

## 第 3 章 仿真问题

**背景：**对动态系统模型的实验是系统设计的标准任务之一，这种仿真结果为有深远影响的设计决策奠定了基础。如今，(商用)计算机辅助工具一般提供建模与仿真平台，因而经常使用预先存在的模型库。但是，在这个极端重要的设计阶段，计算机化的仿真模型与用户(极端情况下用户可能是很天真的)经常存在危险的理解差距。在不利情况下，这很容易导致有缺陷的仿真结果。因此，尤其在采用现代仿真工具时，胜任的系统工程师必须了解仿真实现与求解方法的特殊性。系统工程师只有具备这些知识，才有可能识别出潜在的问题并采用合适的措施缓解问题，要么是改进模型，要么采用可用仿真器功能的目标选择与参数化：“欣赏但觉察地使用工具”(using the tool with appreciation and awareness)。

**本章内容：**本章讨论仿真实验的数学模型实现的特定方面以及关于机电系统模型的特殊问题与求解方法。在此范围内，假设读者已具备数值积分与一般仿真方法的基础知识。在简要讨论数值稳定性、积分步长的重要影响以及不同积分方法的性质之后，本章将介绍多体系统(表现为具有明显特征模态和弱以至无阻尼特征模态的刚性系统结构)仿真的典型问题与求解方法。对于线性高阶多体模型(例如通过有限元方法产生的那些模型)，介绍了使用状态转移矩阵的高效并精确的积分步骤。应用基本概念阐明了微分代数方程(DAE)系统的非平凡(non-trivial)数值积分与混杂现象的处理。最后，通过一个实例说明 DAE 系统的闭式(closed-form)建模及其仿真实现。

### 3.1 系统工程背景

#### 建模与仿真

系统设计(基于模型的设计)包含两个紧密交织的任务：建模与对模型的实验(仿真)。从图 2.3 可以清楚看出，仿真结果的预测能力(即其在多大程度上代表了实际系统的行为)取决于建模误差与仿真误差之和。特定类型模型的选择决定了仿真任务的难度以及最终的仿真误差。与通过面向对象建模获得的高冗余 DAE 系统相比，采用最小坐标的常微分方程组形式的简洁模型更易实现和计算。所以，总是需要对建模工作量、期望的模型精度以及仿真所需的工作量三者进行折衷考虑。

#### 计算机辅助仿真

与计算机辅助建模一起，现代设计工具使我们能够几乎毫不费力地进行仿真实验。这种便利性当然是用户所期望的。但是，若所考虑的模型具有某些不良性质时，现代设计工具会隐藏巨大的危险性。尽管事实上好的计算机工具有很多内置的主要功能正常性检查(sanity checks)，但一个有缺陷的求解算法参数化就可能导致完全错误的仿真结果。在特别有害的情况下(如复杂模型)很难检测出这些错误。计算机辅助工具通常仅检查模型的句

法和参数以及实验参数。原则上模型的语义仍未被监测,因而可能是一个潜在的错误来源。

### 仿真工具的妙用

本章将特别关注常微分方程组与 DAE 系统数值求解方法的语义,即求解算法(数值积分方法)及其重要参数(步长、阶次等)的意义。这些背景的目的是使我们能够在行地选用在当前的商用计算机工具中实现的那些常用方法。

### 预备知识

假设读者熟悉数值积分的基本概念(例如,显式与隐式法、单步与多步法、龙格-库塔法、基于误差监控的自适应步长等),推荐需要更新或复习有关基础知识的读者参考数值分析领域的有关文献(例如,Faires、Burden 2002)。关于直接适合动态系统仿真的方法,可以参见专著(Cellier、Kofman 2006)。

## 3.2 数值积分的基础

### 3.2.1 微分方程的数值积分

#### 仿真实验

为了进行计算机辅助仿真实验,需要基于内在的数学模型计算感兴趣系统变量的近似解。于是,可以认为“在对数学模型进行仿真”。

为此,首先考虑如下具有单输入  $u(t)$  和单输出  $y(t)$  的普通非线性状态空间模型(见图 3.1)<sup>1</sup>:

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}(\mathbf{x}, u, t) \quad (3.1)$$

$$\dot{\mathbf{y}} = \tilde{\mathbf{g}}(\mathbf{x}, u, t) \quad (3.2)$$

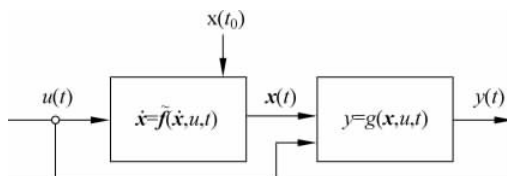


图 3.1 单输入单输出(SISO)动态系统的状态空间模型

当对上述系统进行仿真时,一般关心的是在有限时间区间  $[t_0, t_f]$  内解  $\mathbf{x}(t)$  或  $y(t)$  随时间变化情况。在这种情况下,可以假设输入  $u(t)$  在  $[t_0, t_f]$  内的变化情况是已知的。

为了采用式(3.2)计算输出  $y(t)$ ,只需要确定  $n$  个一阶微分方程组(3.1)在时间区间  $[t_0, t_f]$  的解  $\mathbf{x}(t)$ 。

给定上述假设条件,可以提出如下微分方程的数值积分这一基本问题:找到如下常微分方程组的解  $\mathbf{x}(t)$  的近似  $\hat{\mathbf{x}}(t)$ <sup>2</sup>

<sup>1</sup> 该模型是一个指标为 0 的 DAE 系统(参见 2.4 节)。3.6 节将讨论高指标 DAE 系统的求解。

<sup>2</sup> 假设向量场  $\mathbf{f}(\cdot)$  是光滑的。若  $\mathbf{f}(\cdot)$  存在不连续性(例如,输入激励函数或状态变量  $\mathbf{x}(t)$  存在阶跃变化),则须作出特别规定,见 3.7 节。

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \in R^n \quad (3.3)$$

### 单步法：显式与隐式

使用微分方程(3.3)的差分近似或相应的积分方程,可获得方程(3.3)的近似解(即有限个值 $\hat{\mathbf{x}}(t_k)$ )。然后,为了只根据上次计算值 $\hat{\mathbf{x}}(t_k)$ 来计算新的近似值 $\hat{\mathbf{x}}(t_{k+1})$ ,采用“单步法”可得下列的一般递推关系式:

$$\hat{\mathbf{x}}(t_{k+1}) = \hat{\mathbf{x}}(t_k) + h\varphi(\hat{\mathbf{x}}(t_k), \hat{\mathbf{x}}(t_{k+1}), t_k, h) \quad (3.4)$$

其中, $\varphi(\cdot)$ 为增量函数, $h$ 为步长。若增量函数不依赖于 $\hat{\mathbf{x}}(t_{k+1})$ ,则称该方法为显式(如欧拉法、龙格-库塔法),否则称其为隐式(如梯形法)(Faires、Burden 2002)。

增量函数 $\varphi(\cdot)$ 与步长 $h$ 的不同选择决定了近似精度(图3.2)。

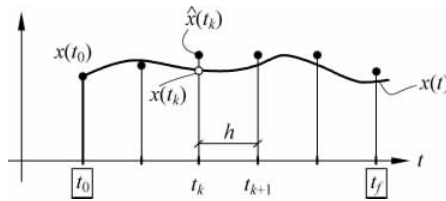


图 3.2 数值积分: 微分方程的近似解

### 3.2.2 稳定性的概念

#### 定义 3.1

局部离散化误差: 在 $t_{k+1}$ 时刻显式单步法<sup>3</sup>的局部离散化误差(LDE)是下面的值:

$$\mathbf{d}_{k+1} := \{\mathbf{x}(t_{k+1}) - \mathbf{x}(t_k)\} - h \cdot \varphi(\mathbf{x}(t_k), t_k, h)$$

上式右端第一项为真实解的单步变化,第二项为应用积分算法后相对于真实解 $\mathbf{x}(t_k)$ 的单步变化。 $\text{LDE}_{k+1}$ 表示积分方法与真实解在单步上的偏差。因此,LDE 衡量方程(3.4)给出的解与真实解 $\mathbf{x}(t_k)$ 接近的程度。

#### 定义 3.2

全局离散化误差: 在固定时刻 $t_k$ 的全局离散化误差(GDE)是下面的值:

$$\mathbf{g}_k := \mathbf{x}(t_k) - \hat{\mathbf{x}}(t_k)$$

因而, $\text{GDE}_{g_i}$ 表示近似解 $\hat{\mathbf{x}}(t_k)$ 与真实解 $\mathbf{x}(t_k)$ 之间的偏差,而且特别包含所有以前 $k$ 步( $j=0,1,\dots,(k-1)$ )的累积误差(LDE 与 GDE)。

#### 定义 3.3

一致性: 求解初值问题的数值积分方法被称为一致的,如果在步长趋于0时局部离散化误差之和 $R_{\text{LDE}}$ 也趋于0,即有下面的条件:

$$\lim_{h \rightarrow 0} \left( \frac{1}{h} R_{\text{LDE}} \right) = 0$$

#### 定义 3.4

收敛性: 求解初值问题的数值积分方法被称为收敛的,如果当步长趋于0时全局离散

<sup>3</sup> 对于隐式法和多步法等其他方法,LDE 也可以类似地定义。

化误差在整个积分区间上也趋于 0,即有下面的条件:

$$\lim_{h \rightarrow 0} (\hat{x}_k - x_k) = \lim_{h \rightarrow 0} g_k = 0, \quad \forall k, \text{ i. e. } t \in [t_0, t_f]$$

### 稳定性

需要区分下列类型的稳定性:

系统模型的固有稳定性

所使用的稳定性概念包括输入输出稳定性(如 BIBO 稳定性)或(渐近)状态稳定性(Ogata 2010)。如果系统模型在上述意义下稳定,则称系统是(固有)稳定的。

积分算法的数值稳定性

求解初值问题的数值积分方法被称为是“数值稳定的”,如果被积值  $\hat{x}_k$  的“微小误差”也只在此后计算  $\hat{x}_{k+1}$  时产生“微小误差”(即有足够的误差抑制)(Faires, Burden 2002)。

给定以上定义,于是有下面的基本定理:

#### 定理 3.1

一个数值积分方法是收敛的,当且仅当其是一致的并是数值稳定的。

所以,收敛性、一致性与数值稳定性紧密联系在一起,这些性质是执行仿真实验的基本性质。尽管商品仿真工具一般将大量具有一致性的积分方法作为内置功能(单单如此甚至也是有意义的!),但是会得到收敛的近似解并非定论(虽然从合理的仿真实验中所期望的一样也不少!)。根据定理 3.1,数值稳定性也是必须的,这基本上取决于步长  $h$ 。若缺乏对  $h$  作用与影响的了解,作为一个可自由选择的仿真参数,  $h$  也可能被错误地任意设定(见 3.2.3 节)。

显然,为了获得高精度的近似解,步长  $h$  应当选择得尽可能小。另一方面,对于一个固定的积分区间,这会增大计算需求(更多次的递推),为了快速计算而希望取更大可能的步长。所以,在任何具体情况下,选择积分步长  $h$  时总需要在精度与计算负荷之间做出折中。

### 3.2.3 数值稳定性

#### 线性测试系统:初值问题

一个数值积分过程可表达为由一组非线性差分方程组表示的离散时间动态系统。这使得我们能采用熟知的稳定性概念和判据来分析其数值稳定性。

为了讨论,考虑如下的线性(固有稳定的)测试初值问题:

$$\dot{x} = \lambda x, \quad \text{其中 } x(0) = x_0, \lambda < 0 \quad (3.5)$$

对于欧拉法

$$\hat{x}_{k+1} = \hat{x}_k + h \cdot f(\hat{x}_k)$$

并考虑式(3.5),得线性一阶差分方程:

$$\hat{x}_{k+1} = (1 + h \cdot \lambda) \hat{x}_k \quad (3.6)$$

方程(3.6)的通解为:

$$\hat{x}_{k+1} = (1 + h \cdot \lambda)^{k+1} x_0 \quad (3.7)$$

当  $k \rightarrow \infty$  时如果式(3.7)的近似值序列  $(\hat{x}_k) = (x_0, \hat{x}_1, \hat{x}_2, \dots)$  收敛到真实解的稳态终值

$x_\infty=0$ , 则会有数值稳定性, 即数值稳定性条件为:

$$|1 + h \cdot \lambda| < 1 \quad (3.8)$$

式(3.8)的条件对应于我们所熟知的线性差分方程(3.6)的离散时间稳定性判据“特征值的幅度小于1”, 参见(Franklin et al. 1998)。

### 固有稳定系统

在一个固有稳定系统中( $\lambda < 0$ )中, 数值稳定性条件式(3.8)满足, 当且仅当:

$$h\lambda < 0 \text{ 且 } h < \frac{2}{-\lambda} = h_{crit} \quad (3.9)$$

所以容许步长的上界为  $h_{crit}$ 。

#### 例 3.1 显式欧拉积分

图 3.3 表明了积分步长  $h$  对仿真解的影响。根据条件式(3.9)可知, 当  $h \geq 2$  且  $\lambda = -1$  时数值积分是不稳定的。

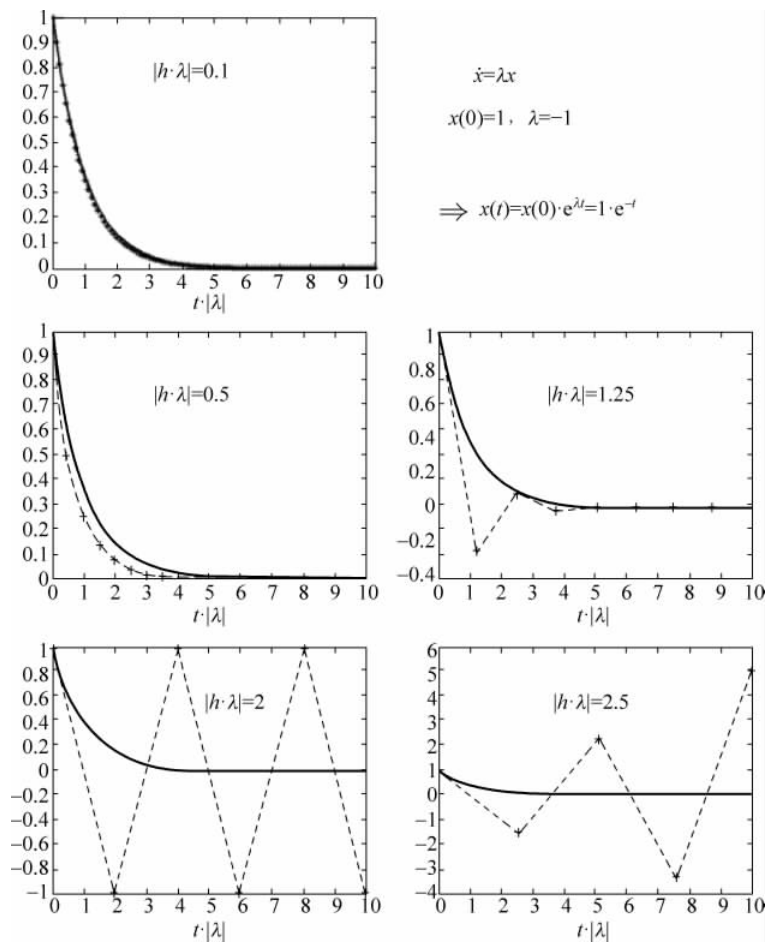


图 3.3 数值稳定性随步长的变化情况

## 绝对数值稳定性

## 定义 3.5

绝对稳定性：对于测试初值问题

$$\dot{x} = \lambda x, \quad x(t_0) = x_0 \quad (3.10)$$

应用单步法得递推关系式：

$$\hat{x}_{k+1} = \varphi_R(h\lambda) \hat{x}_k \quad (3.11)$$

则称集合  $B := \{\mu \in \mathbb{C}, |\varphi(\mu)| < 1\}, \mu = h\lambda$  为绝对稳定域。

因此，总是需要选择积分步长，使得当  $\text{Re}(\lambda) < 0$  时  $(h \cdot \lambda) \in B$  总成立。

当应用到测试线性初值问题式(3.5)时，具有  $p$  阶逼近精度的不同显式与隐式单步法的关于  $\mu = h\lambda$  的特征多项式如表 3.1 所示。

表 3.1 单步法的绝对稳定域

| 显式单步法, $p$ 阶                         | $\varphi_R(\mu), \mu = h\lambda$                                                         |
|--------------------------------------|------------------------------------------------------------------------------------------|
| $p=1$ : 前向欧拉法(EUL)                   | $1 + \mu$                                                                                |
| $p=2$ : 中点法(MIP)                     | $1 + \mu + \frac{1}{2}\mu^2$                                                             |
| $p=3$ : 3 阶龙格-库塔法(RK3)               | $1 + \mu + \frac{1}{2}\mu^2 + \frac{1}{3!}\mu^3$                                         |
| $p=4$ : 4 阶龙格-库塔法(RK4)               | $1 + \mu + \frac{1}{2}\mu^2 + \frac{1}{3!}\mu^3 + \frac{1}{4!}\mu^4$                     |
| $p=5$ : 5 阶龙格-库塔法(RK5)               | $1 + \mu + \frac{1}{2}\mu^2 + \frac{1}{3!}\mu^3 + \frac{1}{4!}\mu^4 + \frac{1}{5!}\mu^5$ |
| $p \rightarrow \infty$ : 状态转移矩阵(LIN) | $e^\mu$                                                                                  |
| 隐式单步法, $p$ 阶                         | $\varphi_R(\mu), \mu = h\lambda$                                                         |
| $p=2$ : 梯形法(TRA)                     | $\frac{1 + \frac{1}{2}\mu}{1 - \frac{1}{2}\mu}$                                          |

图 3.4 给出了龙格-库塔法在复平面上适用的稳定域(对应于复数特征值)。表 3.2 给出了实特征值的稳定区间。可以证明，递推系数  $\varphi_R(\mu)$  是指数函数泰勒展开的前  $p$  项(其中  $p$  为相应积分算法的积分精度的阶次)，当取极限  $p \rightarrow \infty$  时  $\varphi_R(\mu)$  为指数函数本身。

特别应该注意显式积分方法取极限  $p \rightarrow \infty$  时的数值稳定域(状态转移矩阵 LIN)以及隐式方法(如表 3.1 底部的梯形法 TRA)的数值稳定域。在两种情况下，绝对稳定域都是整个左半开平面。所以，对于固有稳定系统( $\lambda < 0$ )，任意  $h > 0$  都会导致稳定的积分算法(这是对于常微分方程刚性系统的一个明显优势，见 3.3 节)。

表 3.2 龙格-库塔法的实稳定区间(参见图 3.4)

| $p$ 阶            | 1           | 2           | 3            | 4            | 5            |
|------------------|-------------|-------------|--------------|--------------|--------------|
| 实 $h\lambda$ 的区间 | $[-2.0, 0]$ | $[-2.0, 0]$ | $[-2.51, 0]$ | $[-2.78, 0]$ | $[-3.21, 0]$ |

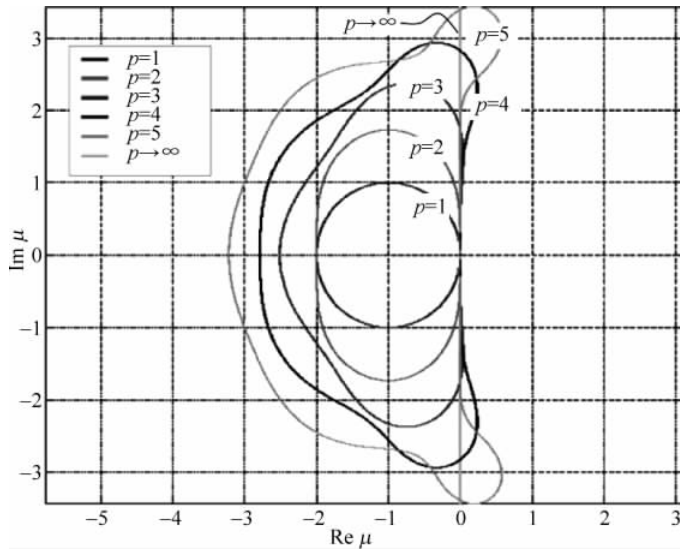


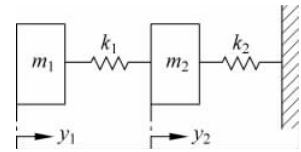
图 3.4 龙格-库塔法的稳定域(Potthoff 2003)

### 3.3 刚性系统

#### 技术背景

实际工程系统的复杂模型的一般特征是组成子系统的时间常数相差很大。这种系统被称作刚性系统。

可以用多体系统来说明这个概念。图 3.5 描绘了一个具有不同弹簧常量的双质体系统。刚度更大的弹簧( $k_2$ )导致波幅更大的谐振频率 $\omega_2$ ,对应于纯虚数特征值 $\lambda_2 = \pm j\omega_2$ 。等价地,大幅度的实数特征值(或小的时间常数)也被称作“刚性的”。

图 3.5  $k_2 \gg k_1$  情况下的刚性多体系统

#### 局部线性化：雅可比矩阵

在形如下式的非线性系统描述的情况下：

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, t)$$

通过 $\mathbf{x}(t_k)$ 附近的轨迹局部线性化可导出在固定时刻 $t_k$ (在时间区间 $[t_k, t_k + h]$ 上近似)系统动态特性的定量描述。为此,考虑雅可比矩阵 $\mathbf{J}$ :

$$\mathbf{J}(t_k) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{t=t_k, \mathbf{x}=\mathbf{x}(t_k)}$$

## 刚度

$J(t_k)$  的特征值  $\lambda_j (j=1, \dots, n)$  描述了系统在  $t_k$  时刻的动态特性。系统的刚度  $S$  由下面的商来表征:

$$S = \frac{\max(|\lambda_j|)}{\min(|\lambda_j|)}, \quad j = 1, \dots, n$$

对于刚性系统  $S \gg 1$ 。

## 线性测试系统: 不同方法的比较

对于这些比较, 考虑下列线性系统:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}, \quad \mathbf{x}(0) = \mathbf{x}_0$$

- 中点法(MIP)(阶次  $p=2$ )

$$\mathbf{k}_1 = \mathbf{A} \hat{\mathbf{x}}_i$$

$$\mathbf{k}_2 = \mathbf{A} \left( \hat{\mathbf{x}}_k + \frac{h}{2} \mathbf{A} \hat{\mathbf{x}}_k \right)$$

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + h \left( \mathbf{A} \hat{\mathbf{x}}_k + \frac{h}{2} \mathbf{A}^2 \hat{\mathbf{x}}_k \right) = \left( \mathbf{I} + h\mathbf{A} + \frac{h^2}{2} \mathbf{A}^2 \right) \hat{\mathbf{x}}_k = \Phi_{MIP} \hat{\mathbf{x}}_k$$

- 梯形法(TRA)(阶次  $p=2$ )

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + \frac{h}{2} (\mathbf{A} \hat{\mathbf{x}}_k + \mathbf{A} \hat{\mathbf{x}}_{k+1})$$

$$\hat{\mathbf{x}}_{k+1} = \left( \mathbf{I} - \frac{h}{2} \mathbf{A} \right)^{-1} \left( \mathbf{I} + \frac{h}{2} \mathbf{A} \right) \hat{\mathbf{x}}_k = \Phi_{TRA} \hat{\mathbf{x}}_k$$

对于  $\mathbf{A}$  为对角阵的特例(如果使用模态变换, 是可能的), 对于二阶系统有:

$$\mathbf{A} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}, \quad \lambda_1 \leq \lambda_2 \text{ 为系统特征值}$$

$$\Phi_{MIP} = \begin{bmatrix} 1 + h\lambda_1 + \frac{h^2}{2}\lambda_1^2 & 0 \\ 0 & 1 + h\lambda_2 + \frac{h^2}{2}\lambda_2^2 \end{bmatrix}$$

$$\Phi_{TRA} = \begin{bmatrix} \frac{1 + \frac{h}{2}\lambda_1}{1 - \frac{h}{2}\lambda_1} & 0 \\ 0 & \frac{1 + \frac{h}{2}\lambda_2}{1 - \frac{h}{2}\lambda_2} \end{bmatrix}$$

为了保证数值稳定性, 应该这样选取步长  $h$ , 使得矩阵  $\Phi_{MIP}$  和  $\Phi_{TRA}$  的所有对角线元素的模都小于 1 (参见绝对稳定性)。

所以, 对于显式 MIP 方法, 最大模特征值  $\lambda_2$  (解的“最快速”分量) 决定了步长大小。在刚性系统中, 也必须使用这个“小”步长对余下的“慢”分量 ( $\lambda_1$ ) 进行积分。显然, 这样做的代价是降低了计算速度。

这里隐式 TRA 方法的优点是明显的,因为选择较大的步长虽然增大了快分量的误差,但是一般能保证积分的绝对稳定性。

### 例 3.2

刚性二阶系统

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{\left(1 + \frac{s}{a_1}\right)\left(1 + \frac{s}{a_2}\right)} = \frac{a_1 a_2}{s^2 + (a_1 + a_2)s + a_1 a_2}$$

$$\text{状态空间模型} \quad \dot{\mathbf{x}} = \begin{pmatrix} 0 & 1 \\ -a_1 a_2 & -(a_1 + a_2) \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$$

$$y = (a_1 a_2 \quad 0) \mathbf{x}$$

特征值  $\lambda_1 = -a_1, \lambda_2 = -a_2$

阶次  $p=1$ (如 EUL)或  $p=2$ (如 MIP)时显式单步法的容许积分步长,即  $\mu = h\lambda \in [-2.0, 0]$ 。

$$a_1 = 1 \Rightarrow h_{1,\max} < 2$$

数值例子

$$a_2 = 0.01 \Rightarrow h_{2,\max} < 200$$

→最大容许步长:  $h_{\max} = \min(h_{1,\max}, h_{2,\max}) = 2!$

### 过长计算时间的避免

为了避免过长的计算时间,可采用下列方法:

- 采用绝对稳定的隐式积分方法(如 TRA)。优点:可选择任意的大步长;缺点:误差取决于步长大小。
- 采用状态转移矩阵的线性仿真(见 3.5 节)。优点:无论步长多大都能获得高精度;缺点:仅适用于线性系统,对输入时间行为的限制(分段定常的输入模型)。
- 模型修正:用比例增益或非线性传递元件来近似“刚性”分量。

### 模型修正:模型简化

我们应将上述的最后一种可能性(即模型修正)视为一种严肃的候选方法。但是,我们也应该批评性地仔细检查这种模型修正是否是真正允许的,即改进的模型是否保持了期望的系统动态特性表示水平。在很多情况下,实际上可在一个匹配频带内(在控制器带宽内)用增益元件来近似强阻尼低通元件。这种类型的单元替换被称作模型简化(图 3.6,例如快速执行器、宽带测量放大器)。

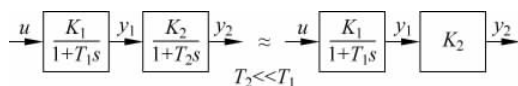


图 3.6 宽带低通元件情况下的模型修正

### 多体系统的高频特征模态

但是,在处理多体系统(MBS)的高频特征模态时需要格外小心。在很多情况下,控制器设计忽略了这些特征模态(信息漏失或溢出效应(spillover effect),参见 10.3 节)。但是,

正是仿真需要表明这种模型不确定性对完整的控制回路及其鲁棒稳定性没有不利影响。在这种意义下,在仿真中隐瞒这些高频特征模态是完全不可接受的。对于这种情况而言,采用状态转移矩阵(因为 MBS 一般是线性描述,这是可行的,例如通过有限元方法)的仿真提供了一种理想的替代解决办法。

### 3.4 弱阻尼系统

#### 多体系统的特征模态

一个经常被忽视的困难出现在弱阻尼(或极端情况下的无阻尼)振荡系统(谐振子)仿真过程中。在为了检验多体系统的鲁棒稳定性而进行的灵敏度与鲁棒性研究期间,特别要考察具有微小阻尼的高频特征模态。

#### 谐振子

图 3.7 给出了谐振子(无阻尼振荡)的两种典型情况。在该例中,绝对稳定性概念表明,对于复共轭特征值  $\lambda_2 = \pm j\omega_2$ ,应该减小步长  $h$  直至  $\mu_2 = \pm jh\omega_2$  位于绝对稳定域内。因为这对于复共轭特征值  $\lambda_1 = \pm j\omega_1$  是成立的,自然出现了这样的问题:采用  $\mu_1 = \pm jh\omega_1$  的仿真能得到正确结果吗?

这个问题的答案是否定的。谐振子的真实解为  $y(t) = y(0) \cdot \sin(\omega_1 t + \varphi_0)$ (即无阻尼谐波振荡)。然而,绝对稳定域保证使用式(3.11)的  $y(t)$  近似离散解将是一个衰减序列,因为差分方程特征值的模均小于 1。

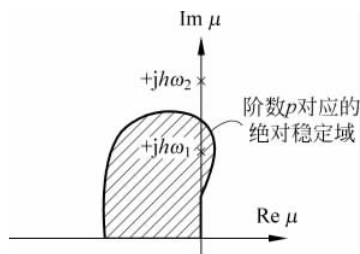


图 3.7 显式单步法下的无阻尼特征模态(只显示了上半平面)

#### 显式单步法的问题

这意味着,使用显式单步法在根本上并不能正确求解谐振子的边际稳定情况(也不能正确求解具有不稳定特征模态的一般线性模型)。由于积分过程而出现阻尼,因此不能正确复制稳态振荡特性<sup>4</sup>。只有对于阶次  $p \rightarrow \infty$  的情况(即状态转移矩阵方法 LIN),才会产生真正的稳态振荡。这再次表明了针对多体系统采用状态转移矩阵方法的优点(参见 3.5 节)。

#### 误差方程

具有有限阶次  $p$  的方法的这种不良行为可以使用误差方程容易地检验(Potthoff 2003)。将测试初值问题(3.10)的真实解  $x(t) = x(0) \cdot e^{t\lambda}$  与(3.11)的近似解  $\hat{x}_{k+1} = \varphi_R(h\lambda)$   $\hat{x}_k$  进行比较,可得全局离散化误差(GDE)方程:

$$\Delta x_{k+1} = x((k+1) \cdot \mu) - \hat{x}_{k+1} = (e^{(k+1) \cdot \mu} - \varphi_R^k(\mu)) \cdot x(0)$$

<sup>4</sup> 这对于不稳定系统(即  $\text{Re} \lambda_i > 0$  的那些系统)一般是成立的。

例如,当  $p=1$  时,误差方程为:

$$\Delta x_{k+1} = (e^{(k+1)\cdot\mu} - (1+\mu)^k) \cdot x(0)$$

在不稳定动态(如  $\text{Re } \mu > 0$ )下,得到一个发散的全局离散化误差(GDE)方程:

$$\text{当 } \mu > 0 \text{ 时, } \lim_{k \rightarrow \infty} \left[ \left( 1 + \mu + \frac{1}{2!}\mu^2 + \cdots \right)^{k+1} - (1+\mu)^k \right] = \infty$$

对于其他方法和阶次,可以用同样的方式来研究动态特性。

那么,存在哪些可能性能来正确实现这种实际重要情况的仿真呢?

### 可能的解决办法

对于线性模型,存在前面提到的状态转移矩阵方法(见 3.5 节)。如果绝对需要非线性仿真(以及一种通用的数值积分方法),只能通过限定全局离散化误差  $GDE=O(h^p)$  来解决该问题。这要求选择一个“足够小”的步长。这里“充分小”意味着  $\mu=\lambda h$  不但必须位于绝对稳定域内,而且要置于原点附近。这又意味着所需计算时间的显著增加。

#### 例 3.3 谐振子

无阻尼的质体—弹簧系统代表一个谐振频率  $\omega_0 = \sqrt{k/m}$  的谐振子。在没有外部激励的情况下,该系统可通过下列自治微分方程来描述:

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{bmatrix} \mathbf{x}, \quad \mathbf{x}(0) = (x_{10} \quad x_{20})^T$$

如果给定数值  $\omega_0=1$  与  $\mathbf{x}(0)=(1 \quad 0)^T$ ,则真实解为  $x(t)=\cos(t)$ 。

图 3.8 给出这个真实解以及不同(固定)步长  $h$  下采用 4 阶龙格-库塔法(RK4)与状态转移矩阵法(LIN)得到的近似解。容易看出, RK4 方法只在步长充分小时复现稳态振荡(以及充分小的 GDE),而 LIN 方法无论积分步长如何都非常精确地模拟了实际动态特性。对于机械结构的高频特征模态, RK4 实现所需的步长不但必须位于绝对稳定域内而且要充分小,以避免不良的阻尼效应(这将导致甚至更长的计算时间!)

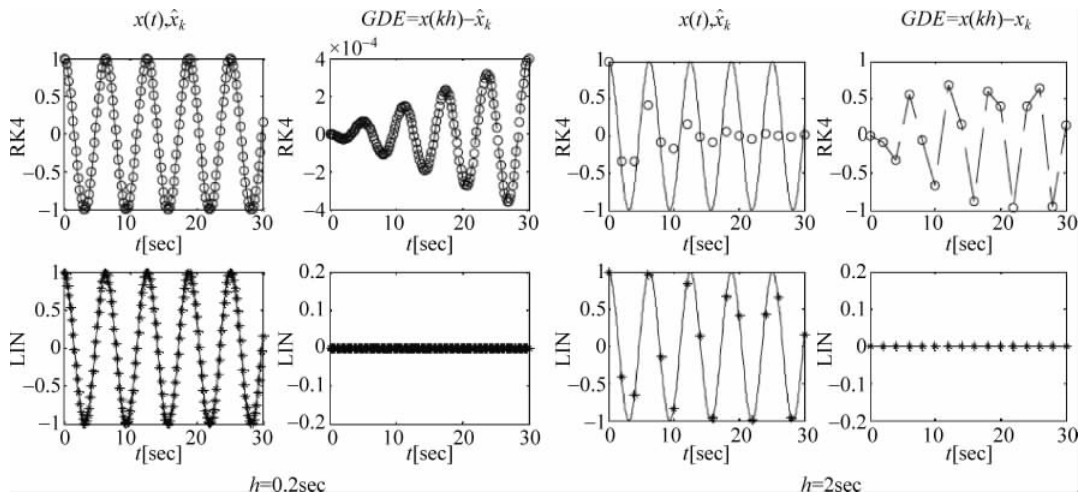


图 3.8 谐波振子的近似解 ( $\omega_0=1$ )

### 3.5 高阶线性系统

#### 多体系统

本书所介绍的与机电系统设计相关的物理现象动态模型主要是线性的。特别地,线性形式(FEM)的多体系统运动方程一般是可以得到的,但是为了高精度建模,这些模型具有较高的阶数。如前面已经多次讨论的,必须将全部特征模态(也包括那些高频且常常是弱阻尼的特征模态)结合考虑在仿真中。这一般会形成线性、弱阻尼、刚性、高阶系统。前面几节已经详细探讨了这种系统模型数值积分方法的一般局限性和困难。

#### 求解方法: 状态转移矩阵

本节将提出另一种动态系统方程数值求解的方法。该方法通过状态转移矩阵使用线性常微分方程系统的解析解。这种方法的优点是显著降低的计算负荷(计算时间)和任意高的计算精度。

在下面的讨论中,我们考虑下列线性时不变(LTI)系统:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad \mathbf{x}(t_0) = \mathbf{x}_0 \in R^n \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}\end{aligned}\quad (3.12)$$

其中  $\mathbf{x} \in R^n, \mathbf{u} \in R^m, \mathbf{y} \in R^p, \mathbf{A} \in R^{n \times n}, \mathbf{B} \in R^{n \times m}, \mathbf{C} \in R^{p \times n}, \mathbf{D} \in R^{p \times m}$ 。

#### 3.5.1 通用的数值积分方法

##### 显式单步法

如果将一种常用的单步数值积分方法用于上述的 LTI 系统,则可得以下高效算法:

- 前向欧拉法:

$$\begin{aligned}\hat{\mathbf{x}}_{k+1} &= \{\mathbf{I} + h \cdot \mathbf{A}\} \cdot \hat{\mathbf{x}}_k + h \cdot \mathbf{B} \cdot \mathbf{u}_k \\ \hat{\mathbf{x}}_{k+1} &= \tilde{\Phi}^{(1)} \cdot \hat{\mathbf{x}}_k + \tilde{\mathbf{H}}^{(1)} \cdot \mathbf{u}_k\end{aligned}\quad (3.13)$$

- 四阶龙格-库塔法:

$$\begin{aligned}\hat{\mathbf{x}}_{k+1} &= \left\{ \mathbf{I} + h\mathbf{A} + \frac{h^2}{2}\mathbf{A}^2 + \frac{h^3}{6}\mathbf{A}^3 + \frac{h^4}{24}\mathbf{A}^4 \right\} \hat{\mathbf{x}}_k + \left\{ h\mathbf{B} + \frac{h^2}{2}\mathbf{A}\mathbf{B} + \frac{h^3}{6}\mathbf{A}^2\mathbf{B} + \frac{h^4}{24}\mathbf{A}^3\mathbf{B} \right\} \mathbf{u}_k \\ \hat{\mathbf{x}}_{k+1} &= \tilde{\Phi}^{(4)} \cdot \hat{\mathbf{x}}_k + \tilde{\mathbf{H}}^{(4)} \cdot \mathbf{u}_k\end{aligned}\quad (3.14)$$

##### 时不变递推

将这些常用的数值积分方法用于线性系统后,就可将常微分方程(3.12)简化为简单的递推式(3.13)或(3.14)。值得注意的是,由于式(3.12)的时不变性,这些递推式的系数矩阵在所有时刻都是恒定的,这也意味着在实际仿真运行之前它们只需要计算一次。

### 3.5.2 通过状态转移矩阵的求解方法

#### 解析解

为了更深入地理解该问题,可以与使用状态转移矩阵 $\Phi(t)$ (也称为基本矩阵或矩阵指数)求得的常微分方程组(3.12)的如下精确解析解进行比较(见(Ogata 2010)):

$$\mathbf{x}(t) = \Phi(t - t_0) \cdot \mathbf{x}(t_0) + \int_{t_0}^t \Phi(t - \tau) \cdot \mathbf{B} \cdot \mathbf{u}(\tau) d\tau \quad (3.15)$$

将式(3.15)中的积分区间选为 $[t_k, t_k + h]$ ,可得:

$$\mathbf{x}(t_k + h) = \Phi(h) \cdot \mathbf{x}(t_k) + \int_{t_k}^{t_k + h} \Phi(t_k + h - \tau) \cdot \mathbf{B} \cdot \mathbf{u}(\tau) d\tau \quad (3.16)$$

如上式满足如下假设条件:

$$\mathbf{u}(t) = \mathbf{u}(t_k) = \text{常数} \quad \text{对 } t \in [t_k, t_k + h) \quad (3.17)$$

可得任意步长  $h$  下精确解的递推公式:

$$\mathbf{x}(t_k + h) = \Phi(h) \cdot \mathbf{x}(t_k) + \left[ \int_0^h \Phi(\tau) \cdot \mathbf{B} \cdot d\tau \right] \cdot \mathbf{u}(t_k) \quad (3.18)$$

$$\mathbf{x}(t_k + h) = \Phi(h) \cdot \mathbf{x}(t_k) + \mathbf{H}(h) \cdot \mathbf{u}(t_k) \quad (3.19)$$

其中,  $\mathbf{H}(h) := \int_0^h \Phi(\tau) \cdot \mathbf{B} \cdot d\tau$ 。

对于状态转移矩阵 $\Phi(h)$ 和离散输入矩阵 $\mathbf{H}(h)$ ,有下面的著名关系成立:

$$\Phi(h) := e^{\mathbf{A} \cdot h} = \mathbf{I} + h \cdot \mathbf{A} + \frac{h^2}{2!} \cdot \mathbf{A}^2 + \frac{h^3}{3!} \cdot \mathbf{A}^3 + \dots \quad (3.20)$$

$$\mathbf{H}(h) = \left\{ h \cdot \mathbf{I} + \frac{h^2}{2!} \cdot \mathbf{A} + \frac{h^3}{3!} \cdot \mathbf{A}^2 + \frac{h^4}{4!} \cdot \mathbf{A}^3 + \dots \right\} \cdot \mathbf{B} \quad (3.21)$$

#### 对通用数值积分方法的价值

##### 近似状态转移矩阵

将式(3.20)、式(3.21)与式(3.13)、式(3.14)相比较可见,此处提出的数值积分方法使用了对真实解的近似。也就是说,在这些方法中,用(很少)数量的求和项来近似状态转移矩阵 $\Phi(h)$ 和离散输入矩阵 $\mathbf{H}(h)$ 。

这显然与方法的阶次  $p$  直接有关,例如, RK4 方法的阶数  $p=4$ , 局部离散化误差 LDE =  $O(h^5)$ , 这意味着 $\tilde{\Phi}^{(4)}$ 将精确的状态转移矩阵 $\Phi(h)$ 近似到 4 阶  $h^4$ 。隐式方法(如梯形法)则通过 Pade 逼近来近似 $\Phi(h)$ (见表 3.1)。

##### 单步法的递推通式

于是,线性时不变系统单步积分法的递推通式可以简单地表示为:

$$\begin{aligned} \hat{\mathbf{x}}(t_k + h) &= \tilde{\Phi}(h) \cdot \hat{\mathbf{x}}(t_k) + \tilde{\mathbf{H}}(h) \cdot \mathbf{u}(t_k) \\ \hat{\mathbf{y}}(t_k) &= \mathbf{C} \cdot \hat{\mathbf{x}}(t_k) + \mathbf{D} \cdot \mathbf{u}(t_k) \end{aligned} \quad (3.22)$$

### 3.5.3 仿真解的精度

#### 影响仿真精度的因素

状态转移矩阵仿真解  $\hat{\mathbf{x}}(t_k)$  的精度在根本上只取决于以下两个因素:

- $\tilde{\Phi}(h)$ 、 $\tilde{\mathbf{H}}(h)$  的近似精度。
- 假设条件式(3.17)的正确性(是否成立),即输入  $\mathbf{u}(t)$  在积分区间  $[t_k, t_k + h]$  内是否是恒定的。

因此,积分步长  $h$  的选择不再像通用数值积分方法情况下对仿真精度有根本的显著影响。

#### $\tilde{\Phi}$ 、 $\tilde{\mathbf{H}}$ 的离线计算

根据式(3.20)和式(3.21)确定足够精确的近似矩阵  $\tilde{\Phi}$ 、 $\tilde{\mathbf{H}}$  完全与实际仿真算法式(3.22)相分离。因此,即使为了高精度积分,仿真的计算负荷仅取决于系统阶次(即系统矩阵  $\mathbf{A}$ 、 $\mathbf{B}$ 、 $\mathbf{C}$ 、 $\mathbf{D}$  或  $\tilde{\Phi}$ 、 $\tilde{\mathbf{H}}$  的阶数)。

$\tilde{\Phi}$ 、 $\tilde{\mathbf{H}}$  的计算可以使用合适的数值稳定的算法离线进行。

然而,在时变系统(如变结构系统),系统参数矩阵  $\mathbf{A}$ 、 $\mathbf{B}$  发生改变就必须重新计算  $\tilde{\Phi}$ 、 $\tilde{\mathbf{H}}$ 。

#### 恒定输入

影响状态转移矩阵方法精度的第二个根本因素是积分区间内输入的恒定性。一般情况下,这个条件当然不可能精确满足。在这种情况下,如果给定输入变化率和总体所需精度,步长  $h$  应选得足够小。

但是,在以下几种结构中,式(3.17)的要求得到精确满足,从而可以获得任意精确的解:

(1)  $u_i(t)$  是阶跃输入(在  $t=0^+$  时刻后恒定)(见图 3.9),即  $u_i(t) = c_i \cdot \sigma(t)$ ,  $i=1, \dots, m$ 。此处对积分步长  $h$  没有限制。

(2)  $u_i(t)$  为(分段恒定的)的阶梯形函数,即  $u_i(t) = c_i$ ,  $t \in [t_k, t_k + h]$ ,  $i=1, \dots, m$ 。

例如,采样数据系统满足这个条件,该系统中  $u_i(t)$  为数字控制器的输出(见图 3.10 中零阶保持器是离散时间子系统与连续时间子系统之间的接口)。在这种情况下,可以将相应的系统采样周期用作最大积分步长。如果有若干个不同的采样时间  $T_{a,j}$ (例如级联控制器),应该这样选择积分步长使得:

$$h_{\max} = \text{最大公因子}(T_{a,j}), j = 1, 2, \dots$$

从计算精度的角度来看,采用较小的积分步长也可以精确地模拟(仿真)采样时刻之间的系统行为。

(3)  $u_i(t)$  的模型是线性微分方程的解(信号发生器,信号模型)。此时,使用这些信号发

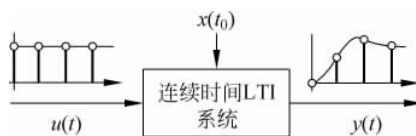


图 3.9 恒定输入

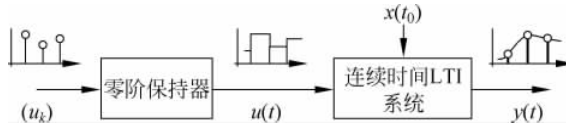


图 3.10 采样控制器产生的阶梯形输入

生器将常微分方程系统进行扩展,可通过选择合适的初值将信号随时间的变化参数化,并根据式(3.22)对该扩展系统进行仿真(见图 3.11)。

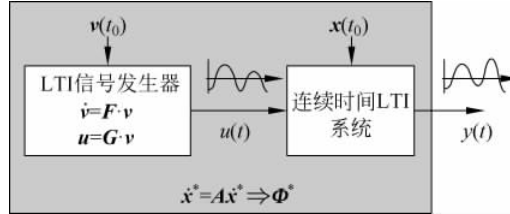


图 3.11 带有信号发生器的扩展系统

通用 LTI 信号发生器:

$$\begin{aligned}\dot{v} &= F \cdot v \\ u &= G \cdot v\end{aligned}$$

扩展系统:

$$\begin{aligned}x^* &:= (x \quad v)^T, \\ \dot{x}^* &= \underbrace{\begin{pmatrix} A & B \cdot G \\ 0 & F \end{pmatrix}}_{A^*} x^*, \quad x^*(0) = \begin{pmatrix} x_0 \\ v_0 \end{pmatrix}\end{aligned}$$

此时,仿真中必须使用状态转移矩阵 $\Phi^*(t) = e^{A^* t}$ 。注意到,此时仅需计算系统的齐次解,这里也没有对积分步长 $h$ 的限制。然而,输入 $u$ 的结构性变化绝对要求重新计算 $\Phi^*(h)$ 。

#### 例 3.4 谐波激励的信号发生器

$$u(t) = U_0 \cdot \sin(\omega_0 t + \phi), \quad t \geq 0$$

LTI 信号模型:

$$\begin{aligned}v_1(t) &:= u(t) = U_0 \cdot \sin(\omega_0 t + \phi), \\ v_2(t) &:= \dot{v}_1(t) = U_0 \cdot \omega_0 \cdot \cos(\omega_0 t + \phi), \\ \dot{v}_2(t) &:= -U_0 \cdot \omega_0^2 \cdot \sin(\omega_0 t + \phi) = -\omega_0^2 \cdot v_1(t)\end{aligned}$$

可推出:

$$\begin{aligned}v &:= (v^1 \quad v^2)^T \\ \dot{v} &= \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{pmatrix} v, \quad v(0) = \begin{pmatrix} U_0 \sin \phi \\ U_0 \cdot \omega_0 \cdot \sin \phi \end{pmatrix} \\ u &= (1 \quad 0) v\end{aligned}$$

## 3.6 DAE 系统的数值积分

### 参考模型

为了便于表述和理解,本节考虑如下只包含一个状态变量和一个代数变量的标量系统:

$$\dot{x} = f(x, z) \quad (3.23)$$

$$0 = g_1(x, z) \quad (3.24)$$

$$\text{或 } 0 = g_2(x) \quad (3.25)$$

向量取值系统也可以类似地处理。

在数值积分的每一步,对应于微分方程(3.23)的差分方程和代数方程(3.24)、(3.25)必须同时得到满足。

### DAE 直接求解

本节将研究 DAE 系统直接数值求解的若干基本方法。直接求解的意思是,无需对模型进行额外修正,直接对原始的 DAE 系统(具有一定的微分指标,见 2.4.1 节)进行积分。

确实,一种做法(2.4.3 节中介绍过)是使用指标约减将 DAE 系统转化为一组常微分方程。这种方法存在的问题已经在之前章节中详细讨论过。

关于该方法的扩展,读者可参考专著(Brenan et al, 1996)、(Cellier、Kofman, 2006)和针对具体问题的论文(Otter, 1999)、(Otter、Bachmann, 1999)、(Otter、Bachmann, 1999)。

### 3.6.1 显式积分法

#### 显式解

按照定义,显式积分法的特征是微分方程的近似解可使用一组递推方程来计算,即可以用恰当的形式顺序地求解这些方程(意味着较轻的计算负担)。

DAE 系统显式积分方法的基本问题已经可以通过简单的显式欧拉法来清晰说明。

#### 指标-1 系统

根据定义,  $\partial g / \partial z \neq 0$ , 即  $g := g_1(x, z)$ 。

给定(任意)初值  $x_0$ , 欧拉递推为:

$$k = 0: 0 = g_1(x_0, \hat{z}_0) \Rightarrow \hat{z}_0 \quad (3.26)$$

$\Rightarrow$  通过已知  $x_0$  的代数方程(3.24)求得合适的  $\hat{z}_0$ 。

$$k = 1: \begin{cases} \hat{x}_1 = x_0 + f(x_0, \hat{z}_0) \Rightarrow \hat{x}_1 \\ 0 = g_1(\hat{x}_1, \hat{z}_1) \Rightarrow \hat{z}_1 \end{cases} \quad (3.27)$$

$$(3.28)$$

$\Rightarrow$  通过已知  $x_0, \hat{z}_0$  的微分方程(3.27)求得  $\hat{x}_1$ ,

$\Rightarrow$  通过已知  $\hat{x}_1$  的代数方程(3.28)求得  $\hat{z}_1$ 。

使用显式积分方法,通过序贯地求解差分方程( $\rightarrow \hat{x}_{k+1}$ )和代数方程( $\rightarrow \hat{z}_{k+1}$ ),便可以求解指标-1 系统。

额外任务:根据式(3.26)确定具有一致性的初值  $x_0$  和  $\hat{z}_0$ 。

显示积分方法可照此使用。但是,在每个积分步,必须用一个代数方程求解对其进行补充。

### 指标-2 系统

根据定义,  $\partial g / \partial z \neq 0$ , 即  $g := g_2(x)$ 。

给定合适的初值  $x_0$ , 欧拉递推为:

$$k = 0; 0 = g_2(x_0) \Rightarrow x_0 \quad (3.29)$$

$\Rightarrow x_0$  不能自由选择,而必须满足代数方程(3.25)!

$$k = 1: \left\{ \begin{array}{l} \hat{x}_1 = x_0 + f(x_0, z_0) \\ 0 = g_2(\hat{x}_1) \end{array} \right\} \Rightarrow \hat{x}_1, z_0 \quad (3.30)$$

$$(3.31)$$

$\Rightarrow \hat{x}_1, z_0$  不是顺序可解的。式(3.30)与式(3.31)是一种隐含的非线性代数方程组,其解需要隐式解法,如牛顿-拉夫森法(Faires, Burden, 2002)。

指标大于或等于 2 的 DAE 系统原则上不能通过显式积分方法求解。

需要一种隐式方程解法来同时求解差分方程与代数方程。

## 3.6.2 隐式积分法

### 隐式解

正如前面所讨论的,隐式数值积分方法必须含有一个隐式方程求解法。即使对于较大的积分步长也能保证数值积分的绝对数值稳定性(这一点对于刚性系统是很重要的),因而相应增大的计算负担是可接受的。所以,采用 DAE 系统数值求解的隐式积分方法来同时求解差分方程与代数约束条件似乎是方便的。然而,下面我们将看到该方法仅适用于特定类型的 DAE 系统。

### 梯形法求解 DAE 系统

#### 隐式递推

对于式(3.23)的 DAE 系统和式(3.24)及式(3.25)的代数方程  $g=0$ , 通过积分步长为  $h$  的数值积分梯形算法<sup>5</sup> TRA(Faires, Burden, 2002)可得:

$$\begin{aligned} \hat{x}_{k+1} &= \hat{x}_k + \frac{h}{2} [f(\hat{x}_k, \hat{z}_k) + f(\hat{x}_{k+1}, \hat{z}_{k+1})] \\ 0 &= g(\hat{x}_{k+1}, \hat{z}_{k+1}) \end{aligned} \quad (3.32)$$

<sup>5</sup> 由于其简单性,此处选取梯形方法作为一个例子。其他隐式单步法有类似的特性。

### 牛顿-拉夫森迭代

整理式(3.32)可得用以确定新的未知近似值 $\hat{x}_{k+1}, \hat{z}_{k+1}$ 的非线性代数方程组:

$$\boldsymbol{\varphi}_{k+1}(\mathbf{p}_{k+1}) := \begin{bmatrix} \varphi_{1,k+1} \\ \varphi_{1,k+1} \end{bmatrix} = \begin{bmatrix} \hat{x}_{k+1} - \hat{x}_k - \frac{h}{2}[f(\hat{x}_k, \hat{z}_k) + f(\hat{x}_{k+1}, \hat{z}_{k+1})] \\ g(\hat{x}_{k+1}, \hat{z}_{k+1}) \end{bmatrix} \quad (3.33)$$

其中,  $\mathbf{p}_{k+1} := (\hat{x}_{k+1} \quad \hat{z}_{k+1})$

$$\Rightarrow \varphi_{k+1}(\mathbf{p}_{k+1}) = 0 \quad (3.34)$$

非线性方程组(3.34)的解可使用著名的牛顿-拉夫森方法得到,从而得非线性递推规则:

$$\mathbf{p}_{k+1,i+1} = \mathbf{p}_{k+1,i} - \mathbf{J}(\mathbf{p}_{k+1,i})^{-1} \cdot \boldsymbol{\varphi}_{k+1}(\mathbf{p}_{k+1,i}) \quad (3.35)$$

当  $i$  满足  $\|\mathbf{p}_{k+1,i+1} - \mathbf{p}_{k+1,i}\| \leq \varepsilon$  时,迭代终止。

### 数值收敛性

下面的雅可比矩阵对式(3.35)牛顿-拉夫森迭代的收敛性起到核心作用<sup>6</sup>:

$$\mathbf{J}(\mathbf{p}) = \begin{bmatrix} \frac{\partial \varphi_1}{\partial x_{i+1}} & \frac{\partial \varphi_1}{\partial x_{i+1}} \\ \frac{\partial \varphi_2}{\partial x_{i+1}} & \frac{\partial \varphi_2}{\partial z_{i+1}} \end{bmatrix} = \begin{bmatrix} 1 - \frac{h}{2} \frac{\partial f}{\partial x} & -\frac{h}{2} \frac{\partial f}{\partial z} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial z} \end{bmatrix} \quad (3.36)$$

根据式(3.35),牛顿-拉夫森迭代在积分步 $(k+1)$ 的数值收敛性要求雅可比矩阵(3.36)的逆必须存在,即下式必须成立:

$$\det \mathbf{J}(\mathbf{p}) = \left(1 - \frac{h}{2} \frac{\partial f}{\partial x}\right) \frac{\partial g}{\partial z} + \frac{h}{2} \frac{\partial f}{\partial z} \frac{\partial g}{\partial x} \neq 0 \quad (3.37)$$

下面将在不同 DAE 指标和任意小积分步长  $h \rightarrow 0$  情况(是高数值精度所必需的)下检验基本条件(3.37)。

### 指标-1 系统

根据定义,满足指标-1 条件,因此 $\partial g_1 / \partial z \neq 0$ (参见式(2.46))而且:

$$\lim_{h \rightarrow 0} (\det \mathbf{J}) = \frac{\partial g_1}{\partial z} \neq 0 \quad (3.38)$$

指标-1 系统一般可使用隐式方法(TRA)很好地求解,且与积分步长的选取无关。

### 指标-2 系统

根据定义,不满足指标-1 条件,于是 $\partial g_2 / \partial z = 0$ (见式(2.48))而且:

$$\det \mathbf{J} = \frac{h}{2} \frac{\partial f}{\partial z} \frac{\partial g_2}{\partial x} \neq 0. \quad \text{当 } h \neq 0 \text{ 时} \quad (3.39)$$

然而,注意到当积分步长非常小时,有:

<sup>6</sup> 在仿真工具中,雅可比矩阵既可以数值计算,也可以使用一个现成的解析函数来计算。

$$\det \mathbf{J} \approx 0, \lim_{h \rightarrow 0} (\det \mathbf{J}) = 0. \quad (3.40)$$

只有当积分步长足够大时,指标-2 系统才能使用隐式方法(TRA)很好地求解。

当积分步长很小时,雅可比矩阵变为病态的(或非良定的)( $\det \mathbf{J} \approx 0$ ),在牛顿-拉夫森迭代中不再能精确求逆。

### 指标-3 系统

根据定义,不满足指标-2 条件式(2.48),因此,可得与积分步长无关的下式:

$$\det \mathbf{J} = \frac{h}{2} \frac{\partial f}{\partial z} \frac{\partial g_2}{\partial x} = 0$$

使用隐式方法(TRA)基本上不能求解指标-3 系统。

### DAE 系统积分算法的小结:

当指标 $\leq 2$ 时:总可以使用隐式方法。对于小步长,可能需要对代数变量进行量化(参见 3.6.3 节)。

当指标 $= 3$ 时:通过约减掉 1 个指标,即微分约束(Pfaffian 形式),转化为指标-2 系统,进而按照上述步骤进行。

### 3.6.3 指标-2 系统的量化

#### 避免雅可比矩阵的奇异性

在积分步长很小情况下指标-2 系统雅可比矩阵的病态可通过对代数变量的合适量化得到缓解,量化方案如下:

#### 解决方法

在固定时标  $k$ (这里的记法中略去)式(3.35)牛顿-拉夫森迭代可以重写为:

$$\mathbf{J}(\mathbf{p}) \cdot (\mathbf{p}_{i+1} - \mathbf{p}_i) = \boldsymbol{\varphi}(\mathbf{p}_i)$$

从而,对如(3.23)和(3.25)所示的指标-2 系统可得:

$$\tilde{\mathbf{J}} \cdot \tilde{\mathbf{p}} = \begin{pmatrix} 1 - \frac{h}{2} \frac{\partial f}{\partial x} & -\frac{h}{2} \frac{\partial f}{\partial z} \\ \frac{\partial g_2}{\partial x} & 0 \end{pmatrix} \begin{pmatrix} p_{1,i+1} - p_{1,i} \\ h \cdot (p_{2,i+1} - p_{2,i}) \end{pmatrix} \quad (3.41)$$

此处,将雅可比矩阵右上角元素乘以积分步长  $h$  等价于将参数向量第二行乘以  $h$ 。因此,代数变量  $z$  被量化了  $h$  倍。这种技巧使得修改后的雅可比矩阵  $\tilde{\mathbf{J}}$  的行列式与积分步长  $h$  无关,即:

$$\Rightarrow \det \tilde{\mathbf{J}} = \frac{\partial f}{\partial z} \frac{\partial g}{\partial x}$$

因此,即使积分步长很小,牛顿-拉夫森迭代也是良定的。

### 3.6.4 具有一致性的初值

#### 条件

为了满足代数约束(3.24),必须选择一致性初值  $x(0)$  和  $z(0)$ 。此时,应该区分如下两种情况:

- 指标为 1 的 DAE 系统

$\partial g/\partial z$  是非奇异的(即  $g=g_1(x,z)$ ),从而可自由选择  $x(0)$ ,并且可通过求解代数约束  $g_1(x(0),z(0))=0$  来确定一致性初值  $z(0)$ 。

- 指标大于或等于 2 的 DAE 系统

$\partial g/\partial z$  是奇异的(即  $g=g_2(x)$ ),因此,由于  $z(0)$  不再只由代数约束确定, $x(0)$  不再能自由选取。因此,需要从微分方程导出额外的约束(关于  $x,z$  的代数方程)(见 3.6.4 节)。

#### 不一致性带来的问题

不一致性初值轻则导致仿真开始的误差,重则导致完全错误的解。一些仿真工具会自动对一致性初值进行检验。但一般情况下,对于复杂问题,一致性初值的确定是一个重要但困难的任务。因此,关于这方面,总要细致监测计算机辅助的解。

#### 漂离(drift-off)现象

即使选择了一致性初值,由于数值积分期间的数字取整(舍入)误差,代数约束一般并不能精确满足(不断增大的解发散引起的问题)。

DAE 系统数值积分存在着特殊的镇定方法。一种基本方法称为 Baumgarte 镇定(Baumgarte 1972)(关于其实际实现,参见(Ascher et al,1994))采用原始代数约束与求微分后(根据指标约减)代数约束的一种线性组合。其他可能的做法是在每个积分步将近似答案投影到在代数约束流形上,因此这些方法被称作投影法(例如,Eich,1993)。这些方法之间的比较可见(Burgermeister et al,2006)等文献。

## 3.7 混杂现象仿真的实现方法

#### 不连续性

对不连续性(切换、状态不连续性)的识别与处理是仿真实实现的一个最大挑战。基于局部离散化误差的估计使用自适应步长选取就可以处理简单的不连续性(例如,龙格-库塔-费尔伯格(Runge-Kutta-Fehlberg)方法(Faires,Burden,2002))。然而,若不准确了解实际实现的步长算法,总应该抱着很大的怀疑态度来评价这种类型仿真的结果。混杂现象的最精确仿真需要采用专门的预防措施和做法,例如,(Otter et al,1999);(Otter et al,1999)。

### 3.7.1 不连续性的处理

#### 基本问题

基本问题如图 3.12 所示。在第  $k+1$  个积分步的开始(即  $t_k$  时刻),向量场  $f^1(x, u, t)$  (常微分方程的右端)是激活的。若采用步长  $h > h_s$  执行积分步骤,结果会是仅使用  $f^1(x, u, t)$  计算得到的近似值  $\hat{x}^1(t_k + h)$ 。但是,实际上,当  $t > t_s$  时,应该在积分中考虑到向量场  $f^2(x, u, t)$ ,从而得到正确答案  $\hat{x}^2(t_k + h)$ 。一般情况下  $\hat{x}^1 \neq \hat{x}^2$  (亦可参见图 2.62)。

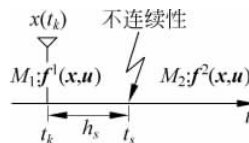


图 3.12 依赖于事件的步长选取

#### 原理性解决方法

为了避免这类错误的仿真结果,推荐采用依赖于事件的步长选取的原理性方法,其算法如下<sup>7</sup>:

- 依赖于事件的步长选择算法

|     |                                                                                                                                                                                           |
|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | 识别出下一个积分区间中的不连续性                                                                                                                                                                          |
| (2) | 确定 $t_s$                                                                                                                                                                                  |
| (3) | 匹配步长: $h_s = t_s - t_k$                                                                                                                                                                   |
| (4) | 积分步 $k+1$ : $\hat{x}(t_k) \xrightarrow{h_s} \hat{x}(t_s - 0)$                                                                                                                             |
| (5) | 评估在不连续时刻 $t_s$ 的不连续性条件:<br>(a) 不连续输入: $u(t_s + 0) = \text{new\_value}$ ,<br>(b) 连续状态: $\hat{x}(t_s + 0) = \hat{x}(t_s - 0)$ (如输入跳变),<br>(c) 不连续状态: $\hat{x}(t_s + 0) = \text{new\_value}$ |
| (6) | 下一个积分步: $\hat{x}(t_s + 0) \xrightarrow{h_{\text{NEW}}} \hat{x}(t_s + h_{\text{NEW}})$ ,<br>其中 $h_{\text{NEW}}$ 根据给定的要求和约束(局部离散化误差、数值稳定性等)来确定                                              |

### 3.7.2 事件检测

#### 事件的时间

在上面给出的事件依赖型步长选择算法中,步骤(1)与步骤(2)提出了识别事件并确定事件的时间  $t_s$  的实际挑战。

切换与状态不连续性等混杂现象一般可使用如下形式的不连续性超曲面及一个阈值检验来建模:

$$m(x, u, t) = 0 \quad (3.42)$$

<sup>7</sup> “良好”的仿真工具一般实现了这种算法,但是在任何具体情况下,都应该对实现的质量进行检验。

对于网络状态模型(见 2.5.3 节),可以用式(3.42)简洁描述连续/离散接口(C/D 接口,状态集及其边界曲面)。于是,可以使用监测函数的变号将离散事件信号  $v_D(t)$  描述成二值信号:

$$z(t_s) = m(\mathbf{x}(t_s), \mathbf{u}(t_s), t_s) \stackrel{!}{=} 0 \quad (3.43)$$

例如,

$$v_D(t) = \begin{cases} 0, & |z(t)| \leq \epsilon \\ 1, & |z(t)| > \epsilon \end{cases} \quad (3.44)$$

其中  $\epsilon$  表示一个适当小的阈值。

通过图 3.13 的二分(bisection)法和下列关系式就可以迭代地确定事件的时间  $t_s$ :

$$\hat{h}_s = h \frac{|z^+|}{|z^+| + |z^-|}, \quad \text{sign } z^+ \neq \text{sign } z^-$$

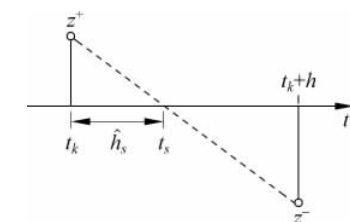


图 3.13 使用监测函数的事件检测

### 在仿真工具中的实现

在好的仿真工具中这类事件检测作为分立的功能模块(如 SIMULINK: hit crossing)或离散事件模块的子功能(如 SIMULINK/STATEFLOW: edge-triggered signal inputs)而存在。在所有情况下,积分算法或其步长选择总是可以访问的<sup>8</sup>。

## 3.8 仿真实例:理想单摆

### 系统结构与问题描述

考虑图 3.14 所示的理想单摆这一机械 DAE 系统的代表性例子,初看起来其动力学特性非常简单。通过这个例子,本节将说明这种 DAE 系统建模与仿真的最重要步骤,将提出基于信号的方法和基于方程的方法。

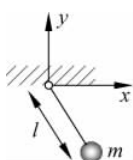


图 3.14 理想单摆

对于图 3.14 所示的理想单摆,需要找到:

- 包括连杆约束力的运动方程(一个 DAE 系统)。
- 基于信号的仿真模型(面向模块,如 MATLAB/SIMULINK)。
- 基于方程的仿真模型(面向对象,如 MODELICA)。

#### 解法 1: 运动方程为 DAE 系统

使用广义坐标  $\mathbf{q} = (x \ y)^T$  和广义速度  $\dot{\mathbf{q}} = \mathbf{v} = (v_x \ v_y)^T$ , 可得系统的共动能(kinetic co-energy)为  $T^* = m/2 \cdot (v_x^2 + v_y^2)$ 。完整约束条件形为  $f(\mathbf{q}) = x^2 + y^2 - l^2 = 0$ 。作用于该系统的唯一外力是重力  $\mathbf{F} = (0 \ -mg)^T$ 。

第一类欧拉-拉格朗日方程的一般形式为(参见 2.3.2 节,这里作了简化):

<sup>8</sup> 再次指出,必须对任何具体实现的正常运行进行细致测试。仿真工具的用户手册中所承诺的性能实际上并不总能交付提供,例如,读者可以参考文献(Buss 2002)中 SIMULINK /STATEFLOW 事件检测模块的性能,6.3 节。

$$\frac{d}{dt} \frac{\partial T^*}{\partial \dot{\mathbf{q}}} - \frac{\partial T^*}{\partial \mathbf{q}} = F + \lambda \cdot \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix}, \quad (3.45)$$

$$f(\mathbf{q}) = 0,$$

其中  $\lambda$  为对单摆连杆的约束力进行建模的拉格朗日乘子(仍为完整约束)。

求式(3.45)的值可得:

$$\begin{pmatrix} m & 0 \\ 0 & m \end{pmatrix} \cdot \dot{\mathbf{v}} = \begin{pmatrix} 0 \\ -mg \end{pmatrix} + \lambda \begin{pmatrix} 2x \\ 2y \end{pmatrix}$$

$$x^2 + y^2 - l^2 = 0$$

将  $\lambda = -\tilde{\lambda} \cdot m/2$  代入,可得半显式 DAE 系统( $u(t)=g=\text{常数}$ ):

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \tilde{\lambda}, u) \quad \text{或} \quad \dot{\mathbf{x}} = \mathbf{v}_x \\ \dot{v}_x &= -\tilde{\lambda} x \end{aligned} \quad (3.46)$$

$$\dot{\mathbf{y}} = \mathbf{v}_y$$

$$\dot{v}_y = -\tilde{\lambda} y - g$$

$$0 = g(x) \quad \text{或} \quad x^2 + y^2 - l^2 = 0 \quad (3.47)$$

### 解法 2: 指标约减与一致性初值

包含质量  $m$  与约束元件间刚性耦合的机械结构看起来是一个指标-3 系统,通过指标检验容易证实这一猜测(见 2.4.2 节)。为了确定面向模块的仿真的一致性初值,下面按照 2.4.3 节进行系统性的指标约减。

根据原始 3-指标系统的代数条件,可得第一个初值方程:

$$x^2 + y^2 - l^2 = 0 \quad (3.48)$$

通过对式(3.48)求导可获得第一次指标约减:

$$0 = \frac{d}{dt}(x^2 + y^2 - l^2) = 2x \dot{x} + 2y \dot{y}$$

将式(3.46)代入可得 2-指标系统的代数条件和第二个初值方程:

$$0 = xv_x + yv_y \quad (3.49)$$

对(3.49)求导可获得第二次指标约减:

$$0 = \frac{d}{dt}(xv_x + yv_y) = x \dot{v}_x + \dot{x}v_x + \dot{y}v_y + y \dot{v}_y$$

再次将式(3.46)代入上式,可得 1-指标系统的代数条件和第三个初值方程:

$$0 = v_x^2 + v_y^2 - gy - \tilde{\lambda} l^2 \quad (3.50)$$

然后对式(3.50)求导可获得第三次指标约减:

$$0 = \frac{d}{dt}(v_x^2 + v_y^2 - gy - \tilde{\lambda} l^2) = 2v_x \dot{v}_x + 2v_y \dot{v}_y - g \dot{y} - \dot{\tilde{\lambda}} l^2$$

最后再次将式(3.46)代入上式,得所期望的关于代数变量  $\tilde{\lambda}$  的微分方程(等价于引入一个新状态变量  $\tilde{\lambda}$ ):

$$\dot{\tilde{\lambda}} = -\frac{2\tilde{\lambda}xv_x + 2\tilde{\lambda}yv_y + 3gv_y}{l^2} \quad (3.51)$$

观察式(3.46)和(3.51)发现,所得结果是指标为 0 的 DAE 系统(即一组一阶常微分

方程)。

通过非线性方程组(3.48)至(3.50)可确定一致性初值  $x(0), v_x(0), y(0), v_y(0)$  和  $\tilde{\lambda}(0)$ 。

### 解法 3: 基于信号的仿真模型

微分方程(3.46)和(3.51)可直接在一个基于信号的仿真模型中实现。图 3.15 给出了使用 MATLAB/SIMULINK 的实现例子。

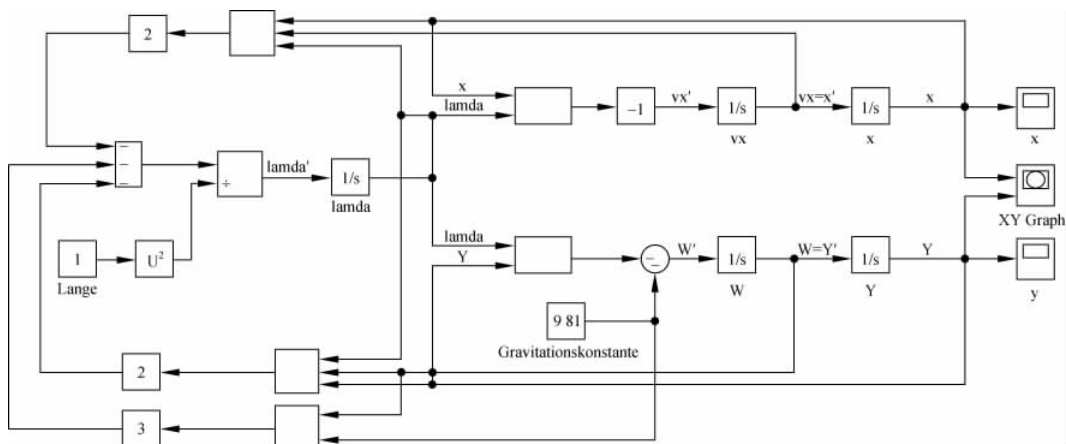


图 3.15 理想单摆的 SIMULINK 模型

### 解法 4: 仿真实验与基于信号的模型

#### 初始的水平位置

若连杆长度  $l=1$ , 连杆的水平初始位置  $y_0=0$ , 根据方程式(3.48)~式(3.50)得一致性初值:

$$\mathbf{x}_0 = (x_0 \quad v_{x0} \quad y_0 \quad v_{y0} \quad \tilde{\lambda}_0)^T = (1 \quad 0 \quad 0 \quad 0 \quad 0)^T$$

图 3.16 所示为时间信号的结果。

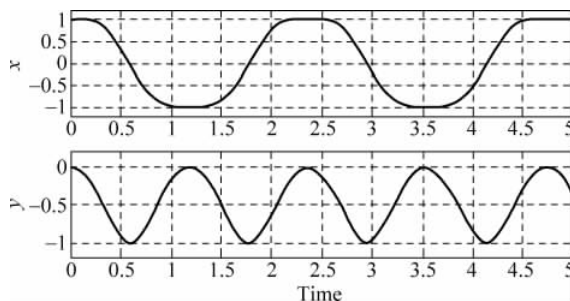


图 3.16 理想单摆: 基于信号的仿真, 初始的水平位置

#### 45°初始位置

当单摆初始位置  $\varphi_0=45^\circ$  时, 所得的一致性初值为:

$$\mathbf{x}_0 = (x_0 \quad v_{x0} \quad y_0 \quad v_{y0} \quad \tilde{\lambda}_0)^T = \left( \frac{1}{\sqrt{2}} \quad 0 \quad -\frac{1}{\sqrt{2}} \quad 0 \quad 6.9367 \right)^T \quad (3.52)$$

容易验证, 下列非一致性初值

$$\mathbf{x}_0 = (x_0 \quad v_{x0} \quad y_0 \quad v_{y0} \quad \tilde{\lambda}_0)^T = \left( \frac{1}{\sqrt{2}} \quad 0 \quad -\frac{1}{\sqrt{2}} \quad 0 \quad 10 \right)^T \quad (3.53)$$

不满足方程式(3.48)~式(3.50)。

图 3.17 绘出了这两组不同初值下随时间的变化情况。与图 3.17(a)中正确的解轨迹相比,容易发现图 3.17(b)中非一致初值条件下错误的解轨迹<sup>9</sup>。

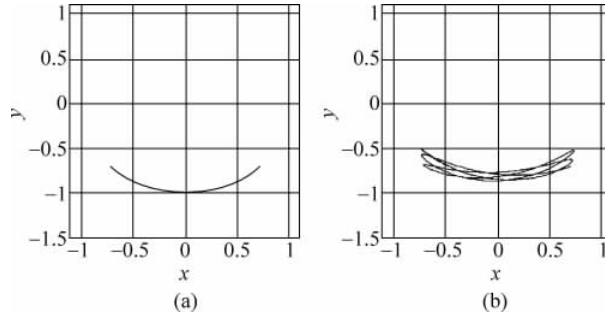


图 3.17 理想单摆: 基于信号的仿真, 初始位置  $\varphi_0 = 45^\circ$

(a) 一致的初值; (b) 不一致的初值

#### 解法 5: 基于方程的仿真模型

DAE 系统式(3.46)和式(3.47)可直接(无需进一步的变换)在基于方程的仿真工具中实现。图 3.18 给出了基于 MODELICA 的这种实现例子。大多数基于 MODELICA 的仿真工具也能够自动生成一致性初值。精确实现的仿真结果与图 3.16 和图 3.17(a)中给出的结果一致。

```

model pendulum
parameter Real l = 1;           /* pendulum length */
constant Real g = 9.81;         /* acceleration of gravity */

Real x;                         /* x coordinate */
Real y;                         /* y coordinate */
Real lambda;                   /* Lagrange multiplier */
Real vx;                       /* x component of velocity */
Real vy;                       /* y component of velocity */

equation
  der(x) = vx;
  der(vx) = -x * lambda;
  der(y) = vy;
  der(vy) = -g - y * lambda;
  0 = x^2 + y^2 - l^2;
end pendulum;

```

图 3.18 理想单摆: MODELICA 中基于方程的实现

<sup>9</sup> 并不总是如此明显,例如在复杂模型情况下。

### 其他可能的模型：常微分方程组

#### 极坐标与直角坐标

目前所采用的单摆模型是用直角坐标  $x, y$  来表示(见图 3.14)。有的读者可能会想到, 极坐标下的单摆模型形为如下无另外代数条件的常微分方程组(指标为 0 的 DAE 系统):

$$\begin{aligned}\dot{\varphi} &= \omega, \\ \dot{\omega} &= -\frac{g}{l} \sin \varphi\end{aligned}\quad (3.54)$$

使用普通的显式积分算法可以毫无困难地对上面的模型进行仿真。

#### 建模与仿真方面

最后,除了上面的模型处理之外,需要说明的是,在系统设计中,总需要在仿真与建模之间进行折中考虑。通过对模型做合适的修改,甚至可以避免出现很多仿真难题。在上面的情况(坐标变换)下,这一点是成功的,并没有产生运动变量(角度、角速度)建模精度的任何损失。但是修改后的模型(3.54)中缺了约束力。若我们不关心后一情况(即约束力的缺失),从实现的观点来看第二种模型(极坐标模型)是更为可取的。

## 本章参考书目

- Ascher, U. M., H. Chin and S. Reich (1994). "Stabilization of DAEs and invariant manifolds." *Numerische Mathematik* 67: 131-149.
- Baumgarte, J. (1972). "Stabilization of constraints and integrals of motion in dynamical systems." *Computer Methods in Applied Mechanics and Engineering* 1: 1-16.
- Brenan, K. E., S. L. Campbell and L. R. Petzold (1996). *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM.
- Burgermeister, B., M. Arnold and B. Esterl (2006). "DAE time integration for real-time applications in multi-body dynamics." *ZAMM-Z. Angew. Math. Mech.* 86(10): 759-771.
- Cellier, F. E. and E. Kofman (2006). *Continuous System Simulation*. Berlin. Springer.
- Eich, E. (1993). "Convergence Results for a Coordinate Projection Method Applied to Mechanical Systems with Algebraic Constraints." *SIAM Journal on Numerical Analysis* 30(5): 1467-1482.
- Faires, J. D. and R. L. Burden (2002). *Numerical Methods*. Brooks/Cole.
- Franklin, G. F., J. D. Powell and M. L. Workman (1998). *Digital Control of Dynamic Systems*. Addison-Wesley.
- Ogata, K. (2010). *Modern Control Engineering*. Prentice Hall.
- Otter, M. (1999). "Objektorientierte Modellierung Physikalischer Systeme, Teil 4." *at-Automatisierungstechnik* 47(4): A13-A16.
- Otter, M. and B. Bachmann (1999). "Objektorientierte Modellierung Physikalischer Systeme, Teil 5." *at-Automatisierungstechnik* 47(5): A17-A20.
- Otter, M. and B. Bachmann (1999). "Objektorientierte Modellierung Physikalischer Systeme, Teil 6." *at-Automatisierungstechnik* 47(6): A21-A24.
- Otter, M., H. Elmqvist and S. E. Mattson (1999). "Objektorientierte Modellierung Physikalischer

Systeme, Teil 7. " at-Automatisierungstechnik 47(7): A25-A28.

Otter, M. , H. Elmqvist and S. E. Mattson (1999). " Objektorientierte Modellierung Physikalischer Systeme, Teil 8. " at-Automatisierungstechnik 47(8): A29-A32.

Potthoff, U. (2003). Zur Stabilität von numerischen Integrationsverfahren. Interner Bericht, Institut für Re.