

STRAT V7.0 隔震、减震结构计算与分析——理论依据

目 录

一、使用振型叠加法的动力分析	2
1.1 求解的方程	2
1.2 变换为模态方程	2
1.3 仅由初始条件引起的响应	3
1.4 对任意荷载的通解	3
二、快速非线性分析	7
2.1 基本平衡方程	7
2.2 非线性力的计算	7
2.3 变换到模态坐标	7

一、使用振型叠加法的动力分析

1.1 求解的方程

动力平衡方程可以按下列方式重写为一组 N_d 二阶微分方程：

$$\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = \mathbf{F}(t) = \sum_{j=1}^J \mathbf{f}_j g_j(t) \quad (1.1)$$

所有可能的时间相关荷载类型（包括风、波浪和地震）可以表示为 J 空间向量 \mathbf{f}_j （该向量不是随时间变化的函数）之和以及第 j 时间函数。

动力自由度的数量等于系统中集中质量的数量。许多出版物主张在求解方程(1.1)前，通过静力凝聚消除所有无质量位移。静力凝聚法减少了所要求解的动力平衡方程的数量；然而，它明显增加了凝聚后刚度矩阵的密度和带宽。在建筑结构中，每个隔板只有三个集中质量，这种方法很有效并应用于建筑分析程序中。

然而，对于任意结构系统的动力求解，消除无质量位移法通常在数值上并不是有效的。因此，现代版本的 SAP 程序不使用静力凝聚以保持刚度矩阵的稀疏性。

1.2 变换为模态方程

求解方程(1.1)的基本数学方法是分离变量法。这种方法假设求出的解可以表现为以下形式：

$$\mathbf{u}(t) = \boldsymbol{\phi} \mathbf{Y}(t) \quad (1.2a)$$

其中 $\boldsymbol{\phi}$ 是一个 $N_d \times N$ 矩阵，该矩阵包含 N 个非时间函数的空间向量，而 $\mathbf{Y}(t)$ 是一个包含 N 个时间函数的向量。从方程(1.2a)可以导出：

$$\dot{\mathbf{u}}(t) = \boldsymbol{\phi} \dot{\mathbf{Y}}(t) \quad (1.2b)$$

$$\ddot{\mathbf{u}}(t) = \boldsymbol{\phi} \ddot{\mathbf{Y}}(t) \quad (1.2c)$$

在求解之前，要求空间函数满足下列质量和刚度正交性条件：

$$\boldsymbol{\phi}^T \mathbf{M} \boldsymbol{\phi} = \mathbf{I} \quad (1.3a)$$

$$\boldsymbol{\phi}^T \mathbf{K} \boldsymbol{\phi} = \boldsymbol{\Omega}^2 \quad (1.3b)$$

其中 \mathbf{I} 为一个对角单位矩阵， $\boldsymbol{\Omega}^2$ 是一个对角项为 ω_n^2 的对角矩阵。 ω_n 项的单位为弧度/秒，而且可以是或不是自由振动频率。应该注意除了正交属性之外，这些向量在数学原则上没有限制。本书中每个空间函数向量 $\boldsymbol{\phi}_n$ 总是正规化的，因此广义质量等于一，即 $\boldsymbol{\phi}_n^T \mathbf{M} \boldsymbol{\phi}_n = 1.0$ 。

把方程(1.2a)至(1.2c)带入方程(1.1)，然后前乘 $\boldsymbol{\phi}^T$ ，产生以下 N 个矩阵方程：

$$\mathbf{I} \ddot{\mathbf{Y}}(t) + \mathbf{d} \dot{\mathbf{Y}}(t) + \boldsymbol{\Omega}^2 \mathbf{Y}(t) = \sum_{j=1}^J \mathbf{p}_j g_j(t) \quad (1.4)$$

其中 $\mathbf{p}_j = \boldsymbol{\phi}^T \mathbf{f}_j$ 并定义为荷载函数 j 的模态参与系数。 p_{nj} 项与第 n 个振型相关。注意对于每个空间荷载条件 \mathbf{f}_j 都有一组 N 模态参与系数。

对于所有真实结构， $N \times N$ 的矩阵 \mathbf{d} 不是对角的，然而，要解耦模态方程，需要假设在振型之间无耦合的典型阻尼。因此，模态阻尼的对角项由下式定义：

$$d_{nn} = 2\xi_n \omega_n \quad (1.5)$$

其中 ξ_n 定义为第 n 振型中的阻尼与该振型的临界阻尼之比。一个非耦合的典型线性结构系统的模态方程为以下形式：

$$\ddot{y}_n(t) + 2\xi_n \omega_n \dot{y}_n(t) + \omega_n^2 y_n(t) = \sum_{j=1}^J p_{nj} g_j(t) \quad (1.6)$$

对于三维地震运动，此方程可以写成：

$$\ddot{\mathbf{y}}_n(t) + 2\xi_n \omega_n \dot{\mathbf{y}}_n(t) + \omega_n^2 \mathbf{y}_n(t) = p_{nx} \ddot{u}(t)_{gx} + p_{ny} \ddot{u}(t)_{gy} + p_{nz} \ddot{u}(t)_{gz} \quad (1.7)$$

其中三维模态参与系数或在此工况中的地震激励系数，由 $p_{ni} = -\boldsymbol{\phi}_n^T \mathbf{M}_j$ 定义，其中 j 是 x 、 y 或 z ，而 n 是振型数。注意本书中的所有振型是正规化的，因此 $\boldsymbol{\phi}_n^T \mathbf{M} \boldsymbol{\phi}_n = 1$ 。

1.3 仅由初始条件引起的响应

在对于各种类型的荷载给出方程(1.6)的解之前，可以方便的定义在表 1.1 中总结的附加常量与函数。这可以将本书其它部分的许多方程以一个紧凑的形式写出来。同样，符号表示法可以化简各种方程的代数推导与验证。此外，它可以将方程以一种便于编程与校验的形式来表示。

如果去掉下标 n ，对于典型振型，方程(1.6)可以写为：

$$\ddot{y}(t) + 2\xi \omega \dot{y}(t) + \omega^2 y(t) = 0 \quad (1.8)$$

其中初始模态位移 y_0 和速度 \dot{y}_0 指定为之前施加在结构上荷载的结果。注意在表 1.1 中给出的函数 $S(t)$ 与 $C(t)$ 是方程(1.8)的解。

方程(1.8)的解现在可以写为一下紧凑形式：

$$y(t) = A_1(t)y_0 + A_2(t)\dot{y}_0 \quad (1.9)$$

由于该解满足方程(1.8)以及初始条件，所以很容易验证。

用于动力相应方程的符号表示法摘要

表 1.1

常量：	$\omega_D = \omega \sqrt{1 - \xi^2}$	$\bar{\omega} = \omega \xi$	$\bar{\xi} = \frac{\xi}{\sqrt{1 - \xi^2}}$
	$a_0 = 2\xi\omega$	$\alpha_1 = \omega_D^2 - \bar{\omega}^2$	$\alpha_2 = 2\bar{\omega}\omega_D$
函数	$S(t) = e^{-\xi\omega t} \sin(\omega_D t)$	$C(t) = e^{-\xi\omega t} \cos(\omega_D t)$	
	$\dot{S}(t) = -\bar{\omega}S(t) + \omega_D C(t)$	$\dot{C}(t) = -\bar{\omega}C(t) - \omega_D S(t)$	
	$\ddot{S}(t) = -a_1 S(t) - a_2 C(t)$	$\ddot{C}(t) = -a_1 C(t) - a_2 S(t)$	
	$A_1(t) = C(t) + \bar{\xi}S(t)$	$A_2(t) = \frac{1}{\omega_D} S(t)$	

1.4 对任意荷载的通解

有许多不同方法可以求解这些典型模态方程。然而已经发现，在较小时间增量内使用多项式近似出来的荷载精确解，是计算机程序对此方程进行数值求解的最经济和最精确的方法。它没有稳定性问题，也不会引入数值阻尼。因为多数地震地面加速度在 0.005s 间隔内定义为线性的，所以对于所有频率的该类荷载而言，此方法是准确的。同样，如果位移用作基本输入，从线性加速度导出的荷载函数在每个时间间隔内为三次函数，如附录 J 所示。

为了简化符号表示法，所有荷载合成一起形成下列形式的典型模态方程：

$$\ddot{y}(t) + 2\xi \omega \dot{y}(t) + \omega^2 y(t) = R(t) \quad (1.10)$$

其中模态荷载 $R(t)$ 是一个分段多项式函数，如图 1.1 所示。注意可以使用附录 J 中概述的数值方法计算三次荷载函数需要的高阶导数。因此，在时间间隔 $i-1$ 至 i 内对于线性及三次荷载函数，需

求解的微分方程为下列形式：

$$\ddot{R}(t) + 2\dot{R}(t) + R(t) = \ddot{R}_i + \frac{2}{\Delta t} \dot{R}_i + \frac{3}{6} \ddot{R}_i \quad (1.11)$$

$$R(t) = R_{i-1} + t\dot{R}_{i-1} + \frac{t^2}{2}\ddot{R}_{i-1} + \frac{t^3}{6}\ddot{\ddot{R}}, \text{ 在间隔 } i-1 \text{ 到 } i$$

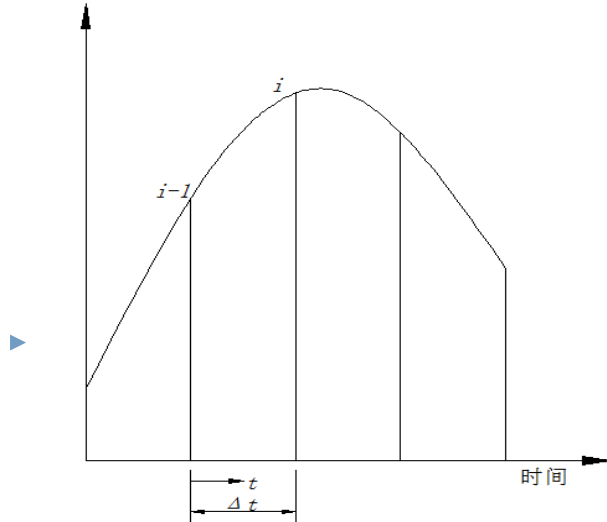


图 1.1 模态荷载函数

对于间隔内的线性荷载

$$\ddot{\ddot{R}}_i = 0$$

$$\ddot{R}_i = 0$$

$$\dot{R}_{i-1} = \frac{R_i - R_{i-1}}{\Delta t}$$

对于间隔内的三次曲线

其中 R_i 和 \dot{R}_i 是指定的

$$\ddot{R}_i = \frac{6}{\Delta t^2}(R_i - R_{i+1}) + \frac{2}{\Delta t}(\dot{R}_{i+1} + 2\dot{R}_i)$$

$$\ddot{\ddot{R}} = \frac{\ddot{R}_i - \ddot{R}_{i-1}}{\Delta t}$$

根据线性微分方程的基本理论，方程(1.11)的通解是一个齐次解和一个特解的和，并且为下列形式：

$$y(t) = b_1 S(t) + b_2 C(t) + b_3 + b_4 t + b_5 t^2 + b_6 t^3 \quad (1.12a)$$

与此解相关的速度与加速度为：

$$\dot{y}(t) = b_1 \dot{S}(t) + b_2 \dot{C}(t) + b_4 + 2b_5 t + 3b_6 t^2 \quad (1.12b)$$

$$\ddot{y}(t) = b_1 \ddot{S}(t) + b_2 \ddot{C}(t) + 2b_5 + 6b_6 t \quad (1.12c)$$

这些方程概括为下列矩阵方程：

$$y_i = \begin{bmatrix} y_i \\ \dot{y}_i \\ \ddot{y}_i \end{bmatrix} = \begin{bmatrix} S(t) & C(t) & 1.0 & t & t^2 & t^3 \\ \dot{S}(t) & \dot{C}(t) & 0 & 1.0 & 2t & 3t^2 \\ \ddot{S}(t) & \ddot{C}(t) & 0 & 0 & 2.0 & 6t \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} = \mathbf{B}(t)\mathbf{b} \quad (1.13)$$

现在可以求解常量 b_i 。 $t=0$ 时的初始条件为 $\dot{y}(0) = \dot{y}_{i-1}$ 和 $y(0) = y_{i-1}$ 。因此，从方程(1.12a) 与 (1.12b)得到：

$$\dot{y}_{i-1} = \omega_D b_1 - \bar{\omega} b_2 + b_4 \quad (1.14a)$$

$$y_{i-1} = b_2 + b_3 \quad (1.14b)$$

将方程(1.12a)、(1.12b)和(1.12c)代入方程(1.11)并设定多项式各项的系数相等，可以产生下列 4 个方程：

$$1: \quad R_{i-1} = \omega^2 b_3 + a_0 b_4 + 2b_5 \quad (1.15a)$$

$$t: \quad \dot{R}_{i-1} = \omega^2 b_4 + 2a_0 b_5 + 6b_6 \quad (1.15b)$$

$$t^2: \quad \ddot{R}_{i-1} = 2\omega^2 b_5 + 6a_0 b_6 \quad (1.15c)$$

$$t^3: \quad \dddot{R}_{i-1} = 6\omega^2 b_6 \quad (1.15d)$$

由方程(1.14)和(1.15)给出的这 6 个方程，可以写成如下矩阵方程：

$$\begin{bmatrix} \dot{y}_{i-1} \\ y_{i-1} \\ R_{i-1} \\ \dot{R}_{i-1} \\ \ddot{R}_{i-1} \\ \dddot{R}_{i-1} \end{bmatrix} = \begin{bmatrix} \omega_D & -\bar{\omega} & 0 & 1.0 & 0 & 0 \\ 0 & 1.0 & 1.0 & 0 & 0 & 0 \\ 0 & 0 & \omega^2 & a_0 & 2.0 & 0 \\ 0 & 0 & 0 & \omega^2 & 2a_0 & 6.0 \\ 0 & 0 & 0 & 0 & 2\omega^2 & 6a_0 \\ 0 & 0 & 0 & 0 & 0 & 6\omega^2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} \quad (1.16a)$$

或用矩阵表示为：

$$\bar{R}_{i-1} = C^{-1} b \quad (1.16b)$$

因此：

$$b = C\bar{R}_{i-1} \quad (1.17)$$

可以形成上三角矩阵 C 的逆，或很容易用计算机程序进行求逆。因此，模态方程在时间 i 处由时间步长内的一个三次荷载所产生的精确解为：

$$\bar{y}_i = B(\Delta t)C\bar{R}_{i-1} = A\bar{R}_{i-1} \quad (1.18)$$

方程(1.18)是一个非常简单但又很强大的递归关系。表 1.2 中概述了线性或三次荷载的完整算法。注意对于每个振型 3×6 的 A 矩阵只需计算一次。因此，对于每个时间增量，大约需要 20 次乘法运算与 16 次加法运算。现代廉价的个人计算机可以在大约 10^{-6} s 内完成一次乘法运算和一次加法运算。因此，对于一个持续时间为 50s 的地震，每秒求解 200 步所需的计算时间大约是 0.01s；或者说 100 个模态方程可以在 1s 内解出。因此，没有必要考虑其他的数值方法，比如近似的快速 Fourier 变换法或 Duhamel 积分数值计算，来求解这些方程。由于此精确分段多项式技术方法速度快，因此它也可以用来很少的计算机时间建立精确地震反应谱。

模态方程求解的高阶递归算法

表 1.2

1. 要解的方程：	$\ddot{y}(t) + 2\xi\omega\dot{y}(t) + \omega^2 y(t) = R_{i-1} + t\dot{R}_{i-1} + \frac{t^2}{2}\ddot{R}_{i-1} + \frac{t^3}{6}\dddot{R}_{i-1}$		
2. 初始计算：	$\omega_D = \omega\sqrt{1-\xi^2}$	$\bar{\omega} = \omega\xi$	$\bar{\xi} = \frac{\xi}{\sqrt{1-\xi^2}}$
	$a_0 = 2\xi\omega$	$a_1 = \omega_D^2 - \bar{\omega}^2$	$a_2 = 2\bar{\omega}\omega_D$
	$S(\Delta t) = e^{-\xi\omega\Delta t} \sin(\omega_D\Delta t)$	$C(\Delta t) = e^{-\xi\omega\Delta t} \cos(\omega_D\Delta t)$	
	$\dot{S}(\Delta t) = -\bar{\omega}S(\Delta t) + \omega_D C(\Delta t)$	$\dot{C}(\Delta t) = -\bar{\omega}C(\Delta t) + \omega_D S(\Delta t)$	
	$\ddot{S}(\Delta t) = -a_1 S(\Delta t) - a_2 C(\Delta t)$	$\ddot{C}(\Delta t) = -a_1 C(\Delta t) - a_2 S(\Delta t)$	
$B(\Delta t) =$	$\begin{bmatrix} S(\Delta t) & C(\Delta t) & 1.0 & \Delta t & \Delta t^2 & \Delta t^3 \\ \dot{S}(\Delta t) & \dot{C}(\Delta t) & 0 & 1.0 & 2\Delta t & 3\Delta t^2 \\ \ddot{S}(\Delta t) & \ddot{C}(\Delta t) & 0 & 0 & 2.0 & 6\Delta t \end{bmatrix}$		

$$C = \begin{bmatrix} \omega_D & -\bar{\omega} & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \omega^2 & a_0 & 2.0 & 0 \\ 0 & 0 & 0 & \omega^2 & 2a_0 & 6.0 \\ 0 & 0 & 0 & 0 & 2\omega^2 & 6a_0 \\ 0 & 0 & 0 & 0 & 0 & 6\omega^2 \end{bmatrix}$$

$$A = B(\Delta t)C$$

3. 递归求解: $i=1, 2, 3, \dots$

$$(a) \ddot{R}_i = \frac{6}{\Delta t^2}(R_i - R_{i+1}) + \frac{2}{\Delta t}(R_{i+1} + 2R_i)$$

$$(b) \ddot{R}_{i-1} = \frac{\ddot{R}_i - \ddot{R}_{i-1}}{\Delta t}$$

$$(c) \bar{y}_i = A\bar{R}_{i-1}$$

(d) $i = i + 1$ 并返回到 3(a)。

二、快速非线性分析

2.1 基本平衡方程

FNA 方法是一种简单方法，这种方法能满足基本力学方程（平衡、力-变形和协调性）的要求。在时间 t 处，结构的计算机模型精确的力平衡由下列矩阵方程表示：

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) + R_{NL}(t) = R(t) \quad (2.1)$$

其中 M 、 C 与 K 分别是质量矩阵、比例阻尼矩阵和刚度矩阵。这三个方阵的大小等于未知节点位移 N_d 的总数。弹性刚度矩阵 K 忽略非线性单元的刚度。时间相关的向量 $\ddot{u}(t)$ 、 $\dot{u}(t)$ 、 $u(t)$ 与 $R(t)$ 分别是节点加速度、速度、位移和外部荷载。而 $R_{NL}(t)$ 来源于非线性单元力总和的整体节点力向量，是通过在每个时间点上的迭代计算出来的。

如果没有非线性单元计算机模型不稳定，可在非线性单元的位置添加任意刚度的“有效弹性单元”。如果在方程(2.1)的两边加上这些有效力 $K_e u(t)$ ，精确的平衡方程可写为：

$$M\ddot{u}(t) + C\dot{u}(t) + (K + K_e)u(t) = R(t) - R_{NL}(t) + K_e u(t) \quad (2.2)$$

其中 K_e 是任意值的有效刚度。因此，非线性计算机模型的精确动力平衡方程可写为：

$$M\ddot{u}(t) + C\dot{u}(t) + \bar{K}u(t) = \bar{R}(t) \quad (2.3)$$

弹性刚度矩阵 \bar{K} 等于 $K + K_e$ ，并且是已知的。有效外部荷载 $\bar{R}(t)$ 等于 $R(t) - R_{NL}(t) + K_e u(t)$ ，必须以迭代方法计算。如果对有效弹性刚度进行较好的估计，收敛速度就可能加速，这是因为未知的荷载项 $-R_{NL}(t) + K_e u(t)$ 很小。

2.2 非线性力的计算

在任何时刻，非线性单元内的 L 个非线性变形 $d(t)$ 可从下面的位移变换方程计算出来：

$$d(t) = bu(t) \quad (2.4)$$

同时，在非线性变形中与时间相关的变化率 $\dot{d}(t)$ 可由下式得出：

$$\dot{d}(t) = b\dot{u}(t) \quad (2.5)$$

注意对于小的位移，位移变换矩阵 b 不是时间的函数并且是精确的。

如果所有非线性单元中的时称变形和速度是已知的，在任何时刻非线性单元的非线性力 $f(t)$ 可由每个非线性单元的非线性材料属性精确的计算出来。很显然，这只能通过每个时间点上的迭代来完成。

2.3 变换到模态坐标

求解方程(18.3)的第一步是要计算一组 N 个正交荷载相关的 Ritz 向量 Φ ，使其满足下列方程：

$$\Phi^T M \Phi = I \quad (2.6a)$$

和

$$\Phi^T K \Phi = \Omega^2 \quad (2.6b)$$

其中 I 为单位矩阵，而 Ω^2 为对角矩阵，在该对角矩阵中，对角项被定义为 ω_n^2 。

系统的响应现在可以通过引入下面的矩阵变换用向量来表示：

$$u(t) = \Phi Y(t) \quad \dot{u}(t) = \Phi \dot{Y}(t) \quad \ddot{u}(t) = \Phi \ddot{Y}(t) \quad (2.7)$$

把这些方程代入方程式(18.1)并在方程式两边同乘以 Φ^T ，这将产生一组可由下面矩阵方程表示的 N 个非耦合方程：

$$I\ddot{Y}(t) + \Lambda\dot{Y}(t) + \Omega^2 Y(t) = F(t) \quad (2.8)$$

其中线性和非线性模态力由下式得出：

$$F(t) = \Phi^T \bar{R}(t) = \Phi^T R(t) - \Phi^T R_{NL}(t) + \Phi^T K_e u(t) \quad (2.9)$$

阻尼矩阵可对角化的假设是和传统标准叠加法一致的，用这种叠加法可在模态水平用临界阻尼百分率来指定阻尼值。矩阵 Λ 的对角项为 $2\xi_n \omega_n$ ，其中是 ξ_n 是振型 n 的阻尼比。应该注意在结构中的任意位置上与集中阻尼器相关的力可作为非线性力向量的一部分。

同样，如果所用的 LDR 向量数等于自由度的总数 N_d ，方程(2.8)在时间 t 处就是精确的。因此，如果使用非常小的时间步长并在每个时间步长中都使用迭代，此方法会收敛到精确解。LDR 向量的使用可明显减少所需的振型数量。

因此 $u(t) = \Phi Y(t)$ ，非线性单元中的变形可按模态坐标直接表示为：

$$d(t) = B Y(t) \quad (2.10)$$

其中单元变形-模态坐标变换矩阵由下式定义：

$$B = b \Phi \quad (2.11)$$

特别要注意 $L \times N$ 矩阵 B 不是时间的函数并且相对较小，它在模态方程积分之前只需计算一次。

在任何时刻，只要给出了非线性单元的变形与行为历史，可根据基本的非线性属性以及单元的变形历史来计算非线性单元中的里 $f(t)$ 。根据虚功基本原理，非线性模态力由下式计算：

$$F_{NL}(t) = B^T f(t) \quad (2.12)$$

有效弹性力也可重写成：

$$F_e(t) = \Phi^T K_e u(t) = \Phi^T b^T k_e b u(t) = B^T k_e d(t) \quad (2.13)$$

其中 k_e 是非线性单元局部参考系的有效线性刚度矩阵。

2.4 非线性模态方程的求解

在求解模态方程之前，第一个步骤需要计算无非线性单元结构的荷载相关向量(Load Dependent Vectors)。在开始逐步求解之前，对变形-模态变换矩阵 B 也同样只需计算一次。一个典型的模态方程是：

$$\ddot{y}_n(t) + 2\xi_n \omega_n \dot{y}_n(t) + \omega_n^2 y_n(t) = \bar{f}_n(t) \quad (2.14)$$

其中 $\bar{f}_n(t)$ 是模态荷载，并且对于非线性单元而言，它是在同一时间点上的所有其它模态响应的函数。因此，模态方程组必须同时积分，而且需要通过迭代以获得在时间 t 处的所有模态方程。方程(1.13)总结了在某一时间步长中对荷载线性或三次变量的模态方程的精确解，并以指数、平方根、正弦和余弦函数表示。然而，表 1.2 给出的这些计算量大的函数，对所有振型而言都是预先计算的，并且作为常量用于每个时间步长内的积分。除此之外，这种精确的分段积分法允许较大的时间步长。

表 2.1 中概述了以迭代形式写出的完整非线性求解算法。

非线性求解算法一览

表 2.1

1. 初始计算——在逐步求解之前

(a) 为无非线性单元的结构计算 N 个荷载相关的 Ritz 向量 Φ 。这些向量具有 N_d 位移自由度。

(b) 计算 $N \times L$ 矩阵 B 。其中 L 是所有非线性单元中的自由度总数。

(c) 计算积分常量 $A_1 \dots$ 用于每个振型的模态方程的分段精确积分。

2. 非线性求解——在时间 Δt 、 $2\Delta t$ 、 $3\Delta t \dots$

(a) 在时间 t 处使用泰勒级数近似求解。

$$\mathbf{Y}(t) = \mathbf{Y}(t - \Delta t) + \Delta t \dot{\mathbf{Y}}(t - \Delta t) + \frac{\Delta t^2}{2} \ddot{\mathbf{Y}}(t - \Delta t)$$

$$\dot{\mathbf{Y}}(t) = \dot{\mathbf{Y}}(t - \Delta t) + \Delta t \ddot{\mathbf{Y}}(t - \Delta t)$$

(b) 对于迭代 i ，计算 L 个非线性变形和速度。

$$\mathbf{d}(t)^i = \mathbf{B}\mathbf{Y}(t)^i \quad \text{和}$$

$$\dot{\mathbf{d}}(t)^i = \mathbf{B}\dot{\mathbf{Y}}(t)^i$$

(c) 根据非线性单元中的变形和速度历史，计算 L 个非线性力 $\mathbf{f}(t)^i$ 。

(d) 计算新模态力向量。

$$\bar{\mathbf{F}}(t)^i = \mathbf{F}(t) - \mathbf{B}^T [\mathbf{f}(t)^i - \mathbf{k}_e \mathbf{d}(t)^i]$$

(e) 为下次迭代使用分段精确积分法求解模态方程。

$$\mathbf{Y}(t)^i, \dot{\mathbf{Y}}(t)^i, \ddot{\mathbf{Y}}(t)^i$$

(f) 计算误差范数：

$$E_{rr} = \frac{\sum_{n=1}^N |\bar{f}_n(t)^i| - \sum_{n=1}^N |\bar{f}_n(t)^{i-1}|}{\sum_{n=1}^N |\bar{f}_n(t)^i|}$$

(g) 检收敛——此处公差 Tol 是指定的。

如果 $E_{rr} > Tol$ ，则转到步骤 2b， $i = i + 1$

如果 $E_{rr} < Tol$ ，则转到步骤 2a， $t = t + \Delta t$

三、使用数值积分的动力分析——Newmark 方法族

1959 年, Newmark[1959]为了解决冲击波和地震荷载的结构动力问题提出了单步积分法族。在过去的 40 年间, Newmark 方法已经应用于许多实际工程结构的动力分析中。除此之外, 许多其他研究者对其也进行了修改和改进。为了说明数值积分法族的使用, 考虑下列形式写出的线性动力平衡方程的解:

$$M\ddot{\mathbf{u}}_t + C\dot{\mathbf{u}}_t + K\mathbf{u}_t = \mathbf{F}_t \quad (3.1)$$

直接使用泰勒级数是一种严格的方法得到下列两个额外方程:

$$\mathbf{u}_t = \mathbf{u}_{t-\Delta t} + \Delta t \dot{\mathbf{u}}_{t-\Delta t} + \frac{\Delta t^2}{2} \ddot{\mathbf{u}}_{t-\Delta t} + \frac{\Delta t^3}{6} \dddot{\mathbf{u}}_{t-\Delta t} + \dots \quad (3.2a)$$

$$\dot{\mathbf{u}}_t = \dot{\mathbf{u}}_{t-\Delta t} + \Delta t \ddot{\mathbf{u}}_{t-\Delta t} + \frac{\Delta t^2}{2} \dddot{\mathbf{u}}_{t-\Delta t} + \dots \quad (3.2b)$$

Newmark 截短了这些方程并以下列形式来表达这些方程:

$$\mathbf{u}_t = \mathbf{u}_{t-\Delta t} + \Delta t \dot{\mathbf{u}}_{t-\Delta t} + \frac{\Delta t^2}{2} \ddot{\mathbf{u}}_{t-\Delta t} + \beta \Delta t^3 \ddot{\mathbf{u}} \quad (3.2c)$$

$$\dot{\mathbf{u}}_t = \dot{\mathbf{u}}_{t-\Delta t} + \Delta t \ddot{\mathbf{u}}_{t-\Delta t} + \gamma \Delta t^2 \ddot{\mathbf{u}} \quad (3.2d)$$

如果假定加速度在时间步长内是线性的, 下列方程可写为:

$$\ddot{\mathbf{u}} = \frac{(\ddot{\mathbf{u}}_t - \ddot{\mathbf{u}}_{t-\Delta t})}{\Delta t} \quad (3.3)$$

将方程(3.3)代入方程(3.2c 和 3.2d), 可产生标准形式的 Newmark 方程:

$$\mathbf{u}_t = \mathbf{u}_{t-\Delta t} + \Delta t \dot{\mathbf{u}}_{t-\Delta t} + \left(\frac{1}{2} - \beta\right) \Delta t^2 \ddot{\mathbf{u}}_{t-\Delta t} + \beta \Delta t^2 \ddot{\mathbf{u}}_t \quad (3.4a)$$

$$\dot{\mathbf{u}}_t = \dot{\mathbf{u}}_{t-\Delta t} + (1 - \gamma) \Delta t \ddot{\mathbf{u}}_{t-\Delta t} + \gamma \Delta t \ddot{\mathbf{u}}_t \quad (3.4b)$$

Newmark 为结构系统的每个位移自由度的每个时间步长, 通过迭代来求解方程(3.4a、3.4b 和 3.1)。通过用自由度相关的质量除以从方程(3.1)中获得 $\ddot{\mathbf{u}}_t$ 项。

在 1962 年, Wilson 把 Newmark 方法写成矩阵形式, 添加了刚度与质量比例阻尼, 并通过在每个时间步长对方程直接求解来消除了对迭代的需要。这要求方程(3.4a 和 3.4b)以下列形式重写:

$$\ddot{\mathbf{u}}_t = b_1(\mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + b_2 \dot{\mathbf{u}}_{t-\Delta t} + b_3 \ddot{\mathbf{u}}_{t-\Delta t} \quad (3.5a)$$

$$\dot{\mathbf{u}}_t = b_4(\mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + b_5 \dot{\mathbf{u}}_{t-\Delta t} + b_6 \ddot{\mathbf{u}}_{t-\Delta t} \quad (3.5b)$$

其中常量 b_1 到 b_6 在表 3.1 中定义的。将方程(3.5a)和(3.5b)代入方程(3.1), 可在时间 t 处按未知节点位移 \mathbf{u}_t 的形式写出系统动力平衡方程。即:

$$(b_1 \mathbf{M} + b_4 \mathbf{C} + \mathbf{K})\mathbf{u}_t = \mathbf{F}_t + \mathbf{M}(b_1 \mathbf{u}_{t-\Delta t} - b_2 \dot{\mathbf{u}}_{t-\Delta t} - b_3 \ddot{\mathbf{u}}_{t-\Delta t}) + \mathbf{C}(b_4 \mathbf{u}_{t-\Delta t} - b_5 \dot{\mathbf{u}}_{t-\Delta t} - b_6 \ddot{\mathbf{u}}_{t-\Delta t}) \quad (3.6)$$

在表 3.1 中总结了 Newmark 直接积分算法。注意常量 b_i 只需计算一次。同样, 对于线性系统, 只进行一次有效动力刚度矩阵 $\bar{\mathbf{K}}$ 的形成和三角分解。

Newmark 直接积分法的概述

表 3.1

<p>1. 初始计算</p> <p>(a) 形成静力刚度矩阵 \mathbf{K}、质量矩阵 \mathbf{M} 以及阻尼矩阵 \mathbf{C}</p> <p>(b) 指定积分参数 β 与 γ</p> <p>(c) 计算积分常数</p>

$$b_1 = \frac{1}{\beta \Delta t^2}$$

$$b_2 = \frac{1}{\beta \Delta t}$$

$$4b_3 = \beta - \frac{1}{2}$$

$$b_4 = \gamma \Delta t b_1$$

$$b_5 = 1 + \gamma \Delta t b_2$$

$$b_6 = \Delta t (1 + \gamma b_3 - \gamma)$$

(d) 形成有效刚度矩阵 $\bar{\mathbf{K}} = \mathbf{K} + b_4 \mathbf{C} + b_1 \mathbf{M}$

(e) 三角分解有效刚度矩阵 $\bar{\mathbf{K}} = \mathbf{LDL}^T$

(f) 指定初始条件 \mathbf{u}_0 、 $\dot{\mathbf{u}}_0$ 、 $\ddot{\mathbf{u}}_0$

2. 对于每个时间步长 $t = \Delta t$ 、 $2\Delta t$ 、 $3\Delta t \dots$

(a) 计算有效荷载向量

$$\bar{\mathbf{F}}_t = \mathbf{F}_t + \mathbf{M}(b_1 \mathbf{u}_{t-\Delta t} - b_2 \dot{\mathbf{u}}_{t-\Delta t} - b_3 \ddot{\mathbf{u}}_{t-\Delta t}) + \mathbf{C}(b_4 \mathbf{u}_{t-\Delta t} - b_5 \dot{\mathbf{u}}_{t-\Delta t} - b_6 \ddot{\mathbf{u}}_{t-\Delta t})$$

(b) 在时间 t 对节点位移求解

$$\mathbf{LDL}^T \mathbf{u}_t = \bar{\mathbf{F}}_t \quad \text{仅向前消去及后退代入}$$

(c) 在时间 t 计算节点速度与加速度

$$\dot{\mathbf{u}}_t = b_4 (\mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + b_5 \dot{\mathbf{u}}_{t-\Delta t} + b_6 \ddot{\mathbf{u}}_{t-\Delta t}$$

$$\ddot{\mathbf{u}}_t = b_1 (\mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + b_2 \dot{\mathbf{u}}_{t-\Delta t} + b_3 \ddot{\mathbf{u}}_{t-\Delta t}$$

(d) 转到步骤 2(a), $t = t + \Delta t$