$$q_1q_2(q_3-2q_3)+q_1q_3(2q_2-q_2)+2q_3q_2(q_1-q_1)=-q_1q_2q_3+q_1q_2q_3=0 \quad (2.2.58)$$

Thus, the differential form of Example 2.2.7 is integrable, even though it is not exact.

2.2.7 Newton-Raphson Method for Solution of Nonlinear Equations

At the intersection of multivariable calculus and matrix theory are iterative methods for solving nonlinear algebraic equations. Let **x** be an n-vector of unknowns and

$$\mathbf{F}(\mathbf{x})=\mathbf{0}\in R^n \quad (2.2.59)$$

be a system of n nonlinear equations to be solved for **x**, where the vector function has continuous derivatives and the *Jacobian matrix* $\mathbf{F}_\mathbf{x}(\mathbf{x})$ is nonsingular in an open set D of R^n. With an estimate $\mathbf{x}^0\in D$ of the solution, a first order Taylor expansion that seeks a perturbation $\Delta\mathbf{x}^0$ of $\mathbf{x}^0$ that better approximates the solution as $\mathbf{x}^1=\mathbf{x}^0+\Delta\mathbf{x}^0$ is

$$\mathbf{F}(\mathbf{x}^0)+\mathbf{F}_\mathbf{x}(\mathbf{x}^0)\Delta\mathbf{x}^0=\mathbf{0}$$

After j iterations, the incremental equation is

$$\begin{aligned}\mathbf{F}_\mathbf{x}(\mathbf{x}^\mathbf{j})\Delta\mathbf{x}^\mathbf{j}&=-\mathbf{F}(\mathbf{x}^\mathbf{j})\\ \mathbf{x}^{j+1}&=\mathbf{x}^j+\Delta\mathbf{x}^j\end{aligned} \quad (2.2.60)$$

which is continued until $\left\|\mathbf{F}(\mathbf{x}^{j+1})\right\|\le \mathrm{Tol}$, where Tol is a solution error tolerance.

This is the *Newton-Raphson method* (Atkinson, 1989) for solution of Eq. (2.2.59). The method has the attractive property that if $\mathbf{x}^0$ is sufficiently close to the solution, the method converges quadratically to the solution; i.e.,

$$\left\|\bar{\mathbf{x}}-\mathbf{x}^{j+1}\right\|\le k\left\|\bar{\mathbf{x}}-\mathbf{x}^j\right\|^2 \quad (2.2.61)$$

where $\bar{\mathbf{x}}$ is the solution and k is a constant. However, a good initial estimate is often difficult to obtain, and the method may diverge for a poor initial estimate. Fortunately, in kinematic and dynamic simulation applications on a time grid $t_{i+1}=t_i+h$, i=1,2, ... , with h small, the solution at t_i can be used as the initial estimate at t_{i+1} and convergence is likely.

2.2.8 Matrix Condition Number

The Jacobian $\mathbf{F}_\mathbf{x}(\mathbf{x})$ being nonsingular is not adequate to assure good performance of the Newton-Raphson method. The *condition number* of the Jacobian plays a crucial role (Strang, 1980). With the norm of a vector defined in Eq. (2.2.45), the matrix norm of the Jacobian is

$$\left\|\mathbf{F}_\mathbf{x}(\mathbf{x})\right\|_M=\max_{\mathbf{x}\neq\mathbf{0}}\left(\left\|\mathbf{F}_\mathbf{x}(\mathbf{x})\mathbf{x}\right\|_M/\left\|\mathbf{x}\right\|\right) \quad (2.2.62)$$

and the condition number of the Jacobian is defined as (Strang, 1980)

$$c=\left\|\mathbf{F}_\mathbf{x}(\mathbf{x})\right\|_M\left\|\mathbf{F}_\mathbf{x}(\mathbf{x})^{-1}\right\|_M \quad (2.2.63)$$

Fortunately, there are methods to obtain good approximations of the condition number, without calculation of the inverse (Atkinson, 1989).

To see the significance of the condition number, consider the case in which the error in evaluating the right side of the matrix equation $\mathbf{Ax} = \mathbf{b}$ is $\delta\mathbf{b}$. A bound on the magnitude of the perturbation $\delta\mathbf{x}$ in the resulting solution is (Strang, 1980)

$$\|\delta\mathbf{x}\| / \|\mathbf{x}\| \leq c\left(\|\delta\mathbf{b}\| / \|\mathbf{A}\|_{M}\right) \tag{2.2.64}$$

If the condition number c of matrix $\mathbf{A}$ is small, the solution is not significantly affected by the perturbation in the right side. If c is large, however, large error and possibly divergence can occur. Similarly, if the matrix is perturbed by $\delta\mathbf{A}$, a bound on solution error is (Strang, 1980)

$$\|\delta\mathbf{x}\| / \|\mathbf{x} + \delta\mathbf{x}\| \leq c\left(\|\delta\mathbf{A}\|_{M} / \|\mathbf{A}\|_{M}\right) \tag{2.2.65}$$

As above, if the condition number c is small, the solution is not significantly affected by the perturbation in the Jacobian. If c is large, however, large error and possibly divergence can occur.

The condition number of a matrix $\mathbf{A}$ in an equation $\mathbf{Ax} = \mathbf{b}$ thus serves as a check on accuracy of solution. If $c = \|\mathbf{A}\|_{M}\|\mathbf{A}^{-1}\|_{M}$ is large, problems occur. Unfortunately, if A is a scalar, $\|A\|_{M} = |A|$ and $\|A^{-1}\|_{M} = 1/|A|$. Thus, $c = |A|/|A| = 1$ and no information is obtained. In this case, $c \equiv 1/|A|$ serves as a check on accuracy of the solution. If $1/|A|$ is large, problems arise. This serves as an extension of condition number when A is a scalar.

As may be noted in MATLAB code presented in the text for iterative solution of nonlinear equations, the condition number is used to control error in the solution of equations of kinematics and those of dynamics and to avoid divergence of the Newton-Raphson algorithm.

Matrix algebra and multivariable calculus provide indispensable tools for mechanical system kinematics and dynamics. Both are ideally suited for digital computation. While the Lagrange multiplier theorem of linear algebra has a somewhat abstract proof, it is easily applied as a basic tool of multibody dynamics.

The Euclidean space R^n, with the definition of scalar product, norm, and distance between vectors, is indispensable in definition of continuity of functions and is the basis for differential geometry, the foundation of kinematics and dynamics of mechanical systems.

The implicit and inverse function theorems and criteria for exactness and integrability of differential forms are key in assuring existence of solutions of nonlinear equations and understanding the difference between algebraic and differential kinematic constraints. Finally, the Newton-Raphson iterative solution algorithm and the matrix condition number play pivotal roles in numerical solution of equations of kinematics and dynamics.

Key Formulas

$(\mathbf{AB})^{T} = \mathbf{B}^{T}\mathbf{A}^{T}$ $\quad (\mathbf{A}^{-1})^{T} = (\mathbf{A}^{T})^{-1}$ $\quad (\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$ (2.2.16) (2.2.22) (2.2.23)

$$a_{\mathbf{q}} \equiv \frac{\partial a}{\partial \mathbf{q}} = \left[\frac{\partial a}{\partial q_j} \right]_{1\times k} \qquad \mathbf{f}_{\mathbf{q}} \equiv \frac{\partial \mathbf{f}}{\partial \mathbf{q}} = \left[\frac{\partial f_i}{\partial q_j} \right]_{n\times k} \qquad (2.2.34) \qquad (2.2.35)$$

$$\left(\mathbf{g}^{T}\mathbf{h}\right)_{\mathbf{q}} = \mathbf{h}^{T}\mathbf{g}_{\mathbf{q}} + \mathbf{g}^{T}\mathbf{h}_{\mathbf{q}} \qquad \left(\Phi\left(\mathbf{g}\left(\mathbf{q}\right)\right)\right)_{\mathbf{q}} = \Phi_{\mathbf{g}}\mathbf{g}_{\mathbf{q}} \qquad (2.2.36) \qquad (2.2.37)$$

$$R^{n} = \left\{ \mathbf{q} = \left[q_1 \quad \cdots \quad q_n \right]^{T} : q_i \text{ real, } i = 1, \cdots, n \right\} \qquad (2.2.41)$$

$$\left\langle \mathbf{r}, \mathbf{s} \right\rangle \equiv \mathbf{r}^{T}\mathbf{s} = \sum_{l=1}^{n} r_i s_i \qquad d\left(\mathbf{r}, \mathbf{s}\right) \equiv \left\| \mathbf{r} - \mathbf{s} \right\|_{n} \qquad \left\| \mathbf{q} \right\|_{n} \equiv \sqrt{\left\langle \mathbf{q}, \mathbf{q} \right\rangle} \qquad (2.2.43) \qquad (2.2.44) \qquad (2.2.45)$$

$$B_{\varepsilon}(\mathbf{q}^{0}) = \left\{ \mathbf{q} \in R^{n} : \left\| \mathbf{q} - \mathbf{q}^{0} \right\| < \varepsilon \right\} \qquad (2.2.47)$$