**ROB 501 Handout: Grizzle**

**Newton Raphson Algorithm**

Let $h : \mathbb{R}^n \to \mathbb{R}^n$ be continuously differentiable, and satisfy

$$\boxed{\det\left(\frac{\partial h}{\partial x}(x)\right) \neq 0 \quad \forall x \in \mathbb{R}^n}$$

**Problem:** For $y \in \mathbb{R}^n$ fixed, find a solution of $y = h(x)$; i.e, find $x^* \in \mathbb{R}^n$ s.t. $y = h(x^*)$. We note that this is equivalent to $h(x^*) - y = 0$. In other words, we are looking for a root of the equation $h(x) - y = 0$,

**Approach:** Find a convergent sequence $x_k \to x^*$ such that

$$\boxed{\lim_{k\to\infty} h(x_k) - y = h(x^*) - y = 0}$$

that is, $x^* = \lim\limits_{k\to\infty} x_k$ is a root of $h(x) - y = 0$

**Idea:** Write $x_{k+1} = x_k + \Delta x_k$. We want

$$h(x_{k+1}) - y = h(x_k + \Delta x_k) - y \approx 0.$$

What should $\Delta x_k$ look like ?

Apply Taylor's Theorem, to get

$$h(x_k) + \frac{\partial h}{\partial x}(x_k)\Delta x_k - y \approx 0$$

$$\therefore \frac{\partial h}{\partial x}(x_k)\Delta x_k \approx -(h(x_k) - y)$$

$$\Delta x_k \approx -\left(\frac{\partial h}{\partial x}(x_k)\right)^{-1}(h(x_k) - y)$$

Recalling that $x_{k+1} = x_k + \Delta x_k$, we arrive at Newton's Algorithm,

$$\boxed{x_{k+1} = x_k - \left(\frac{\partial h}{\partial x}(x_k)\right)^{-1} (h(x_k) - y)}$$

In practice, the change in $x_k$ given by $\Delta x_k = -\left(\frac{\partial h}{\partial x}(x_k)\right)^{-1}(h(x_k) - y)$ is often too large. Hence, one uses the so-called <u>Damped Newton Algorithm</u>

$$\boxed{x_{k+1} = x_k - \epsilon\left(\frac{\partial h}{\partial x}(x_k)\right)^{-1} (h(x_k) - y)}$$

where $\epsilon > 0$ provides step size control!

<u>**Remark:**</u> Looking ahead to our discussion of contraction mappings, let's rewrite the algorithm as the iteration of a mapping $x_{k+1} = P(x_k)$

$$\boxed{P(x) := x - \epsilon\left(\frac{\partial h}{\partial x}(x)\right)^{-1} (h(x) - y)}$$

A solution of $h(x) - y$ is a <u>fixed point</u> of $P(x)$. Indeed,

$$\begin{gathered}
x^* = P(x^*) \\
\Updownarrow \\
x^* = x^* - \epsilon\left(\frac{\partial h}{\partial x}(x^*)\right)^{-1} (h(x^*) - y) \\
\Updownarrow \\
0 = -\epsilon\left(\frac{\partial h}{\partial x}(x^*)\right)^{-1} (h(x^*) - y) \\
\Updownarrow \\
0 = (h(x^*) - y).
\end{gathered}$$

It can be shown that $P$ is a <u>local contraction</u> on an open ball around a solution of $h(x) - y = 0$.

**Example** Find the solution to the coupled NL equations

$$0 = h(x) = \begin{pmatrix} x_1 + 2\,x_2 - x_1\,(x_1 + 4\,x_2) - x_2\,(4\,x_1 + 10\,x_2) + 3 \\ 3\,x_1 + 4\,x_2 - x_1\,(x_1 + 4\,x_2) - x_2\,(4\,x_1 + 10\,x_2) + 4 \\ \sin(x_3)^7 + \frac{\cos(x_1)}{2} \\ {x_4}^3 - 2\,{x_2}^2\,\sin(x_1) \end{pmatrix}$$

**Initial Guess:** $x_0 = \begin{bmatrix} 7 \\ 8 \\ 9 \\ 10 \end{bmatrix}$

We do 16 iterations of Newton's Algorithm (a nonlinear root finding algorithm) and we obtain:

$$x^* = \begin{pmatrix} -2.25957308738366677539068499960 \\ 1.75957308738366677539068499960 \\ 189.50954100613333978330549312824 \\ -1.68458069860197189523093013800 \end{pmatrix}$$

**And the error is:**

$$h(x^*) = \begin{bmatrix} 3.6734198 \times 10^{-39} \\ 2.9387359 \times 10^{-39} \\ 1.2765134 \times 10^{-38} \\ -2.5915832 \times 10^{-32} \end{bmatrix}$$