16.070 Introduction to Computers and Programming

5 April **Recitation 8** **Spring 2001**

Topics

Procedure for the Simulation of Differential Equations (By Example):

- Concept
- Equations of motion
- Continuous State Space
- Discrete State Space
- Algorithm
- Coding
- Analysis

Procedure for the Simulation of Differential Equations

Concept

A Mars Lander(/rock/ball) is suspended 500m from the surface of the earth and released. Simulate the dynamics of the problem as the vehicle falls to earth. Only 1-DOF is required: the vertical altitude of the vehicle above the surface of the earth. Assume a constant gravitational acceleration of g=9.81m/s/s. The vehicles mass is 500kg (does this matter?). The vehicle does not thrust and drag is neglected. Write the vehicles state history to a file, in the format:

Equations of motion

Newton's second law: $F = ma = m\ddot{x}$

Initial Conditions:

$$x(0) = 500$$
$$\dot{x}(0) = 0$$

Forces acting on the vehicle: $F = -mg$

Complete simplified equation on motion: $m\ddot{x} = -mg$

Continuous State Space

The states are similar to the specific of initial conditions that are required:

State Vector:

$$\mathsf{X} = \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix}$$

Continuous State Space Representation:

$$\dot{\mathsf{X}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathsf{X} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} (-mg)$$

Discrete State Space

Approximate $\frac{dx}{dt}$ with $\frac{x_{n+1} - x_n}{\Delta t}$

then we get the following Discrete State Space result:

$$\begin{Bmatrix} \dfrac{x_{1_{n+1}} - x_{1_n}}{\Delta t} \\ \dfrac{x_{2_{n+1}} - x_{2_n}}{\Delta t} \end{Bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x_{1_n} \\ x_{2_n} \end{Bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} (-mg)$$

$$\begin{Bmatrix} x_{1_{n+1}} \\ x_{2_{n+1}} \end{Bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} x_{1_n} \\ x_{2_n} \end{Bmatrix} + \begin{bmatrix} 0 \\ \frac{\Delta t}{m} \end{bmatrix} (-mg)$$

Algorithm

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ \dfrac{\Delta t}{m} \end{bmatrix}$$

Variables and constants:

$$t_0 = 0$$

$$\Delta t = 0.01$$

$$t_{\max} = 20$$

$$\mathrm{X}_n$$

$$\mathrm{X}_{n+1}$$

$$F = -mg$$

Basic Pseudo-code (A flowchart will be added):

MAIN MODULE

[0]	Start	
[1]	Declare all the variables/constants above	
[2]	Open an output file for telemetry	
[3]	Set initial conditions	SEPARATE MODULE
[4]	Write initial states to a file	SEPARATE MODULE
[5]	Increment time (t=t+dt)	
[6]	If t>tmax, goto XX (IMPLEMENTED WITH A WHILE LOOP)	
[7]	Update State (Xn+1=AXn+BF)	SEPARATE MODULE
[8]	Write states to file	SEPARATE MODULE
[9]	Advance the state Xn=Xn+1	
[10]	Goto [6]	
[11]	Close all open files	
[12]	Exit	

Code