University of Cincinnati

ENVE 6026: Env/Hydr System Analysis

Optimization of Model Parameters for Theis Equation

Term Project Report

Submitted By

Gaurav Atreya, Aakriti Regmi

M14001485, M14645790

November 29, 2021

Executive Summary

This project uses a non-linear optimization method known as Genetic Algorithm to find the groundwater aquifer properties. Since we can't know what inside of the ground, we try to construct some windows to the underground (wells) to measure the response of the system to some actions (like pumping) and then interpret the data obtained to qualitatively and quantitatively analyze the system.

Using the data obtained from a pumping test, we made a model using the known theoretical solution of the same system and found out the model parameters using Genetic Algorithm.

As Genetic Algorithm is a non-linear method which can find the global solution independent of the initial condition, and is stochastic in nature, it was suitable for this problem. We compared the results from the Genetic Algorithm and other methods that are commonly used by experts in the field and found out that the results generated by the Genetic Algorithm is comparable to the industry standard techniques.

Though still more observations are needed, specially in case of unconfined and leaky aquifer to see if the Genetic Algorithm performs similarly in those models.

The results obtained also shows the importance of **Mutation** parameter in the Genetic Algorithm.

Contents

1. Introduction

Optimization problem is defined as the type of problem where you have to determine the decision variables so that it satisfies the constraints while also giving you the optimal results. The optimal results could be maximizing profits, minimizing loss, minimizing errors, having something zero, etc.

An Optimization problem is created from a system of parameters and equations, which are decision variables, constraints, objectives etc. Depending upon the characteristics of the equations and the variables there are many techniques available for solving an optimization problems.

1. Linear $\rightarrow$ All the constraints and the objectives are linear.

2. Non-Linear $\rightarrow$ At least one of the constraints or the objectives are non-linear.

And depending on what type of Optimization we want to perform there are a lot of different techniques available to us:

Linear:	Linear Programming, Inverse Matrix, Brute Force
Non-Linear:	Gradient Based Methods, Particle Swarm,
	Ant Colony, **Genetic Algorithm**

The one we are most familiar with is the **Linear Programming** method. For Linear programming, there are some requirements, like the parameters need to be continuous and the equations need to be linear. Though there are many derivatives of the Linear programming methods like the Integer programming, which can be used to solve the non-linear problems to some extent. The Linear programming method still has its limitations.

Among the non-linear programming techniques, there are gradient descent, particle swarm, ant colony, genetic algorithm, etc. Though they can solve non linear problems, they also come up with their own set of advantages and disadvantages. For example Gradient descent method has the limitations that it doesn't work on discontinuous space, and the solution depends on the initial guess, hence there is a chance of being stuck at a local minima.

Among the methods mentioned, **Genetic Algorithm** has the advantage that it doesn't get stuck in the local minima or depend on the initial values like gradient descent method. But it is lot more computationally intensive.

1.1 Problem Statement

Fitting models to the observed data is one of the most important things we do in a lot of fields related to data forecasting like hydrology. For fitting models to the data we need to find the model parameters that work the best, which is an optimization problem where we try to minimize the error, sometimes single sometimes multiple. And after model is prepared we use that model to estimate the response of the system in places or times where we don't have the observation.

Groundwater systems has one problem that's very common. That is we can't see what's inside the ground, so we have many different methods of probing for some samples, or probing the ground for some response of an action to understand it better. And since the properties of the ground varies by space, we have to do the testing for each place we want data on.

One of such method being the Pumping test, through which we can understand the hydraulic properties of the aquifer. In this project we have a pumping test data for a confined aquifer, and we'll estimate its Storativity and the Transmissivity from that data.

We have some field data of measurement of time and the drawdown level at that time, the data was taken when the pumping rate of the well was $8.155 ft^3/min$ and the observation well was at a distance of $582 ft$ from the pumping well.

$$\begin{aligned} \text{Pumping Rate, } Q &= 8.155 ft^3/min \\ \text{Distance to OW, r} &= 582 ft \end{aligned}$$

Table 1: Observed Data for Drawdown Vs Time

SN (i)	time (t_i)	drawdown (s_i)	SN (i)	time (t_i)	drawdown (s_i)
1	1.93	0.11	11	14.73	0.31
2	2.98	0.14	12	17.75	0.34
3	4.00	0.15	13	35.22	0.40
4	5.12	0.18	14	41.41	0.42
5	5.95	0.20	15	53.65	0.45
6	6.88	0.22	16	59.51	0.47
7	8.05	0.23	17	97.06	0.53
8	8.88	0.24	18	117.06	0.54
9	10.97	0.27	19	193.64	0.59
10	12.73	0.29	20	280.46	0.64

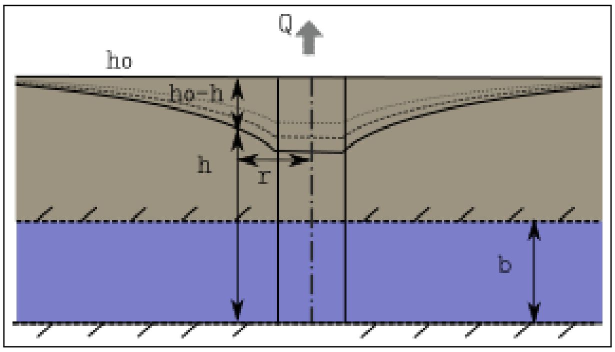


Figure 1: Drawdown Vs Distance

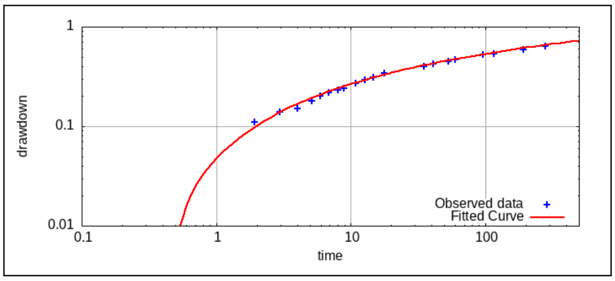


Figure 2: Drawdown Vs Time Plot

Using the formula for the Drawdown from Theis (1952), we can find the aquifer properties, which gives us the curve as shown in Figure 2.

Theis equation is used as an analytical solution in the field of transient groundwater hydraulics, it can give us the drawdown in terms of distance from the well (r) and the time (t).

The formula for the Drawdown from Theis (1952) can be written as:

$$s(r,t) = \frac{Q}{4\pi T} W(u) \tag{1}$$

Where,
$s(r,t)$ is the drawdown at distance(r) at time (t) after the start of pumping,
Q is the discharge rate, and
$W(u)$ is the well function of Theis
Which is further defined by:

$$W(u) = \int_u^\infty \frac{e^{-y}}{y} dy \tag{2}$$

$$= -\log(u) - E + \frac{u^1}{1*1!} - \frac{u^2}{2*2!} + \frac{u^3}{3*3!} - ... \tag{3}$$

Where,

$$u = \frac{r^2 S}{4Tt} \tag{4}$$

$$E = 0.57721 \tag{5}$$

1.2 Background

Finding the properties of aquifer is the most important part of the groundwater modeling. Without knowing the properties we can't predict how it'll behave in the situations we are interested in. And due to limitations in the techniques to study the underground, and the expensive natures of such techniques, we have to find the properties using just the amount of data we are able to obtain using simple tests.

Although the governing equations of the groundwater flow are simple (**Darcy's Law** & **Continuity Equations**), there aren't enough theoretical solutions for all the conditions.

The theoretical solution for **Confined Aquifer** system given by Theis (1952) assumes that the aquifer is homogeneous and isotropic. Even though it might rarely be the case, that assumption does help us in modeling with less difficulty.

Although even with knowing the equations, there comes the problem of inverse problems. Calculating Drawdowns from time and other parameters might be simple as we can just substitute the values in the equations. If some parameters are unknown and the equation can't be solved with just a simple substitution and algebra, we have to use solving techniques.

Due to non-linearity of the equations, we cannot use linear techniques for this. Traditionally **Graphical Method** was used extensively for this purpose, which was approximate. But with increase in computational powers many non-linear algorithms has in the rise. But again the problem with non-linear techniques being that it can get stuck in local minima. Methods that don't get stuck are preferred, mainly **Ant Colony Optimization** and **Genetic Algorithm** (Bateni et al., 2015).

Prasad and Rastogi (2000), Eryigit (2021), Liao et al. (2015), Chintalapati and Mathur (2012), ElHarrouni et al. (1997), etc explore many other techniques like **Simulation based** parameter estimation, **neural network** based and so on.

Besides all the available techniques Samuel and Jha (2003) and Bateni et al. (2015) show how Genetic Algorithm performs better for all 3 types of aquifer system (Confined, Unconfined & Leaky).

1.3 Objectives

The main objective of this project is to perform the optimization problem of curve fitting to the confined aquifer field observation data with the help of suitable non-linear optimization method.

2. Methodology

In this project, the main objective being the fitting of observed data to a theoretical solution, we made an optimization model based on the theoretical solution and the data with the error function as the objective function to minimize. Then we used the Genetic Algorithm as the non-linear optimization technique to solve that optimization problem.

The main equation model along with the functions to calculate the errors were created as part of the model, and then genetic algorithm was applied to that model. A lot of different parameters for genetic algorithm were tested out to find the best solution.

2.1 Optimization Model

Using the equation (1) as the basis, a model was created which would have the following **Model Parameters**.

1. Aquifer Storativity, S
2. Aquifer Transmissivity, T

As listed above, the Model has two independent parameters, that are linked to the equation (1), and the observation data. So, for each set of model parameters the estimated drawdown and the error from the observed data can be calculated. As for the parameter constraints, the ranges of values from Freeze and Cherry (1979) for different types of aquifer materials are used to get the possible range of values.

The **Constraints** can be summarized as follows:

1. $S \geq 1 \times 10^{-14}, S \leq 0.01$
2. $T \geq 0.001, T \leq 1000.0$

The **Objective Function** was chosen to **Minimize** mean absolute difference between predicted value and actual measurement.

$$Z = \frac{1}{N} \sum_{i} |s_i - s(r, t_i)| \tag{6}$$

Where,
s_i is the observed drawdown, and
$s(r,t_i)$ is the drawdown calculated using Equation (1).

2.2 Genetic Algorithm

Genetic Algorithm (GA) is among the algorithms that were designed to imitate the Nature. Since the nature already have so many complex things going on as a result of simple governing rules. GA tries to imitate it to solve problems or to understand the natural systems using the system of **Organisms**, their **Genes** and the **Evolution of Population**.

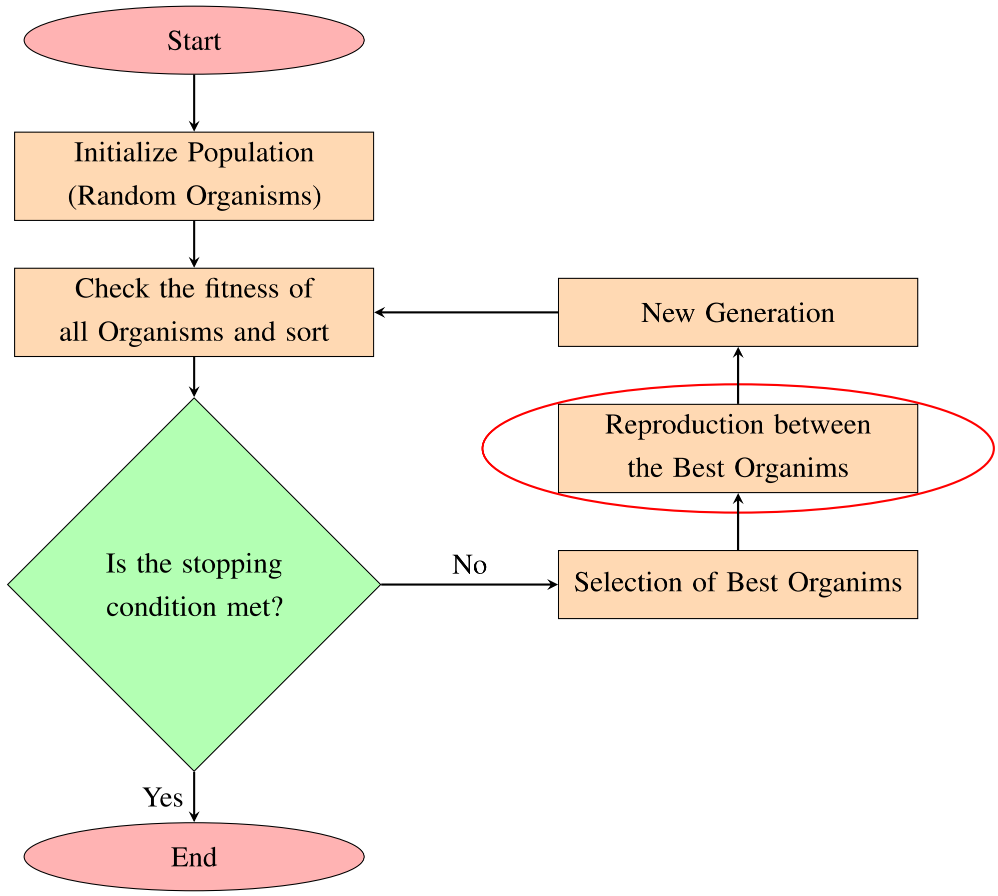


Figure 3: Flowchart for Genetic Algorithm

The main core part of Genetic Algorithm was just the part where we calculate the next generation from the current one, and all other parts can be considered the supporting structure to work around that principle.

Figure 4 shows how the reproduction is done between two organisms to make a offspring, that principle was used in the parents which were selected from the elites of the generation to get the population of next generation.

We used stochastic model using various random parameters in each calculations so that different children from the same set of parents would have different genes even if they resembled their parents (to imitate the nature better).

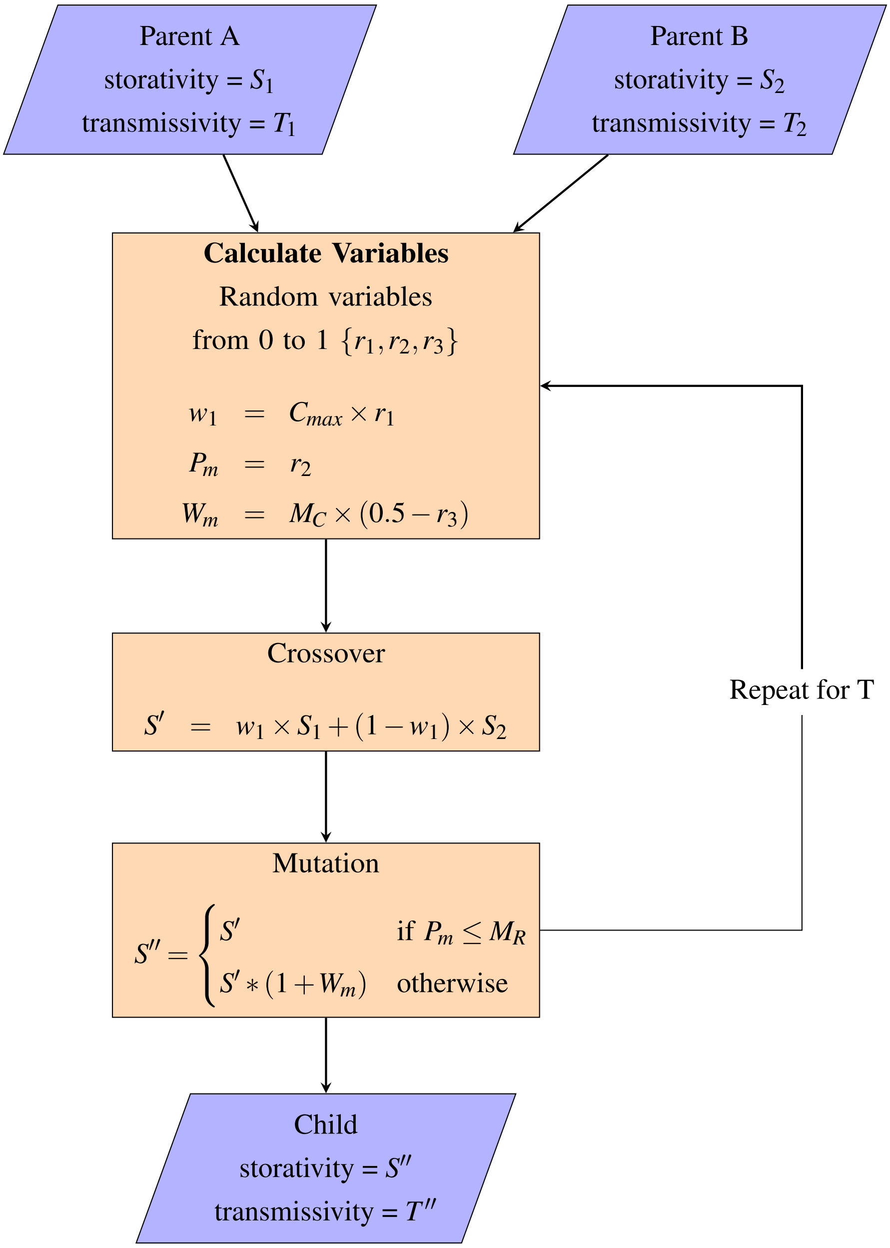


Figure 4: Reproduction

This concludes the core algorithm of our model. The main genetic algorithm was just used as the core while other modules were generated to compare the results as well as to improve the results from the genetic algorithm.

For example without the **Visualizer module** it was being really hard to debug and know if things are going well or not just from just the numerical results.

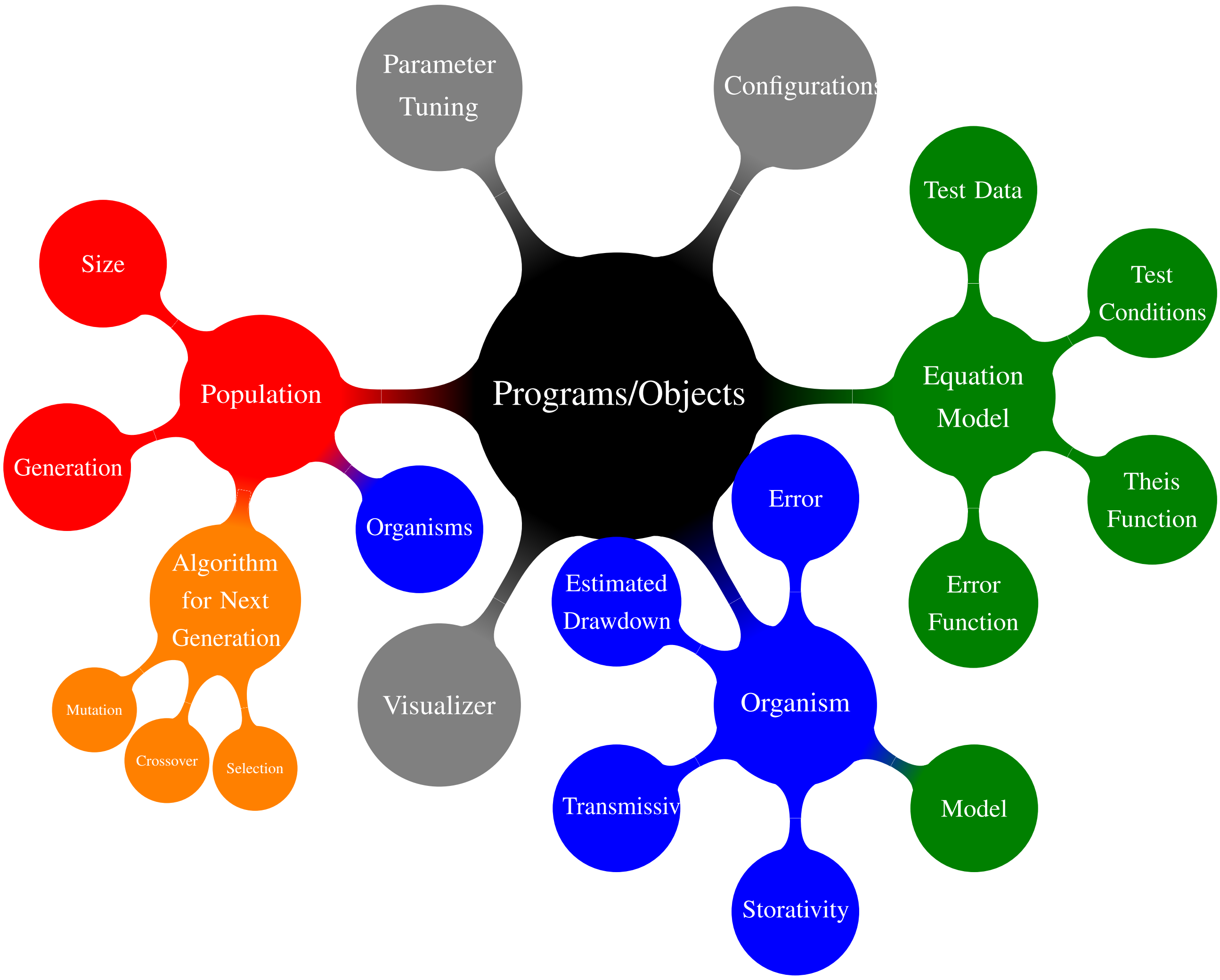


Figure 5: Objects Model for GA

2.3 Parameters Tuning

There were mainly 5 independent parameters spread throughout the implementation of the Genetic Algorithm. To find out the best set of parameters for our model, a process called parameter tuning was completed. Giving the parameters different possible values and running the model many times using the different combinations of those parameters we were able to find the parameter values which worked best.

Figure 6 shows the best 5 and the worst 5 models and the parameter values for their model.

Lots of models with comparatively wide variety of parameter values were able to reach the minimum error, so even if there are just best 5 shown in this figure, many other models have given the similar results.

Figure 7 also shows the effects of mutation parameters in the error and number of generations. Among all the 3125 models that were run, most models well successful. The unsuccessful models were only present in the rows and columns where the value of mutation parameters were zero. It is discussed further in Section 3.

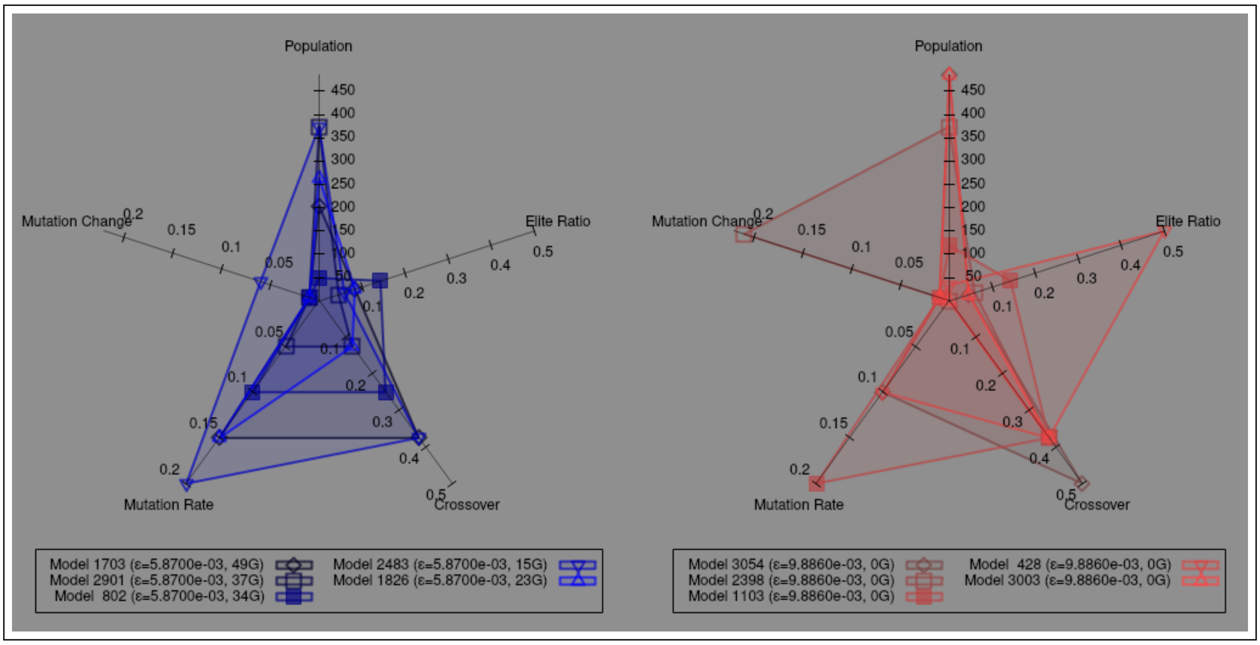


Figure 6: 5 Best and Worst Models and Their GA Parameters

3. Results and Discussions

As the model was stochastic it was run over and over again, and in all of those run, using the same parameters resulted in similar results.

Table 2: Comparison of Results from GA With Other Methods

	Parameters		Errors		
Method	Storativity	Transmissivity	M.A.E.	Mean	Std. Dev.
Graphical	3.8×10^{-05}	5.7020	8.12×10^{-03}	5.31×10^{-03}	8.63×10^{-03}
AQTsolve	4.2331×10^{-05}	5.5483	6.08×10^{-03}	-5.47×10^{-05}	7.86×10^{-03}
GA	4.2316×10^{-05}	5.5657	5.87×10^{-03}	-7.26×10^{-04}	7.87×10^{-03}

Looking at the results from Table 2 the Mean Absolute Error (MAE) for the solution obtained from the GA is better among other methods, while the mean error is less for AQTSOLVE, and the standard deviation of the errors is similar for both. Though both are clearly better than the **graphical** method, we can't say for sure which is better. The inconclusiveness comes because the AQTSOLVE used a different method of solving and was reducing different error than our **GA** algorithm.

Although not shown here, the AQTsolve program depends on the initial guess while GA doesn't, so it may be a good point against GA. Again the number of iteration it took for GA is lot higher than that for AQTsolve (See Appendix).

And since GA's advantage lies in the ability of the population to adapt to the change, it can't show its full potential in a static environment where the solution doesn't change. We made the changes in the representation and the mutation to remove the diversity of the population as the environment was static, it still took a good amount of processing before it was able to find the solution.

Without even that modification, using a binary genetic algorithm we had to use a large population, or a large number of generations to get the similar results.

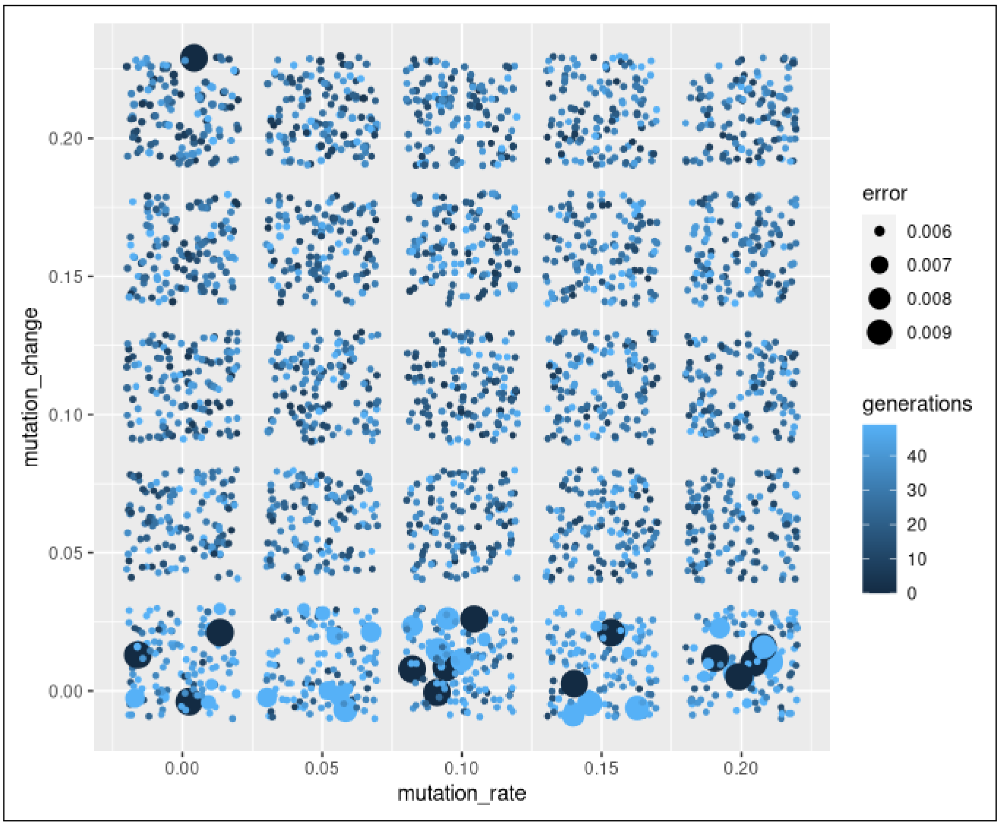


Figure 7: Role of GA Parameters on Solution Convergence

Figure 7 shows that, most of the solutions that were unable to reach the solution were either the ones with zero mutation change, or zero mutation rate (top left). In all other cases the model was successful. And as for the ones which did succeeded despite the zero mutation change, the number of generation is higher (you can see the lighter color on the bottom row). So it shows the importance of the mutations in the model.

4. Conclusions

From this experiment we were able to learn that Genetic Algorithm does perform well in terms of accuracy when used for the determining the aquifer properties of confined aquifer. The time taken for the solution was larger than the industry standard software (AQTSOLVE), but we need to use some better method to quantify that.

Also, even if the accuracy is higher, the error produced by both are too small, so even if it may be able to accurately measured by the field instruments that sort of variation may occur due to other reasons hence we cannot say with confidence which one is more accurate.

And since this approach seems to be converging faster than the binary genetic algorithm, we can conclude that it is better suited for the current problem as the solution field is static and does not have local solutions. However we do still need to perform the same modeling for **Unconfined Aquifer**, and the **Leaky Aquifer** types to see if the same thing applies for those too. It'll be even more important if those aquifer's models have local minima, which we were unable to text with confined aquifer system.

References

Bateni, S. M., Mortazavi-Naeini, M., Ataie-Ashtiani, B., Jeng, D. S., and Khanbilvardi, R. (2015). "Evaluation of methods for estimating aquifer hydraulic parameters." *Applied Soft Computing*, 28, 541–549.

Chintalapati, D. S. and Mathur, S. (2012). "Particle swarm optimization trained neural network for aquifer parameter estimation." *KSCE Journal of Civil Engineering*, 16.

ElHarrouni, K., Ouazar, D., Wrobel, L. C., and Cheng, A. H. D. (1997). "Groundwater parameter estimation by optimization and DRBEM." *Engineering Analysis with Boundary Elements*, 19(2), 97–103 Place: Oxford Publisher: Elsevier Sci Ltd WOS:A1997XV06100003.

Eryigit, M. (2021). "Estimation of parameters in groundwater modelling by modified Clonalg." *Journal of Hydroinformatics*, 23(2), 298–306 Place: London Publisher: Iwa Publishing WOS:000630000200007.

Freeze, R. A. and Cherry, J. A. (1979). *Groundwater*. Prentice-Hall Google-Books-ID: feVOAAAAMAAJ.

Liao, H.-S., Sampath, P. V., Curtis, Z. K., and Li, S.-G. (2015). "Hierarchical Modeling and Parameter Estimation for a Coupled Groundwater-Lake System." *Journal of Hydrologic Engineering*, 20(11), 04015027 Place: Reston Publisher: Asce-Amer Soc Civil Engineers WOS:000363089300012.

Prasad, K. L. and Rastogi, A. K. (2000). "Estimation of hydrogeological parameters in groundwater modeling by genetic algorithm." *Ground Water Updates*, K. Sato and Y. Iwasa, eds., Tokyo, Springer-Verlag Tokyo, 405–410, <https://www.webofscience.com/wos/woscc/summary/2f15549e-b03c-468a-9e6b-a5fe11430692-160e395c/relevance/1>. WOS:000088358300067.

Samuel, M. P. and Jha, M. K. (2003). "Estimation of Aquifer Parameters from Pumping Test Data by Genetic Algorithm Optimization Technique." *Journal of Irrigation and Drainage Engineering*, 129(5), 348–359 Publisher: American Society of Civil Engineers.

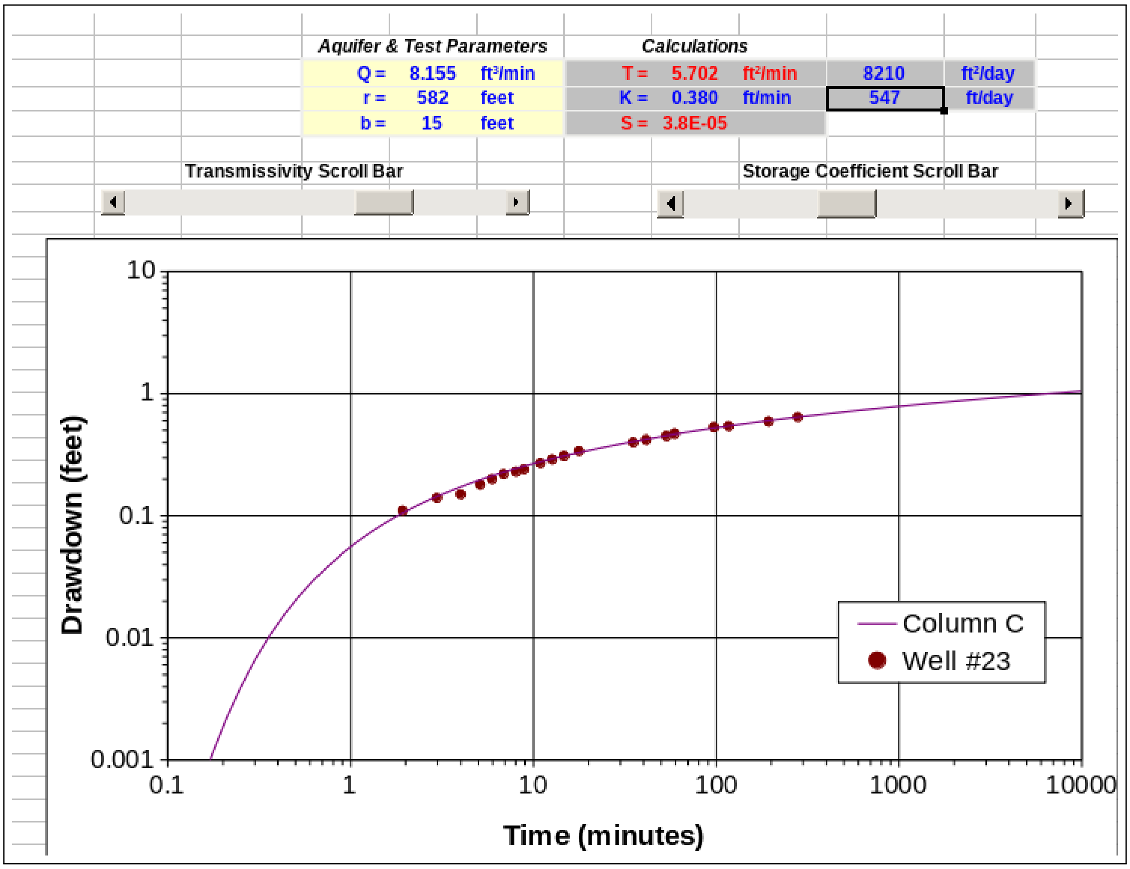
Theis, C. V. (1952). "The relation between the lowering of the piezometric surface and the rate and duration of discharge of a well using ground-water storage." Vol. 16, 519–524, <https://pubs.er.usgs.gov/publication/70198446>. Issue: 2.

Appendix

Graphical Method

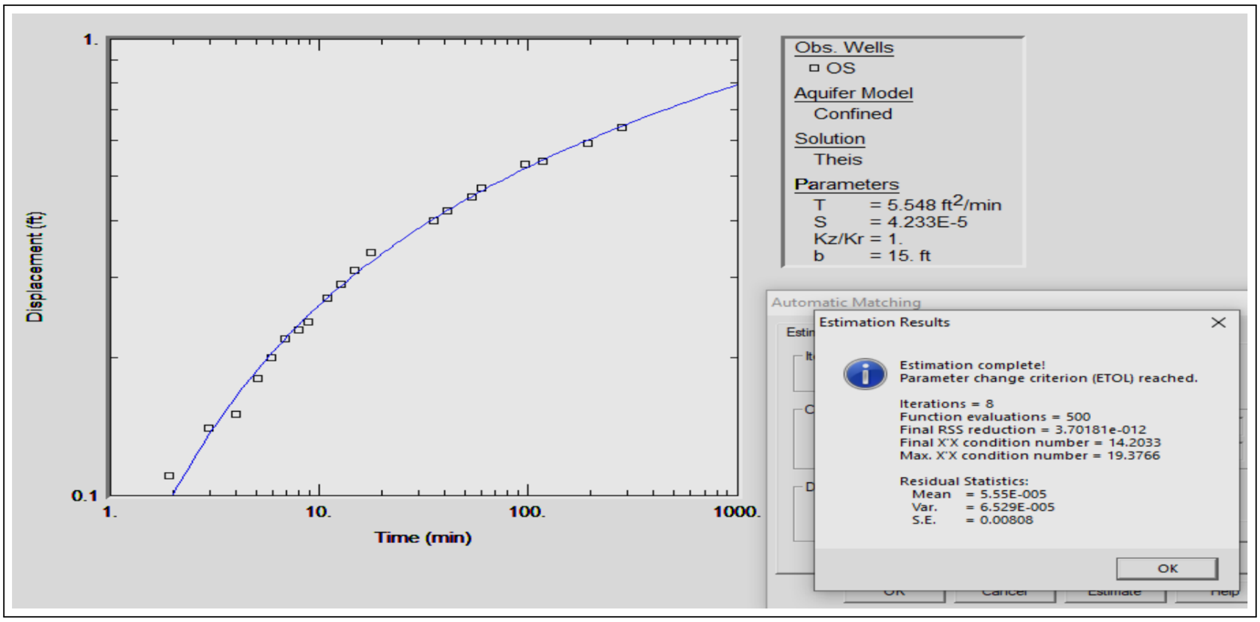
We used a spreadsheet to solve the same problem graphically, the accuracy was shown in the section 3 along with other methods.

Since this method is mostly manual, us just scrolling the values of S & T till we were satisfied visually, the accuracy was rather low compared to other two, and it was also time consuming.



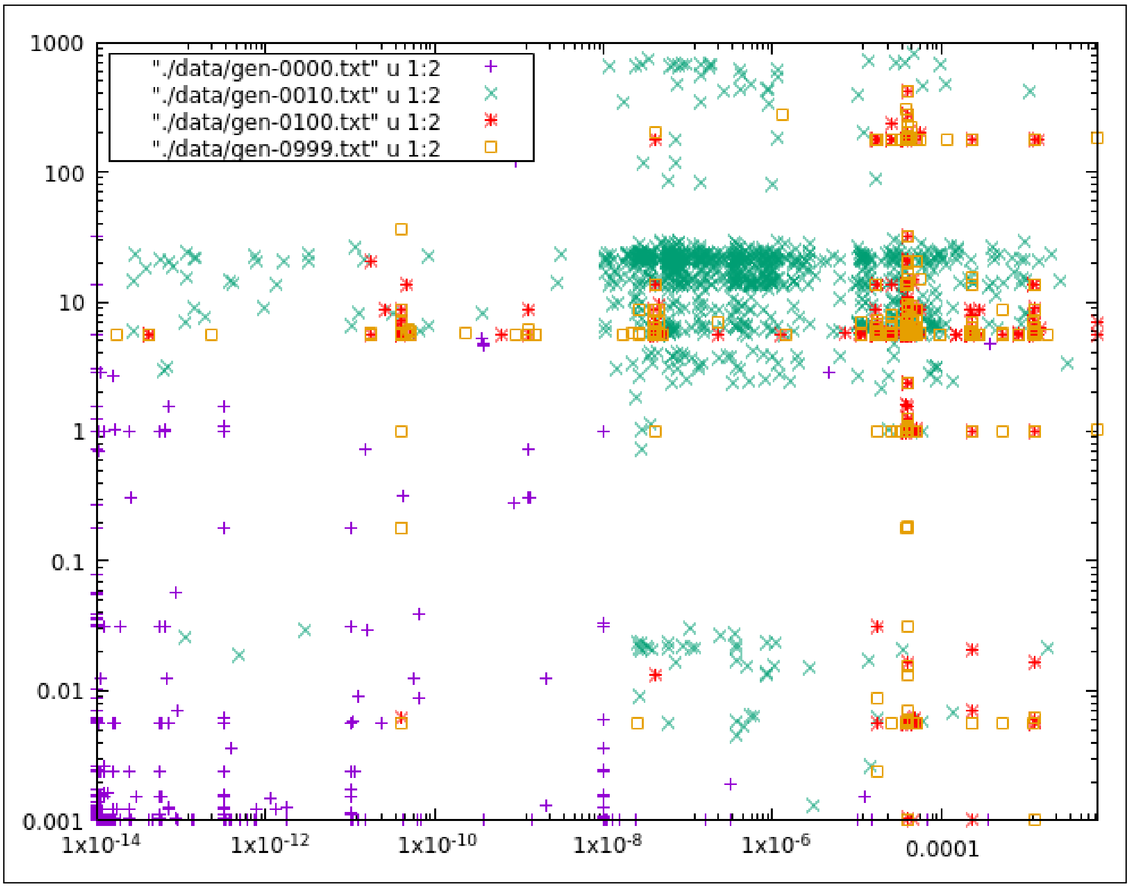
AQTsolve

We also used AQTsolve software to solve the same problem, the accuracy was shown in the section 3 along with other methods.



Binary Genetic Algorithm

Since our model was running using the numerical values as the genes instead of a gene string, we also tried a binary Genetic Algorithm with a binary genes, half the genes were for storativity and half for transmissivity. Running that model, the results obtained were not that great, as the length of string determines the resolution of search, and the diversity is high in any population, which is good for dynamic environment but since ours is static we ended up not choosing this model.



Resources

All the codes for the genetic algorithm and the plots generated along with instructions on how to use them are available at:
`https://github.com/Atreyagaurav/aquifer-properties-ga/`.