SOME APPLICATIONS TO MINE GEOTECHNICAL DESIGN USING MATLAB®

Esteban Hormazabal Zúñiga
Senior Geotechnical Engineer, MSc. SRK Consulting.
ehormazabal@srk.cl

Pamela Ortubia Fernández
Geotechnical Engineer. SRK Consulting.
portubia@srk.cl

Francisco Rovira Frez
Geotechnical Engineer. SRK Consulting.
frovira@srk.cl

Matilde Salamanca 736 Piso 6 Providencia Santiago CHILE.
Tel: (56-2) 269 0353 Fax: (56-2) 269 0322.

ABSTRACT.

The objective of this paper is to use MatLab® as a computer tool to solve geotechnical problems involved in underground mine and open pit design. Regional tectonic, in situ stresses and other geological variables control the stress field in the rock surrounding a mine opening. Knowledge of the magnitudes and directions of these in situ and induced stresses is an essential component of underground excavation design. Large sets of field measurements (overcoring methods and hydrofracturing) can be analyzed and in situ stress tensor can be estimated for underground mine design.

1. - INTRODUCTION MatLab® is a commercial package for numerical analysis. It is based on the concept that data can be represented as arrays of numbers, either in rows or columns (one dimensional arrays are called vectors) or both (two dimensional arrays are called matrices). Then most of the operations of numerical analysis can be carried out using linear (or matrix) algebra. Some of these operations are quite complicated to program using ordinary programming languages like BASIC, FORTRAN, Pascal, or C - but they have been incorporated as single commands in MatLab®.

The objective of this paper is to use MatLab® as a computer tool to solve well known problems encountered in geotechnical practice. For the most part the underlying mathematics of such problems has been understood for many years, but the practical evaluation of solutions, even when they are in the form of charts, can be tedious and time-consuming. The main benefit of using MatLab® applications with such problems is the ability to obtain accurate results rapidly and thus to perform parametric studies quickly enough to be useful in a design context. Three generic cases were analyzed:

1. Principal stresses determination for an underground mine.
2. Estimation of quantity of flow for an impervious dam.
3. Determination of hydraulic parameters from a pumping test.

A detailed mathematical background is out of the scope of this paper and just an overview is provided.

2. METHODOLOGY.

2.1. PRINCIPAL STRESSES: The rock in which excavation occurs is stressed by gravitational, tectonic and others forces and methods exist for determining the stresses at a mine site. There is a pre-existing stress state in the ground that need to be understood, both directly and as the state stress applies to analysis and design. Stress is a tensor quantity with magnitude and direction, and with reference to the plane it is acting across (e.g. stress, strain and permeability).

The stress at a point within a rock mass has three normal stress components acting perpendicular to the faces of a small cube, and six shear stress components acting along the faces (see Figure 1), a total of nine stress components (Hudson & Harrison, 1997).

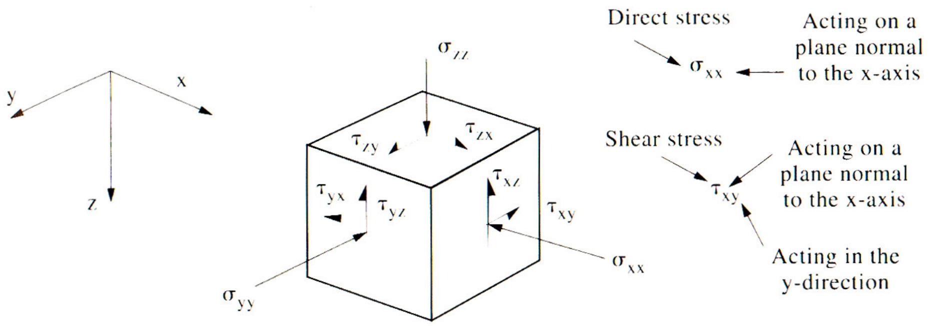

Figure 1.- The normal and shear stress components on an infinitesimal cube in the rock aligned with the Cartesian axes (taken from Hudson & Harrison, 1997).

The individual stress components are listed in the stress matrix in Figure 2. The elemental cube shown in Figure 1 is in equilibrium and, by taking moments about the axes, the complementary shear stresses are found to be equal. This means that the nine component stress tensor has six independent components. Hence, whenever the rock stress is specified, six independent pieces of information must be given (Hudson et al., 2003). The stress state is specified either by: (a) the three normal stresses and the three shear stresses acting on the three specified orthogonal planes determined by a set of x; y and z axes; or (b) the magnitudes and directions of the three principal stresses. When the elemental cube shown in Figure 1 is rotated, the stress components on the faces change in value. There is always one, and only one, cube orientation at which all the shear stress component values are zero. When this occurs, the cube faces represent the principal stress planes. The normal stresses on these planes are the principal stresses (see Figure 3).

$$\left[\begin{array}{c:cc} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \hdashline \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{array}\right]$$

The components in a row are the components acting on a plane; for this top row, the plane on which σ_{xx} acts.

The components in a column are the components acting in one direction; for this column, the x direction.

Figure 2.- The components of the stress matrix referred to given x, y and z axes (modified from Hudson et al. 2003).

The stress state at a given point in a rock mass is generally presented in terms of the magnitude and orientation of the principal stresses. Any system utilized for estimating the in situ stress state must involve a minimum of six independent measurements. The four main methods recommended by the International Society for Rock Mechanics (ISRM) are shown in Figure 4.

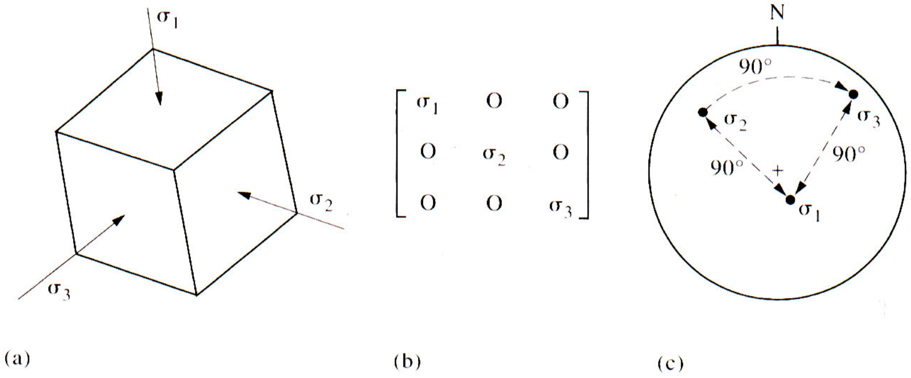

Figure 3.- (a) Principal stresses acting on a small cube. (b) Principal stresses expressed in a matrix form. (c) Principal stress orientation shown on a hemispherical projection (taken from Hudson & Harrison, 1997).

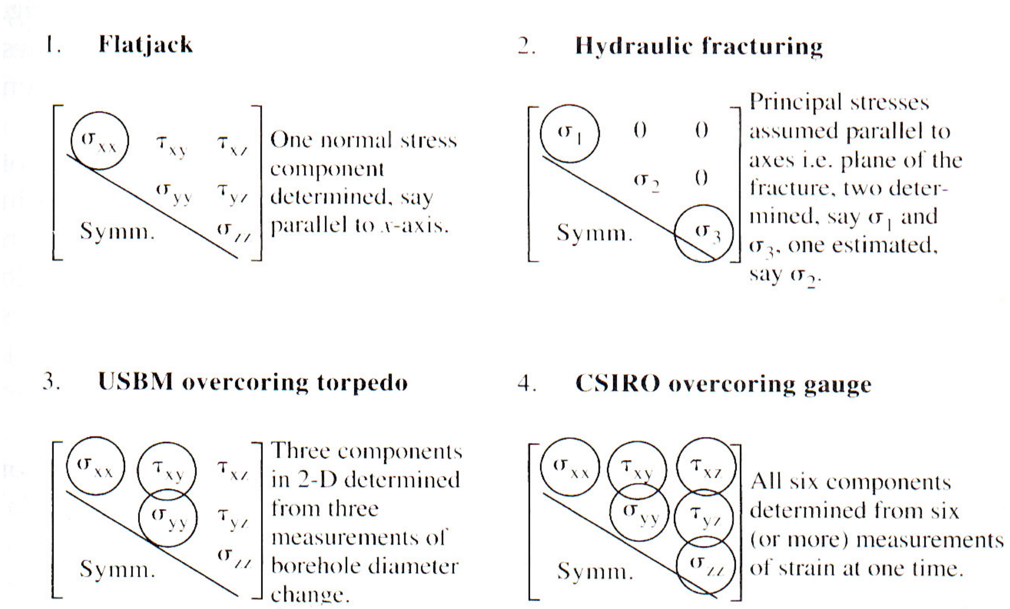

Figure 4.- The four ISRM suggested methods for rock stress determination and their ability to determine the components of the stress tensor (taken from Hudson & Harrison, 1997).

When a full description of the state of stress requires that all three components of the coordinate system be considered, the algebra becomes considerably more complicated. The stress equation to conduct stress transformation is as follows (see Goodman, 1989, Hudson & Harrison, 1997, Brady & Brown, 2004, among others):

$$\begin{bmatrix} \sigma_l & \tau_{ml} & \tau_{nl} \\ \tau_{lm} & \sigma_m & \tau_{nm} \\ \tau_{ln} & \tau_{mn} & \sigma_n \end{bmatrix} = \begin{bmatrix} l_x & l_y & l_z \\ m_x & m_y & m_z \\ n_x & n_y & n_z \end{bmatrix} \begin{bmatrix} \sigma_x & \tau_{yx} & \tau_{zx} \\ \tau_{xy} & \sigma_y & \tau_{zy} \\ \tau_{xz} & \tau_{yz} & \sigma_z \end{bmatrix} \begin{bmatrix} l_x & m_x & n_x \\ l_y & m_y & n_y \\ l_z & m_z & n_z \end{bmatrix} \quad \text{or}$$

$$[\sigma]_{lmn} = [R][\sigma]_{xyz}[R]^T \qquad \text{Eq.1}$$

Eq.1 means that, if we know the stresses relative to the xyz axes (i.e. σ_{xyz}) and the orientation of the *lmn* axes relative to the *xyz* axes (i.e. *R*), we can then compute the stresses relative to the *lmn* axes (i.e. σ_{lmn}).

Where the stress components are assumed known in the x-y-z coordinate system and are required in another coordinate system *l-m-n* inclined with respect to the first. The term l_x is the direction cosine of the angle between the *x*-axis and *l*-axis. Physically, it is the projection of a unit vector parallel to *l* on to the *x*-axis, with the other terms similarly defined. Expanding this matrix equation in order to obtain expressions for the normal component of stress in the l-direction and shear on the l-face in the m-direction gives:

$$\sigma_l = l_x^2\sigma_{xx} + l_y^2\sigma_{yy} + l_z^2\sigma_{zz} + 2(l_x l_y \sigma_{xy} + l_y l_z \sigma_{yz} + l_z l_x \sigma_{zx}) \qquad \text{Eq.2}$$

$$\sigma_{lm} = l_x m_x \sigma_{xx} + l_y m_y \sigma_{yy} + l_z m_z \sigma_{zz} + (l_x m_y + l_y m_x)\sigma_{xy} + (l_y m_z + l_z m_y)\sigma_{yz} + (l_z m_x + l_x m_z)\sigma_{zx}$$

Eq.3

The other four necessary equations (i.e. for σ_y, σ_z, τ_{yz} and τ_{zx}) are found using cyclic permutation of the subscripts in the equations above (*x:l; y:m* and *z:n*).

It is generally most convenient to refer to the orientation of a plane on which the components of stress are required using Dip Direction/Dip angle notation (α, β). The dip direction is measured clockwise bearing from North and the dip angle is measured downwards from the horizontal plane. If it used a right-handed coordinate system with x=north, y=east and z=down, and take n as the normal to the desired plane, then:

$n_x = \cos\alpha_n \cos\beta_n$;

Eq.4

$n_y = \sin\alpha_n \cos\beta_n$;

Eq.5

$n_z = \sin\beta_n$

Eq.6

and the rotation matrix becomes (see Goodman, 1989, Hudson & Harrison, 1997, Brady & Brown, 2004, among others):

$$\begin{bmatrix} l_x & l_y & l_z \\ m_x & m_y & m_z \\ n_x & n_y & n_z \end{bmatrix} = \begin{bmatrix} \cos\alpha_l \cos\beta_l & \sin\alpha_l \cos\beta_l & \sin\beta_l \\ \cos\alpha_m \cos\beta_m & \sin\alpha_m \cos\beta_m & \sin\beta_m \\ \cos\alpha_n \cos\beta_n & \sin\alpha_n \cos\beta_n & \sin\beta_n \end{bmatrix} \qquad \text{Eq.7}$$

Thus at least one orthogonal coordinate system (*l*, *m*, *n*) for which all the shear components vanish and only the three normal components remain (see Figure 3). It can be shown that the principal stresses are the three values of σ_p that satisfy the equation:

$$(\sigma_{xyz} - \sigma_p I)v_i = 0 \quad \text{or} \begin{bmatrix} \sigma_{xx} - \sigma_p & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} - \sigma_p & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} - \sigma_p \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = [0] \qquad \text{Eq.8}$$

Where the σ_i are the eigenvalues and the v_i are the eigenvectors. The solution to Eq.7 is obtained by finding the roots of the determinant:

$$\left| \sigma_{xyz} - \sigma_i I \right| = 0 \qquad \text{Eq.9}$$

Solution of the Eq.9 by some general methods, such as a complex variable method, produce three real solutions for the principal stresses and each one is related to a principal stress axis, whose direction cosines can be obtained directly from Eq.8. Thus, the procedure for calculating the principal stresses and the orientations of the principal stress axes is simply the determination of the eigenvalues of the stress matrix, and the eigenvector for each eigenvalue.

A computer MatLab® script was written to calculate eigenvalues of the stress matrix, and eigenvector for each eigeinvalue. These functions are included in MatLab® entering the following command:

```
[V, D] = eig(R)
```

where R is the stress matrix. V is the matrix of eigenvectors-each column gives the cosine directions for the corresponding principal components- and D is a diagonal matrix with the eigenvalues in the principal diagonal and zeros elsewhere. Table 1.1 shows different in situ measurements of stress and the principal stress tensor was obtained. The eigenvalues are shown in matrix (1), and the corresponding eigenvectors are shown in matrix (2). MatLab® script is provided in Appendix A.

Site	σ_{xx}	σ_{yy}	σ_{zz}	τ_{xy}	τ_{xz}	τ_{yz}
1	34.1	25.3	33.7	4.2	9.7	-10.5
2	17.9	26.4	29.8	-2.2	6.1	-18.3
3	24.1	20.7	46.9	1.9	5.9	2.3
4	30.9	42.8	24.4	6.6	-0.2	4.0

Table 1.1- Data from different sites.

$$D = \begin{bmatrix} 20.486 & 0 & 0 \\ 0 & 30.406 & 0 \\ 0 & 0 & 38.358 \end{bmatrix} \times (MPa) \tag{1}$$

$$V = \begin{bmatrix} -0.66423 & 0.55129 & 0.50486 \\ 0.68423 & 0.72036 & 0.11362 \\ 0.30104 & -0.42091 & 0.85569 \end{bmatrix} \tag{2}$$

The trends and plunges of the principal axes relative to a bearing reference frame with y north, x east, and z up are 324.4/58.8, 223.5/6.5 and 129.7/30.3, respectively. The trend azimuth is measured relative to the north-south (y) axis, the algebraic signs of *x* and *y* determines the appropriate quadrant. Note that principal stresses are shown in matrix (1), in reverse order according to MatLab® convention.

2.2. ESTIMATION OF QUANTITY OF FLOW FOR AN IMPERVIOUS DAM: Muskat (1936) used the Schwarz-Christoffel transformation[1] (see Weaver, 1932; Harr, 1991 among others) and elliptic integrals[2] to find the closed form solution of seepage under an impervious dam over a finite thickness layer (Figure 5):

$$\frac{\mathrm{Q}}{\mathrm{kh}} = \frac{1}{2}\frac{K'(m')}{K(m)}; \qquad \text{Eq.10}$$

$$m = \tanh\frac{\pi w}{4b} \qquad \text{Eq.11}$$

$$m' = \sqrt{1-m^2} \qquad \text{Eq.12}$$

where:

Q is the quantity of flow under an impervious dam.
h is the head loss in the dam.
k is the hydraulic conductivity of the soil.
w is the width of the dam.
b is the soil layer thickness.

---

[1] This transformation is a special conformal mapping technique for the determination of a function which will transform a problem from a geometrical domain within which a solution is sought one within which the solution is known using elements of complex variable theory.

[2] In groundwater problems, as a consequence of the Schwarz-Christoffel transformation, often encounters integrals of the form:

$$\int R[x, \sqrt{P(x)}]dx$$

where *R(x)* is a rational function of *x*, and *P(x)* is a polynomial in x. In particular, if *P(x)* is of third or fourth degree in *x*, and the integral cannot be expressed by elementary functions alone, this equation is called an *elliptic integral*.

K is the complete elliptical integral of the first kind with modulus m[3].
K' is the complement of the elliptical integral of the first kind with modulus m'.

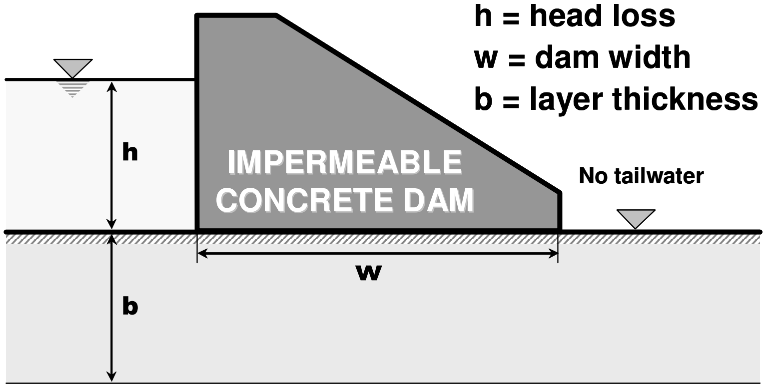

Figure 5.- Cross section of an impermeable concrete dam on layer of finite depth.

A computer MatLab® script was written to calculate the complete elliptic integral K and K' with modulus m and m'. These functions are included in MatLab® entering the following command:

K1 =ellipke(m1)

where K1 is the complete elliptical integral of the first kind with modulus m1.

Normalized seepage by head loss and soil hydraulic conductivity is shown in Figure 6, as a function of w/b, the ratio of the dam width to the thickness of soil layer. It will be seen that, seepage decreases with increasing w/b from infinitely large values for w/b ~ 0 to vanishing values for w/b → ∞. MatLab® script is provided in Appendix A.

---

[3] Complete tables of K and K' may be found in Groundwater and Seepage by M. Harr (1991).

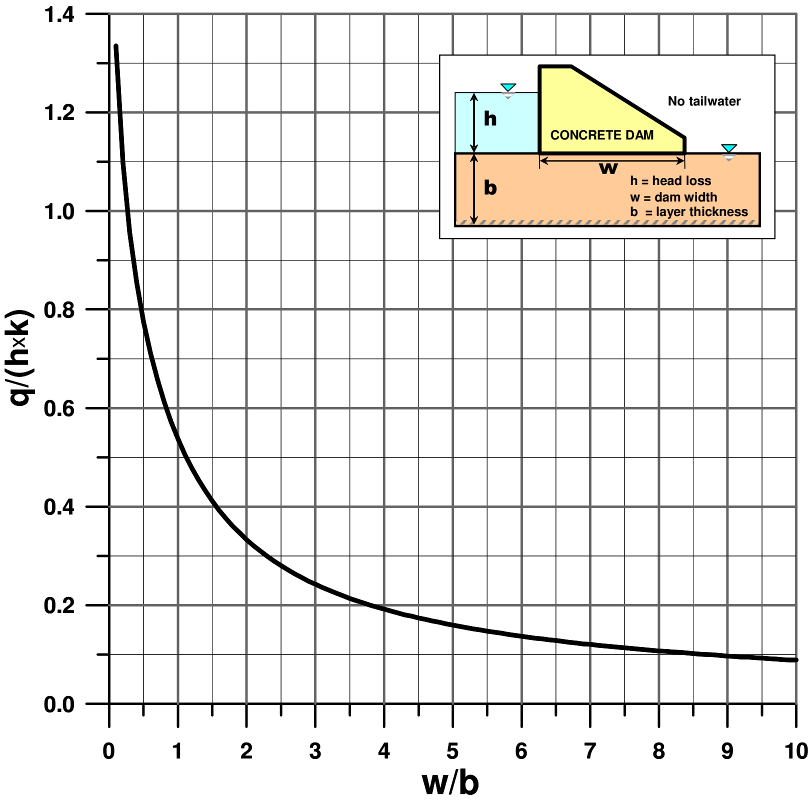

Figure 6.- Seepage under an impermeable concrete dam based on the closed form solution derived by Muskat (1936).

2.3. DETERMINATION OF HYDRAULIC PARAMETERS FROM A PUMPING TEST: Determination of aquifer parameters by pump test methods is a standard procedure in groundwater problems. The pump test concept is simple. A production well is discharged at a known rate and the effect of the pumping is measured in the observation wells at certain time intervals. The time-drawdown data are plotted on logarithmic graph paper. A common method of interpreting the data involves matching of the data curve with the type curve or the family of type curves. When the “best” match position is selected, the coordinates of the match-point are entered in the approximate equations for calculating the aquifer parameters. A constant term relates drawdown and W(u) and time and 1/u. These relationships are the basis for the curve matching technique. These values are then used to compute the transmissivity (T) and storativity (S).

The described methods of pump test data analysis are based on graphical procedures and as such are subject to errors in matching curves and in reading the coordinates of the chosen match-point. The procedure, which is done after the pump test, is tedious because it requires plotting data by hand.

The Theis equation for unsteady drawdown in a confined aquifer is written as (Theis, 1935):

$$\Delta h = \frac{Q}{4\pi T} W(u) \qquad \text{and} \qquad u = \frac{r^2 S}{4Tt} \qquad \text{Eq.13}$$

where,

$$W(u) = \int_u^\infty \frac{e^{-u}}{u} du = -\gamma - Ln(u) - \sum_{n=1}^{25} \frac{(-1)^n x^n}{nn!} \qquad \text{Eq.14}$$

where,

γ = 0 0.57721566490153 (Abramowitz & Stegun, 1964)
Δh = drawndown at the observation point, [L].
Q = discharge rate, [L^3T^{-1}].
r = distance from pumped well to observation point, [L].
S = storativity or storage coefficient [dimensionless].
T = transmissity, [L^2T^{-1}].
t = time since pumping started, [T].

The Theis equation is applicable if the following conditions are satisfied: aquifer is of infinite areal extent, aquifer is confined, homogeneous, isotropic, of uniform thickness, the well is pumped at a constant rate, the pumping well and the observation wells fully penetrate the aquifer, water is released from storage instantaneously, the diameter of the pumped well is small, i.e., the well storage can be neglected.

A computer MatLab® script was written to calculate transmissity knowing the storativity (storage coefficient). The well function W(u) is included in MatLab® entering the following command:

W(u) = Ei(u) = expint(u)

where u[4] is the expression shown in Eq. 13. MatLab® script is provided in Appendix A.

---

[4] An initial guess is required and can be based on Theim (1906) equation for steady-state confined flow:

$$T = \frac{Ln(r)}{2\pi\Delta h}$$

3. CONCLUSIONS AND FUTURE WORK.

This paper has described some functions included in MatLab® for use in three types of problems that arise frequently in geotechnical practice. It has also explained some of the analytical concepts underlying these tools.

The first example provides a simplification of a complicated engineering problem, for several in situ tests, estimate in situ stress field assuming that rock is not linearly elastic and isotropic; and not uniform.

The second example shows the closed form solution for seepage under an impervious dam considering an isotropic and homogeneous soil. This equation includes the effects of dam width and depth of layer under an impervious dam founded on a single pervious layer. Further future work with this topic should focus on a closer examination of the effects of anisotropy as well as effects of non-homogeneous soils (or multiple layers with different permeabilities). Also, pore pressure acting on the base of the dam should be analyzed for stability purposes.

The third example provides a good method to estimate transmissity assuming mainly that the aquifer is of infinite areal extent, is confined, homogeneous, isotropic and uniform thickness. Effects of fractures or joints on permeability should be considered for hydrogeologic purposes.

4. ACKOWLEDGEMENTS.

The author says a special thanks to Françoise Baramdyka from SRK Consulting Chile for meticulous proofreading.

5. – REFERENCES.

Abramowitz, M. and I. A. Stegun (1965). “Handbook of Mathematical Functions”. Chapter 5, New York: Dover Publications.

Brady B.H. & Brown E.T. (2004). “Rock Mechanics for Underground Mining”, 3rd ed., Kluwer Academic Publishers, London.

Goodman R. (1989). “Introduction to Rock Mechanics”, 2nd ed., John Willey & Sons, New York.
Harr, M.E. (1962). “Groundwater and Seepage”, 2nd ed., Dover Publications, New York.

Hudson J.A. & Harrison, J.P. (1997). “Engineering Rock Mechanics”, 2nd ed., Pergamon, UK.

Hudson, J.A., Cornet, F. H. & Christiansson, R. (2003). “ISRM Suggested Methods for Rock Stress Estimation-Part 1: Strategy for Rock Stress Estimation”. International Journal of Rock Mechanics & Mining sciences, 40 pp 991 – 998.

Muskat, M. (1936). “The Flow of Homogeneous Fluids through Porous Media”, McGraw-Hill Book Company, New York, 1937; reprinted by J. W. Edwards, Publisher, Inc., Ann Arbor.

Theim G. (1906). “Hydrologische Methoden”. J.M. Gebhart, Leipzig. 56.

Theis, C.V. (1935). “The relation between the lowering of the piezometric surface and the rate and duration of discharge of well using groundwater storage”. Trans. Am. Geophys. Un. 16, pp. 519 – 524.

Weaver, W. (1932). “Uplift pressure on Dams”. *J. Math. & Physics.* 11, 114.

APPENDIX A

Example 1

```
%Program to determine principal Stresses
S = xlsread('Stress_Data');
T = [mean(S(:,1)) mean(S(:,4)) mean(S(:,5)); mean(S(:,4)) mean(S(:,2)) mean(S(:,6));
mean(S(:,5)) mean(S(:,6)) mean(S(:,3))];
[W,D] = eig(T);
V=W'
for ii=1:3
   B(ii,1) = asin((V(3,ii)/(sqrt(V(ii,1).^2+V(ii,2).^2+V(ii,3).^2)))).*180/pi;
   A(ii,1) = atan(V(ii,1)/V(ii,2)).*180/pi;
   M(ii,1) = asin(V(ii,3)).*180/pi;
   ii=ii+1;
end
```

Example 2

```
%Program to determine Q, Muskat (1936)
woverb = 0:0.1:10;
m1 = tanh((pi/4)*woverb);
m2 = sqrt(1-m1.^2);
K1 =ellipke(m1);
K2 =ellipke(m2);
Q =(K2)./(2*K1);
plot(woverb,Q);
```

Example 3

```
% Program to determine Transmissivity when Storativity is known
x0 = [0.023; 14.697];          % Make a starting guess at the solution using Theim equation
options = optimset('Display','iter');   % Option to display output
[x,fval] = fsolve(@fun1,x0,options)  % Call optimizer

%fun1 is used in the example 3

function F = fun1(x)
r = 100; % (m)
t = 15; % (min)
h = 2.0; % (m)
q = 40.104; %(m^3/min)
s = 0.002;
F = [x(1) - (s*r^2)/(4*t*x(2));
    h/q - expint(x(1))/(4*pi*x(2))];
```