted to find a favorable vertex (1.688, 0.7). This process of reflection and contraction continues until the minimum of the objective function is reached. Only the objective function value is calculated at each vertex and compared with the other vertices in every simplex. The minimum point could not be shown clearly due to clustering of small simplexes near the minimum.

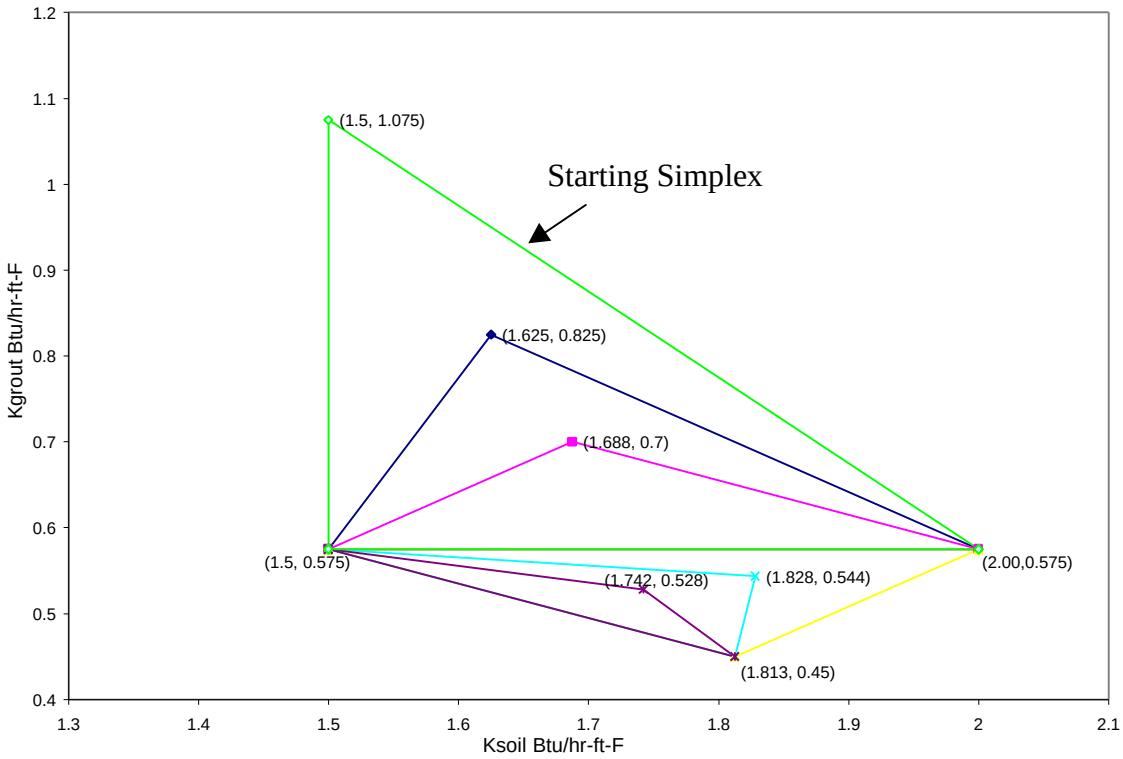

Figure 4-2. Movement of Simplex towards the minimum for SiteA5_93

4.2. Results from O'Neill's implementation of Nelder Mead Simplex

As mentioned earlier, this method is same as Nelder Mead Simplex except that the convergence criterion is different. Also, this algorithm restarts with a smaller step size if the small excursions at the converged point in the coordinate directions show a lower objective function value than the current minimum. A summary of results from this method is shown in Table 4-3.

In some cases, the algorithm does not restart since the small excursions don't provide the objective function value less than the previously found minimum. Figure 4-3 shows the small excursions in the coordinate directions.

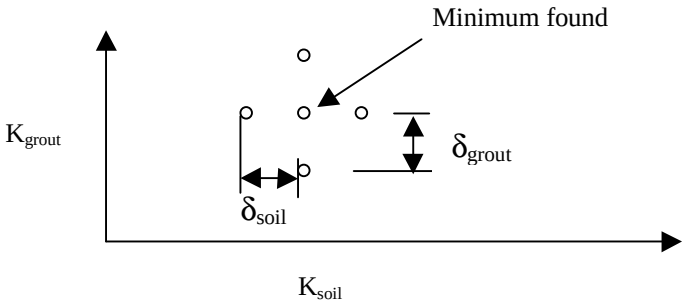

Figure 4-3. Small excursions in both the coordinate directions

Table 4-3. Summary of results for O'Neill's Simplex Algorithm for 50 hours of data

Site	Ksoil	Kgrout	Estimated Standard Deviation	Function Evaluations
SiteA1_98	1.368	0.562	0.25764	54
SiteA2_114	1.450	0.866	0.22356	59
SiteA2_170	1.432	1.158	0.17610	83
SiteA5_93	1.426	0.585	0.15725	47
SiteA6_240	1.317	0.782	0.09832	74
Chickasha	1.478	0.764	0.15523	58
Weatherford	1.634	1.193	0.12062	55

This algorithm was started with small simplex (with step size = 0.05) in both the K_{soil} and K_{grout} directions to see the effects of different starting simplexes. The summary of results is shown in Table 4-4. It is observed that the number of objective function evaluations is, in some cases, less than the 0.5 step size case and more in some other cases without any specific order.

When this algorithm runs, a log file is created which keeps track of the Simplex vertices. When these vertices are plotted, it was observed that since the valley of the objective function is flat, the simplex rolls down the valley and lot of objective function

values are evaluated in the flat part. Hence, to make the valley deeper, the sum of the fourth power of the error values was calculated and minimized, or

$$\text{ERROR} = \sum_{n=1}^{N} \left(T_{\text{experiment al, n}} - T_{\text{numerical, n}}\right)^4 \qquad \textbf{(4-2)}$$

Table 4-5 shows a summary of the results thus obtained. It is observed that soil and grout conductivity values converge to a different number other than their true minimum values.

Table 4-4. Summary of results for O'Neill's Simplex algorithm for step size=0.05 in both K_{soil} and K_{grout} directions

Site	Ksoil	Kgrout	Estimated Standard Deviation	Function Evaluations
SiteA1_98	1.371	0.561	0.25762	73
SiteA2_114	1.459	0.863	0.22390	56
SiteA2_170	1.416	1.177	0.17654	69
SiteA5_93	1.443	0.577	0.15877	38
SiteA6_240	1.312	0.782	0.09844	66
Chickasha	1.496	0.751	0.15575	45
Weatherford	1.640	1.191	0.12064	49

Table 4-5. Summary of results from O'Neill's Simplex Algorithm for the sum of the fourth power of the error for all the test sites

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations
SiteA1_98	1.251	8.557	0.609	0.48034	84
SiteA2_114	1.335	7.903	0.978	0.37408	50
SiteA2_170	1.282	10.471	1.420	0.42543	64
SiteA5_93	1.385	2.859	0.604	0.19956	98
SiteA6_240	1.316	0.081	0.781	0.13809	43
Chickasha	1.547	-4.639	0.713	0.34211	40
Weatherford	1.596	2.342	1.214	0.24472	46

Other exponents of the error, 0.8, 1.5, 1.8, were also tried and the results did not converge to the true parameter values. For comparison purposes, Estimated Standard Deviation was calculated in each case and then compared with true minimum. Figure 4-4 shows the exponent of the error with respect to the Estimated Standard Deviations for four sites. It can be concluded that the Estimated Standard Deviation is minimum for the case of exponent of 2. Other cases provide higher Estimated Standard Deviation than the minimum.

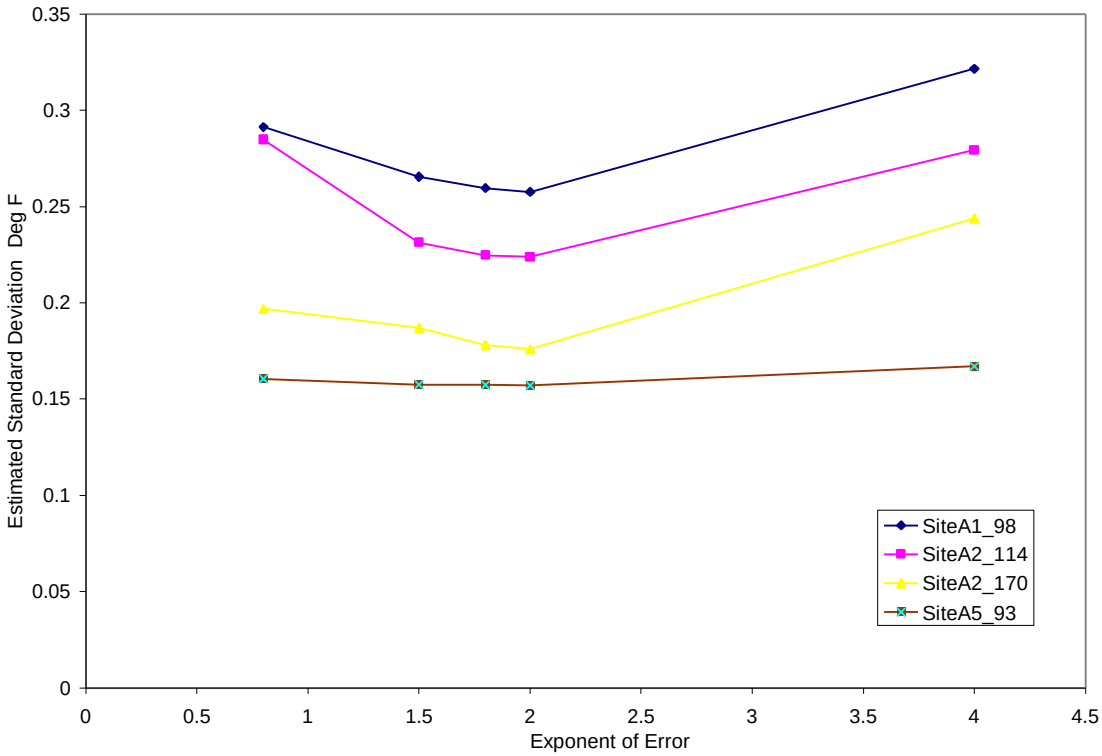

Figure 4-4. Estimated Standard Deviation vs. Exponent of error

4.3. Results from Box's Complex method

The algorithm of Box's method is similar to Nelder Mead Simplex except that it has 2*ndim+1 vertices instead of ndim+1 vertices. Box's method combines the expansion and reflection steps into one by reflecting the vertex by an amount α greater than 1. It then contracts if the objective function value comes out to be more than the maximum objective function value of the complex. The parameters that can be varied are the value of α, the value of α_{low} and the number of vertices which can be higher or lower than 2*ndim+1. Table 4-6 shows the comparison of results for SiteA1_98, for 50 hours of data with different values of parameters.

It is observed that if the value of α_{low} is changed from 0.000001 to 0.001, the number of function evaluations get reduced by about 6%, without effecting the results up to 3 significant digits in the conductivity values and SSQERR value. This is because when the contraction about the centroid is tried, 0.001 times the distance between the

reflected point and the centroid is considered sufficiently close. Going very close to the centroid of the complex is not that significant. Hence, a value of α_{low} = 0.001 is sufficient. As the value of α is changed, for some values of α, parameter values did not converge to the right numbers (especially where α values were less than 1) for six vertices in the complex. When the number of vertices in the complex were reduced from 6 to 5, the number of function evaluations decreased by 36% for other parameters being constant.

Table 4-6. Comparison of results from Box's Complex method for SiteA1_98 with different parameter values

Alpha	$Alpha_{low}$	Ksoil	Kgrout	Estimated Standard Deviation	Function Evaluations	No of Vertices
0.6	**0.001**	**1.560**	**0.518**	**0.36987**	**74**	**6**
0.8	0.001	1.389	0.561	0.27417	85	5
0.8	**0.001**	**1.463**	**0.529**	**0.34738**	**83**	**4**
0.9	0.001	1.378	0.568	0.27265	128	6
1	0.000001	1.359	0.573	0.27341	167	6
1.1	0.001	1.384	0.567	0.27315	125	6
1.3	0.000001	1.374	0.568	0.27254	192	6
1.3	0.001	1.374	0.568	0.27254	182	6
1.3	0.001	1.381	0.570	0.27317	123	5
1.6	0.000001	1.381	0.567	0.27286	241	6

Table 4-7 shows the results for other sites for the best set of parameters from Table 4-6 (α=0.8, α_{low}=0.001, generation=5).

Table 4-7. Results from Box's Complex method for all the sites with the best set of parameter values

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to True Minimum
SiteA1_98	1.389	-1.535	0.561	0.27417	85	Yes
SiteA2_114	1.886	-30.067	0.621	0.73046	140	No
SiteA2_170	2.000	-39.665	0.760	0.46847	74	No
SiteA5_93	1.463	-2.574	0.569	0.16151	297	Yes
SiteA6_240	1.909	-44.925	0.616	0.50817	353	No
Chickasha	1.789	-21.009	0.598	0.29678	392	No
Weatherford	2.000	-22.399	0.974	0.27314	158	No

This method does not always converge to the true parameter values for reflection coefficient less than 1. Different reflection coefficients were tried and Table 4-8 shows that the value of the parameters converged in all cases for reflection coefficient 1.3 which is greater than 1. In fact, it can be concluded from Table 4-6 that the complex always converges for the values of reflection coefficient greater than or equal to one. Hence, the reflection coefficient with a value greater than 1 is required to guarantee that the complex will converge to true parameter values, although the complex might converge for lower value of reflection coefficient too.

Figure 4-5 shows the reflection process for reflection coefficient of 0.8. The vertex corresponding to the worst objective function value in the complex ABCDE is B, which is reflected to produce vertex B'. The new complex is ACDEB'. The size of the complex ACDEB' is less than the size of the complex ABCDE because vertex B is reflected with reflection coefficient of 0.8 which is less than 1. If this process is continued further, the size of the complex will keep reducing. There is a possibility that the final complex eventually becomes so small that the objective function values at all the

vertices fall within the user-specified tolerance without converging to the true minimum. Hence, in some cases the complex may not converge to the true minimum for reflection coefficient less than one.

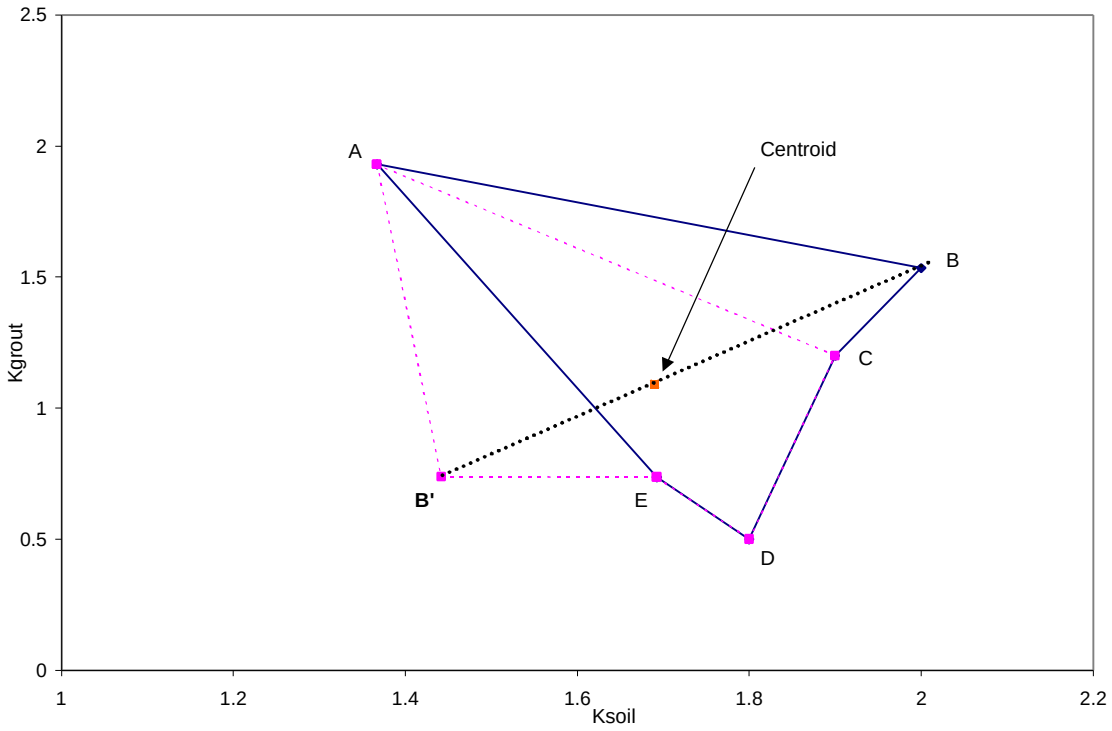

Figure 4-5. Reflection of the worst point for reflection coefficient of 0.8

Table 4-8. Results from Box's Complex method for reflection coefficient of 1.3

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to True Minimum
SiteA1_98	1.381	-0.924	0.570	0.27317	123	Yes
SiteA2_114	1.455	-0.313	0.865	0.22158	116	Yes
SiteA2_170	1.437	-0.346	1.132	0.17631	154	Yes
SiteA5_93	1.432	-0.454	0.584	0.15720	168	Yes
SiteA6_240	1.324	-0.525	0.783	0.10153	155	Yes
Chickasha	1.476	0.166	0.766	0.15530	170	Yes
Weatherford	1.635	-0.087	1.194	0.12068	174	Yes

4.4. Results from Hooke and Jeeves method

Hooke and Jeeves method is well known for problems where the domain is a steep, turning valley. The advantage of this method is that it takes larger steps in the favorable direction called pattern moves and if the valley is turning, the direction also keeps turning due to the exploratory steps.

Since our problem domain is also a deep turning valley, we implemented this method to see if we can reduce the time taken to minimize the objective function or the number of objective function evaluations. Table 4-9 shows the results for all the sites using this method. Figure 4-6 shows the steps taken by this method for SiteA1_98. Results for all the sites converge to their true minimum values. But if we compare it with O'Neill's Simplex method, this method reaches the minimum by evaluating more number of objective function values than O'Neill's Simplex method for all the test sites.

The reason behind the higher number of objective function evaluations is the small excursions, which do not result in a favorable objective function value. Another source might be evaluating the objective function values when step size is reduced in all the coordinate directions. All 2*ndim+1 objective function evaluations of the previous step are wasted because no favorable vertex is obtained during this step.

Table 4-9. Results from Hooke and Jeeves method for all the sites

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to True Minimum
SiteA1_98	1.402	-2.454	0.550	0.25978	113	Yes
SiteA2_114	1.453	-0.215	0.866	0.22368	150	Yes
SiteA2_170	1.451	-1.298	1.125	0.17746	126	Yes
SiteA5_93	1.457	-2.204	0.576	0.15952	79	Yes
SiteA6_240	1.326	-0.667	0.777	0.09910	154	Yes
Chickasha	1.489	-0.748	0.755	0.15535	113	Yes
Weatherford	1.624	0.623	1.198	0.12116	149	Yes

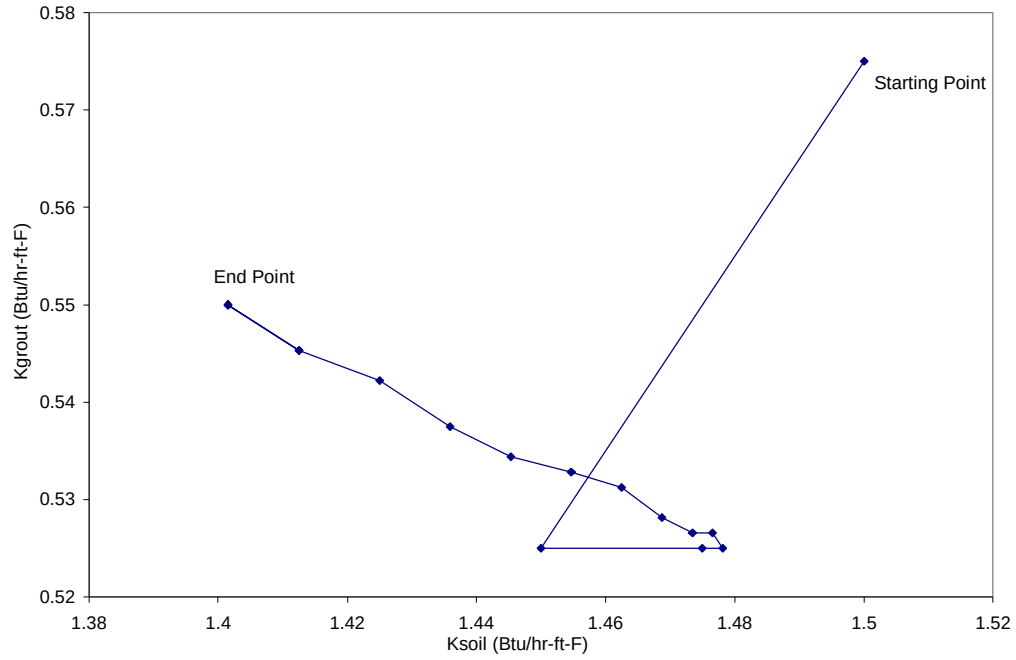

Figure 4-6. Steps taken by Hooke and Jeeves method for SiteA1_98

4.5. Results from Powell's method

Powell's method is supposed to be very efficient in minimizing the sum of the square of the error. This method consists of line minimization in the coordinate directions

in the beginning and then generates directions depending upon the maximum decrease in the objective function value along a particular direction. This process is repeated until the minimum of the valley of the objective function is reached. Results from this method have been summarized in Table 4-10.

It can be observed from Table 4-10 that the parameter values do not necessarily reach the true minimum for all the test cases. For some cases the number of objective function evaluations are quite large (e.g. SiteA2_170, 561 objective function evaluations) and still the method does not reach the true minimum. Hence, it is not guaranteed with this method, that the minimum reached will be the true minimum after one pass of the method, even after large number of objective function evaluations. The reason for not converging to true minimum can be attributed to stopping prematurely at a point where the convergence criteria is satisfied. Since the convergence criteria can not be improved, the only possibility of obtaining a converged value is by restarting the method with previous best point. This works for some cases and we obtained a converged value of parameters close to the optimum. But the number of objective function evaluations is unusually high hence a restart was not performed for all the cases. According to Powell, it takes more than n*(n+1) line minimization for a non-quadratic objective function to reach minimum, where n is the number of coordinate directions. That is why this method takes very large number of objective function evaluations.

Table 4-10. Results from Powell's method for all the Sites

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to True Minimum
SiteA1_98	1.334	2.489	0.574	0.26255	124	Yes
SiteA2_114	2.186	-50.730	0.583	0.70103	159	No
SiteA2_170	1.576	-10.039	0.961	0.24088	561	No
SiteA5_93	1.458	-2.234	0.575	0.15944	123	Yes
SiteA6_240	1.553	-17.911	0.684	0.27855	319	No
Chickasha	1.494	-1.070	0.752	0.15563	462	Yes
Weatherford	4.506	-175.781	0.608	0.93958	226	No

Figure 4-7 shows the steps taken by Powell's method for the test site SiteA2_170.

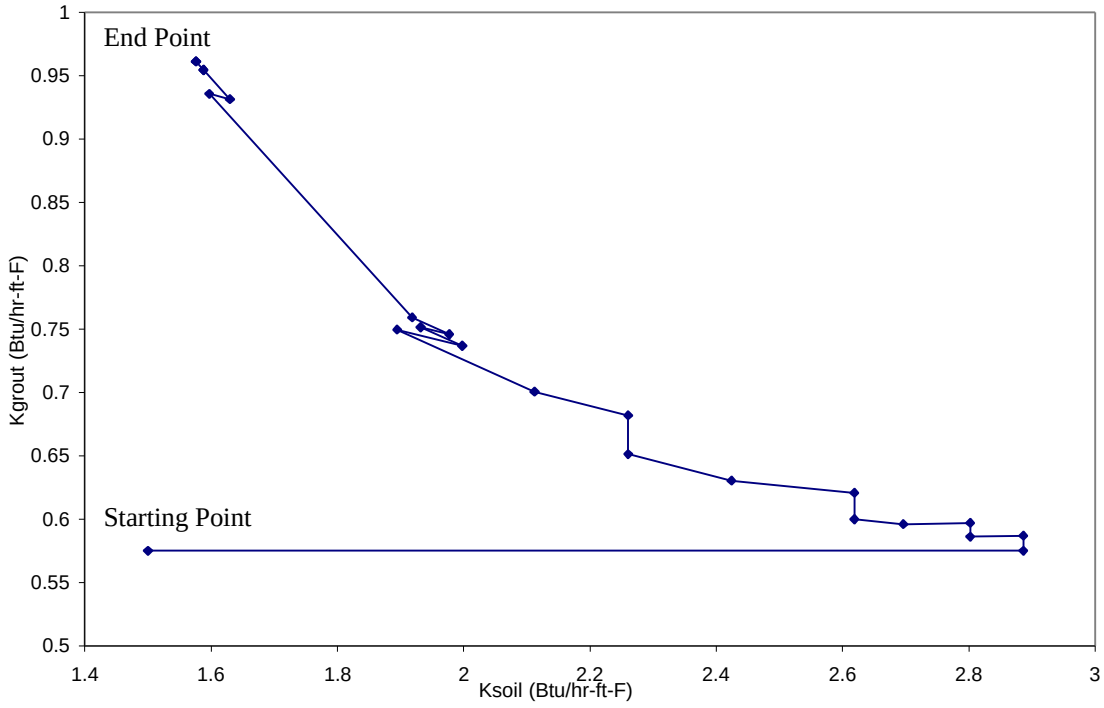

Figure 4-7. Steps taken by Powell's method for SiteA2_170

4.6. Results from BFGS method

BFGS method is a well-known gradient based method for minimization. The inverse of Hessian matrix is updated with some approximation rather than evaluating the Hessian matrix whose components are second partial derivative of objective function. Since, it is already difficult to calculate the gradients due to uncertainty in step size, it will even be more difficult to calculate the second partial derivatives. Hence, this method is better than some other Quasi-Newton methods that require accurate computation of inverse of Hessian matrix. Once the direction is computed in which the objective function has to be minimized, golden section search is used to minimize the objective function in that particular direction. Table 4-11 summarizes the results for all the sites for this method.

Table 4-11. A summary of results from BFGS method for all the sites

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations*	Close to True Minimum
SiteA1_98	1.466	-7.147	0.531	0.29286	672	No
SiteA2_114	1.484	-2.349	0.841	0.22771	419	Yes
SiteA2_170	1.434	-0.153	1.149	0.17753	988	Yes
SiteA5_93	1.438	-0.809	0.580	0.15949	412	Yes
SiteA6_240	1.443	-9.560	0.736	0.16581	326	No
Chickasha	1.759	-19.021	0.600	0.27369	498	No
Weatherford	2.551	-56.147	0.797	0.52859	220	No

* Program was stopped and the convergence criteria was not satisfied

From Table 4-11, it is observed that this method does not always converge to the true minimum even after a large number of objective function evaluations. For the cases this method provide parameter values close to the minimum but the program did not stop

due to convergence and was stopped by us, it took a very large number of objective function evaluations as compared to other methods like Nelder Mead Simplex and O'Neill Simplex.

The reason for not converging to true minimum can be attributed to inaccurate computation of the gradient of the objective function due to numerical errors. Susceptibility to divergence is caused if step lengths during line minimization are not found accurately. Also the inverse of Hessian matrix looses its positive definiteness after a lot of updates which is only approximate. Figure 4-8 shows the steps taken by this method for SiteA6_240. The initial direction takes the starting point directly into the valley but the future steps just hover around the minimum without satisfying the convergence criteria.

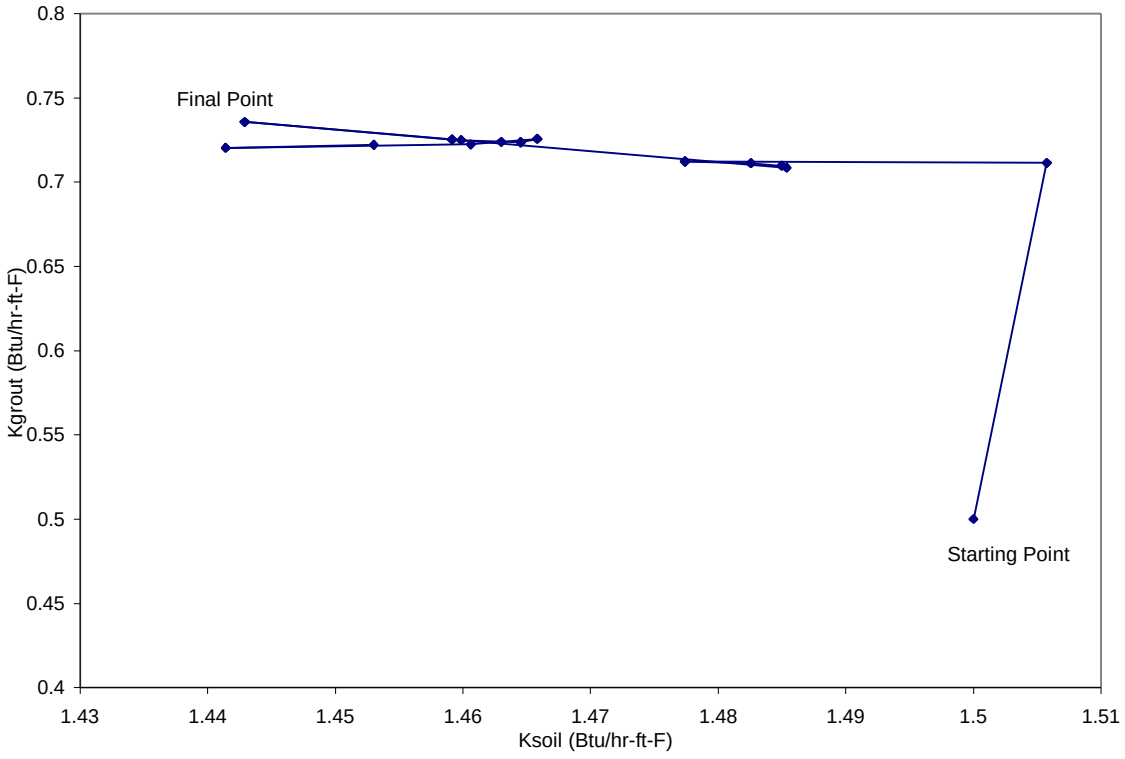

Figure 4-8. Steps taken by BFGS method for SiteA6_240

4.7. Results from Genetic Algorithms

This stochastic based method is a global search method in a domain. Mutation and Crossover are the two main features of these classes of algorithms. This search uses random number generation. David L. Carroll (1996), Ph.D. at the University of Illinois, Urbana-Champaign, developed the FORTRAN code for this algorithm. Table 4-12 shows the results for all the sites using genetic algorithms for our problem.

From Table 4-12 it can be concluded that for maximum of 100 generations, genetic algorithms does not always reach true minimum but *it is close to the minimum*. So, it is expected that if we run the program for more than 100 generations, it might reach the true minimum. This has been verified for some test cases. We chose 100 generations due to the time constraint because there is no specific convergence criteria in these kinds of algorithms that use random numbers. Hence, the minimum obtained is optimum only in probabilistic sense. These methods are suitable only where the objective function evaluations are not the performance criteria and obtaining the global optimum parameters are more important.

Hence this method is not recommended for this type of problem. Deterministic methods will be more appropriate.

Table 4-12. A summary of results from Genetic Algorithms for all the sites for maximum of 100 generations

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to true minimum
SiteA1_98	1.363	0.388	0.563	0.26040	500	Yes
SiteA2_114	1.478	-1.940	0.850	0.22655	500	Yes
SiteA2_170	1.480	-3.355	1.084	0.18363	500	No
SiteA5_93	1.431	-0.352	0.584	0.15983	500	Yes
SiteA6_240	1.411	-7.117	0.739	0.14521	500	No
Chickasha	1.359	8.079	0.899	0.20967	500	No
Weatherford	1.749	-7.019	1.095	0.15066	500	No

4.8. Results from Quadratic fit method

This method is based on line minimization and minimization along a quadratic curve. The line minimization, which is performed by fixing K_{soil}, produces a point that has minimum objective function value along the K_{grout} direction. Three-line minimization at three different K_{soil} values produce three different points on the floor of the valley. A quadratic polynomial is fit using those three points and the objective function is minimized along the quadratic. Table 4-13 shows the results for all the sites using this method. The main reason of not converging to true optimum parameter values is that the fitting of quadratic may not pass through the optimum point for all the test cases and valley is not truly quadratic. It can provide only a set of parameters that are close to the optimum. Hence, the errors encountered range from less than 1% to as high as 13% in the soil conductivity values. Number of objective function evaluations present a great advantage of this method and this method might give a set of parameters close to optimum if the valley turns out to be close to quadratic for any test case. A further investigation into this method is required to fully realize the advantages of this method

and variations of this method. No recommendation on whether to use this method or not can be made at this stage.

Table 4-13. Results of all the sites using Quadratic fit method

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to True Minimum
Site A1_98	1.416	-3.539	0.548	0.27136	26	No
Site A2_114	1.444	0.395	0.841	0.33753	24	Yes
Site A2_170	1.489	-4.005	1.074	0.18727	22	No
Site A5_93	1.556	-9.097	0.550	0.20628	27	No
Site A6_240	1.500	-13.895	0.720	0.21934	25	No
Chickasha	1.427	3.447	0.779	0.23929	23	No
Weatherford	1.496	8.450	1.357	0.17969	22	No

4.9. Results from O'Neill's implementation of Nelder Mead Simplex method with exploratory search

O'Neill's Simplex method provides good results in the sense that it converges for all initial guesses and even for small starting simplexes. But if we can start with an initial guess that is close to the minimum, the objective function evaluations can be reduced to some extent. Hence, an exploratory search is performed which is based on the line source method. This exploratory search gives a better starting point than just any arbitrary guess. Since we have a better guess of parameters, we can start with a smaller simplex that we hope will reduce the number of objective function evaluations and which in turn reduce the time taken to minimize the objective function. Table 4-14 shows the results of just the exploratory search for all the sites.

From Table 4-14, it can be seen that the values of K_{soil} and K_{grout} are closer to the optimum values than our starting guess of K_{soil} =1.5 Btu/hr-ft-F and K_{grout} = 0.575 Btu/hr-ft-F. It is also observed that the average deviation from the optimum values is between 0.1–0.15 Btu/hr-ft-F except for one case. Hence, we can take a step size of 0.1 in both the parameter directions to construct the simplex.

Table 4-14. Results of exploratory search for all the sites

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Exploratory Function Evaluations
SiteA1_98	1.127	17.642	0.651	0.67879	69
SiteA2_114	1.354	6.655	0.797	0.40956	59
SiteA2_170	1.352	5.613	0.941	0.27989	45
SiteA5_93	1.264	11.370	0.632	0.37550	47
SiteA6_240	2.182	-65.674	0.625	1.12559	107
Chickasha	1.351	8.578	0.879	0.15427	67
Weatherford	2.057	-25.862	0.918	0.61358	114

Table 4-15 provides the results for all the sites by O'Neill's Simplex method using exploratory search results as the starting guesses. The average number of objective function evaluations without exploratory search and with exploratory search is 62 and 49 respectively. The average is calculated based on 7 sites used to compare the results. Hence, we can say that there is an improvement of about 20% in the number of objective function evaluations.

Table 4-15. Results of O'Neill Simplex method with exploratory search for all the sites

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation	Function Evaluations	Converged to True Minimum
SiteA1_98	1.363	0.357	0.561	0.25731	60	Yes
SiteA2_114	1.469	-1.343	0.853	0.22592	36	Yes
SiteA2_170	1.441	-0.658	1.147	0.17574	52	Yes
SiteA5_93	1.430	-0.256	0.585	0.15728	40	Yes
SiteA6_240	1.309	0.618	0.786	0.09875	49	Yes
Chickasha	1.485	-0.466	0.759	0.15520	57	Yes
Weatherford	1.636	-0.130	1.190	0.12083	49	Yes

4.10. Comparison of results

The basis of comparing the results from all the methods is the number of objective function evaluations. This is because the number of objective function evaluations is directly proportional to the time taken to obtain the optimum parameter values. All the methods are compared against Nelder Mead Simplex / O'Neill Simplex method, which consistently provides the best solution. Another form of comparison is made between using an exploratory search and not using an exploratory search for the same method. Genetic Algorithms do not require a starting guess, hence exploratory search can not be applied for this method.

Table 4-16 shows the average number of objective function evaluations for all the methods used. For the cases in which the method does not converge to the true minimum it is assumed that a restart of the method might allow it to converge. However, the number of objective function evaluations will only add to the current average value. It can also be concluded from Table 4-16 that the O'Neill Simplex method takes the minimum average number of objective function evaluations. Although the Quadratic fit method takes less objective function evaluations compared to O'Neill Simplex method, it

does not converge to true minimum in all cases. Hence, the O'Neill Simplex method is considered the best optimization method for our problem of steep, turning valley.

Table 4-16[2]. Comparison of methods based on average number of objective function evaluations

Method	Average Function Evaluations
Nelder Mead Simplex	79
O'Neill Simplex	62
Box Complex	152
Powell*	> 292
Hooke and Jeeves	127
BFGS*	> 505
Genetic Algorithm*	> 500
Quadratic Fit*	25

Nelder Mead Simplex method and Hooke and Jeeves method are the two potential methods that might be further considered. Hence, these methods are rerun with exploratory search so that they can start with a better initial guess. The results are shown in Table 4-17.

[2] * Indicates that the method does not converge to the true minimum in all the test cases.

Table 4-17. Comparison of results with exploratory search

Method	Average Function Evaluations
Nelder Mead Simplex	68
O'Neill Simplex	49
Hooke and Jeeves	138

From Table 4-16 and Table 4-17 it can be concluded that O'Neill Simplex method with exploratory search produces the best results and minimizes the objective function in an average of 49 objective function evaluations.

Although the Quadratic fit method does not converge to true optimum parameter values, this method may have the potential to reduce the number of objective function evaluations significantly if we use exploratory search to obtain an initial guess of parameters. This option was realized towards the latter part of the project and could not be considered in a greater detail. Hence, a further investigation will establish whether this method has the potential to reduce the number of objective function evaluations or not.

5. Online Parameter Estimation

Off-line parameter estimation has the disadvantage that the temperature and power data is first collected in the field and then taken to the computer to estimate ground thermal properties. This estimation takes some computer time before we can get ground thermal property values. Another disadvantage of off-line parameter estimation is that if there is some error while collecting the data in the field, may it be experimental error, it can not be detected.

Disadvantages of off-line parameter estimation can be overcome by some other means and one of them is online parameter estimation. Online parameter estimation, as the name implies, is a parameter estimation method by which the values of ground thermal properties can be estimated as the data is collected in the field. A graph of parameters can be plotted to see where the values of parameters might be converging. This is very advantageous because if the value of parameters are not changing much (if they are within some tolerance), then the experiment can be stopped. Hence, this provides a better measurement of the length of the experimental test.

Online parameter estimation is also called Sequential Non-Linear Estimation in some literature (Sorenson, 1980). The term sequential refers to updating the previous parameter estimate using the newly collected data sequentially. Two approaches to estimate the parameters are considered here:

- Non-linear Recursive Estimator.
- Application of non-linear optimization method.

Sorenson (1980) and Salsbury (1996) presented Non-linear recursive algorithm for general optimization problems. Jang et al (1986) presented extended Kalman filter approach to account for general nonlinear systems and standard optimization method approach which they call *Horizon approach*. They compared both the approaches for their problem of chemical process.

5.1. Non-linear Recursive Estimator

Salsbury (1996) applied the nonlinear recursive estimator equations for the problem of fault detection in HVAC systems. The nonlinear recursive estimator equations presented are in the following form:

$$e_k = f_k(\boldsymbol{\theta}_{k-1}) - y_k \quad \textbf{(5-1)}$$

$$\boldsymbol{\psi}_k = \nabla f_k(\boldsymbol{\theta}_{k-1}) \quad \textbf{(5-2)}$$

$$\mathbf{P}_k = \left\{ \sum_{i=1}^{k} \boldsymbol{\psi}_i \boldsymbol{\psi}_i^T \right\}^{-1} \quad \textbf{(5-3)}$$

$$\Delta\boldsymbol{\theta}_k = \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1} \quad \textbf{(5-4)}$$

Where, $\boldsymbol{\theta}_k$ is the vector containing parameters at k_{th} step, $\boldsymbol{\psi}_k$ is the gradient of objective function at k_{th} step, **P** contains curvature information and y_k is the experimental value of temperature.

The complete algorithm is stated as:

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} - \mathbf{L}_k \cdot e_k \quad \textbf{(5-5)}$$

$$\mathbf{L}_k = \mathbf{P}_{k-1}\boldsymbol{\psi}_k (I + \boldsymbol{\psi}_k^T \mathbf{P}_{k-1} \boldsymbol{\psi}_k) \quad \textbf{(5-6)}$$

$$\mathbf{P}_k = (I - \mathbf{L}_k \boldsymbol{\psi}_k^T)\mathbf{P}_{k-1} \quad \textbf{(5-7)}$$

$\mathbf{L}_k$ is called gain vector, **P** matrix is the inverse of Hessian matrix and -**y** is the direction of steepest descent according to Salsbury.

The algorithm starts with a positive definite symmetric matrix $\mathbf{P}_0$ of ndim x ndim dimensions. It is taken as Identity matrix for the first step. $\boldsymbol{\theta}_0$ contains an initial guess of parameters. $\boldsymbol{\psi}_k$ is calculated for the next step from equation (5-2) and $\mathbf{L}_k$ is computed from equation (5-6). $\boldsymbol{\theta}_k$ is then updated using equation (5-5) and $\mathbf{P}_k$ is then updated for the next step. This process is repeated until the parameters are converged.

5.2. Application of Non-linear optimization method

The basic idea behind online parameter estimation method is that if we compute the parameter values for M points, can this value be utilized for M+N points where N is the number of additional points collected within the time period of estimation of parameters for M points. This value will certainly be closer to the optimum value instead of any initial guess for the next set of points. It has already been established from off-line parameter estimation that the number of objective function evaluations is reduced if we have an initial guess of parameters which is closer to optimum values of those parameters.

Another issue that is worth addressing is how much information or how many hours of data are needed before updating the parameters. Should we update the parameters for every collected data point or should parameters be updated after a set of data points have been collected?

It mainly depends on the information that is coming from the experiment. If the experimental data is changing slowly with time, probably we do not need to update the parameters for each collected data point. But if the incoming information is changing rapidly with time, we need to update the parameters for each collected point.

An extensive analysis of the data collected from the seven test sites has been done and online parameter estimation has been performed using different strategies. Nelder Mead Simplex algorithm is used to perform the online parameter estimation. Four different alternatives were investigated.

- Construct a starting simplex for a fixed number of data points, increase the number of points by one at each reflection, contraction and expansion, evaluate the average SSQERR and find the best set of parameters at every step.

- Construct a starting simplex for a fixed number of data points, increase the number of points by a fixed number at each reflection, contraction and expansion, evaluate the average SSQERR and find the best set of parameters at every step.

- Start with a fixed number of data points. Get the best set of parameter values by optimizing for a fixed time. Increase the number of points by a fixed number, optimize for a fixed amount of time using the previous set of parameters as initial guess to this set.

- Start with a fixed number of points. Get the optimum parameter values. Then take this optimum value as initial guess for the next set of points collected during that time period. Construct the initial simplex and minimize. In other words, minimize the average SSQERR for a variable number of experimental data points.

5.3. Results from Salsbury's method

Salsbury's method of recursive parameter estimation was applied for our non-linear problem of parameter estimation. This approach yields parameter values that are quite off from the true optimum parameter values obtained from off-line parameter estimation. The main reason for this method to fail to converge appear to be:

- This method requires an initial guess of parameter values to begin with. It is desirable to choose a guess of parameters, which is closer to the true values. This depends upon the experience of the problem at hand. The performance of the algorithm is strongly influenced by the initial guess. Inability to start with a better initial guess might be a reason for the divergence of the method.

- To compute the gradient of SSQERR function, a finite difference approach is used. Since the problem is highly non-linear, the uncertainty in step size in both the parameter directions can cause an inaccurate first derivative of the objective function. The objective function is linearized using Taylor series given in equation (5-8) and

this linear approximation is used to derive the recursive form of equations, which introduces a significant amount of error for highly non-linear problems like ours.

$$f_k(\boldsymbol{q}) \approx f_k(\boldsymbol{q}_0) + [\boldsymbol{q} - \boldsymbol{q}_0]^T \nabla f_k(\boldsymbol{q}_0) \qquad \textbf{(5-8)}$$

5.4. Results from application of optimization method

The four variations in application of off-line optimization method applied to online parameter estimation presented above are analyzed for the seven test sites. For the case of updating the parameters after every data point, since one objective function evaluation takes somewhere between one to three minutes depending on the number of points, the program has to wait till the next data point is collected. In our case, since we already had the data, we assumed that each data point is coming only after 2.5 minutes and ran the simulation for 50 hours of data. Table 5-1 shows the results for all the test sites for this case.

This process of updating the parameters at each collected point does not always lead to the optimum value of parameters as seen by Table 5-1. The errors in the conductivity values are as high as 33%, as compared to the values obtained from off-line Nelder Mead Simplex method. The reason might be that if the starting guess is far away from the minimum point, then the chances are more that it will converge at some point other than the optimum.

Table 5-1. Comparison of results for six test-sites for updating the parameter after every collected data point

Site	Ksoil	Kgrout	% Ksoil Difference from True Minimum	Estimated Standard Deviation
SiteA1_98	0.921	0.695	-32.643	0.35612
SiteA2_114	1.289	0.977	-11.712	0.19322
SiteA2_170	1.427	1.156	0.005	0.17539
SiteA5_93	1.168	0.650	-18.587	0.08935
SiteA6_240	1.279	0.795	-3.003	0.06387
Chickasha	1.843	0.667	23.657	0.28906
Weatherford	1.310	1.393	-20.028	0.13172

Taking a fixed number of points for updating the parameters at each reflection, contraction and expansion process also has the same problem as updating the parameters at each collected points has. Table 5-2 shows the comparison of results for this case where the number of collected points is equal to 15. Table 5-2 also shows the percent difference of K_{soil} and K_{grout} from off-line parameter estimation. It is observed that the percentage error can be as high as 10-12% in both soil conductivity and grout conductivity after 50 hours of test. Hence, the method does not lead to the true optimum point in some cases.

Table 5-2. Comparison of results for six test-sites for updating the parameter after every 15 collected data points

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation
SiteA1_98	1.336	-2.269	0.571	0.26544
SiteA2_114	1.305	-10.662	0.962	0.19972
SiteA2_170	1.411	-1.093	1.184	0.17671
SiteA5_93	1.329	-7.363	0.612	0.12200
SiteA6_240	1.300	-1.386	0.785	0.08352
Chickasha	1.484	-0.415	0.759	0.15644
Weatherford	1.550	-5.368	1.254	0.13788

The previous two methods don't necessarily lead to the minimum point. Hence, a better method is needed, which can provide a better update of the values of the parameters at each step. A method, which takes a fixed number of collected data points and minimizes with those set of points till we receive the next set of data points will be more suitable. In our case, we tried 15 experimental data points to be the fixed number of points for which the minimization process continues. The time for the process is fixed since we have fixed the number of data points and each data point is collected after 2.5 minutes. Hence, total time will be 15*2.5=37.5 minutes before the minimization starts for the next set of points. This gives us a better estimate of parameters since at each step we get a value of parameters which is closer to optimum for those set of points. A new simplex is constructed after every 37.5 minutes and minimization is done during that time starting with the best guess from the previous step. Table 5-3 gives a comparison of the results obtained from this method at the end of 50 hours of test length. Figure 5-1 shows how the parameters vary as the experimental data is collected for SiteA1_98.

Table 5-3. Comparison of results for updating parameters during a fixed amount of time

Site	Ksoil	% Ksoil Difference from True Minimum	Kgrout	Estimated Standard Deviation
SiteA1_98	1.364	-0.164	0.562	0.26117
SiteA2_114	1.450	-0.728	0.868	0.22820
SiteA2_170	1.424	-0.189	1.161	0.17764
SiteA5_93	1.431	-0.259	0.584	0.16118
SiteA6_240	1.318	-0.034	0.781	0.10373
Chickasha	1.476	-0.980	0.769	0.15442

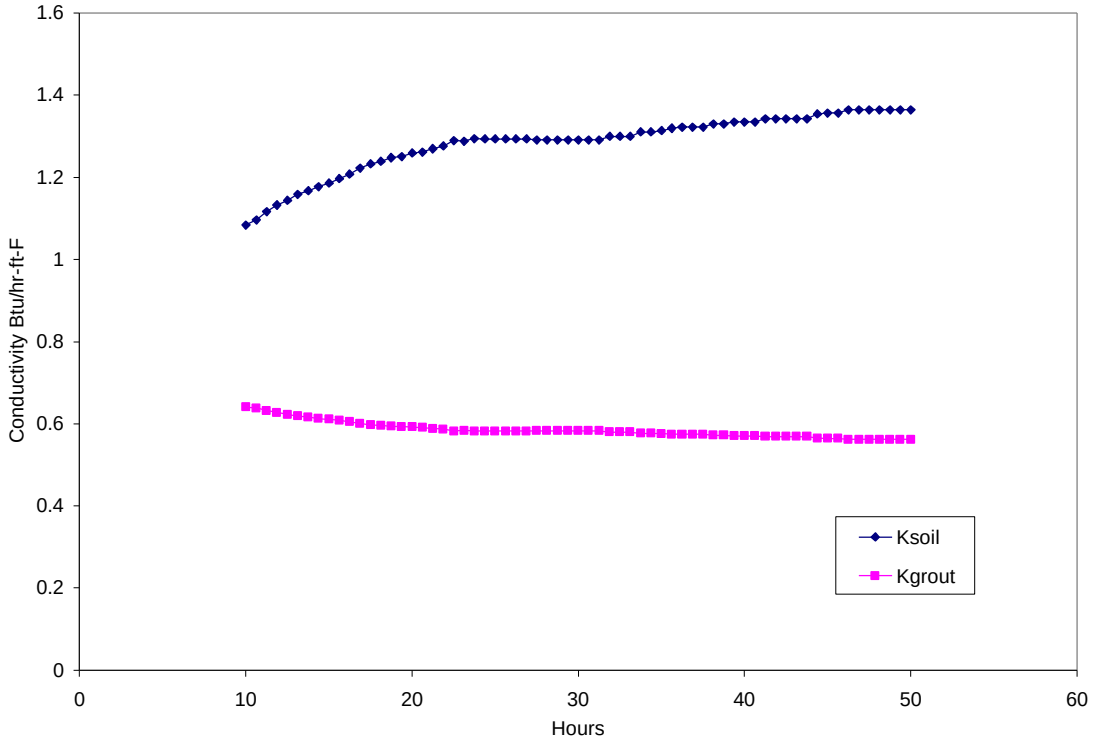

Figure 5-1. Variation of updated parameters for a fixed interval of optimization time as the test for SiteA1_98 progresses

If we want to find out whether the experiment should be stopped or continued to collect more data, we have to look at the plot which shows the variation of the parameters. If the parameters were not changing significantly then we would like to stop the experiment. The method which updates parameters for a fixed interval of time will not give us optimum value at each step since the optimization process may not have reached the true optimum within that interval. With this we can not find out whether we should stop the experiment or not since we are not dealing with optimum parameter values.

To avoid this and get an optimum value at each step, we start with 5 hours of data. Construct the simplex, minimize using Nelder Mead Simplex and get the optimum set of parameters. This process takes some time and during that time we must have collected some data from the experiment. We take those data points, add them in the previous set and again minimize for that set of data points starting with the optimum value of the parameters from the previous step. Table 5-4 shows the comparison of results obtained from this method and off-line Nelder Mead Simplex method. The difference between the Nelder Mead Simplex values and values obtained from online parameter estimation are within 1% for 50 hours of data. The variation of parameters is shown in Figure 5-2 for SiteA1_98, Figure 5-3 for SiteA2_114, Figure 5-4 for SiteA2_170, Figure 5-5 for SiteA5_93, Figure 5-6 for SiteA6_240, Figure 5-7 for Chickasha, Figure 5-8 for Weatherford.

Table 5-4. Comparison of results obtained from minimization at each step and off-line Nelder Mead Simplex method

Site	Ksoil	Ksoil Obtained from NM Simplex	% Ksoil Difference from True Minimum	Kgrout
SiteA1_98	1.373	1.367	0.472	0.558
SiteA2_114	1.453	1.460	-0.476	0.865
SiteA2_170	1.432	1.427	0.355	1.153
SiteA5_93	1.435	1.435	0.027	0.584
SiteA6_240	1.323	1.319	0.319	0.779
Chickasha	1.485	1.490	-0.364	0.760
Weatherford	1.634	1.638	-0.238	1.194

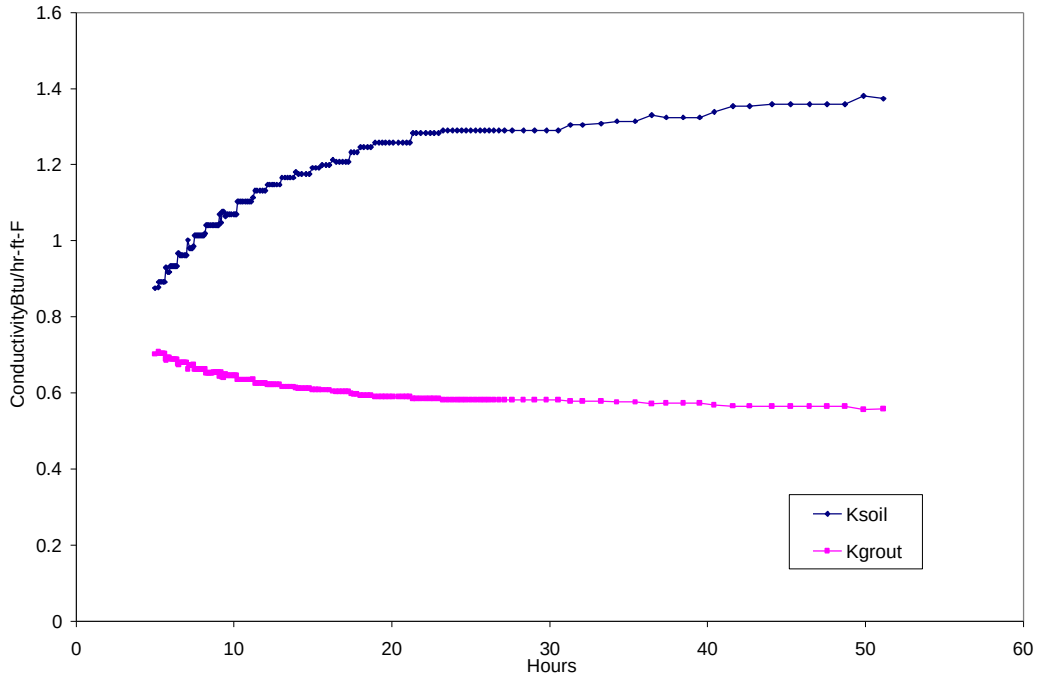

Figure 5-2. Variation of estimated parameters for minimization at each data set for SiteA1_98

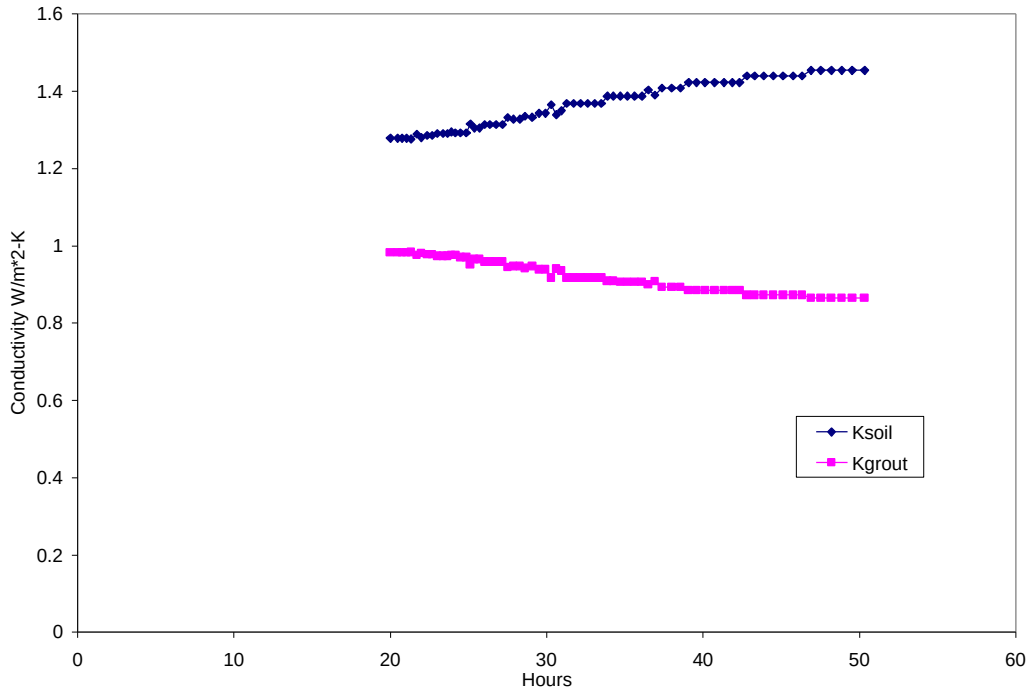

Figure 5-3. Variation of estimated parameters for minimization at each data set for SiteA2_114

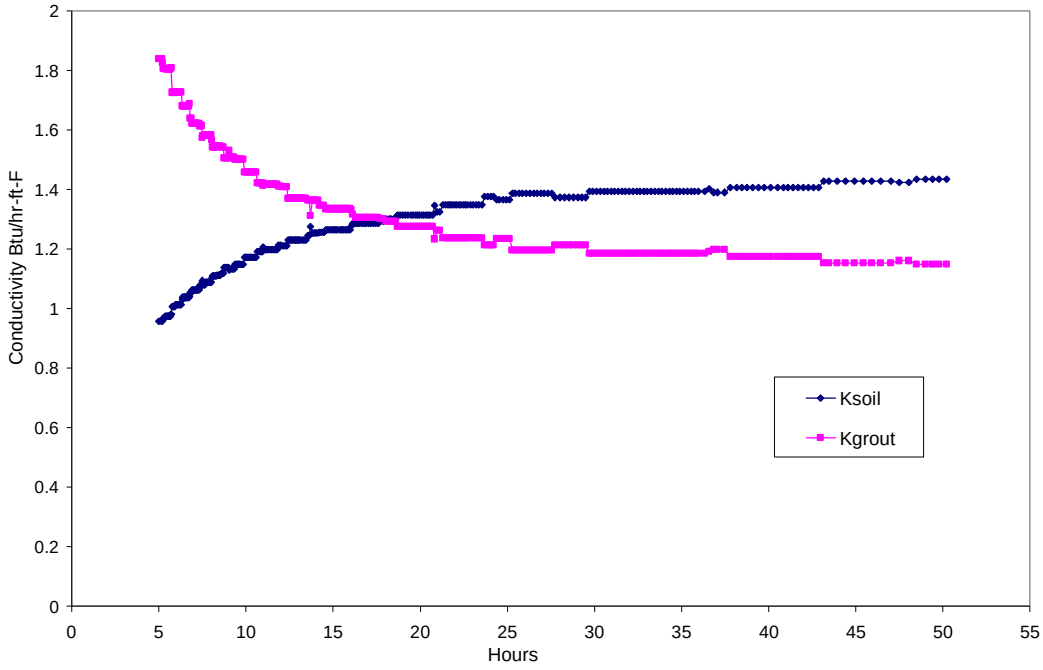

Figure 5-4. Variation of estimated parameters for minimization at each data set for SiteA2_170

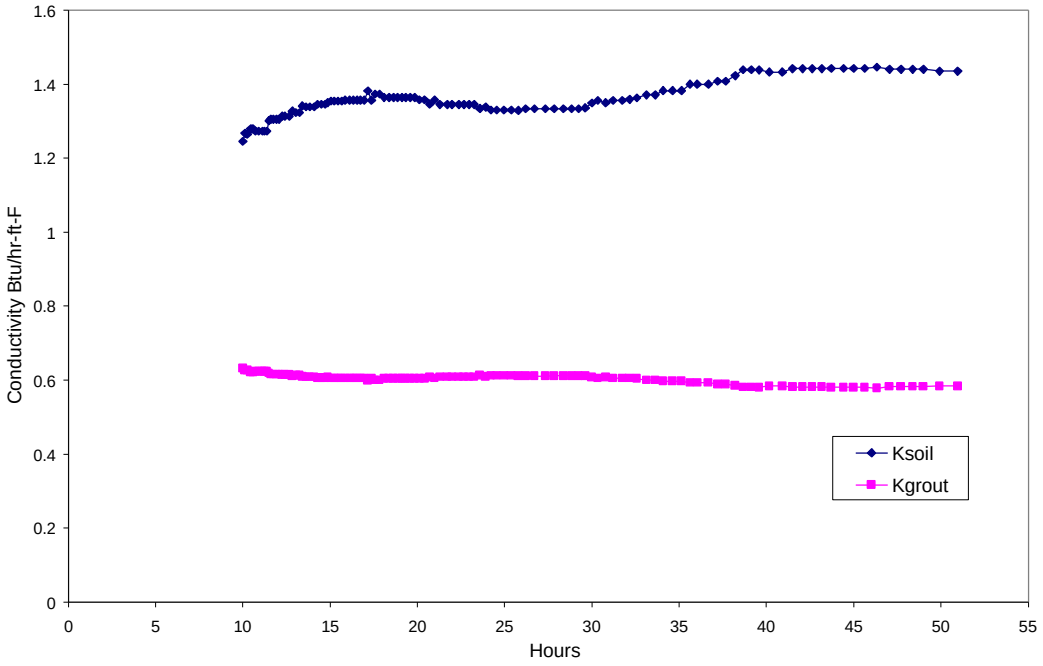

Figure 5-5. Variation of estimated parameters for minimization at each data set for SiteA5_93

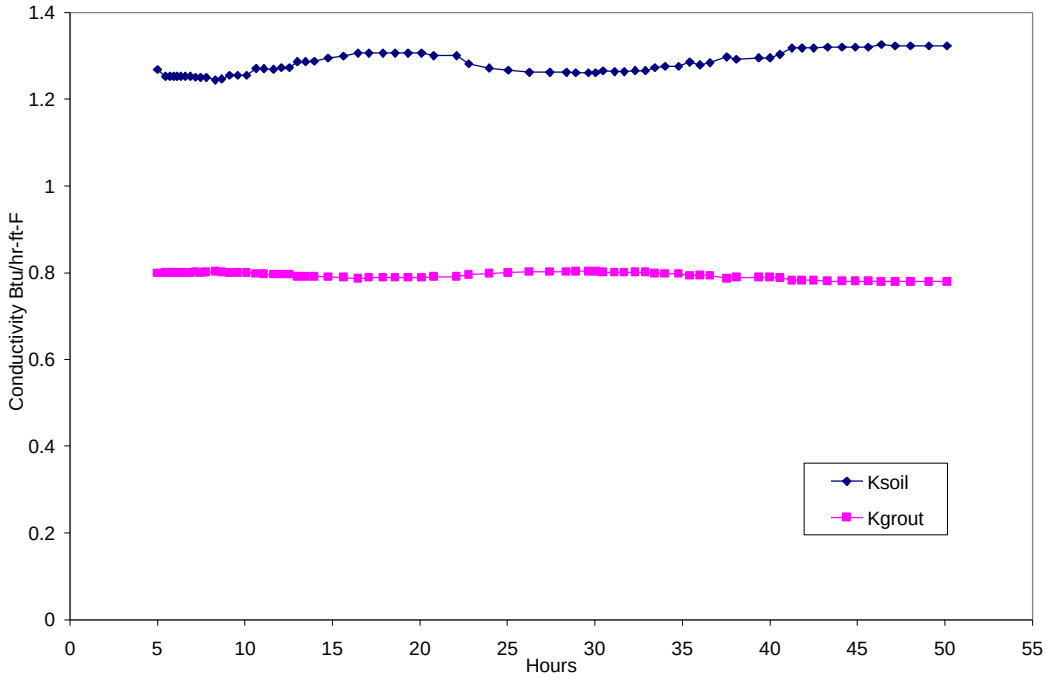

Figure 5-6. Variation of estimated parameters for minimization at each data set for SiteA6_240

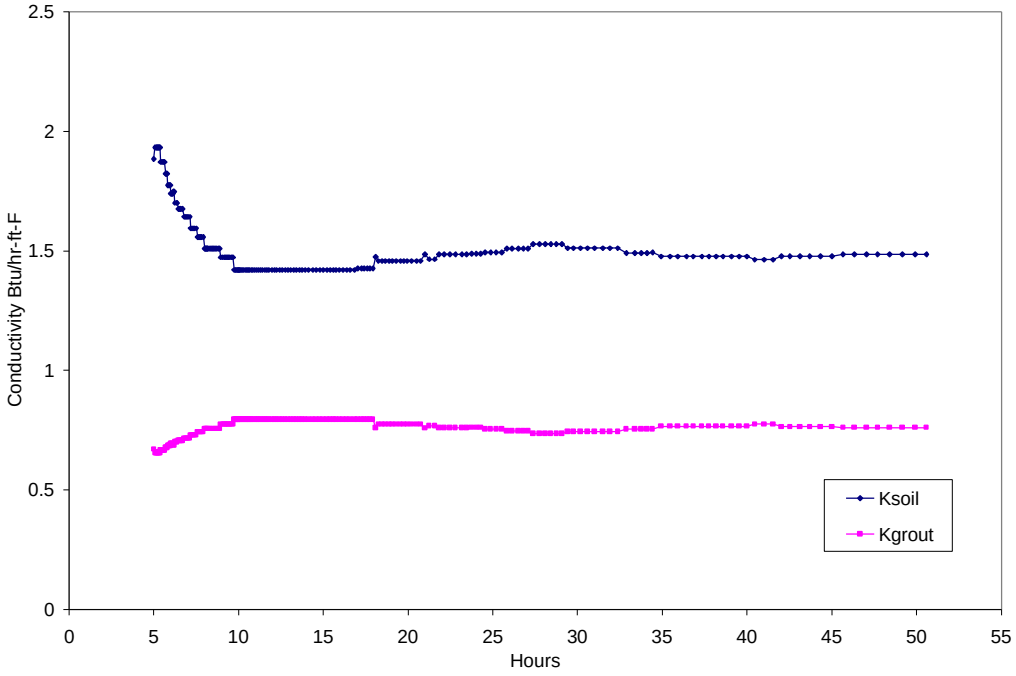

Figure 5-7. Variation of estimated parameters for minimization at each data set for Chickasha data

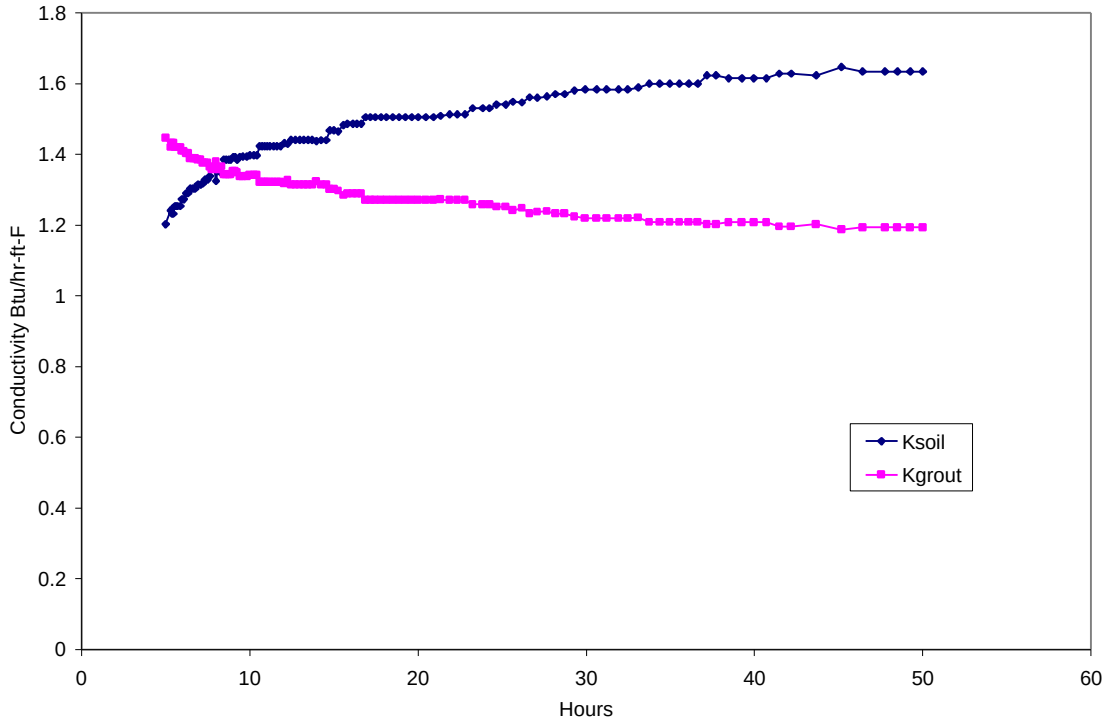

Figure 5-8. Variation of estimated parameters for minimization at each data set for Weatherford data

5.5. Determination of the length of the experimental test

As the experiment is performed and parameters are updated, one is interested in knowing how long the test has to be run to get the set of parameter values close to the true minimum within some error band. The analysis of online parameter estimation results obtained from optimization algorithm for variable point has been performed for the seven test sites. The convergence criterion is if there is a variation of less than X% in the parameter values for Y hours of collected data, then the experiment is stopped and the length of the experiment obtained is sufficient for practical purposes. The variation is calculated as:

$$X = \left(\sqrt{\frac{\sum_{i=1}^{N} (K_{soil,\,i} - K_{soil,\,mean})^2}{N}} \right) \bullet 100 \qquad \textbf{(5-9)}$$

Several convergence criteria are applied to get an optimum length of the experimental tests by changing the X and Y values. Some of the criteria did not converge for all the test sites, hence are not discussed here. The criteria were applied only within 50 hours of experimental test. The possibility that a particular criterion will converge after 50 hours of experimental data can not be avoided. The best set of values are X = 0.5% and Y = 5 hours, which are obtained by trial and error. Table 5-5 shows a summary of the length of the experimental test for all the seven test sites.

Table 5-5. A summary of length of experiment for all the test sites

Site	Ksoil	Ksoil from O'Neill	% Difference Ksoil	Kgrout	Max Time Taken(Hrs)
SiteA1_98	1.290	1.367	5.624	0.581	26.46
SiteA2_114	1.453	1.450	-0.202	0.865	48.21
SiteA2_170	1.393	1.432	2.670	1.185	34.50
SiteA5_93	1.334	1.426	6.467	0.612	28.38
SiteA6_240	1.256	1.317	4.672	0.800	10.08
Chickasha	1.420	1.478	3.890	0.794	14.79
Weatherford	1.513	1.634	7.403	1.270	21.88

From Table 5-5, it can be observed that in the worst case, soil conductivity is off by 7.5% from the true minimum obtained from O'Neill Simplex method. The length of the experimental test can be reduced as low as 10 hours in some cases with only 4.6% of error in soil conductivity. The average length of the test turns out to be approximately 27 hours.

6. Conclusions and Recommendations

The main objective of this project, as stated earlier, is to investigate the best parameter optimization algorithm which will reduce the time taken to estimate ground thermal properties using the numerical model which best fits the experimental data. Investigation of online parameter estimation method is also performed which will eliminate the need for off-line parameter estimation and help in determining the length of the experimental test so that unnecessary collection of data can be avoided.

6.1. Off-line parameter estimation

Several parameter optimization methods have been applied to see whether the number of objective function evaluations can be reduced because each objective function evaluation takes about 3 minutes on a Pentium II 233 MHz computer for 50 hours of experimental data. Hence, a lot of computer time is required if number of function evaluations are quite large.

Both gradient and non-gradient based, deterministic and stochastic methods have been applied for 50 hours of experimental data. Non-gradient and deterministic methods perform the best if comparison is made in terms of number of objective function evaluations.

Both Nelder Mead Simplex and O'Neill simplex algorithm perform in a similar manner except that convergence criterion is different. O'Neill simplex method has advantages of built-in construction of simplex and restart of the algorithm. There are always slight differences in the optimum parameter values depending upon the starting guess because of the basic nature of search being at discrete points and no absolute convergence criteria. Hence differences of about 1% in the optimum parameter values are ignored for comparison purposes and results from other optimization methods are compared with O'Neill's Simplex method. These two implementations minimize the SSQERR in the least number of function evaluations. Nelder Mead Simplex takes about 79 and O'Neill's implementation requires about 62 objective function evaluations.

Box's Complex method is also a "simplex" method with larger number of vertices, 2*n+1, as opposed to n+1 in Nelder Mead Simplex. This method has been investigated for various values of reflection coefficient greater than 1 as well as less than 1. For some cases of reflection coefficient, the parameters do not converge to true minimum. Results obtained for an appropriate reflection coefficient show that the average number of objective function evaluations is 152, higher than Nelder Mead simplex. This can be attributed to multiple contractions of the reflected point in the case when reflection is not successful. Contraction is tried until a favorable point is obtained which involves waste of objective function evaluations.

Hooke and Jeeves' method performs next best to Nelder Mead Simplex in terms of number of objective function evaluations. This method has the advantage of increasing the step length and turning of pattern direction as the favorable points are obtained. Higher number of function evaluations is due to an unsuccessful exploratory step, which causes at least 5 objective function evaluations per unsuccessful exploratory step. The average number of objective function evaluations for this case is 127, a bit higher than Nelder Mead simplex but less than all the other methods.

Powell's method is considered very efficient in minimizing sum of the square of the error for quadratic functions. But due to non-quadratic nature of the objective function surface, this method requires very large number of objective function evaluations. Each line minimization requires on an average of 15 objective function evaluations, which means 30 function evaluations for each cycle of minimization for two dimensions. According to Powell, more than n*(n+1), 6 in our case, line minimization are required for non-quadratic functions. Hence, this method becomes very inefficient for our purposes. The average number of objective function evaluations for this case is 292, considerably higher than Nelder Mead Simplex.

BFGS method, which is based on calculating the gradient of objective function, is tried because if successful, the gradient method can minimize the objective function quickly. For our problem, this method could not converge for several test sites. This can be attributed to uncertainties in computing the gradient of objective function. This

method is susceptible to divergence if step lengths during line minimization is not found accurately. The inverse of Hessian matrix looses its positive definiteness after a lot of updates. For the cases where this method converged, it took about 505 objective function evaluations.

Genetic algorithms are stochastic based optimization methods, which require random number generation, manipulating the bits of a parent by mutation and crossovers and evaluation of objective function at those points. The optimum values found from this method are probabilistic. Even after 100 generations, 500 objective function evaluations the minimum is not reached for some test sites. But it is observed that the optimum for those sites can be reached if we run the simulation for more number of generations. Overall this method presents a hopeless case for our problem.

The Quadratic fit method, which is based on experiences from exhaustive search, gives parameters values close to optimum for some cases but it is off as high as 14% for the other test cases. The only advantage is number of objective function evaluations, which are not useful at the cost of accuracy.

Exploratory search is applied to obtain a better set of initial guess of parameters because several methods require a good initial guess. The line source method is implemented and a good starting guess is obtained for both the parameters. Then O'Neill's Simplex method is applied to get the optimum value. The number of objective function evaluations reduce to about 49 as opposed to 62 without exploratory search. This saves about (62-49)*3.0=39 minutes of computer time which is about 21% of actual time. Hence, the **O'Neill's implementation of Nelder Mead Simplex method with exploratory search** is the best method for our problem of turning valley.

6.2. Online parameter estimation

Online parameter estimation was performed to get the parameter values, as the experiment commences which saves time to estimate the parameters after the test. Optimum length of the test can be obtained by applying the convergence criteria on the estimated parameters with some error. Two conventional estimation methods are applied

for our case. Salsbury's method diverges due to its approximation of non-linear objective function as linear. Another method, which is actually application of an off-line optimization method, is successful in predicting the optimum parameters and is useful in determining the length of the test. Nelder Mead Simplex optimization algorithm is applied to get the optimum parameters for variable number of data points. The convergence criterion for online parameter estimation, obtained by trial and error, *is 0.5% variance in conductivity values applied on 5 hours of estimated parameter values.*

The convergence criteria when applied to all the test cases, provides that the required length of the test varies from 10 hours to 48 hours depending upon the specific test case. But the average length is 27 hours. The maximum error in the predicted soil conductivity values for all the test cases is 7.5%.

6.3. Recommendations

- Apply more optimization algorithms that are gradient based and are more robust than BFGS method to investigate how gradient-based methods fair for this problem.

- A more robust online parameter estimation technique should be applied rather than the off-line optimization method applied for online parameter estimation, which updates the parameters in a manner that a true optimum can be obtained successfully at a particular data set. This will help in determining the length of the experimental test with further confidence.

- Numerical model should be modified in such a way so that the time taken for each objective function evaluation is reduced.

- More test sites should be investigated to gain confidence in the average number of objective function evaluations for off-line parameter estimation case.

- A plot of parameters can be obtained which will help in observing where the parameter values are converging and when to stop the experimental test.

- Quadratic search method has to be refined to get closer estimates of parameter values because this method has the potential to reduce number of objective function evaluations.

- Implement combined data acquisition and online parameter estimation on site.

References

Austin Trey 1998, Development of an In Situ System for Measuring Ground Thermal Properties, Masters Thesis, Oklahoma State University, Stillwater Oklahoma. (Also available at http://www.mae.okstate.edu/Faculty/spitler/Austin_thesis.pdf.)

Austin W. A., Yavuzturk C. and Spitler J. D. 2000, Development of an In Situ System for Measuring Ground Thermal Properties, Draft submitted to ASHRAE Transactions.

Box M. J. 1965, A New Method of Constrained Optimization and a Comparison with Other Methods, Computer Journal, Vol. 8, No. 1, pp. 42-52.

Broyden G. G. 1970, The Convergence of a Class of Double-Rank Minimization Algorithms, Parts I and II, *Journal of the Institute of Mathematics and Its Applications*, Vol. 6, pp. 76-90, 222-231

Carroll D. L. 1996, Chemical Laser Modeling with Genetic Algorithms, *AIAA Journal*, Vol. 34, 2, pp. 338-346.

Carroll D. L. 1999, FORTRAN Genetic Algorithm (GA) Driver, Online. Available: http://www.staff.uiuc.edu/~carroll/ga.html 11 March 1999.

Fletcher R. 1970, A New Approach to Variable Metric Algorithms, Computer Journal, Vol. 13, pp. 317-322.

Goldfarb D. 1970, A Family of Variable Metric Methods Derived by Variational Means, *Mathematics of Computation*, Vol. 24, pp. 23-26.

Goldberg, D.E. 1989, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., pp. 412.

Hart D. P. and Couvillion R. C., Earth-coupled heat transfer, National Water Well Association, 1986.

Hooke R. and T. A. Jeeves 1961, Direct Search Solution of Numerical and Statistical Problems, *Journal of the ACM*, Vol. 8, No. 2, pp. 212-229.

Jang S. S., Joseph B. and Mukai H. 1986, Comparison of Two Approaches to On-Line and State Estimation of Nonlinear Systems, Ind. Eng. Chem. Process Des. Dev. Vol. 25, pp. 809-814.

Nelder J. A. and Mead R. 1965, A Simplex Method for Function Minimization, *Computer Journal*, Vol. 7, pp. 308.

O'Neill R. 1971, Function Minimization using a Simplex Algorithm, J. R. Statist. Soc. C Vol. 20, No. 3, pp. 338-45.

Powell M. J. D. 1964, An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives, *Computer Journal*, Vol. 7, No. 4, p. 303-307.

Rao, S. S. 1996, Engineering Optimization, Theory and Practice, Third Edition, John Wiley & Sons, Inc.

Salsbury T. I., Fault Detection and Diagnosis in HVAC Systems using Analytical Models, Ph.D. thesis 1996, Loughborough University, UK.

Scheaffer R. L. and McClave J. T. 1995, Probability and Statistics for Engineers, 4th Edition, Duxbury Press.

Shanno, D. F. 1970, Conditioning of Quasi-Newton Methods for Function Minimization, *Mathematics of Computation*, Vol. 24, pp. 647-656.

Spendley W., Hext G. R. and Himsworth F. R 1962, Sequential Application of Simplex Designs in Optimization and Evolutionary Operation, *Technometrics*, Vol. 4, pp. 441.

Yavuzturk C., Spitler J. D. and Rees S.J. 1999, A Transient Two-Dimensional Finite Volume Model for the Simulation of Vertical U-Tube Ground Heat Exchangers, to be published in *ASHRAE Transactions*.

VITA

Nagendra Kumar Jain

Candidate for the Degree of

Master of Science

Thesis: PARAMETER ESTIMATION OF GROUND THERMAL PROPERTIES

Major Field: Mechanical Engineering

Biographical:

Education: Graduated from Indian Institute of Technology, Bombay India in May 1997 with a Bachelor of Technology Degree. Completed the requirements for Master of Science degree with a major in Mechanical Engineering from Oklahoma State University, Stillwater OK in May 1999.

Experience:

Worked for a manufacturing plant on the project “Inventory Analysis”. Employed as Teaching Assistant and Research Assistant at Oklahoma State University from January 1998 to current.

Professional Membership:

American Society of Heating, Refrigeration and Air-Conditioning Engineers (ASHRAE).