



普通高等教育“十一五”规划教材
21世纪大学数学创新教材

丛书主编 陈 化

数学软件与

(第二版)

数学实验

王正东 主编



科学出版社
www.sciencep.com

启动仿真后,显示器上的读数表明借贷人的欠款额为 14 345.46 元,示波器显示的欠款数额随时间变化曲线如图 4.17 所示。

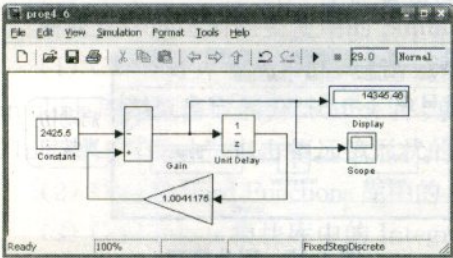


图 4.16 借贷还款问题的仿真模型

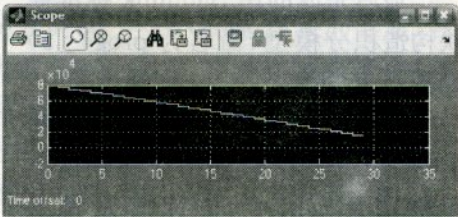


图 4.17 30 个月的欠款阶梯形曲线

4.2.2 蹦极系统的仿真

例 4.7 研究蹦极者的落体运动方程,在指定的参数条件下,用仿真模型讨论蹦极系统的安全性。

1) 蹦极系统的数学模型

蹦极是一种挑战自我的活动,蹦极者身系一根弹力绳从高高的桥梁(或悬崖)向下跳。在下落的过程中,蹦极者几乎处于失重状态。按牛顿运动规律,自由下落物体的位置由下式确定:

$$mx'' = mg - a_1x' - a_2x' | x' |$$

其中 m 为物体的质量, g 为重力加速度, $x = x(t)$ 是物体在时刻 t 的位置坐标,第二项与第三项表示空气阻力。设定 x 轴的正方向向下,物体系在一个劲度系数为 k 的弹力绳索上,取绳索末端为坐标原点,则其对落体位置的影响为

$$b = b(x) = \begin{cases} -kx, & x > 0 \\ 0, & x \leq 0 \end{cases} \quad (k > 0) \tag{4.19}$$

若绳索长 30 m,蹦极者起跳位置的坐标为 -30 m,整个蹦极系统的数学模型描述为

$$\begin{cases} mx'' = mg + b(x) - a_1x' - a_2x' | x' | \\ x'(0) = 0, x(0) = -30 \end{cases} \tag{4.20}$$

此系统为一个具有连续状态的非线性连续系统。设桥面距离地面 50 m,其余的参数分别是 $k = 20$, $a_1 = 1.6$, $a_2 = 1.0$, $m = 70$ kg, $g = 10$ m/s²。

建立蹦极系统的仿真模型,在如上的参数下对系统进行仿真,分析此蹦极系统对体重为 70 kg 的蹦极者是否安全。

2) 建立蹦极系统的仿真模型

将模型改写成

$$\begin{cases} x'' = \frac{1}{m}[mg + b(x) - a_1x' - a_2x' | x' |] \\ x'(0) = 0, x(0) = -30 \end{cases}$$

用积分模块构造微分方程求解模型的思路是：括号内的各项完成代数和运算以后，再乘以因子 $1/m$ ，就是 x'' ，而对 x'' 积分一次得到 x' ，对 x' 再积分一次就得 x 。在蹦极系统的仿真模型中，使用的系统模块有：

(1) Continuous 模块库中的 Integrator 积分模块，用来实现系统中的积分运算，本模型中的两个积分模块的标签分别改写为“Velocity”和“Position”。

(2) User-Defined Functions 模块库的 Fcn 模块，用来实现空气阻力函数关系。

(3) Commonly Used Blocks 模块库中：① 求和模块 Sum，实现求和运算；② 增益模块 Gain，分别提供变量的因子系数 k 以及 a_1 和 a_2 ；③ 两个输入的选择模块 Switch，用来实现系统中弹力绳索的函数关系式(4.19)；④ 接地模块 Ground，联结 Switch 模块一个空闲的输入端口。

(4) Sources 库中的两个 Constant 常量模块，一个模块提供模型中常数项 mg 之值；另一模块取值“50”，表明蹦极者相对于地面的距离。

(5) Sinks 库中的 Scope 示波器模块，以图形方式输出蹦极系统的仿真结果。蹦极系统的仿真系统如图 4.18 所示(命名为 prog4_7)。

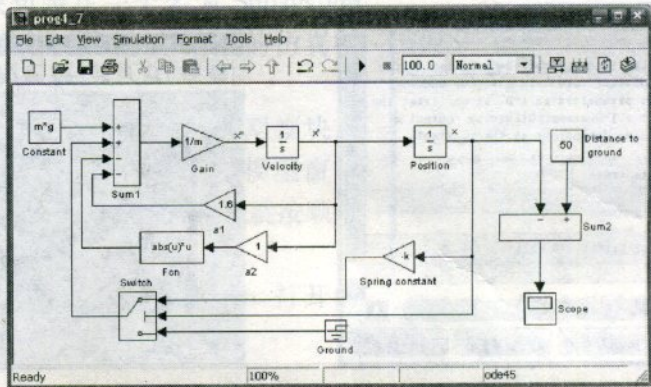


图 4.18 蹦极系统的仿真系统

注 本系统的输出模块 Scope 显示的是蹦极者的绝对位置，即相对于地面的距离，并不是相对于桥面的距离。

3) 各模块的参数配置

在建立求解方程(4.20)的模型后，同样需要设置模型中各个模块的参数。这里仅给出积分模块 Velocity 和 Position 初始值参数的设置，如图 4.19 所示。

在连续系统中，不能忽略的积分模块的初始值参数的设置，一般来说，初始条件决定了运动规律。

在本例中，蹦极者质量 m 和重力加速度 g 之积 mg 的值作为模型(4.20)的常数项由 Constant 模块提供，其参数设置如图 4.20 所示。其中的 Constant value 栏

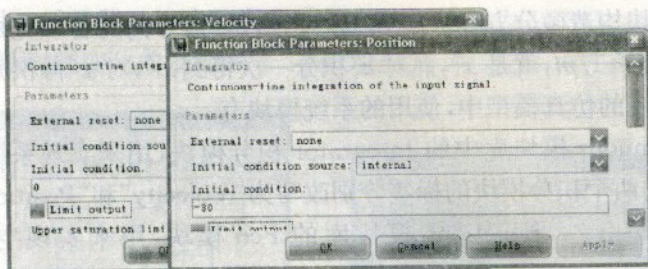


图 4.19 积分模块 Velocity 与 Position 初始值参数的设置

内可填写常数 700, 这里设置的是 $m \cdot g$, m 和 g 的值由 Matlab 工作环境中的变量给出, 与之相类似的还有对弹性常数 k 的设置, 在启动仿真之前必须在工作空间对这三个变量 (m , g 和 k) 赋值, 即先在 Matlab 命令窗口 (Command Window), 键入命令:

```
>> m = 70; g = 10; k = 20; % 向模块变量赋值
```

然后再点击仿真按钮, 运行蹦极系统. 采用命令方式向仿真模块输入参数增加了系统的可理解性, 并且便于在仿真过程中调整参数.

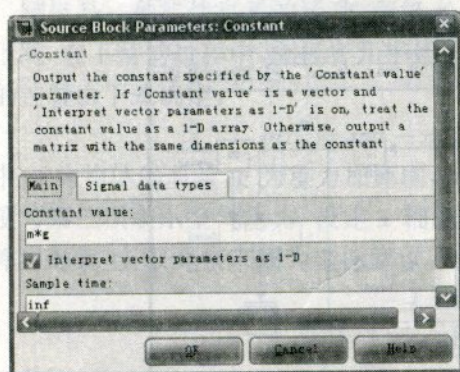


图 4.20 常量模块 Constant 的参数设置

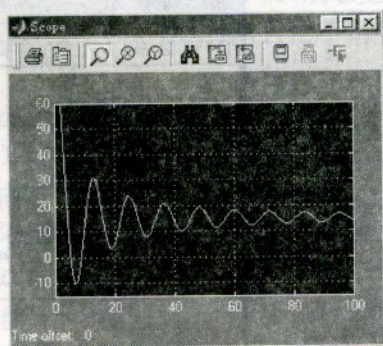


图 4.21 蹦极系统的仿真结果

其他模块参数设置都同前例, 不再赘述.

4) 系统仿真参数设置与仿真结果输出

点击模型窗口菜单 Simulation: Configuration Parameters, 打开仿真参数设置对话框, 选择 Simulation time 栏, 对蹦极系统的仿真参数设置如下: 系统仿真时间为 0~100 s; 其余参数采用系统默认值 (若认为仿真输出曲线不够光滑, 可将最大仿真步长 (Max step size) 由 0.1 改为更小).

启动仿真系统之前, 先在 Matlab 命令窗口键入命令:

```
>> m = 70; g = 10; k = 20;
```

执行完上述命令后, 再启动仿真, 仿真输出方程 (4.20) 的积分曲线 $x = x(t)$ 如图 4.21 所示.

图 4.21 所示为蹦极者与地面之间的距离. 从仿真结果可知:对于体重为 70 kg 的蹦极者来说,此系统是不安全的,因为蹦极者与地面之间的距离出现了负值,也就是说,蹦极者在下落的过程中会触及地面. 因此,必须改用弹力系数更大的弹性绳索,才能保证蹦极者的安全. 读者可以试一试,看看劲度系数 k 为何值时,蹦极者的安全才有保障.

5) 创建子系统

随着研究对象的情况变得更加复杂,直接用基本模块构成的模型也变得更加庞大,其中主要信息的流向就会不易辨识. 此时,若把整个模型按实现功能或对应物理器件的存在划分成一个或几个子系统,将有利于对整个系统的概念抽象. 以下介绍创建子系统(简称“包装”)的操作方法.

方法一 先有内容后加包装.

(1) 将图 4.18 所示的模型复制到一个新的模型窗口. 这一步是必要的,因为模型一旦被“打包”,就不再可能“解包复原”.

(2) 选定打算“包装”的模块,这里是除 Constant、Distance to ground、Scope 和 Sum 2 以外的其余所有模块. 点击模型窗口菜单 Edit>Create:Subsystem,就把前述选定的模块包装在一个名为 Subsystem 的模块中,经必要的整理,建成的仿真模型如图 4.22 所示.

在整理模块的过程中,若要对模块作旋转操作,可先选中模块,点击模型窗口菜单 Format:Rotate Block,模块就按顺时针方向旋转 90° .

双击子系统模块 Subsystem,打开的子系统窗口如图 4.23 所示.

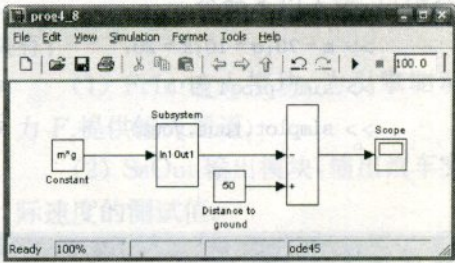


图 4.22 完成创建子系统后的仿真模型

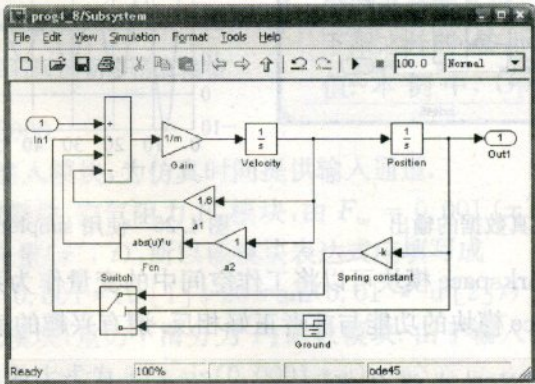


图 4.23 子系统窗口

方法二 先有包装后置内容.

(1) 在新建模型窗内复制 Commonly Used Blocks 库中的标准模块 Subsystem.

(2) 双击模块 Subsystem, 引出一个空白模型窗, 在该窗口内建立如图 4. 23 所示的子系统模型, 关闭子系统窗口, 子系统就制作完成了.

6) 在 Matlab 工作空间与 Simulink 模块之间传递信号

前面已经介绍了可以在 Matlab 工作空间向 Simulink 模型传递模块参数, 当然也可将 Simulink 的仿真结果输出到 Matlab 的工作空间, 可以用人们习惯的行命令方式进行仿真分析, 这样就使得用户可以对模型做更多的控制与修改, 大大方便了用户的操作.

在图 4. 22 模型的基础上增设一个 Out 模块(Commonly Used Blocks 库), 就可以将蹦极者的位置数据输出到工作空间. 新的仿真模型如图 4. 24 所示.

点击模型窗口菜单 Simulation: Configuration Parameters, 打开仿真参数设置对话框, 在 Save to workspace 栏中勾选 Time 和 Output 两项, 其中有两个变量分别是 tout 和 yout, 这表示将 Time 和 Output 两项数据分别以变量名 tout 和 yout 输送到工作空间. 将该模型取名为 prog4_9, 保存后退出模型窗口, 进入 Matlab 命令窗口, 键入以下命令:

```
>> m = 70; g = 10; k = 20;
```

```
>> sim('prog4_9')
```

% 运行名为 prog4_9 的仿真模型

```
>> simplot(tout, yout)
```

% 输出仿真曲线

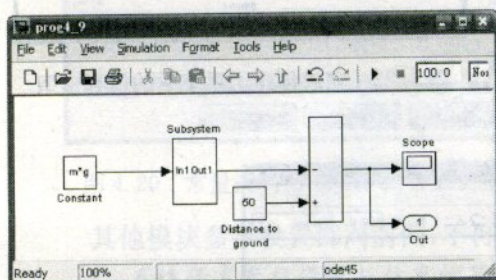


图 4. 24 仿真数据的输出

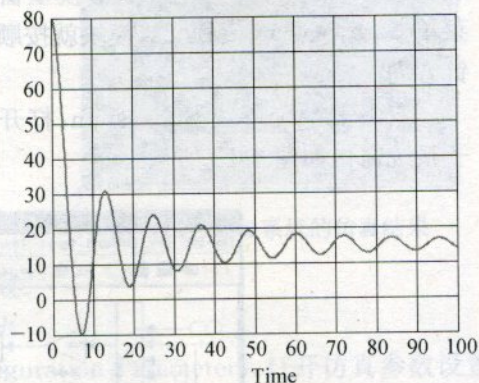


图 4. 25 使用 simplot 命令绘制的仿真结果

使用 From Workspace 模块可以将工作空间中的变量作为系统模型的输入信号, 而 To Workspace 模块的功能与前者正好相反. 请有兴趣的读者自己试试.

4. 2. 3 行驶汽车的动力学仿真

例 4. 8 质量为 m 的汽车沿直线行驶在上山的斜坡路上, 斜坡与水平面所成