Flow Through Porous Media: Recovering permeability data from incomplete information by function fitting

Eleanor Gosling

September 2004

Abstract

The problem of having a restricted amount of data fpr a computational model is one faced by many, including oceanographers, meteorologists and oil companies. How this problem is overcome varies from discipline to discipline. In this dissertation we consider methods for fitting the permeability data associated with a model for flow through porous media. Optimisation techniques namely conjugate gradients and simulated, are presented as a means for solving linear systems of equations which may be under or over determined. We start by considering an inverse polynomial function for the permeability to achieve an equation for the pressure, minimising either the least squares problem $\|A\boldsymbol{x} - \boldsymbol{b}\|$ or the norm $\|\boldsymbol{x}\|^2$ for the under and over determined cases. This, however, is a highly specific case and so the second part of the dissertation is concerned with methods applicable when both the pressure and permeability are approximated on a discrete mesh. This is achieved by first looking at the pressure field and minimising the curvature of the field, [2]. In one dimension we then consider the first order differential equation for the permeability, which can be integrated and used to find an estimate for the permeability. The equivalent method is outlined for the two dimensional case, which involves integrating the permeability field along streamlines.

I confirm that this is my own work and the use of all material from other sources has been properly acknowledged.

Acknowledgments

I would like to thank Professor Mike Baines and Dr Mark Wakefield for their support and approachability throughout this dissertation and the course in general. I would also like to thank Sue Davis, Dr Pete Sweby and the other members of staff for their continued help, making it an enjoyable and relatively stress free year.

I would also like to say thank you to the NERC for their financial support.

Last but by no means least many thanks also go to my family and friends for being there and believing in me throughout the year.

Contents

Chapter 1

Introduction

In many areas where input data is required we often find that the amount of data that we have is much less than the amount required to give a full solution to the problem. In these cases we need to consider the best way to fill in the missing information required with respect to the underlying model. When data is repeatedly received this can be achieved by data assimilation, otherwise interpolation techniques are required, which try to find an optimal solution (numerical or analytical) in some sense. The problem of having minimal data is one faced by oil recovery companies: in order to find the best place to extract the oil the permeability of the rock that it flows through needs to be estimated, often with only a small amount of data which may be at random positions within the field and many miles apart. The problem is governed by flow in a porous medium in which the permeability varies. The primary variable is normally only the pressure but in the situation we are considering here both the permeability and the pressure are required, with relatively little data.

1.1 Building a model for fluid flow through porous media

Consider the equations which govern the flow of fluid through a porous media. The state variables for fluid flow through a porous media are the porosity of the media, ψ, the density, ρ, pressure, p, and velocity $\boldsymbol{u}$ of the fluid. The mass of the fluid is then given by the integral

$$M = \int_\omega \psi(x)\rho(x)d^3x.$$

The mass balance law says that the rate of increase in a volume ω must equal the sum of the rate of decrease by the flux through $\partial\omega$ and the rate of increase by creation. Given that the rate of fluid creation is given by

$$\int_\omega q d^3x,$$

then

$$\begin{aligned}
&\frac{d}{dt}\int_\omega \rho\psi d^3x = -\int_{\partial\omega} \rho\boldsymbol{u}.\boldsymbol{n}dS + \int_\omega q d^3x,\\
&\therefore \int_\omega \frac{\partial}{\partial t}(\rho\psi)d^3x = -\int_\omega \nabla.(\rho\boldsymbol{u})d^3x + \int_\omega q d^3x,\\
&\therefore \int_\omega \left\{\frac{\partial}{\partial t}(\rho\psi) + \nabla.(\rho\boldsymbol{u}) - q\right\} d^3x = 0.
\end{aligned}$$

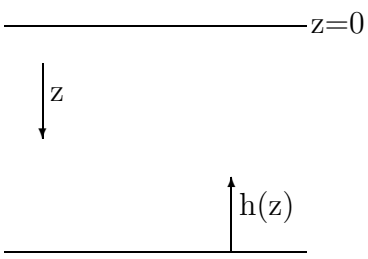
But ω is an arbitrary volume and so

$$\frac{\partial}{\partial t}(\rho\psi) + \nabla.(\rho\boldsymbol{u}) = q,$$

which is the mass balance equation in Eulerian form.

Darcy's Law says that the velocity is proportional to the gradient of the potential

$$\boldsymbol{u} = -\frac{K}{\mu}(\nabla p + \rho g \nabla h),$$



where $K = K(\boldsymbol{x})$ is the permeability of the rock and μ the viscosity of the fluid.

During the course of this dissertation we will be concerned with the special case where there is no creation/destruction of the fluid ($q = 0$). We shall consider a horizontal layer (so we can take $\Delta h = 0$) and take the density of the fluid is constant (ρ = constant). Then

$$\nabla.\boldsymbol{u} = 0, \boldsymbol{u} = -\frac{K}{\mu}\nabla p,$$

$$\text{so} \quad \nabla.\left(\frac{K}{\mu}\nabla p\right) = 0, \tag{1.1}$$

$$\text{or} \quad \nabla.\left(k\nabla p\right) = 0, \tag{1.2}$$

where k is the relative permeability, referred to as the permeability during the rest of this dissertation. Equation (1.1) is the equation we work with.

The problem is then that the permeability is unknown and the pressure is only known at a few positions within the field. The aim of this project is to look at ways of estimating the solution to the oil field based on small amounts of data available.

1.2 Maximum Principle

A maximum principle says that (Protter [6]) if $v(x_1, x_2, \cdots, x_n)$ satisfies the differential inequality

$$L[v] \equiv \sum_{i,j=1}^{n} a_{ij}(\boldsymbol{x})\frac{\partial^2 v}{\partial x_i \partial x_j} + \sum_{i=1}^{n} b_i(\boldsymbol{x})\frac{\partial v}{\partial x_i} \geq 0 \tag{1.3}$$

in a domain D. Where L is uniformly elliptic and the coefficients a_{ij} and b_i are uniformly bounded. Then if v attains a maximum M at a point of D, $v = M$ in D.

A similar result can be obtain for a minimum principle, by taking $-\boldsymbol{v}(x_1, x_2, \cdots, x_n)$ as a solution to (1.3) and so D can not contain either a maximum or minimum of $\boldsymbol{v}$.

Both these principles apply to equation (1.1), which means that the pressure field can not contain a maximum or minimum with in the domain. Realistically this means that the pressure flows through the domain and it contains no sources or sinks. In generating test data for our simulation the maximum principle will be taken into consideration and pressure gradients will be non-zero throughout the domain.

1.3 Remit

Finding the permeability in both one and two dimensions presents itself as a set of linear equations. When this set of equations is square (that is the same number of unknowns as linearly independent equations) the solution can easily be found using a number of methods (including conjugate gradients or gaussian elimination and back substitution). However when the system is either under-determined or over-determined rectangular systems arise and there may be infinitely many solutions or no solution. In the former case each of these solutions would satisfy the equation but we need one solution above all others. During the course of this project we determine permeability and pressure fields or approximations that satisfy the stationary diffusion equation given a small amount of data. First however, in chapters 2 and 3, we shall look at two methods for optimization which can be applied to solving a system of equations $A\boldsymbol{x} = \boldsymbol{b}$, namely Conjugate Gradients and Simulated Annealing. Then we shall move on to look at one and two dimensions for

a specific form of k. Finally we shall consider a numerical solution, first approximating the pressure, using the principle of minimum curvature, and then the permeability using the resulting pressure values.

Chapter 2

Conjugate Gradients

The method of conjugate gradients takes an arbitrary starting point and minimises the quadratic form associated with the system $A\boldsymbol{x} = \boldsymbol{b}$ (Shewchuck [7]). It is an advance on steepest decent methods. In the conjugate gradient method steps are taken down the paraboloid of the quadratic form in which search directions are never repeated and therefore convergence is guaranteed in n steps where n is the size of the problem.

2.1 Quadratic Forms

Conjugate gradients uses the fact that, when the matrix A is a symmetric matrix, a solution to the system $A\boldsymbol{x} = \boldsymbol{b}$ is a critical point of the quadratic form. The quadratic form associated with the system of equations $A\boldsymbol{x} = \boldsymbol{b}$ is given by

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x},$$

which if we differentiate we can see has a critical point at $A\boldsymbol{x} = \boldsymbol{b}$.

$$\begin{aligned} f'(x) &= \frac{1}{2}A^T\boldsymbol{x} + \frac{1}{2}A\boldsymbol{x} - \boldsymbol{b}, \\ &= A\boldsymbol{x} - \boldsymbol{b} \text{ if } A \text{ is symmetric } (A^T = A), \\ &= 0 \text{ for critical points.} \end{aligned}$$

The shape of $f(\boldsymbol{x})$ depends on the form of the matrix A, see figures 2.1, and therefore also effects the ability of conjugate gradients to find a solution. A singular matrix has a line of solutions running through the the bottom of the 'valley'. For an indefinite matrix the quadratic form has a saddle point and as such the methods of steepest decent and conjugate gradients will not work. However, if the matrix A is also positive-definite, then this solution is also a minimum of $f(\boldsymbol{x})$. To see this suppose A is symmetric and let $\boldsymbol{x}$ be a point that satisfies $A\boldsymbol{x} = \boldsymbol{b}$ as well as minimising $f(\boldsymbol{x})$.

$$\begin{aligned} f(\boldsymbol{x} + \boldsymbol{e}) &= \frac{1}{2}(\boldsymbol{x} + \boldsymbol{e})^T A(\boldsymbol{x} + \boldsymbol{e}) - \boldsymbol{b}^T(\boldsymbol{x} + \boldsymbol{e}), \\ &= \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} + \boldsymbol{e}^T A\boldsymbol{x} + \frac{1}{2}\boldsymbol{e}^T A\boldsymbol{e} - \boldsymbol{b}^T\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{e}, \\ &= f(\boldsymbol{x}) + \frac{1}{2}\boldsymbol{e}^T A\boldsymbol{e}. \end{aligned}$$

Provided A is always positive-definite the second term is always positive for $\boldsymbol{e} \neq 0$ and so $\boldsymbol{x}$ minimises f.

2.2 Conjugate Gradient Method

The method of conjugate gradients is a combination of the method of steepest descent and the method of conjugate directions. In the method of steepest descent a series of steps are taken which slide down to the bottom of the paraboloid of the quadratic form,

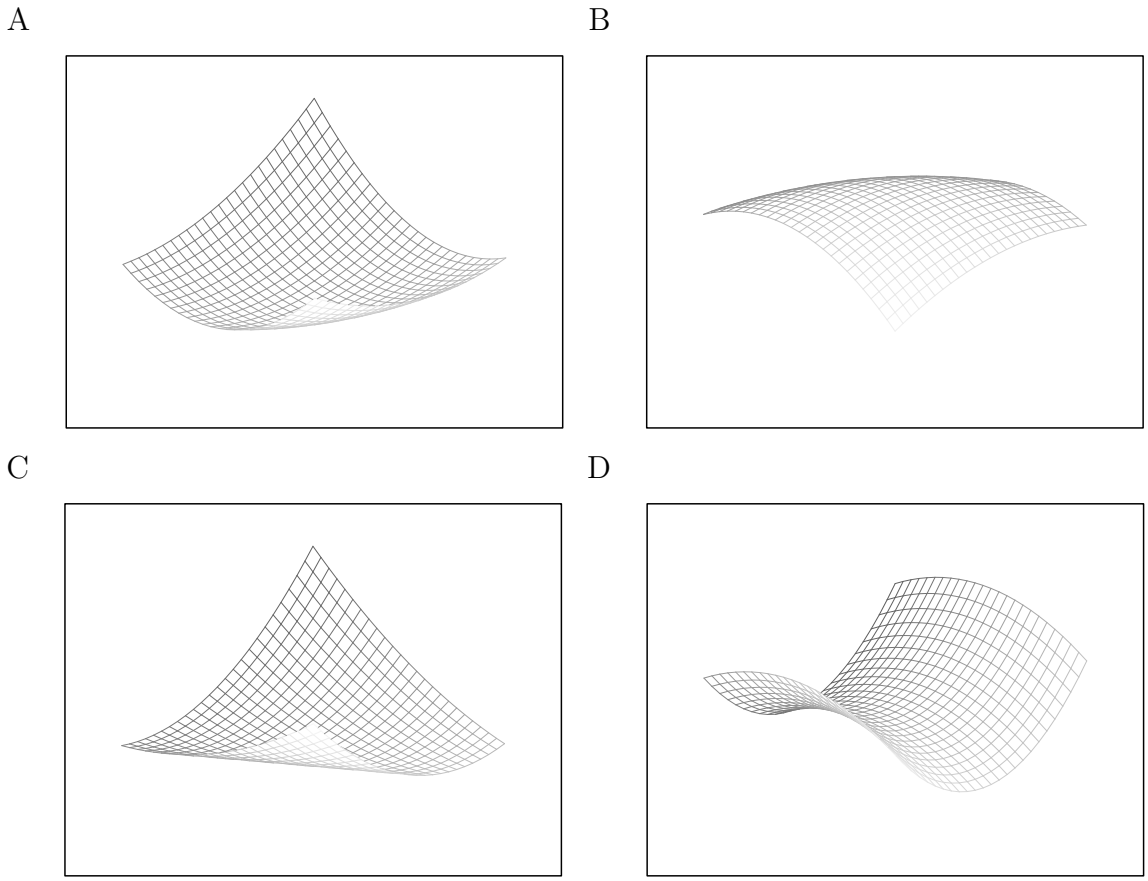


Figure 2.1: *A: positive-definite matrix, B: negative-definite matrix, C: singular matrix and D: indefinite matrix.*

f. Each step is taken in the direction in which f decreases most quickly, which is the direction opposite to $\boldsymbol{\nabla}_{\boldsymbol{x}} f(\boldsymbol{x}_{(i)}) = A\boldsymbol{x}_{(i)} - \boldsymbol{b}$ at the point $\boldsymbol{x}_i$. The direction of steepest decent is also equal to the residual $\boldsymbol{r}_{(i)} = \boldsymbol{b} - A\boldsymbol{x}_{(i)}$ which indicates how far we are from the correct value of b. So if we start at an arbitrary point $x_{(0)}$ then the next step will be given by

$$x_{(1)} = x_{(0)} + \alpha r_{(0)}. \tag{2.1}$$

The size of step, i.e. the size of α, is chosen to mimimise f along a line defined by the directional derivative

$$\begin{aligned} 0 &= \frac{d}{d\alpha} f(x_1), \text{ that is} \\ &= f'(x_{(1)})^T \frac{d}{d\alpha} x_{(1)}, \\ &= f'(x_{(1)})^T r_{(0)}, \end{aligned}$$

that is α is chosen so that the residual and directional derivative are orthogonal.

To determine α, note that $f'(x_{(1)}) = -r_{(1)}$ then we have

$$\begin{aligned} &\boldsymbol{r}_{(1)}^T \boldsymbol{r}_{(0)} = 0, \\ &(\boldsymbol{b} - A\boldsymbol{x}_{(1)})^T \boldsymbol{r}_{(0)} = 0, \\ &(\boldsymbol{b} - A(\boldsymbol{x}_{(0)} + \alpha \boldsymbol{r}_{(0)})^T \boldsymbol{r}_{(0)} = 0 \quad \text{by 2.1}, \\ &(\boldsymbol{b} - A\boldsymbol{x}_{(0)})^T \boldsymbol{r}_{(0)} - \alpha (A\boldsymbol{r}_{(0)})^T \boldsymbol{r}_{(0)} = 0, \\ &(\boldsymbol{b} - A\boldsymbol{x}_{(0)})^T \boldsymbol{r}_{(0)} = \alpha (A\boldsymbol{r}_{(0)})^T \boldsymbol{r}_{(0)}, \\ &\boldsymbol{r}_{(0)}^T \boldsymbol{r}_{(0)} = \alpha \boldsymbol{r}_{(0)}^T (A\boldsymbol{r}_{(0)}) \quad \text{since } \boldsymbol{r}_{(0)} = \boldsymbol{b} - a\boldsymbol{x}_{(0)}, \\ &\alpha = \frac{\boldsymbol{r}_{(0)}^T \boldsymbol{r}_{(0)}}{\boldsymbol{r}_{(0)}^T A \boldsymbol{r}_{(0)}}. \end{aligned}$$

In general the steepest descent method is

$$\begin{aligned}
\boldsymbol{r}_{(i)} &= \boldsymbol{b} - A\boldsymbol{x}_{(i)}, \\
\alpha_{(i)} &= \frac{\boldsymbol{r}_{(i)}^T \boldsymbol{r}_{(i)}}{\boldsymbol{r}_{(i)}^T A \boldsymbol{r}_{(i)}}, \\
\boldsymbol{x}_{(i+1)} &= \boldsymbol{x}_{(i)} + \alpha_{(i)} \boldsymbol{r}_{(i)}.
\end{aligned}$$

The problem with the method of steepest descent is that it can find itself taking directions it has already searched along making it inefficient, which is where the method of conjugate directions improves the algorithm. A set of orthogonal search directions $\boldsymbol{d}_{(0)}, \cdots, \boldsymbol{d}_{(n-1)}$ are generated and exactly one step is taken in each of these directions. Since each direction need only be considered once the method is guaranteed to converge in n steps, where n is the size of the problem (apart from the effects of rounding error).

Consider the two dimensional example shown in figure 2.2. It can be seen that $\boldsymbol{e}_{(1)}$ (the error $\boldsymbol{e}_{(i)} = \boldsymbol{x}_{(i)} - \boldsymbol{x}$) is orthogonal to $\boldsymbol{d}_{(0)}$. In order to find $\alpha_{(i)}$ this fact is used and

$$\alpha_{(i)} = -\frac{\boldsymbol{d}_{(i)}^T \boldsymbol{e}_{(i)}}{\boldsymbol{d}_{(i)}^T \boldsymbol{d}_{(i)}}.$$

Unfortunately, in order to find the size of step needed we need to know the distance from the solution (which if we knew we would know the solution) and so $\alpha_{(i)}$ cannot be computed. This problem can be solved by using a set of directions that are A-orthogonal instead of orthogonal, that is use

$$\begin{aligned}
&\boldsymbol{d}_{(i)}^T A \boldsymbol{d}_{(j)} = 0, \text{ giving} \\
&\alpha_{(i)} = -\frac{\boldsymbol{d}_{(i)}^T \boldsymbol{r}_{(i)}}{\boldsymbol{d}_{(i)}^T A \boldsymbol{d}_{(i)}}
\end{aligned}$$

To compute the set of search directions suppose we have set of n linearly independent vectors $\boldsymbol{u}_{(0)}, \cdots, \boldsymbol{u}_{(n-1)}$. Then $\boldsymbol{d}_{(i)}$ is constructed by taking $\boldsymbol{u}_{(i)}$ and subtracting out any

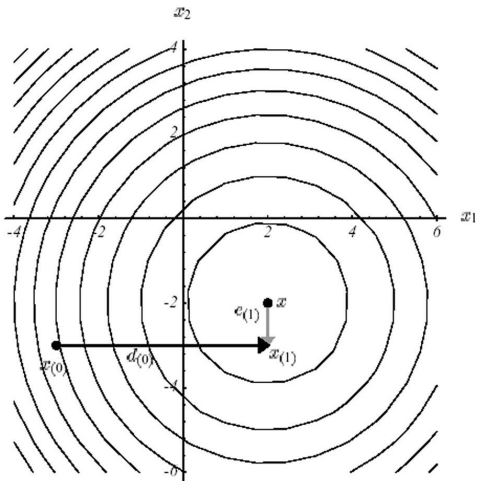


Figure 2.2: *The method of conjugate directions*

components that are not A orthogonal to to the previous $\boldsymbol{d}$ vectors,

$$\begin{aligned}\boldsymbol{d}_{(0)} &= \boldsymbol{u}_{(0)} \\ \boldsymbol{d}_{(i)} &= \boldsymbol{u}_{(i)} + \sum_{k=0}^{i-1} \beta_{ik} \boldsymbol{d}_{(k)},\end{aligned}$$

where β_{ik} are defined for $i > k$ by

$$\begin{aligned}\boldsymbol{d}_{(i)}^T A \boldsymbol{d}_{(j)} &= \boldsymbol{u}_{(i)}^T A \boldsymbol{d}_{(j)} + \sum_{k=0}^{i-1} \beta_{ik} \boldsymbol{d}_{(k)}^T A \boldsymbol{d}_{(j)}, \\ 0 &= \boldsymbol{u}_{(i)}^T A \boldsymbol{d}_{(j)} + \beta_{ij} \boldsymbol{d}_{(j)}^T A \boldsymbol{d}_{(j)}, \quad i > j \\ \beta_{ij} &= -\frac{\boldsymbol{u}_{(i)}^T A \boldsymbol{d}_{(j)}}{\boldsymbol{d}_{(j)}^T A \boldsymbol{d}_{(j)}}.\end{aligned}$$

The method of conjugate gradients is simply the method of conjugate directions where the directions are constructed by the conjugation of the residuals, that is the set of n linearly independent vectors is given by the residuals.

2.3 Proof that Conjugate Gradients Converges in n Steps

Consider the initial error, $\boldsymbol{e}_{(0)}$, as a linear combination of the search directions

$$\boldsymbol{e}_{(0)} = \sum_{j=0}^{n-1} \delta_j \boldsymbol{d}_{(j)}.$$

Pre-multiplying by $\boldsymbol{d}_{(k)}^T A$ and using the fact that the search directions are A-orthogonal

$$\begin{aligned}
\boldsymbol{d}_{(k)}^T A \boldsymbol{e}_{(0)} &= \sum_{j=0}^{n-1} \delta_j \boldsymbol{d}_{(k)}^T A \boldsymbol{d}_{(j)}, \\
&= \delta_k \boldsymbol{d}_{(k)}^T A \boldsymbol{d}_{(k)}, \\
\delta_k &= \frac{\boldsymbol{d}_{(k)}^T A \boldsymbol{e}_{(0)}}{\boldsymbol{d}_{(k)}^T A \boldsymbol{d}_{(k)}}, \\
&= \frac{\boldsymbol{d}_{(k)}^T A (\boldsymbol{e}_{(k)} - \sum_{i=0}^{k-1} \alpha_{(i)} \boldsymbol{d}_{(i)})}{\boldsymbol{d}_{(k)}^T A \boldsymbol{d}_{(k)}}, \\
&= \frac{\boldsymbol{d}_{(k)}^T A \boldsymbol{e}_{(k)}}{\boldsymbol{d}_{(k)}^T A \boldsymbol{d}_{(k)}} = -\alpha_{(k)}.
\end{aligned}$$

Now

$$\begin{aligned}
\boldsymbol{e}_{(i)} &= \boldsymbol{e}_{(0)} + \sum_{j=0}^{i-1} \alpha_{(j)} \boldsymbol{d}_{(j)}, \\
&= \sum_{j=0}^{n-1} \delta_{(j)} \boldsymbol{d}_{(j)} - \sum_{j=0}^{i-1} \delta_{(j)} \boldsymbol{d}_{(j)}, \\
&= \sum_{j=i}^{n-1} \delta_{(j)} \boldsymbol{d}_{(j)},
\end{aligned}$$

and so after n iterations $\boldsymbol{e}_{(n)} = 0$.

2.4 Algorithm For Conjugate Gradients

$$
\begin{aligned}
&\boldsymbol{d}_{(0)} = \boldsymbol{r}_{(0)} = \boldsymbol{b} - A\boldsymbol{x}_{(0)},\\
&\alpha_{(i)} = \frac{\boldsymbol{r}_{(i)}^T \boldsymbol{r}_{(i)}}{\boldsymbol{d}_{(i)}^T A \boldsymbol{d}_{(i)}},\\
&\boldsymbol{x}_{(i+1)} = \boldsymbol{x}_{(i)} + \alpha_{(i)} \boldsymbol{d}_{(i)},\\
&\boldsymbol{r}_{(i+1)} = \boldsymbol{r}_{(i)} - \alpha_{(i)} A \boldsymbol{d}_{(i)}, \quad \text{recursive residual, see below}\\
&\beta_{i+1} = \frac{\boldsymbol{r}_{(i+1)}^T \boldsymbol{r}_{(i+1)}}{\boldsymbol{r}_{(i)}^T \boldsymbol{r}_{(i)}},\\
&\boldsymbol{d}_{(i+1)} = \boldsymbol{r}_{(i+1)} + \beta_{(i+1)} \boldsymbol{d}_{(i)}.
\end{aligned}
$$

At each stage a recursive residual is calculated, the exact value of the residual is $\boldsymbol{b} - A\boldsymbol{x}$, which builds up round off errors with in the algorithm. The build up of the round off errors can be eliminated at regular stages by calculating the exact residual instead alternative the algorithm needs to run for more than n iterations. The algorithm can be run using a termination other than the number of iterations, by terminating once the value of α becomes significantly close to zero.

Chapter 3

Simulating Annealing

Simulated annealing is another minimisation technique, in this case based on the physical process of cooling a molten metal slowly in stages, allowing it to reach thermal equilibrium at each stage before continuing the the cooling. As the metal cools it could take any number of states each with a different energy value. Provided that the molten metal is cooled sufficiently slowly the process produces a crystalised state which has minimum energy. The algorithm that describes this process is the "Metropolis Algorithm", which provides an efficient simulation of a collection of atoms at a given temperature. At each step the atoms are given a small random displacement and the resulting change in energy is calculated, ΔE. If $\Delta E \leq 0$ then the new configuration of atoms is accepted, otherwise the change is considered in a probabilistic way. The new configuration is given a probability of being accepted $P(\Delta E) = \exp(-\Delta E/k_B T)$ (where k_B is the Boltzman constant and T the temperature). If the new configuration is accepted then the process starts again with this new set of positions, if not then the process starts again with the old configuration of atoms. Simulated annealing takes the simulation and applies it to a cost/objective

function instead of energy levels, and k_b and T are parameterised within the simulated annealing algorithm.

3.1 Simulated Annealing Method

The method of simulated annealing is a biased random walk around the n-dimensional space of the objective function, ϕ (Bohachevsky [1]). Starting at a point in the n-dimensional space, with an objective function of ϕ_0, a random direction is chosen and a step of Δr is taken in this direction. Before this step is taken the objective function is calculated at the new point, ϕ_1, with the step chosen based on this value. The step is given a probability, p, of being taken given by

$$p = \begin{cases} 1 & \text{if} \quad \Delta\phi = \phi_1 - \phi_0 \leq 0 \\ \exp(-\beta\Delta\phi) & \text{if} \quad \Delta\phi > 0 \end{cases},$$

where β is a chosen parameter. In this way any step which results in the reduction of the objective function is automatically taken, otherwise the step is taken based on a probability. To decide if a detrimental step is taken (that is one which increases the objective function) a random number $\xi \in [0, 1]$ is chosen. Then if $\xi < p$ the step is taken otherwise it is discarded.

The parameter Δr needs to be chosen so that the walk can easily escape a local minimum in 2 or 3 steps, and β such that $0.5 < \exp(-\beta\Delta\phi) < 0.9$. The latter condition is required for efficient searches. Probabilities close to 1 lead to most detrimental steps being taken, and probabilities less than 0.5 mean climbing out of local extrema requires a lot of function evaluations.

Generally the program is run for sufficiently many iterations required to achieve the stopping criteria that 50 random steps must be rejected at one iteration. If we assume the the objective function to have a minimum value of 0 then we can rewrite the probability choice to

$$p = \begin{cases} 1 & \text{if} \quad \Delta\phi = \phi_1 - \phi_0 \leq 0 \\ \exp(-\beta\phi_0^g \Delta\phi) & \text{if} \quad \Delta\phi > 0 \end{cases},$$

where g is some arbitrary negative number. This means that the probability will be greatly reduced as the objective function approaches zero, giving steps a much lower chance of being accepted. The objective function may not naturally give a minimum value of 0 and so we use $\phi - \phi_{min}$ as our objective function in the simulated annealing algorithm.

3.2 Generalised Simulated Annealing Algorithm

Let ϕ_m be the value of ϕ at the global minimum.

1. Let $\boldsymbol{x}_0$ be the arbitrary starting point.
2. Set $\phi_0 = \Phi(\boldsymbol{x}_0)$. If $|\phi_0 - \phi_m| < \epsilon$, stop.
3. Choose a random direction. Generate n independent random numbers between 0 and 1, $u_1, ..., u_n$, and compute the components of $U_i = u_i/(u_1^2 + ... + u_n^2)^{1/2}$, $i = 1, ..., n$.
4. Set $\boldsymbol{x}^* = \boldsymbol{x}_0 + (\Delta r)U$.
5. If $\boldsymbol{x}^* \notin \Omega$, return to step 3. Otherwise $\phi_1 = \phi(\boldsymbol{x}^*$ and $\Delta\phi = \phi_1 - \phi_0$.
6. If $\phi_1 \leq \phi_0$, set $\boldsymbol{x}_0 = \boldsymbol{x}^*$ and $\phi_0 = \phi_1$. If $|\phi_0 = \phi_m| < \epsilon$, stop. Otherwise, go to step 3.

7. If $\phi_1 > \phi_0$, set $p = \exp(-\beta\phi_0^g \Delta\phi)$.

   a) Generate a random number between 0 and 1, ρ

   b) If $\rho \geq p$, go to step 3.

   c) If $\rho < p$, set $\boldsymbol{x}_0 = \boldsymbol{x}^*$, $\phi_0 = \phi_1$, and go to step 3.

At the start of the of the algorithm we let ϕ_m be the value of ϕ at the global optimum, which assumes that we know the minimum value of the objective function. This is not often the case and so this assumption needs to be removed which can be done by running the random walk with ϕ_m set to some small value, until $\phi - \phi_m < 0$ at which point ϕ_m is lowered and the walk is continued.

Note that putting $g = 0$ gives us the standard simulated annealing method.

3.3 Choice of Parameters, a Simple Cost Function

As mentioned in section 3.1 the choice of the parameters Δr and β are very important, in order to keep the algorithm accurate and efficient. To illustrate this consider the cost function

$$\phi(x, y) = x^2 + 2y^2 - 0.3\cos(3\pi x) - 0.4\sin(4\pi y) + 0.7$$

The maximum and minimum of which can be seen in figure 3.1. For the purposes of this example the algorithm has been executed with 150 iterations, with a maximum of 50 random steps being tested at each iteration before the program is terminated. All other parameters are specified with the results shown in figures 3.2 to 3.4, the blue squares represent accepted steps and red represent rejected steps. By changing the parameters

and recording the steps accepted and rejected we can see the progress made with certain parameter choices.

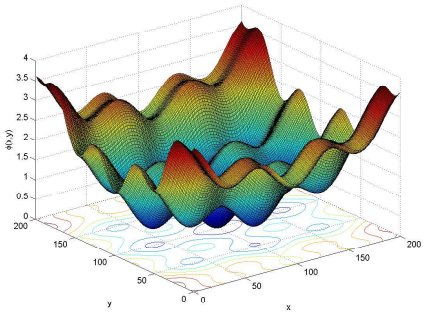


Figure 3.1: *Objective function*

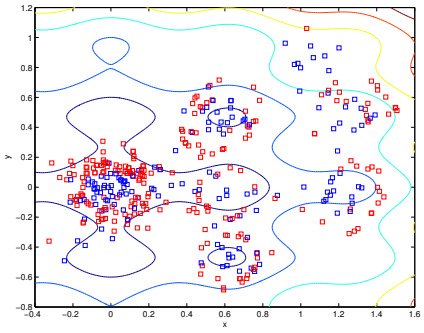


Figure 3.2: *Standard simulated annealing method with* $\Delta r = 0.15$, $\beta = 4.5$. *Algorithm runs to the end (150 iterations) and although finds the global minimum it fails to terminate at this point.*

Notice how the standard method (figure 3.2), although locating the global minimum, fails to stop at this point with the specified 150 iterations, whereas the generalized method terminates there (figure 3.3). A slight change in the β parameter by .5 means that the algorithm no longer finds the global minimum but settles for one of the local values, since

it is unable to escape from this point with in the specified 50 random steps termination criteria.

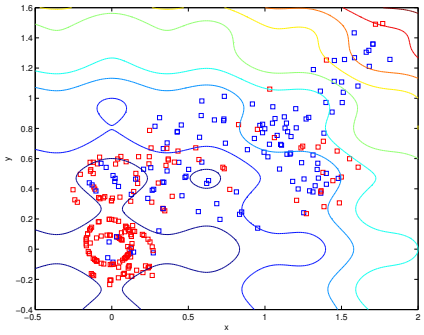


Figure 3.3: *Generalised simulated annealing method with* $g = -1$, $\Delta r = 0.15$, $\beta = 3.5$. *Algorithm temrinates at 124 out of 150 iterations, at the global minimum.*

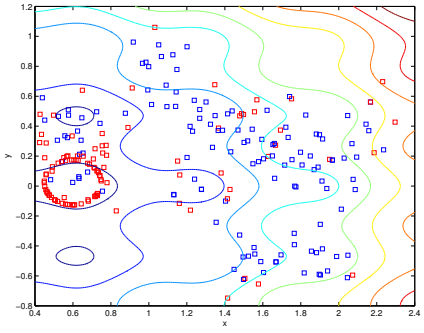


Figure 3.4: *Generalised simulated annealing method with* $g = -1$, $\Delta r = 0.15$, $\beta = 4.0$. *Algorithm terminates at 129 out of 150 iterations, but at a local minimum.*

Chapter 4

An Analytic Form of the Permeability

4.1 One-Dimension

Consider the one-dimensional case putting the permeability equal to

$$k(x) = \frac{1}{1 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \cdots + \alpha_n x^n}.$$

In one-dimension (1.1) becomes

$$\frac{d}{dx}\left(k(x)\frac{dp}{dx}\right) = 0, \tag{4.1}$$

using the boundary conditions

$$\begin{aligned} p(0) &= \eta \\ p(1) &= \gamma \end{aligned} \tag{4.2}$$

Integrating we get

$$k(x)\frac{dp}{dx} = a,$$

where a is a constant. That is

$$\frac{dp}{dx} = a(1 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \cdots + \alpha_n x^n)$$
$$\text{giving} \quad p(x) = a\left(x + \frac{\alpha_1}{2}x^2 + \frac{\alpha_2}{3}x^3 + \frac{\alpha_3}{4}x^4 + \cdots + \frac{\alpha_n}{n+1}x^{n+1}\right) + b, \qquad (4.3)$$

where b is also an unknown constant. Using the boundary conditions (4.2) in (4.3) we find that

$$a = \frac{\gamma - \eta}{1 + \frac{\alpha_1}{2} + \frac{\alpha_2}{3} + \frac{\alpha_3}{4} + \cdots + \frac{\alpha_n}{n+1}}$$
$$b = \eta$$

So the exact solution to the one-dimensional case (4.1) and (4.2) is

$$p(x) = \left(\frac{\gamma - \eta}{1 + \frac{\alpha_1}{2} + \frac{\alpha_2}{3} + \frac{\alpha_3}{4} + \cdots + \frac{\alpha_n}{n+1}}\right)\left(x + \frac{\alpha_1}{2}x^2 + \frac{\alpha_2}{3}x^3 + \frac{\alpha_3}{4}x^4 + \cdots + \frac{\alpha_n}{n+1}x^{n+1}\right) + \eta.$$

Now if $p(x_j) = \bar{p}_j$ for n observations at x_j, $j = 1, ...m$, then

$$\bar{p}_j = \left(\frac{\gamma - \eta}{1 + \frac{\alpha_1}{2} + \frac{\alpha_2}{3} + \frac{\alpha_3}{4} + \cdots \frac{\alpha_n}{n+1}}\right)\left(x_j + \frac{\alpha_1}{2}x_j^2 + \frac{\alpha_2}{3}x_j^3 + \frac{\alpha_3}{4}x_j^4 + \cdots \frac{\alpha_n}{n+1}x_j^{n+1}\right) + \eta,$$

for $j = 1, ..., m$, which can be written as an $m \times n$ system of equations, for the unknowns α_i

$$\begin{pmatrix} \frac{\omega_1}{2} - \frac{x_1^2}{2} & \frac{\omega_1}{3} - \frac{x_1^3}{3} & \cdots & \frac{\omega_1}{(n+1)} - \frac{x_1^{(n+1)}}{(n+1)} \\ \frac{\omega_2}{2} - \frac{x_2^2}{2} & \frac{\omega_2}{3} - \frac{x_2^3}{3} & \cdots & \frac{\omega_2}{(n+1)} - \frac{x_2^{(n+1)}}{(n+1)} \\ \vdots & \vdots & & \vdots \\ \frac{\omega_j}{2} - \frac{x_j^2}{2} & \frac{\omega_j}{3} - \frac{x_j^3}{3} & \cdots & \frac{\omega_j}{(n+1)} - \frac{x_j^{(n+1)}}{(n+1)} \\ \vdots & \vdots & & \vdots \\ \frac{\omega_m}{2} - \frac{x_m^2}{2} & \frac{\omega_m}{3} - \frac{x_m^3}{3} & \cdots & \frac{\omega_m}{(n+1)} - \frac{x_m^{(n+1)}}{(n+1)} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \vdots \\ \alpha_{n-1} \\ \alpha_n \end{pmatrix} = \begin{pmatrix} x_1 - \omega_1 \\ x_2 - \omega_2 \\ \vdots \\ \vdots \\ x_{n-1} - \omega_{n-1} \\ x_n - \omega_n \end{pmatrix} \qquad (4.4)$$

where $\omega_j = \frac{\bar{p}_j - \eta}{\gamma - \eta}$.

If $m = n$, that is the number of observations is equal to the number of unknowns, (4.4) has a unique solution provided that the matrix is non-singular. However, in general there are fewer observational points than unknowns and so the system (4.4) is under-determined and has infinitely many solutions. We therefore seek a solution which gives us something over other solutions of the problem, such as the solution which minimises $||\boldsymbol{\alpha}||^2$, and the problem becomes one of optimisation.

There are many ways of finding a local minimum solution of a function, however the problem here is that of finding the global extreme. Many methods, including conjugate gradient methods (section 2), become stuck in local minimum and are unable to find their way out in order to reach the global minimum but are fine when there is only one. The other method we have looked at simulated annealing (section 3) is able to leave a local minimum in its search for the global minimum and can therefore be used for a wider variety of conditions when choosing one solution over another.

4.1.1 With Conjugate Gradients

The following results are from the least squares problem

$$A^*\boldsymbol{x} = \boldsymbol{b}^*$$

where $A^* = A^T A$ and $\boldsymbol{b}^* = A^T\boldsymbol{b}$, in order to give a square symmetric matrix which is required for the conjugate gradients method. We have A^* symmetric, since

$$(A^*)^T = (A^T A)^T = A^T A = A^*,$$

and positive (indefinite),

$$\boldsymbol{\lambda}^T A^* \boldsymbol{\lambda} = \boldsymbol{\lambda}^T A^T A \boldsymbol{\lambda} = \|(A\boldsymbol{\lambda})\|^2 \geq 0, \quad \text{for any } \boldsymbol{\lambda}.$$

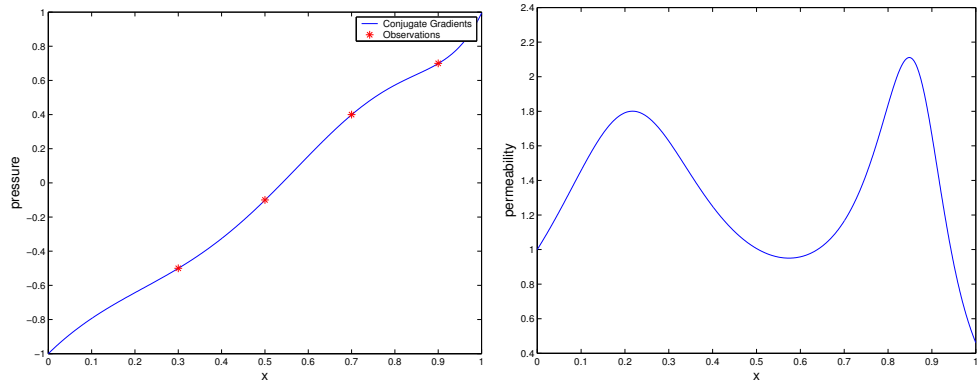


Figure 4.1: *4 observations, 6 unknowns: pressure and permeability results*

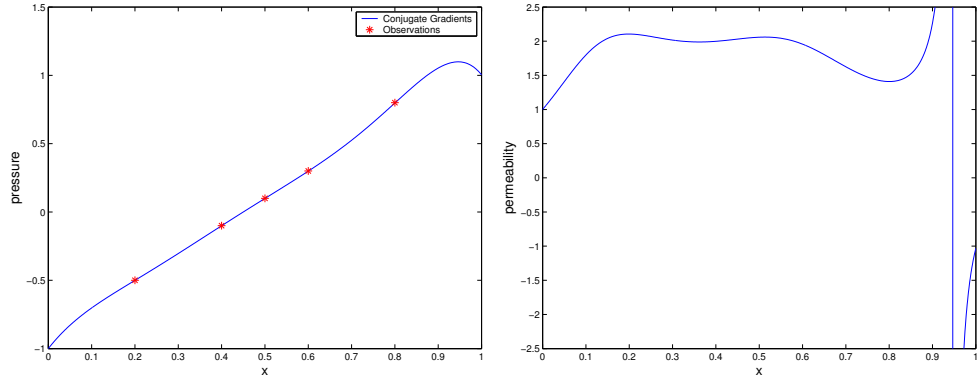


Figure 4.2: *5 observations, 6 unknowns: pressure and permeability results*

While the number of observations is equal to or less than the number of unknowns the pressure curve goes through the observation points. However, for the overdetermined case in figure 4.4A the distance between the observation and pressure returned is minimised

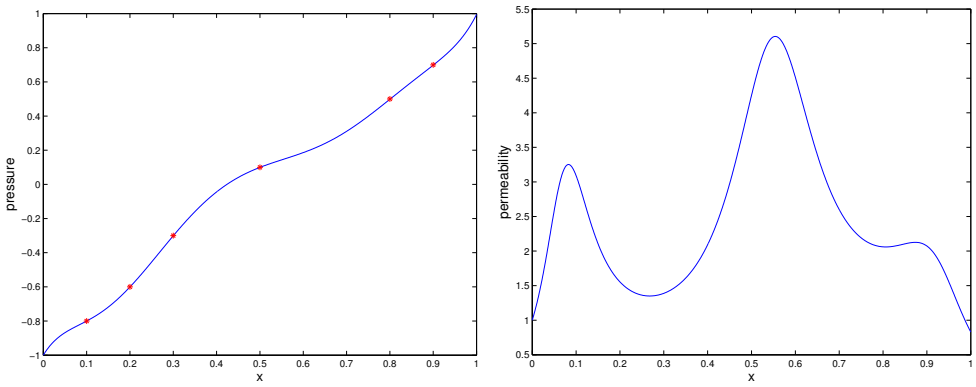
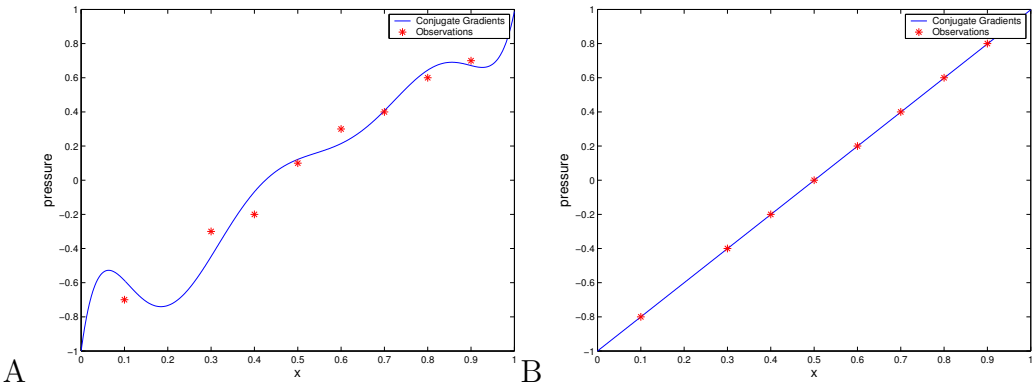


Figure 4.3: *6 observations, 6 unknowns*

in a least squares sense, i.e. the solution is that to the least squares problem.

$$\min_x \|A^* - b^*\|^2.$$



A B

Figure 4.4: *8 observations 6 unknowns*

Although figure 4.4B is also an over determined case and we would therefore not expect the curve to pass through all the observations, the values were chosen so that they were on the line $p = 2x - 1$ and so the program returns α values which would enable the curve to be very close to the straight line.

It is also clear from figure 4.2 that it is possible to obtain a zero pressure gradient, which is a result of the permeability k changing signs. Physically this is unrealistic as the permeability should be strictly positive and hence we expect the maximum principle to hold (section 1.2).

4.1.2 With Simulated Annealing

For simulated annealing we need to decide on an objective function we will aim to minimise for the set of available solutions, one such method is to minimise $\|\boldsymbol{\alpha}\|^2$ subject to the constraints $A\boldsymbol{\alpha} = \boldsymbol{b}$. The choice of $\|\boldsymbol{\alpha}\|^2$ as the cost function is primarily based on ease of manipulation and the physics of the problem may suggest other alternatives.

Consider the problem

$$A\boldsymbol{\alpha} = \boldsymbol{b}, \tag{4.5}$$

with A and $m \times n$ matrix, α an $n \times 1$ and b an $m \times 1$, where $m < n$. Reduce by Gaussian elimination so that the matrix A has the form

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & \cdots & \cdots & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & \cdots & \cdots & a_{2n} \\ \vdots & 0 & a_{33} & \cdots & \cdots & \cdots & a_{3n} \\ \vdots & \vdots & & \ddots & & & \vdots \\ 0 & 0 & \cdots & 0 & a_{mm} & \cdots & a_{mn} \end{pmatrix}$$

Then (4.5) can be rewritten in the form

$$\hat{A}\hat{\boldsymbol{\alpha}} + \tilde{A}\tilde{\boldsymbol{\alpha}} = \boldsymbol{b},$$

where

$$\hat{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & \cdots & a_{1m} \\ 0 & a_{22} & a_{23} & \cdots & a_{2m} \\ \vdots & 0 & a_{33} & \cdots & a_{3m} \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & \cdots & 0 & a_{mm} \end{pmatrix}, \quad \tilde{A} = \begin{pmatrix} a_{1(m+1)} & \cdots & \cdots & a_{1n} \\ a_{2(m+1)} & \cdots & \cdots & a_{2n} \\ a_{3(m+1)} & \cdots & \cdots & a_{3n} \\ \vdots & & & \vdots \\ a_{m(m+1)} & \cdots & \cdots & a_{mn} \end{pmatrix}$$

$$\text{and } \hat{\boldsymbol{\alpha}} = (\alpha_1, \cdots, \alpha_m)^T, \quad \tilde{\boldsymbol{\alpha}} = (\alpha_{m+1}, \cdots, \alpha_n)^T.$$

Now $\hat{\boldsymbol{\alpha}}$ depends directly on $\tilde{\boldsymbol{\alpha}}$ and can be found by solving

$$\hat{A}\hat{\boldsymbol{\alpha}} = \boldsymbol{b} - \tilde{A}\tilde{\boldsymbol{\alpha}}.$$

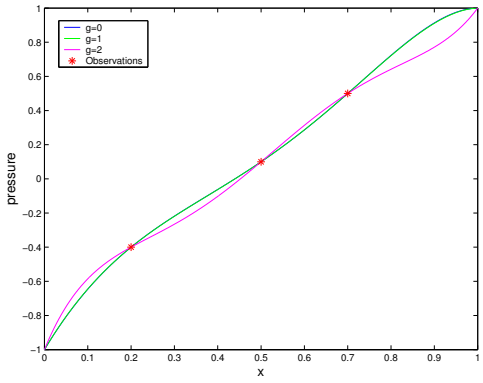


Figure 4.5: *3 observations,* $\Delta r = 0.3$, $\beta = 4.5$

Having found the dependent values of α the objective function

$$\|\boldsymbol{\alpha}\|^2 = \|\hat{\boldsymbol{\alpha}}\|^2 + \|\tilde{\boldsymbol{\alpha}}\|^2,$$

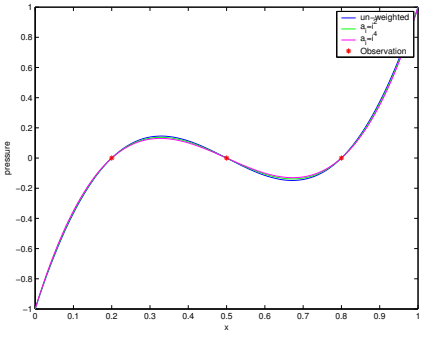
and only the independent α, the $\tilde{\boldsymbol{\alpha}}$, are passed through the random stepping of the simulated annealing algorithm.

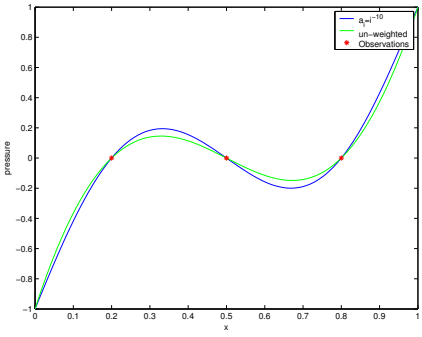
4.1.3 Simulated Annealing and a Smoother Curve

The methods of conjugate gradients, using $A^T A\boldsymbol{x} = \boldsymbol{b}$, and simulated annealing, minimising $\|\boldsymbol{\alpha}\|^2$, yield very similar results for the 3 observations case with 5 unknowns. However the advantage of conjugate gradients is the ease at which we are able to increase the number of observations beyond the number of unknowns which we are unable to do with the simulated annealing method outlined here due to the set up of the objective function. however, we are able to change the way in which the solution is selected with simulated annealing: by changing the objective function for example we can give a higher weighing to the higher order α to produce a smoother curve. Thus, using the objective function

$$\phi = a_1\alpha_1^2 + a_2\alpha_2^2 + a_3\alpha_3^2 + \cdots ,$$

where $a_{i+1} > a_i$ are the weighting constants. Although we have said this is unrealistic in that we have maximum and minimum pressure values inside the domain, for ease of seeing the effect consider the case where we have a cubic pressure field. The results are shown in figure 4.6.

A 



B

Figure 4.6: *comparing a weighted objective function with the unweighted* $\|\boldsymbol{\alpha}\|^2$

4.2 Two-Dimensions

As in for the one-dimensional case in chapter 4.1 we can consider the case where the reciprocal of the permeability, k^{-1}, is polynomial. For example

$$k(x,y) = \frac{1}{a + bx + cy + dxy + ex^2 + fy^2},$$

where a, b, c, d, e, f are constant coefficients. Consider Darcy's law

$$\boldsymbol{u} = k\boldsymbol{\nabla} p$$

Assuming $\boldsymbol{u}$ is constant, say $(1,2)^T$, then

$$k(x,y) \begin{pmatrix} \frac{\partial p}{\partial x} \\ \\ \frac{\partial p}{\partial y} \end{pmatrix} = \begin{pmatrix} 1 \\ \\ 2 \end{pmatrix} \tag{4.6}$$

which we can integrate to find an expression for the pressure, p, at (x,y). Since

$$\begin{aligned} \frac{\partial p}{\partial x} &= a + bx + cy + dxy + ex^2 + fy^2 \\ \Rightarrow \quad p &= ax + b\frac{x^2}{2} + cyx + dy\frac{x^2}{2} + e\frac{x^3}{3} + fy^2x + g(y), \end{aligned} \tag{4.7}$$

where g is an arbitrary function of y. Then comparing

$$\frac{\partial p}{\partial y} = 2(a + bx + cy + dxy + ex^2 + fy^2)$$

with

$$\frac{\partial p}{\partial y} = cx + d\frac{x^2}{2} + 2fxy + g'(y),$$

from (4.7), and equating coefficients we get

$$
\begin{aligned}
& c = 2b, \\
& \frac{d}{2} = 2e, \\
& f = d, \\
& 2(a + cy + fy^2) = g'(y), \\
& \quad \Rightarrow \quad g(y) = 2\left(ay + c\frac{y^2}{2} + f\frac{y^3}{3}\right) + p_1,
\end{aligned}
$$

say. Assuming the point $(0, 0)$ has a pressure value of 1, then $p_1 = 1$ and we get

$$(x + 2y)a + \left(\frac{x^2}{2} + 2yx + 2y^2\right) b + \left(\frac{yx^2}{2} + \frac{x^3}{12} + y^2x + \frac{2y^3}{3}\right) d = p(x, y) - 1. \qquad (4.8)$$

Writing this at the prescribed values of p as a system of m (the number of observations) linear equations in a, b, d we can apply the method of conjugate gradients as in section 4.1.1 to minimise the least squares problem.

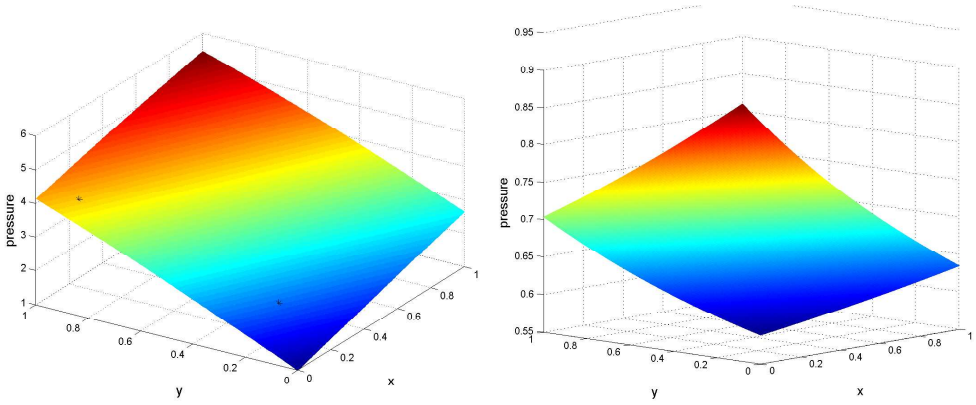


Figure 4.7: *2 observations, under-determined case and the corresponding permeability*

As before we get the solution passing through the observation points for the under-determined and exact cases as expected. Disadvantages of this method include the fact

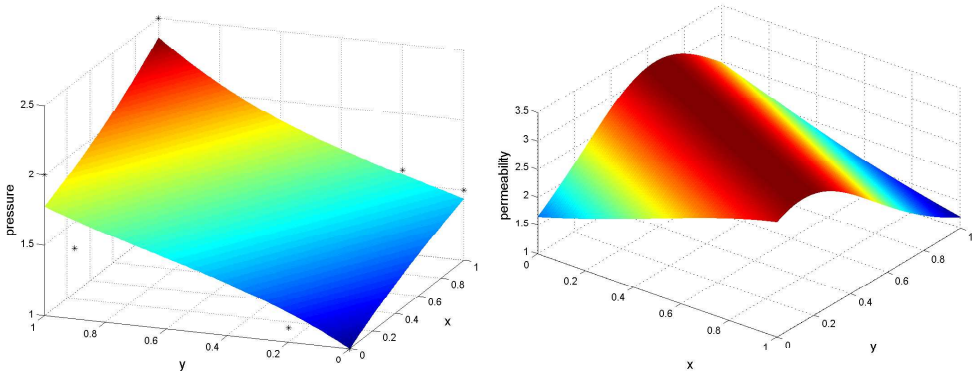


Figure 4.8: *8 observations, over-determined case and the corresponding permeability*

that inverse polynomial approximation is a special case and the possibility of zero pressure gradient occurring.

Chapter 5

Approximating the Pressure Field

In chapter 4 we assumed that we knew the form of the permeability, which was easily integrated to give an equation for the pressure at any point (x, y). However, what happens if we do not wish to make this assumption as to the form of the permeability field? An alternative method of representing the permeability by a finite number of degrees of freedom is to estimate k at discrete nodal values. However, the governing equation (1.1) cannot be easily integrated to produce a closed form equation for both k and p as in section 4. So in order to calculate the permeability we first need to estimate the pressure field.

One method can be found by assuming that the pressure field is comparable to a thin metal sheet that is given a displacement at a fixed number of positions (the observational points), assuming that the pressure field is smooth and continuous. For example in two dimensions the sheet is given displacements at the points (x_n, y_n) of u, with the force acting being f_n. The resulting displacements satisfy the fourth order differential equation

(Briggs citeBriggs)

$$\begin{aligned}\frac{\partial^4 u}{\partial x^4} + 2\frac{\partial^4 u}{\partial x^2 \partial y^2} + \frac{\partial^4 u}{\partial y^4} &= f_n, \quad x = x_n, \;\; y = y_n \\ &= 0, \quad \text{otherwise,}\end{aligned}$$

with boundary conditions

$$\begin{aligned}\frac{\partial}{\partial n}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) &= 0, \\ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} &= 0.\end{aligned}$$

Farmer [4] and Briggs [2] look at solving this equation by minimising the curvature of the field, invoking the principle of minimum curvature.

5.1 Principle of Minimum Curvature

The function $u(x, y)$ that minimises the curvature satisfies the fourth order differential equation

$$\frac{\partial^4 u}{\partial x^4} + 2\frac{\partial^4 u}{\partial x^2 \partial y^2} + \frac{\partial^4 u}{\partial y^4} = 0. \tag{5.1}$$

Proof

Consider the total squared curvature of $u = u(x, y)$

$$C(u) = \iint \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)^2 dxdy.$$

We need to show that if $u(x, y)$ makes C an extremum then it obeys 5.1 and that a function, $u(x, y)$ which obeys 5.1 minimises C.

Let $u(x, y)$ be a function on the region R^2 with boundary B, such that u makes C an extremum, and g an arbitrary function such that

$$g = 0 \quad \text{and} \quad \frac{\partial g}{\partial n} = 0 \quad \text{on } B.$$

Let

$$z(x, y) = u(x, y) + \epsilon g(x, y),$$

where ϵ is a real number. Then, as demonstrated by Briggs,

$$\left.\frac{\partial C(z)}{\partial \epsilon}\right|_{\epsilon=0} = 0,$$

which holds for all functions $g(x, y)$. Now

$$C(z) = \iint (\nabla^2 u)^2 dxdy + 2\epsilon \iint \nabla^2 u \nabla^2 g dxdy + \epsilon^2 \iint (\nabla g)^2 dxdy,$$

and so

$$\begin{aligned} \left.\frac{\partial C(z)}{\partial \epsilon}\right|_{\epsilon=0} &= 2 \iint \nabla^2 u \nabla^2 g dxdy, \quad \text{which using Green's Theorem} \\ &= 2 \left(\iint g \nabla^2 (\nabla^2 u) dxdy + \int_B \nabla^2 u \frac{\partial g}{\partial n} dl - \int_B g \frac{\partial}{\partial n} (\nabla^2 u) dl \right). \end{aligned}$$

This leaves

$$\iint g \nabla^2 (\nabla^2 u) dxdy = 0,$$

holding for all functions g and hence $\nabla^2(\nabla^2 u) = 0$.

Conversely, consider the case where $u(x, y)$ satisfies $\nabla^2(\nabla^2 u) = 0$ and $z(x, y)$ is any other function on the region R^2, with

$$z = u \quad \text{and} \quad \frac{\partial z}{\partial n} = \frac{\partial u}{\partial n} \quad \text{on } B.$$

We need to show that $C(u) \leq C(z)$.

$$\begin{aligned} C(z) - C(u) &= \iint [(\nabla^2 z)^2 - (\nabla^2 u)^2] dxdy, \\ &= \iint (\nabla^2 z - \nabla^2 u)^2 dxdy + \iint \nabla^2 u (\nabla^2 z - \nabla^2 u) dxdy, \\ &= \iint (\nabla^2 z - \nabla^2 u)^2 dxdy + 2 \left(\iint (z-u) \nabla^2 (\nabla^2 u) dxdy \right. \\ &\quad \left. + \int_B \nabla^2 u \frac{\partial}{\partial n}(z-u) dl - \int_B (z-u) \frac{\partial}{\partial n} (\nabla^2 u) dl \right). \end{aligned}$$

This gives

$$C(z) - C(u) = \iint (\nabla^2 z - \nabla^2 u)^2 dxdy \geq 0.$$

Therefore the function $u(x, y)$ which minimises the curvature satisfies 5.1.

5.2 Numerical Solution for One-Dimensions

Here we shall only consider the observation points which lie on grid points. For the equivalent equations when this is not the case see Briggs [2].

In one dimension the total curvature is

$$C = \sum_{i=1}^{I} (c_i)^2,$$

where c_i is the curvature at the point x_i, given by

$$c_i = \frac{u_{i+1} + ui - 1 - 2u_i}{h^2}$$

for points away from the two ends of the field. It is assumed at these points the curvature is zero. We require the minimum of the curvature, i.e

$$\frac{\partial C}{\partial u_i} = 0.$$

There are five cases to consider when calculating the total curvature at each point: the two ends, the two points next to the ends and then the inner points. For the total curvature at the inner points we need to consider for the total curvature are c_{i-1}, c_i, c_{i+1} since these are the only ones which depend on u_i. Therefore the equation for any point i, with $2 < i, I - 2$, is given by

$$\frac{\partial C}{\partial u_i} = \frac{\partial}{\partial u_i}\left((c_{i-1})^2 + (c_i)^2 + (c_{i+1})^2\right) = 0,$$

in other words

$$u_{i-2} + u_{i+2} - 4(u_{i-1} + u_{i+1}) + 6u_i = 0.$$

A system of equations is constructed by sweeping through the grid points with their corresponding equations, an observational grid point having the simple equation $u_i = p$, where p is the pressure value at that point. These can be solved iteratively using conjugate gradients or as outlined in Briggs [2].

5.2.1 One-Dimensional Results for Pressure

Unlike the case in which the form of the permeability function is prescribed the end points are now unconstrained and free to move in such a manner as to smooth the curve. Figures 5.1 and 5.2 show the smoothed pressure data achieved by taking the initial pressure field of a weighted average of the observational data, depending on their distance from the grid point. Figure 5.3 also shows initial pressure with the smoothed data.

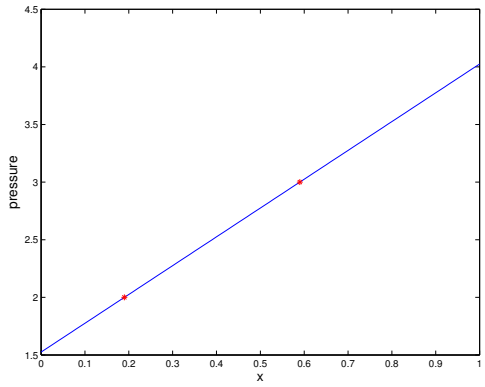


Figure 5.1: *Minimum curvature with 2 observations*

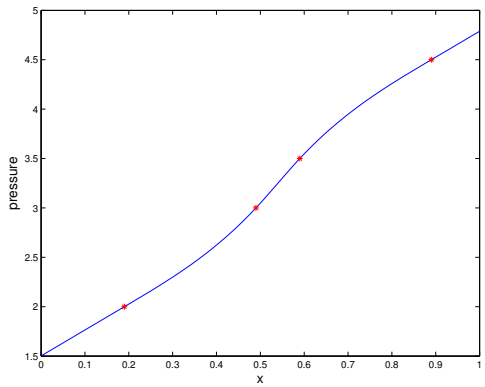


Figure 5.2: *Minimum curvature with 4 observations*

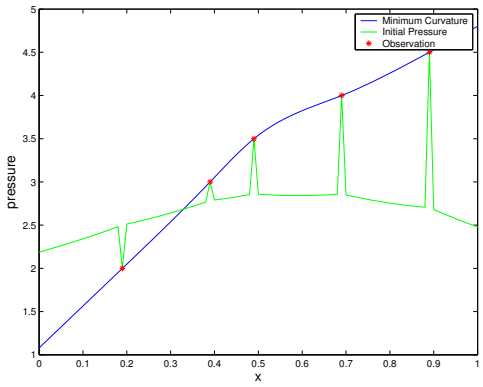


Figure 5.3: *Minimum curvature with 5 observations, compared with initial pressure*

5.3 Numerical Solution for Two-Dimensions

Similarly to before the total curvature is given by

$$C = \sum_{i=1}^{I} \sum_{j=1}^{J} (c_{i,j})^2,$$

where $c_{i,j}$ is the curvature at the point (x_i, y_j). Generally this is given by

$$c_{i,j} = \frac{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}}{h^2}.$$

The curvature at each point is shown in figure 5.4 with the curvature at the corners assumed to be zero.

Away from the edges the grid point value $u_{i,j}$ is present in the expressions for $c_{i,j}$, $c_{i\pm1,j}$, $c_{i,j\pm1}$ and so only these 5 need to be considered when a point away from the edge is being dealt with, ie in the interior

$$\frac{\partial C}{\partial u_{i,j}} = \frac{\partial}{\partial u_{i,j}} \left(c_{i,j} + c_{i+1,j} + c_{i-1,j} + c_{i,j+1} + c_{i,j-1} \right),$$

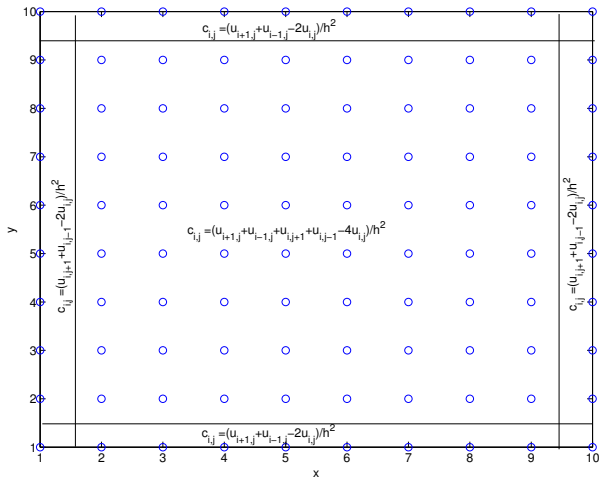


Figure 5.4: *Curvature giving at each point, in terms of surrounding values*

giving the equation for the interior points as

$$u_{i+2,j} + u_{i-2,j} + u_{i,j+2} + u_{i,j-2} + 2(u_{i+1,j+1} + u_{i-1,j+1} + u_{i-1,j-1} + u_{i+1,j-1})$$
$$-8(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) + u_{i,j} = 0.$$

Similar equations can be worked out for the other areas of the domain.

5.3.1 Two-Dimensional Results for Pressure

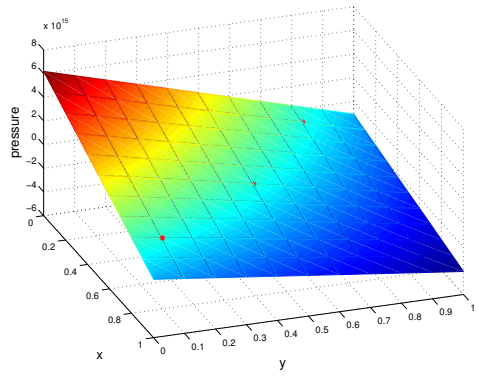


Figure 5.5: *Two dimensional minimum curvature with 3 observations*

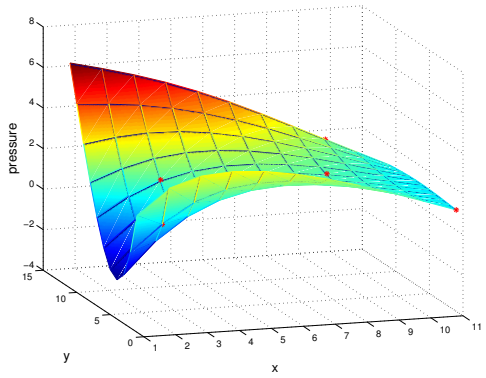


Figure 5.6: *Two dimensional minimum curvature with 5 observations*

Chapter 6

Finding the Permeability

6.1 In One Dimension

Previously in chapter 4 we looked at a given permeability form

$$k(x) = \frac{1}{1 + \alpha_1 x + \alpha_2 x^2 + \cdots},$$

now we look at using the approximated pressure field to find the permeability at grid points, assuming that we do not know the permeability field. Consider the one-dimensional case

$$\begin{aligned} \frac{d}{dx}\left(k\frac{dp}{dx}\right) &= 0 \\ k\frac{dp}{dx} &= c, \text{ a constant} \\ k &= \frac{c}{\frac{dp}{dx}}. \end{aligned} \tag{6.1}$$

Let us assume that $c = 1$ then at the point x_i

$$k_i = \frac{h}{p_{i+1} - p_i}.$$

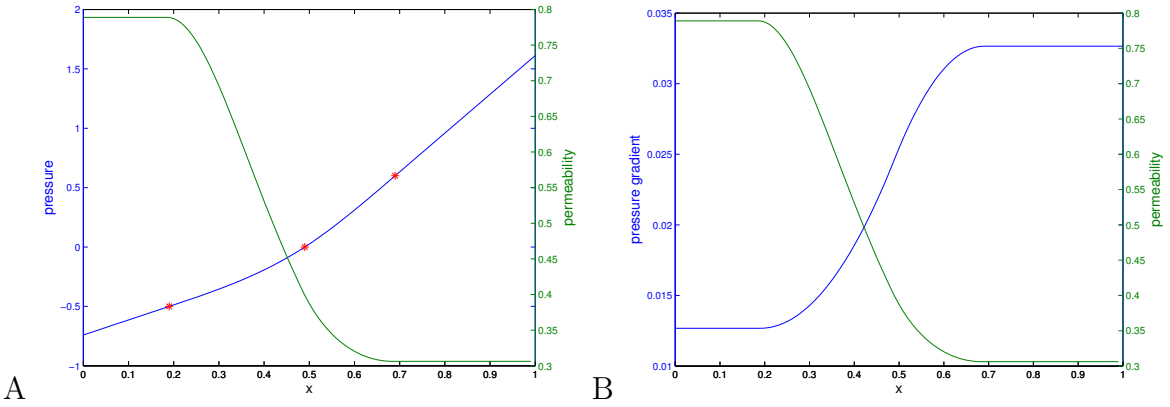


Figure 6.1: *one dimensional pressure and permeability field (A), pressure gradient and permeability (B)*

Notice that whilst the pressure gradient is constant we have a constant permeability, since

$$\frac{dp}{dx} = \varphi \text{ (a constant)} \quad \Rightarrow \quad k = \frac{c}{\varphi},$$

with variations occurring in k when the pressure gradient is also variable. See figures 6.1 and 6.2

As $|\frac{dp}{dx}| \to 0$ $|k| \to \infty$ and so at these points a singularity occurs, shown in figure 6.3. As previously mentioned this occurs due to the change in sign of k, which realistically is always positive leading to the maximum principle.

6.2 In Two-Dimension

In one dimension we were able to integrate equation (1.1) simply to find an equation for the permeability based on the previously found pressure field. Note that there is a first

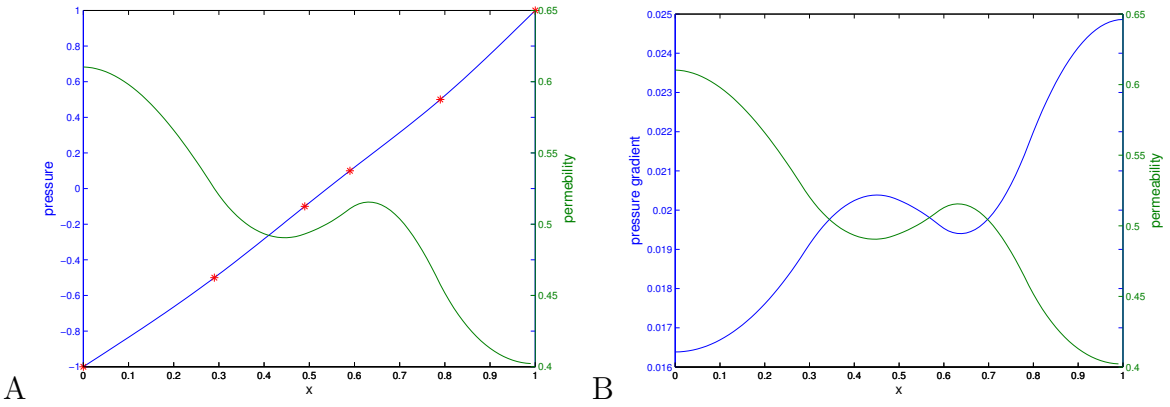


Figure 6.2: *one dimensional pressure and permeability field (A), pressure gradient and permeability (B)*

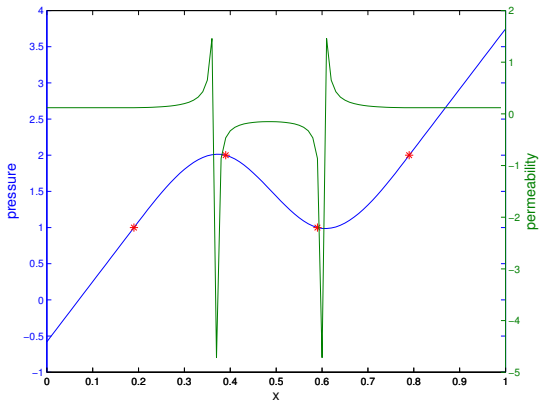


Figure 6.3: *Example of when the pressure has a maximum or minimum inside the domain, approximation of k*

order differential equation for k analogous to equaiotn (6.1).

$$\nabla.(k\nabla p) = 0,$$
$$k\nabla^2 p + \nabla k.\nabla p = 0,$$
$$\nabla \ln k.\overline{\boldsymbol{v}} = r,$$
$$\frac{\partial \ln k}{\partial \overline{\boldsymbol{v}}} = r,$$

where $\overline{\boldsymbol{v}} = \nabla p$ and $r = -\nabla^2 p$. However, unlike before this integration is not simply approximated as now we are required to integrate along the streamline directions $\overline{\boldsymbol{v}}$. Due to the scope of this project we outline an algorithm below by which we can find the permeability based on the pressure fields in section 5.3 and leave this as an area of further research.

Outline of Algorithm

1. Use the minimisation of the curvature to find the two dimensional pressure field.
2. Use the pressure field to determine the streamline directions, $\overline{\boldsymbol{v}} = \nabla p$, and the forcing term $r = -\nabla^2 p$.
3. Use a semi-lagrangian type method to integrate k forwards along the streamlines from specified values at the inflow portions on the boundary.

Chapter 7

Conclusions and Further Work

The focus of this dissertation has been on the problem of fining a numerical solution from a minimal amount data for a large area, with the direct aim of finding the pressure and permeability values across an oil field. A similar situation also arises in other areas such as weather prediction and oceanography but the problem remains essentially the same. Although data assimilation can also be employed we have looked at function fitting and interpolation as a means for generating an overall picture of the quantity in question, leading to cases where we have needed to solve an under-determined set of equations using optimisation techniques such as simulated annealing and conjugate gradients (by solving the equation $A^T A\boldsymbol{x} = A^T\boldsymbol{b}$ to minimise the norm $\|A\boldsymbol{x} - \boldsymbol{b}\|$).

For both one and two dimensions we were able to fit an analytical solution for the permeability to the governing equation. This lead to an equation for the pressure across the domain. As we expected for the case where we had more unknowns than observations (the under-determined situation) the pressure curve passed through them.

In one-dimensions it was shown that the method of Conjugate gradients was able to

do adjust to the changing number of observations easily which we were unable to do with the way the simulated annealing objective function had been set up. However we were able to look at other objective functions and minimise alternatives to $\|\alpha\|^2$, such as using a weighted function of α to give a smoother curve. It would be interesting to see what results other objective functions yield and decisions on the if one gives a more optimal solution over others. Together with the performance of conjugate gradients and simulated annealing against other optimisation techniques such as quadratic programming.

The main disadvantages of fitting a function to the permeability were a) the need to fix at least one point, to calculate constants in the pressure equation, and b) assuming the form of the permeability. An alternative, looked at in chapters 5 and 6, is to estimate the two values at discrete mesh points. Pressure being approximated by minimising the curvature and permeability using equation (1.1) and the previously calculated pressure values. The results in one dimension show the direct link of the permeability to the pressure gradient and the need for this to be non-zero.

A consideration for further work would be to look at what happens when we start with a permeability field, find pressure values from this data, pick out observations,and then try and regenerate the permeability field. We would hope to end with a similar permeability field to the one we started with, however approximation errors are likely to be introduced. The nature of these errors and the effect they have on the final output would be an interesting area for further research. For example, does increasing the number of observation points decrease the error introduced, and is it possible to predict the optimal positions for the observation points.

Bibliography

[1] Bohachevsky, I.O., Johnson, M.E., Stein, M.L. Generalized Simulated Annealing for Function Optimization. Technometrics, vol 28 (1986)

[2] Briggs, I.C. Machine Contouring Using Minimum Curvature. Geophysics, vol 39 No.1 p39-48 (1974)

[3] Brooks, S.P., Morgan, B.J.T. Optimization Using Simulated Annealing. The Statistician, vol 44 No.2 (1995)

[4] Farmer, C.L. Geological Modeling and Reservoir Simulation

[5] Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P. OPtimization by Simulated Annealing. Science, vol 220 (1983)

[6] Protter, M.H., Weinberger, H.F. Maximum Principles in differential Equations. Springer-Verlag (1984)

[7] Shewchuk, J.R. An Introduction to the Conjugate Gradient Method Without the Agonizing Pain. School of Computer Science, Carnegie Mellon University Pittsburgh (1994)

[8] Smith, G.D. Numerical Solution of Partial Differential Equations: Finite Difference Methods. Clarendon Press Oxford (1979)