# 3

# Residual Force Equations

## TABLE OF CONTENTS

Chapters 3 through 6 describe basic properties of systems of algebraic nonlinear equations that depend on one or more control parameters. *Algebraic* means that these systems contain a *finite* number of equations and unknowns.

Those systems result from the *discretization* of the continuum models of nonlinear structures. As mentioned in Chapter 1, the most widely used discretization method is the displacement-based Finite Element Method, or FEM. The description of FEM discretization techniques is deferred until Chapter 8 and following. For the moment it is assumed that the discretization has been carried out.

Physically the algebraic systems represent equilibrium of forces at the discrete level. More specifically, for discrete model coming from the displacement FEM, the equilibrium of nodal forces. These are collectively known as *force residual equations* or *residual equations* for short. In the present Chapter, residual and differential forms of use in later Chapters are presented, and the key concept of *staging*, through which multiple control parameters are reduced to one, is introduced.

## §3.1. Equilibrium Equations

Discrete equilibrium equations encountered in *nonlinear static structural analysis* formulated by the *displacement* method may be presented in the compact *force residual* form

$$\mathbf{r}(\mathbf{u}, \mathbf{\Lambda}) = \mathbf{0}. \tag{3.1}$$

Here **u** is the *state vector* containing the degrees of freedom that characterize the configuration of the structure, **r** is the *residual* vector that contains out-of-balance forces conjugate to **u**, and $\mathbf{\Lambda}$ is an array of assignable *control parameters*.

In structural mechanics, control parameters are commonly mechanical load levels. They may also be, however, prescribed physical or generalized displacements, temperature variations, imperfection amplitudes and even (in design and optimization) geometric dimensions or material properties. The degrees of freedom collected in **u** are usually physical or generalized unknown displacements. The names *behavior* or *configuration* variables are also used, however, to stress a more general significance in other applications. In a general mathematical context, **u** and $\mathbf{\Lambda}$ are called the *active* and *passive* variables, respectively.

The dependence of **r** on **u** and $\mathbf{\Lambda}$ is assumed to be piecewise smooth so that first and second derivatives exist except possibly at isolated critical points. If the system is conservative,[1] **r** is the gradient of the *total* potential energy $\Pi(\mathbf{u}, \mathbf{\Lambda})$ for fixed $\mathbf{\Lambda}$:

$$\mathbf{r} = \frac{\partial \Pi}{\partial \mathbf{u}}, \qquad \text{or} \quad r_i = \frac{\partial \Pi}{\partial u_i}. \tag{3.2}$$

Then (3.1) expresses that equilibrium is associated with a energy-stationarity condition.

An alternative version of (3.1) that displays more physical meaning is the *force-balance* form:

$$\mathbf{p}(\mathbf{u}) = \mathbf{f}(\mathbf{u}, \mathbf{\Lambda}). \tag{3.3}$$

In this form **p** denotes the configuration-dependent *internal forces* resisted by the structure whereas **f** are the control-dependent *external* or *applied forces*, which may also be configuration dependent.

---

[1] A property studied more carefully in Chapter 6.

The residual (3.2) is either $\mathbf{r} = \mathbf{p} - \mathbf{f}$ or $\mathbf{r} = \mathbf{f} - \mathbf{p}$, the two versions being equivalent except for sign. This expression states that if (3.2) is verified, internal forces $\mathbf{p}$ balance the applied forces $\mathbf{f}$. If a total potential energy exists, the decomposition associated with (3.2) is

$$\mathbf{p} = \frac{\partial U}{\partial \mathbf{u}}, \qquad \mathbf{f} = \frac{\partial P}{\partial \mathbf{u}}, \tag{3.4}$$

where $U$ and $P$ are the internal and external energy components, respectively, of $\Pi = U - P$.

The formulation of these discrete equations using the Finite Element Method is treated in Chapter 7 and following. For the moment it is simply assumed that, given $\mathbf{u}$ and $\mathbf{\Lambda}$, a computational method exists that returns $\mathbf{r}$. In addition, most solution methods require derivative information of the type discussed in §3.3.

**Example 3.1**. Consider the following residual equilibrium equations

$$\begin{aligned} r_1 &= 4u_1 - u_2 + u_2 u_3 - 6\Lambda_1 = 0, \\ r_2 &= 6u_2 - u_1 + u_1 u_3 - 3\Lambda_2 = 0, \\ r_3 &= 4u_3 + u_1 u_2 - 3\Lambda_2 = 0. \end{aligned} \tag{3.5}$$

The vector form of (3.5) is

$$\mathbf{r} = \begin{bmatrix} r_1(\mathbf{u}, \mathbf{\Lambda}) \\ r_2(\mathbf{u}, \mathbf{\Lambda}) \\ r_3(\mathbf{u}, \mathbf{\Lambda}) \end{bmatrix} = \begin{bmatrix} 4u_1 - u_2 + u_2 u_3 - 6\Lambda_1 \\ 6u_2 - u_1 + u_1 u_3 - 3\Lambda_2 \\ 4u_3 + u_1 u_2 - 3\Lambda_2 \end{bmatrix} = \mathbf{0}, \quad \text{with} \quad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}, \quad \mathbf{\Lambda} = \begin{bmatrix} \Lambda_1 \\ \Lambda_2 \end{bmatrix}. \tag{3.6}$$

The force-balance vector form of (3.5) is $\mathbf{p} = \mathbf{f}$, in which

$$\mathbf{p} = \begin{bmatrix} p_1(\mathbf{u}) \\ p_2(\mathbf{u}) \\ p_3(\mathbf{u}) \end{bmatrix} = \begin{bmatrix} 4u_1 - u_2 + u_2 u_3 \\ 6u_2 - u_1 + u_1 u_3 \\ 4u_3 + u_1 u_2 \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} f_1(\mathbf{\Lambda}) \\ f_2(\mathbf{\Lambda}) \\ f_3(\mathbf{\Lambda}) \end{bmatrix} = 3 \begin{bmatrix} 2\Lambda_1 \\ \Lambda_2 \\ \Lambda_2 \end{bmatrix}. \tag{3.7}$$

In this particular case $\mathbf{f}$ does not depend on $\mathbf{u}$.

**Remark 3.1**. The function $\mathbf{u}(\mathbf{\Lambda})$ characterizes the *equilibrium surface* of the structure in the space spanned by $\mathbf{u}$ and $\mathbf{\Lambda}$. In a general mathematical context the set of $\{\mathbf{u}, \mathbf{\Lambda}\}$ pairs that satisfies (3.1) is called a *solution manifold*.

**Remark 3.2**. The usefulness of the residual equation (3.1) is not restricted to static problems. It is also applicable to nonlinear dynamical systems

$$\mathbf{r}(\mathbf{u}(\tau), \mathbf{\Lambda}(\tau)) = \mathbf{0}, \tag{3.8}$$

which have been discretized in time $\tau$ by implicit methods.[2] In this case a system of nonlinear equations such as (3.3) arises at each time station.

**Remark 3.3**. Equation (3.1) or its alternative version (3.3) are restricted in that no account for historical effects is made. A more general form would include the *history* of past deformations:

$$\mathbf{r}(\mathbf{u}, \Xi, \mathbf{\Lambda}) = 0,$$

where $\Xi$ is a functional of the past history of the deformation gradients. This generalized form is needed for the treatment of inelastic path-dependent materials. The form (3.0) is sufficient, however, for the class of problems considered here. In addition to geometric and force B.C. nonlinearities, these forms are applicable to nonlinear elasticity and several types of displacement B.C. nonlinearities.

---

[2]  $\tau$ is used throughout this book for real time in lieu of $t$, which is used more extensively to denote pseudo-time.

## §3.2. Stiffness and Control Matrix

Varying the vector $\mathbf{r}$ with respect to the components of $\mathbf{u}$ while keeping $\boldsymbol{\Lambda}$ fixed yields the *Jacobian matrix* $\mathbf{K}$:

$$\mathbf{K} = \frac{\partial \mathbf{r}}{\partial \mathbf{u}}, \quad \text{with entries} \quad K_{ij} = \frac{\partial r_i}{\partial u_j}. \tag{3.9}$$

This is called the *tangent stiffness matrix* in structural mechanics applications. The inverse of $\mathbf{K}$, if it exists, is denoted by $\mathbf{F} = \mathbf{K}^{-1}$; a notation suggested by the name *flexibility matrix* used in linear structural analysis for the reciprocal of the stiffness. If system (3.1) derives from a potential, both $\mathbf{K}$ and $\mathbf{F}$ are symmetric matrices (see Exercise 3.5).

Varying the negative of $\mathbf{r}$ with respect to $\boldsymbol{\Lambda}$ while keeping $\mathbf{u}$ fixed yields

$$\mathbf{Q} = -\frac{\partial \mathbf{r}}{\partial \boldsymbol{\Lambda}}, \quad \text{with entries} \quad Q_{ij} = -\frac{\partial r_i}{\partial \Lambda_j}. \tag{3.10}$$

**Remark 3.4**. There is no agreed upon name for $\mathbf{Q}$. In the sequel it is called the *control matrix*, although the name "loads matrix" is also appropriate. The specialization of $\mathbf{Q}$ to the usual incremental load vector $\mathbf{q}$ is discussed in Chapter 4.

## §3.3. Parametric Representations and Rate Forms

Parametric representations of $\mathbf{u}$ and $\boldsymbol{\Lambda}$ are useful in the description of solution methods as pseudo-dynamical processes. The general form is

$$\boxed{\mathbf{u} = \mathbf{u}(t), \qquad \boldsymbol{\Lambda} = \boldsymbol{\Lambda}(t)}, \tag{3.11}$$

where $t$ is a *dimensionless* time-like parameter. Derivatives with respect to $t$ will be denoted by superposed dots, as in real dynamics. The first two $t$-derivatives of the residual in component form are (with summation convention implied):

$$\dot{r}_i = \frac{\partial r_i}{\partial u_j}\dot{u}_j + \frac{\partial r_i}{\partial \Lambda_j}\dot{\Lambda}_j, \tag{3.12}$$

$$\ddot{r}_i = \frac{\partial r_i}{\partial u_j}\ddot{u}_j + \left[\frac{\partial^2 r_i}{\partial u_j \partial u_k}\dot{u}_k + \frac{\partial^2 r_i}{\partial u_j \partial \Lambda_k}\dot{\Lambda}_k\right]\dot{u}_j + \frac{\partial r_i}{\partial \Lambda_j}\ddot{\Lambda}_j + \left[\frac{\partial^2 r_i}{\partial \Lambda_j \partial u_k}\dot{u}_k + \frac{\partial^2 r_i}{\partial \Lambda_j \partial \Lambda_k}\dot{\Lambda}_k\right]\dot{\Lambda}_j. \tag{3.13}$$

In matrix form,

$$\boxed{\dot{\mathbf{r}} = \mathbf{K}\dot{\mathbf{u}} - \mathbf{Q}\dot{\boldsymbol{\Lambda}},} \tag{3.14}$$

$$\boxed{\ddot{\mathbf{r}} = \mathbf{K}\ddot{\mathbf{u}} + \dot{\mathbf{K}}\dot{\mathbf{u}} - \mathbf{Q}\ddot{\boldsymbol{\Lambda}} - \dot{\mathbf{Q}}\dot{\boldsymbol{\Lambda}}.} \tag{3.15}$$

Note that both $\dot{\mathbf{K}}$ and $\dot{\mathbf{Q}}$ are matrices. Their $(i, j)$ components are shown in square brackets in (3.13). On the other hand, terms such as $\partial^2 r_i / \partial u_j \partial u_k$, are three-dimensional arrays that may be visualized as "cubic matrices." Matrices $\dot{\mathbf{K}}$ and $\dot{\mathbf{Q}}$ are projections of those arrays on the subspace spanned by the directions $\dot{\mathbf{u}}$ and $\dot{\boldsymbol{\Lambda}}$. Often these matrices can be more expediently formed by direct time-differentiation:

$$\boxed{\dot{\mathbf{K}} = \frac{d\mathbf{K}}{dt}, \qquad \dot{\mathbf{Q}} = \frac{d\mathbf{Q}}{dt}.} \tag{3.16}$$

## §3.4.  Reduction to Single Control Parameter:  Staging

Multiple control parameters are quite common in real-life nonlinear problems.  They are the analog of multiple load conditions in linear problems.  But in the linear world, multiple load conditions can be processed *independently* because any load combination is readily handled by superposition.  In nonlinear problems, however, control parameters are *not* varied independently.  This aspect deserves an explanation, as it is rarely mentioned in the literature.

Typically the analysis proceeds as follows.  The user defines the control parameters to the computer program during a model preprocessing phase.  To illustrate the process, let us assume that for the analysis of a suspension bridge (Figure 3.1) there are six control parameters

$$\mathbf{\Lambda} = [\,\Lambda_1, \Lambda_2, \ldots, \Lambda_6\,]^T$$

where parameter $\Lambda_1$ is associated with own weight, $\Lambda_2$ and $\Lambda_3$ with live loads, $\Lambda_4$ with temperature changes, $\Lambda_5$ with foundation settlement and $\Lambda_6$ with wind velocity.

Suppose that $\Lambda_1 = 10$ corresponds to full own weight.  The first nonlinear analysis involves going from

$$\mathbf{\Lambda}_{ref} = [\,0,\ 0,\ 0,\ 0,\ 0,\ 0\,]^T$$

to

$$\mathbf{\Lambda}_W = [\,10,\ 0,\ 0,\ 0,\ 0,\ 0\,]^T$$

Next, assume that the effect of a temperature drop of $-20°C$ is to be investigated.  If a unit increment of $\Lambda_4$ corresponds to $1°C$, then the next nonlinear analysis corresponds to going from $\mathbf{\Lambda}_W$ to

$$\mathbf{\Lambda}_T = [\,10,\ 0,\ 0,\ -20,\ 0,\ 0\,]^T$$

A live load analysis might entail going from $\mathbf{\Lambda}_W$ to

$$\mathbf{\Lambda}_{LL} = [\,10,\ -20,\ 6,\ 0,\ 0\,]^T$$

and so on.

Each of these processes is called an *analysis stage* or simply *stage*.  A stage can be defined as "advancing the solution" from

$$\mathbf{\Lambda}_A \quad \text{to} \quad \mathbf{\Lambda}_B$$

when the solution $\mathbf{u}_A$ is known.  Furthermore, if we assume that the components of $\mathbf{\Lambda}$ will vary proportionally, we can introduce a single control parameter $\lambda$ that varies from 0 through 1 according to

$$\mathbf{\Lambda} = (1 - \lambda)\mathbf{\Lambda}_A + \lambda\mathbf{\Lambda}_B \tag{3.17}$$

This $\lambda$ is called the *stage control parameter*.  The nonlinear equation to be solved in $(A \rightarrow B)$ is

$$\boxed{\mathbf{r}(\mathbf{u}, \lambda) = \mathbf{0}} \tag{3.18}$$

with the initial condition $\mathbf{u} = \mathbf{u}_A$ at $\lambda = 0$.  The solution curve $\mathbf{u} = \mathbf{u}(\lambda)$ is called the *response* of the structure in the $(A \rightarrow B)$ stage.
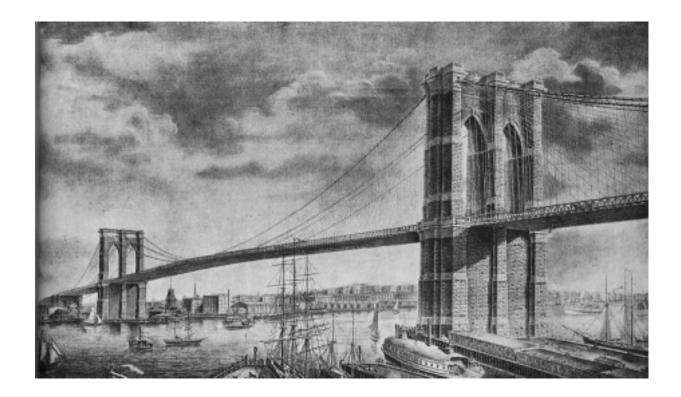
Figure 3.1.   The Brooklyn Bridge in 1876, drawn by F. Hildebrand for the Library of Congress.

The importance of staging in nonlinear static analysis arises from the inapplicability of the super-position principle of linear analysis.  For example, the sequences

$$\Lambda_A \to \Lambda_B \to \Lambda_C \qquad \Lambda_A \to \Lambda_C \tag{3.19}$$

$$\Lambda_A \to \Lambda_B \to \Lambda_A \qquad \Lambda_A \to \Lambda_C \to \Lambda_A \tag{3.20}$$

do not necessarily produce the same final solution.  Of course, this phenomenon is especially important in intrinsically path-dependent problems.

**Remark 3.5**.  In mathematical circles, (3.17) receive names such as *linear homotopy imbedding* and *piecewise linear imbedding*.  The vector $\mathbf{u}_{ref}$ for $\Lambda_{ref} = \mathbf{0}$ is called the *reference* or *initial configuration*.  Commonly $\mathbf{u}_{ref} = \mathbf{0}$, *i.e.*, the reference configuration is the origin of displacements.  This is not necessarily a physically attainable configuration.  In the suspension bridge example, $\mathbf{u}_{ref}$ is fictitious because bridges are not erected in zero gravity fields; on the other hand the own-weight configuration $\mathbf{u}_W$ is physically relevant.

**Remark 3.6**.  Many analysis sequences do start from the same configuration, for example the own-weight solution $\mathbf{u}_W$ in the suspension bridge analysis.  A comprehensive analysis system must therefore provides facilities for saving selected solutions on a permanent database, and the ability to restart the analysis from any saved solution.

**Remark 3.7**. The definition of $\lambda$ as a $[0 \to 1]$ parametrization of the $[A \to B]$ stage is somewhat artificial for the following reasons. First, there is no guarantee that a solution at $\mathbf{\Lambda}_B$ exists, so $\lambda = 1$ may not be in fact attainable. Second, in stability analysis (3.20) defines only a *direction* in control parameter space; in this case the user wants to know the smallest $\lambda$ (or largest $-\lambda$) at which a critical point occurs, and the analysis stops there.

**Remark 3.8**. As the number of control parameters grows, the number of possible analysis sequences increases combinatorially. Given the substantial costs usually incurred in these analyses, the experience and ability of the engineer can play an important role in weeding out unproductive paths. Selective linearization can also reduce the number of cases substantially, because invoking the superposition principle "factors out" certain parameters. For example, if the bridge response to live loads is essentially linear about the own-weight solution, parameters $\Lambda_2$ and $\Lambda_3$ may be removed from the nonlinear analysis, and the dimension of $\mathbf{\Lambda}$ is reduced from 6 to 4.

### §3.4.1. *Implicit Reduction

The most general reduction from multiple control parameters $\mathbf{\Lambda}$ to a single parameter $\lambda$ can be formulated as follows. Suppose that over a stage an implicit vector algebraic relation is established:

$$\mathbf{m}(\mathbf{\Lambda}, \lambda) = \mathbf{0}, \tag{3.21}$$

where $\mathbf{m}$ is a vector of $m$ equations that uniquely defines $\mathbf{\Lambda}$ if $\lambda$ is given. Augmenting the residual with (3.21) one obtains the expanded system

$$\begin{bmatrix} \mathbf{r}(\mathbf{u}, \mathbf{\Lambda}) \\ \mathbf{m}(\mathbf{\Lambda}, \lambda) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{3.22}$$

which implicitly defines $\mathbf{u}$ as a function of $\lambda$. Derivatives of $\mathbf{r}$ with respect to $\lambda$ are obtained by the chain rule; for example

$$\begin{bmatrix} \dot{\mathbf{r}}(\mathbf{u}, \mathbf{\Lambda}) \\ \dot{\mathbf{m}}(\mathbf{\Lambda}, \lambda) \end{bmatrix} = \begin{bmatrix} \mathbf{K}\dot{\mathbf{u}} - \mathbf{Q}\dot{\mathbf{\Lambda}} \\ \mathbf{A}\dot{\mathbf{\Lambda}} - \mathbf{a}^T \dot{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{3.23}$$

in which $\mathbf{A} = \partial \mathbf{m}/\partial \mathbf{\Lambda}$ and $\mathbf{a} = -\partial \mathbf{m}/\partial \lambda$. The linear interpolation (3.17) is a special case of (3.21) for which

$$\mathbf{m}(\mathbf{\Lambda}, \lambda) = \mathbf{\Lambda} - (1 - \lambda)\mathbf{\Lambda}_A - \lambda \mathbf{\Lambda}_B = \mathbf{0}. \tag{3.24}$$

For most applications use of the general form (3.23) is unwarranted. It may come handy, however, in complicated load parameter dependencies; for example, those associated with aerodynamic loads since the physical loads are quadratic functions of the velocity.

### Homework Exercises for Chapter 3

### Residual Force  Equations

**EXERCISE 3.1** [A:15]  For the example system (3.5) find $\mathbf{K}$ and $\mathbf{Q}$.

**EXERCISE 3.2** [A:15]  For the same example system and assuming the parametric representation (3.11) for $\mathbf{u}$ and $\mathbf{\Lambda}$, write down $\dot{\mathbf{r}}$, $\dot{\mathbf{K}}$, $\dot{\mathbf{Q}}$ and $\ddot{\mathbf{r}}$ in matrix/vector form.

**EXERCISE 3.3** [A:15]  Continuing the above exercise: if $\Lambda_1 = 2\lambda$ and $\Lambda_2 = \lambda$, write down $\mathbf{r}(\mathbf{u}, \lambda)$ and $\dot{\mathbf{r}}(\mathbf{u}, \lambda)$ in vector form.

**EXERCISE 3.4** [A:20]  Is (3.5) derivable from a potential energy function so that $\mathbf{r}$ can be represented as (3.2)? Can you guess by inspection what the potential $\Pi$ is?

**EXERCISE 3.5** [A:20]  Show that both $\mathbf{K}$ and $\mathbf{F}$ (assuming the latter exists) are symmetric if the residual is derivable from a potential energy function.