A METHOD FOR DETERMINATION OF THE SURFACE LAYER PARAMETERS USING WIND VELOCITY PROFILE

OMAR M.Y. ANBAR

Department of Meteorology, Faculty of Meteorology, Environment and Arid Land Agriculture, King Abdulaziz University, Jeddah, Saudi Arabia

Abstract

The aerodynamic roughness length, $z_o$, and displacement height, $d$, which obtained from profile wind speed data were calculated using nonlinear system by Newton Method (NM) which solves Jacobean matrices and use gauss elimination in order to find out the unknown values. After comparing some results obtained by the program (NM) with the Haenel (1993) results, it is found that there is no significant difference between the stability parameters because of the nearly similar wind direction $316^\circ$ and $313^\circ$ respectively. The friction velocity $u_*$ in both program and Haenel values were very close to each other.

Introduction

Newton method was used for non linear system, a system of equations include variables and unknowns.

For convergence, in solving the system of non linear equations, iteration was used and solving the system by using gauss elimination. We used (Newton method for systems Algorithm, fixed point iteration for non linear systems).

Many studies have been made to evaluate of roughness parameters over relative smooth surface using analytical or mathematical methods.

Mikami et al (1995), reported that roughness parameters $z_o$ and $d$ determined from vertical profiles of wind speed in the surface boundary layer. They used a capative balloon to obtain the mean vertical profiles of wind velocity, air temperature and humidity. Sets of data contain height, air temperature, wind speed and direction. All these data taken from the study area called Musashi Hills located northwest of the Kanto Plain in Japan. However, They carried out a profile measurement by capitative balloon and determined the values of roughness parameters $z_o$ and $d$ by the maximum correlation method.

2. Methodology

The mean wind profile can be written as

$$\frac{\partial U}{\partial z} = \frac{u_*}{kz}$$

(1)
Monin and Obukhov (1954) represented the deviation from the neutral form of the wind profile by writing

\[ \frac{\partial U}{\partial z} = \frac{u_*}{kz} \phi_u \left( \frac{z}{L} \right) \]  

(2)

where the function \( \phi_u \) is determined by theory or observation (Webb, 1965). Webb (1965) noted that \( \phi_u \) must approach 1 when \( \frac{z}{L} \) becomes small, and is expected to be less than or greater than 1 when \( \frac{z}{L} \) is negative or positive, respectively. Monin and Obukhov suggested that the function \( \phi_u \) may be represented as a power series

\[ 1 + \beta_1 \left( \frac{z}{L} \right) + \beta_2 \left( \frac{z}{L} \right)^2 + \ldots, \] 

and that in conditions not too far removed from neutral, it may be sufficiently accurate to take this only to the linear term, so that (2) becomes

\[ \frac{\partial U}{\partial z} = \frac{u_*}{kz} \left( 1 + \beta \frac{z}{L} \right) \]  

(3)

where \( \beta \) is written for convenience to denote \( \beta_1 \). With this approximation, integration of (2) gives the so called log-linear profile

\[ U = \frac{u_*}{k} \left[ \ln \left( \frac{z}{z_*} \right) + \beta \left( \frac{z - z_o}{L} \right) \right] \]  

(4)

Webb (1965), adopted in case of inversion profiles the value 5 for \( \beta \). This representation applies not only in conditions near to neutral but over a wide range of stable conditions with the limiting value of Richardson number 0.2.

Monin and Obukhov obtained from data of four expeditions the value of \( \beta = 0.62 \). The error in the \( \beta \) value is about 10%.

Brutsaert (1999) noted that the atmospheric surface layer of the vertical structure of the turbulent moments is described well by Monin-Obukhov similarity (MOS). Thus, one has for the profiles of the mean (in turbulent sense) wind speed

\[ u = \frac{u_*}{k} \left[ \ln \left( \frac{z - d}{z_*} \right) + \beta \left( z - z_* \right) \right] \]  

(5)

\[ \theta - \theta_* = \frac{\theta_*}{k} \left[ \ln \left( \frac{z - d}{z_{*h}} \right) + \beta_h \left( z - d - z_{*h} \right) \right] \]  

(6)

where \( z \) is the height above the zero displacement level \( d \); this displacement level is a parameter, which is commonly used to account for the uncertainty regarding the reference level, i.e. the origin of the vertical coordinate, in the application of
similarity above rough surfaces (e.g. Brutsaert, 1982). \( \beta \) was written instead of \( \psi \). Brutsaert (1999) used \( \psi_m \) in Eq.(5) while \( \psi_h \) in Eq.(6). The subscripts \( m \) and \( h \) stands for momentum and heat fluxes.

From Equations (5) and (6), the following equations which include known and unknown variables may be written as follows:

\[
\frac{ku_i}{u_*} = \ln\left(\frac{z_i - d}{z_o}\right) + \frac{\beta_1}{L}[(z_i - d) - z_o] + \frac{\beta_2}{L^2}[(z_i - d) - z_o]^2 + \ldots \quad (7)
\]

\[
\frac{k(\theta_i - \theta_*)}{\theta_*} = \ln\left(\frac{z_i - d}{z_o}\right) + \frac{\beta_1}{L}[(z_i - d) - z_o] + \frac{\beta_2}{L^2}[(z_i - d) - z_o]^2 + \ldots \quad (8)
\]

where \( u_* \), \( \theta_* \) and \( k \) are the friction velocity, friction temperature and the Von Karmen constant (0.41) respectively.

The iterative technique was used and Newton method for finding a solution for a system of non linear equations. The following equation were written showing the function value of several variables.

\[
B_1, B_2, \ldots, B_n = x_1[\ln(z_i - x_2) - \ln x_3] + x_1 x_4 (z_i - x_2 - x_3) + x_5 (z_i - x_2 - z_o)^2 \quad (9)
\]

\[
A_{11} = \frac{\partial f}{\partial x_1} = \ln(z_i - x_2) - \ln x_3 + x_4 (z_i - x_2 - x_3) + x_5 (z_i - x_2 - x_3)^2 \quad (10)
\]

\[
A_{12} = \frac{\partial f}{\partial x_2} = -x_1 \frac{1}{z_i - x_2} (z_i - x_2) - x_1 x_4 (z_i - x_2 - x_3) - 2x_1 x_5 (z_i - x_2 - x_3) \quad (11)
\]

\[
A_{13} = \frac{\partial f}{\partial x_3} = -x_1 \frac{x_4}{x_3} - x_1 x_4 - 2x_1 x_5 (z_i - x_2 - x_3) \quad (12)
\]

\[
A_{14} = \frac{\partial f}{\partial x_4} = x_1 (z_i - x_2 - x_3) \quad (13)
\]

\[
A_{15} = \frac{\partial f}{\partial x_5} = x_1 (z_i - x_2 - x_3)^2 \quad (14)
\]

where \( x_1 = \frac{u_*}{k} \), \( x_2 = d \), \( x_3 = z_o \), \( x_4 = \frac{\beta_1}{L} \) and \( x_5 = \frac{\beta_2}{L^2} \) (\( L \) here is the Monin-Obokhov length) unknown values. The variables and known values in Eq.(7) and Eq.(8) are: the height \( z_i \) and the wind speed \( u_i \). The subscript \( i \) means the level number.
\[ A_{i1}x_1 + A_{i2}x_2 + \ldots + A_{in}x_n = B_i \]
\[ A_{11}x_1 + A_{12}x_2 + \ldots + A_{1n}x_n = B_1 \]
\[ \vdots \]
\[ A_{n1}x_1 + A_{n2}x_2 + \ldots + A_{nn}x_n = B_n \]  

The unknown are denoted by \( x_1, x_2, \ldots, x_n \) and the coefficients a’s and b’s are considered to be given. In matrix form the above system of equations can be written as

\[
\begin{bmatrix}
A_{11} & A_{12} & \cdots & A_{1n} & x_1 & B_1 \\
A_{21} & A_{22} & \cdots & A_{2n} & x_2 & B_2 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
A_{n1} & A_{n2} & \cdots & A_{nn} & x_n & B_n
\end{bmatrix}
\]


Iterative technique may be introduced that extend the fixed point and Newton methods for finding a root of an equation. It is desired to have a method for finding a solution for the system of nonlinear equations

\[ f_1(x, y) = 0, \quad f_2(x, y) = 0 \]  

Each equation in (17) implicity defines a curve in the plane and we want to find their points of interaction.

3.1 Jacobian Matrix

Assume that \( f_1(x, y) \), \( f_2(x, y) \) are functions of the independent variables \( x, y \), then their Jacobian Matrix \( J(x, y) \) is

\[
J(x, y) = \begin{pmatrix}
\frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\
\frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y}
\end{pmatrix}
\]  

It’s generalization as a \((N \times N)\) matrix for a system for \( N \) variables is immediate.

Similarly, if \( f_1(x, y, z) \), \( f_2(x, y, z) \) and \( f_3(x, y, z) \) are functions of the independent variables \( x, y \) and \( z \) then their Jacobian Matrix \( J(x, y, z) \) is
A METHOD FOR DETERMINATION OF THE SURFACE LAYER

\[ J(x, y) = \begin{pmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial z} \end{pmatrix} \]

Other process can be written as

\[ \begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \end{pmatrix} - J^{-1}(x_n, y_n) \begin{pmatrix} f_1(x_n, y_n) \\ f_2(x_n, y_n) \end{pmatrix} \]

Starting with an initial guess \((x_0, y_0)\) and under certain conditions (which are not easy to check and the main disadvantage of the method), it’s possible to show that this process converges to a root of the system.

More details about Newton Method for nonlinear systems will be found in Appendix A.

4. The Procedure

First of all the number of variables written according to the level numbers if we have 4 levels, we should have 4 known and unknown values. This means that the number of variables \(N\) depends on number of equations and unknowns: initial approximation \(X = X(1), X(2), \ldots X(n)\) and tolerance: maximum number of iteration \(N_{\text{MAX}}\). The value of \(X_0(1) = X(1), \ldots X_0(n) = X(n)\). Then the iteration starts to find out \(X(1), \ldots X(n)\). After the calculation of Jacobian and the function values by solving Jacobian matrix \(J(x)\) [Eq.’s between (10) –(14)] and function values \(-f(x)\) [Eq. 9]. Finally, solving Jacobian system \(Y = F\) using gauss elimination.

5. Data Analysis

The following sample of a data set used (Table 1) to find out the unknown values: \(u_*, d, z_*, \frac{\beta_1}{L}\) and \(\frac{\beta_2}{L^2}\) as

<table>
<thead>
<tr>
<th>No.s</th>
<th>Heights z (m)</th>
<th>Wind Speed u (ms(^{-1}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>35</td>
<td>0.6432</td>
</tr>
<tr>
<td>2</td>
<td>50</td>
<td>0.8928</td>
</tr>
<tr>
<td>3</td>
<td>70</td>
<td>0.9291</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>1.15</td>
</tr>
<tr>
<td>5</td>
<td>150</td>
<td>1.156</td>
</tr>
</tbody>
</table>
and after iteration the results were found as

\[ u_\star = 0.079, \quad d = 25.55, \quad z_\star = 0.0388, \quad \frac{\beta_1}{L} = -0.062, \quad \frac{\beta_2}{L^2} = 0.001 \]

A selected data set (Table 2) taken from the observations over the Arabian Sea during the International Indian Ocean (Badgley and Paulson, 1965) also used to test the program and found the following

\[ u_\star = 0.33, \quad d = xx \text{ (small)}, \quad z_\star = 0.19, \quad \frac{\beta_1}{L} = -0.58, \quad \frac{\beta_2}{L^2} = 0.01. \]

A selected data set (Table 3) for a Khulais site in the western region of Saudi Arabia (Anbar Sery 2001) for only on day measurements (June 1, 1998) were used for testing our program and comparing the results with Haenel (1993) findings.

Table 3: The following stability parameters were tabulated in Table 3, Table 4 and Table 5.

<table>
<thead>
<tr>
<th>No.s</th>
<th>Heights z (m)</th>
<th>Wind Speed u (ms(^{-1}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.57</td>
<td>3.40</td>
</tr>
<tr>
<td>2</td>
<td>2.13</td>
<td>3.49</td>
</tr>
<tr>
<td>3</td>
<td>2.96</td>
<td>3.59</td>
</tr>
<tr>
<td>4</td>
<td>4.13</td>
<td>3.65</td>
</tr>
<tr>
<td>5</td>
<td>5.79</td>
<td>3.75</td>
</tr>
</tbody>
</table>

Table 3: Stability parameters calculated from the Khulais data, June 1, 1988.

<table>
<thead>
<tr>
<th>Heights z (m)</th>
<th>Data set 0000 hours W. D. [349°]</th>
<th>Data set at 0300 hours W. D. [123°]</th>
<th>Data set at 0800 hours W. D. [353°]</th>
</tr>
</thead>
<tbody>
<tr>
<td>z (m)</td>
<td>u (ms(^{-1}))</td>
<td>u (ms(^{-1}))</td>
<td>u (ms(^{-1}))</td>
</tr>
<tr>
<td>2.0</td>
<td>1.18</td>
<td>1.99</td>
<td>1.95</td>
</tr>
<tr>
<td>4.0</td>
<td>1.35</td>
<td>2.27</td>
<td>2.06</td>
</tr>
<tr>
<td>8.0</td>
<td>1.70</td>
<td>2.69</td>
<td>2.17</td>
</tr>
<tr>
<td>16.0</td>
<td>2.05</td>
<td>3.15</td>
<td>2.27</td>
</tr>
</tbody>
</table>

Results

<table>
<thead>
<tr>
<th>Stability Parameters</th>
<th>S. P. Values Prog.</th>
<th>S. P. Values Haenel</th>
<th>S. P. Values Prog.</th>
<th>S. P. Values Haenel</th>
<th>S. P. Values Prog.</th>
<th>S. P. Values Haenel</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_\star (ms^{-1}) )</td>
<td>0.053</td>
<td>0.087</td>
<td>0.466</td>
<td>0.085</td>
<td>0.046</td>
<td>0.158</td>
</tr>
<tr>
<td>( d (m) )</td>
<td>1.65</td>
<td>0.0</td>
<td>1.167</td>
<td>0.0</td>
<td>1.81</td>
<td>0.0</td>
</tr>
<tr>
<td>( z_\star (m) )</td>
<td>0.03</td>
<td>0.069</td>
<td>0.00255</td>
<td>0.0006</td>
<td>0.379</td>
<td>0.0047</td>
</tr>
<tr>
<td>( \beta_1 / L )</td>
<td>-0.10</td>
<td>-</td>
<td>1.84</td>
<td>-</td>
<td>2.84</td>
<td>-</td>
</tr>
</tbody>
</table>
Table 4: As in Table 3.

<table>
<thead>
<tr>
<th>Heights $z$ (m)</th>
<th>Data set 1000 hours W. D. [227°]</th>
<th>Data set at 1200 hours W. D. [316°]</th>
<th>Data set at 1300 hours W. D. [313°]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Z$ (m)</td>
<td>$u \ (ms^{-1})$</td>
<td>$u \ (ms^{-1})$</td>
<td>$u \ (ms^{-1})$</td>
</tr>
<tr>
<td>2.0</td>
<td>4.32</td>
<td>5.79</td>
<td>5.65</td>
</tr>
<tr>
<td>4.0</td>
<td>4.69</td>
<td>6.25</td>
<td>6.07</td>
</tr>
<tr>
<td>8.0</td>
<td>5.03</td>
<td>6.72</td>
<td>6.61</td>
</tr>
<tr>
<td>16.0</td>
<td>5.30</td>
<td>7.08</td>
<td>6.93</td>
</tr>
</tbody>
</table>

Results

<table>
<thead>
<tr>
<th>Stability Parameters</th>
<th>S. P. Values</th>
<th>S. P. Values</th>
<th>S. P. Values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prog.</td>
<td>Haenel</td>
<td>Prog.</td>
</tr>
<tr>
<td>$u_\ast \ (ms^{-1})$</td>
<td>0.116</td>
<td>0.352</td>
<td>0.14</td>
</tr>
<tr>
<td>$d \ (m)$</td>
<td>1.82</td>
<td>0.0</td>
<td>1.79</td>
</tr>
<tr>
<td>$z_\ast \ (m)$</td>
<td>0.99</td>
<td>0.008</td>
<td>0.246</td>
</tr>
<tr>
<td>$\frac{\beta_1}{L}$</td>
<td>3.066</td>
<td>-</td>
<td>2.62</td>
</tr>
</tbody>
</table>

Table 5: As in Table 3 and 4.

<table>
<thead>
<tr>
<th>Heights $z$ (m)</th>
<th>Data set (1330) hours W. D. [314°]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$z$ (m)</td>
<td>$u \ (ms^{-1})$</td>
</tr>
<tr>
<td>2.0</td>
<td>5.32</td>
</tr>
<tr>
<td>4.0</td>
<td>5.75</td>
</tr>
<tr>
<td>8.0</td>
<td>6.2</td>
</tr>
<tr>
<td>16.0</td>
<td>6.48</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stability Parameters</th>
<th>S. P. Values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prog.</td>
</tr>
<tr>
<td>$u_\ast \ (ms^{-1})$</td>
<td>0.125</td>
</tr>
<tr>
<td>$d \ (m)$</td>
<td>1.76</td>
</tr>
<tr>
<td>$z_\ast \ (m)$</td>
<td>0.26</td>
</tr>
<tr>
<td>$\frac{\beta_1}{L}$</td>
<td>-</td>
</tr>
</tbody>
</table>
A paired t-test may be worth for both data set at 1200 and 1300 hours calculated by the program to look at the difference between individual pairs of values (A null hypothesis was set up; i.e.: at the 0.05 level) show that there is no significant difference between the stability parameters because of the nearly similar wind direction 316° and 313° successively. Evidence also showing that there no significant difference between the Haenel results at both 1200 and 1300 hours data (Fig.1). Figure 2 shows the relation between 1200 and 1300 hours data using our program and no significant difference between the stability parameters \([u_*(ms^{-1}), d (m) \text{ and } z_*(m)]\).

The same data as above was computed at 1300 and 1330 using our program, show there is no significant difference (t test at the 0.05 level) between the stability parameters \([u_*(ms^{-1}), d (m) \text{ and } z_*(m)]\) because of the similar wind direction 314° (Fig. 3). Figure 4 shows the fitted line plot between both data set \([u_*(ms^{-1}), d (m) \text{ and } z_*(m)]\) calculated by Haenel at 1300 and 1330.

The last test was done (Fig. 5) on both data set at 0300 and 0800 [stability parameters \(u_*(ms^{-1}), d(m), z_*(m) \text{ and } \frac{\beta}{L}\)] shows again no significant difference between them (t test at the 0.05 level) while more precisely the fitted line plot between both data set \([u_*(ms^{-1}), d(m) \text{ and } z_*(m)]\) calculated by Program (Fig. 6).

Fig. 1: Fitted line plot between 1200 and 1300 hours data set \([u_*(ms^{-1}), d (m) \text{ and } z_*(m)]\) using Haenel program on June 1 1998 in Khulais
Fig. 2: Fitted line plot between 1200 and 1300 hours data set [$u_*(m/s)$, $d (m)$ and $z_* (m)$] using Program on June 1 1998 in Khulais.

Fig. 3: Fitted line plot between 1300 and 1330 hours data set [$u_*(m/s)$, $d (m)$ and $z_* (m)$] using Program on June 1 1998 in Khulais.
Fig. 4: Fitted line plot between 1300 and 1330 hours data set \[ u_*(ms^{-1}), d(m), \\
\beta_1 \] using Program on June 1 1998 in Khulais

Fig. 5: Fitted line plot between 0300 and 0800 hours data set \[ u_*(ms^{-1}), d(m) \]
and \[ z_o (m) \] using Program on June 1 1998 in Khulais
A METHOD FOR DETERMINATION OF THE SURFACE LAYER

9. Concluding remarks

It may be intrepretate that the values of zero displacement level \((d)\) in Table 3, 4 and 5 doesn’t coincide with Haenel’s values propably because of difference in heights of the levels and this is true because \(d\) values of Haenel calculated from 6 levels of wind speed while we used 5 levels. Another possibilty is by looking at the formulae between 1 and 3 and \(x_5 = \frac{\beta_2}{L^2}\) Haenel probably didn’t include \(x_5 = \frac{\beta_2}{L^2}\) in his calculation, so that there are an agreement and disagreement with stability parameters between the program values and Haenel’s values. Anyway, more or less the friction velocity \(u_*\) in both program and Haenel values were very close to each other.

Acknowledgments

I would like to thank Dr. M.S. Hamouda from the Department of Mathematics King Abdul Aziz University, Jeddah who assist me to write the above program and to understand it.
Appendix A

It is possible to outline the derivation of the Newton’s method in two dimensions. Newton’s method can be easily extended to higher dimensions. Consider a system

\[ u = f_1(x, y), \quad v = f_2(x, y) \quad (1A) \]

which can be considered a transformation from the xy-plane to the uv-plane. It is interested in the behavior of this transformation near the point \((x_0, y_0)\), whose image is the point \((u_0, v_0)\). If the two functions have continuous partial derivatives, then the differential can be used to write a system of linear approximations that is valid near the point \((x_0, y_0)\):

\[
\begin{pmatrix}
\frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\
\frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y}
\end{pmatrix}
\begin{pmatrix}
\Delta x \\
\Delta y
\end{pmatrix}
= \begin{pmatrix}
\Delta u \\
\Delta v
\end{pmatrix}
\quad (2A)
\]

Then substitute the changes

\[
\Delta x = x - x_o,
\]
\[
\Delta y = y - y_o,
\]
\[
\Delta u = u - u_o
\]
and
\[
\Delta v = v - v_o
\]
for \(dx\), \(dy\), \(du\) and \(dv\) respectively. Then we will have

\[
\begin{pmatrix}
\Delta u \\
\Delta v
\end{pmatrix} = \begin{pmatrix}
\frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\
\frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y}
\end{pmatrix}
\begin{pmatrix}
\Delta x \\
\Delta y
\end{pmatrix}
\quad (3A)
\]

or

\[
\Delta \bar{F} = J(x_o, y_o) \Delta \bar{X}
\quad (4A)
\]

Consider the system \((1A)\) with \(u\) and \(v\) set equal to zero,

\[ 0 = f_1(x, y), \quad 0 = f_2(x, y). \quad (5A) \]

Suppose we are trying to find the solution \((p, q)\) and we start iteration at the nearby point \((p_0, q_0)\), then we can apply \((2A)\) and write

\[
\begin{pmatrix}
f_1(p, q) - f_1(p_0, q_0) \\
f_2(p, q) - f_2(p_0, q_0)
\end{pmatrix} = \begin{pmatrix}
\frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\
\frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y}
\end{pmatrix}
\begin{pmatrix}
\Delta p \\
\Delta q
\end{pmatrix}
\quad (6A)
\]
A METHOD FOR DETERMINATION OF THE SURFACE LAYER

Since \( f_1(p, q) = 0 \), \( f_2(p, q) = 0 \), this becomes

\[
\begin{pmatrix}
- f_1(p, q) \\
- f_2(p, q)
\end{pmatrix}
= \begin{pmatrix}
\frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\
\frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y}
\end{pmatrix}
\begin{pmatrix}
\Delta p \\
\Delta q
\end{pmatrix}
\quad (7A)
\]

or

\[
- \vec{F}(p, q) = J(p, q) \Delta \vec{P}
\quad (8A)
\]

When we solve this latter equation for \( \Delta \vec{F} \), we get

\[
\Delta \vec{F} = -(J(p, q))^{-1} \vec{F}(p, q)
\]

and the next approximation \( \vec{P}_1 = \vec{P}_0 + \Delta \vec{P} \) is

\[
\vec{P}_1 = \vec{P}_0 - (J(\vec{P}_0))^{-1} \vec{F}(\vec{P}_0)
\quad (9A)
\]

References