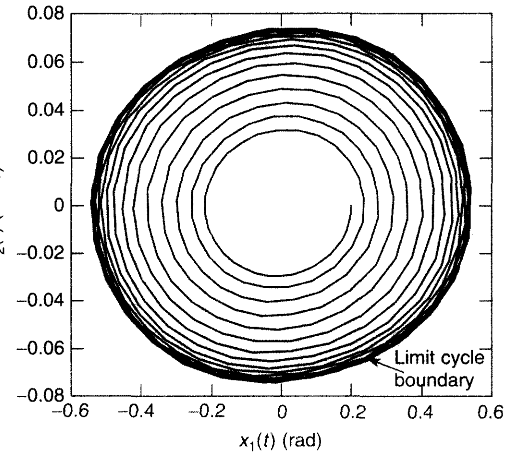

Figure 4.18 Phase-plane plot of $x_2(t)$ vs. $x_1(t)$ for the wing-rock problem of Example 4.13 showing the limit cycle boundary

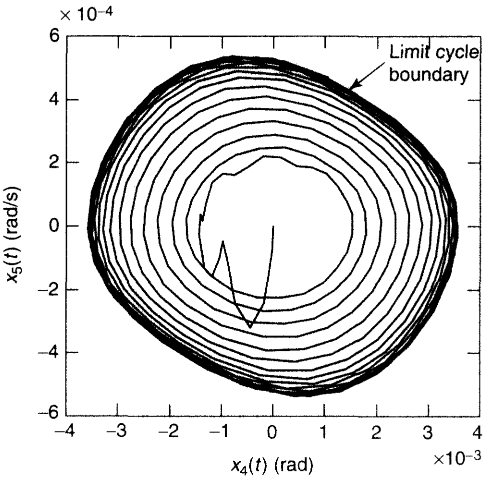

Figure 4.19 Phase-plane plot of $x_5(t)$ vs. $x_1(t)$ for the wing-rock problem of Example 4.13 showing the limit cycle boundary

of the motion has become constant. Similarly, Figure 4.19 shows the phase-plane plot of $x_5(t)$ against $x_4(t)$, displaying the limit cycle boundary.

4.7 Simulating Control System Response with SIMULINK

SIMULINK is very handy in quickly obtaining solutions to linear or nonlinear state-equations resulting from control systems with many sub-systems, using the ordinary differential equation solvers of MATLAB. SIMULINK allows the representation of each system by a set of linear and nonlinear *blocks*, and lets inputs and outputs to be modeled as special blocks called *sources* and *sinks*, respectively. A multivariable system can be represented either using the individual transfer functions between scalar inputs and outputs, or more conveniently by state-space model blocks. Copying the blocks from the SIMULINK *block library* makes simulating a control system very easy. Refer to Appendix B, or SIMULINK *User's Guide* [7] for more information on SIMULINK. Since SIMULINK works seamlessly with MATLAB and the Control System Toolbox (CST), you can draw upon the functions libraries of MATLAB and CST (and any other toolboxes that you may happen to have on your computer). The default simulation of SIMULINK uses the MATLAB function *ode45*, which is a fifth order, variable time-step Runge–Kutta solver (see Section 4.6). This allows a simulation of both linear and nonlinear state-equations.

Example 4.14

Let us simulate the step response of the flexible bomber aircraft (Example 4.7) using SIMULINK. The SIMULINK block diagram and the resulting simulation are shown in Figure 4.20. Note the *state-space* block modeling the aircraft dynamics, which requires that the state-coefficient matrices **A**, **B**, **C**, **D** of the aircraft (Eq. (4.71)) be available in the MATLAB work-space. The source *step input* block applies simultaneous unit step inputs, $u_1(t)$ and $u_2(t)$, to the aircraft, and the resulting output

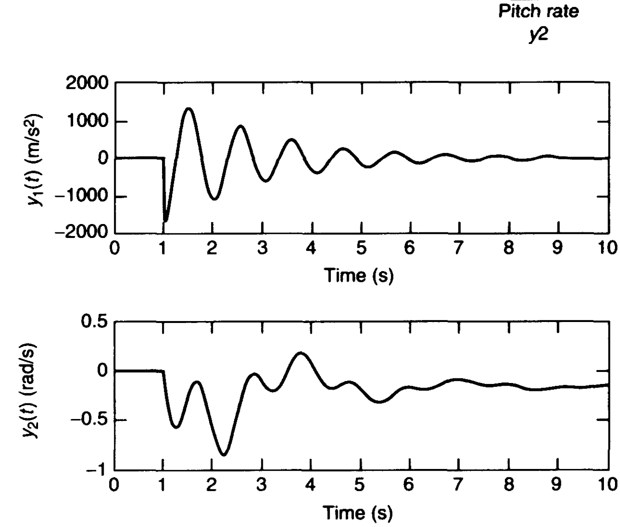

Figure 4.20 Simulation of the flexible bomber's step response using SIMULINK

vector is split into its two scalar elements, $y_1(t)$ (normal acceleration), and $y_2(t)$ (pitch-rate), by a *demux* block. The two simulated outputs are displayed individually on the scope blocks, as shown in Figure 4.20. The simulation is run with default parameters (variable time-step *ode45* solver with relative error tolerance of 0.001 and absolute tolerance of 10^{-6} per element).

A useful feature of SIMULINK is the availability of many nonlinear blocks, such as *dead-zone*, *backlash*, *saturation*, *rate-limiter*, *switch*, *relay*, *coulomb* and *viscous friction*, etc., which are commonly encountered in modeling many practical control systems. The following example illustrates the use of SIMULINK for a nonlinear control system.

Example 4.15

Let us simulate a nonlinear control system for controlling the roll dynamics of a fighter aircraft (Example 2.24). The *aileron input*, $\delta(t)$, to the aircraft is modeled

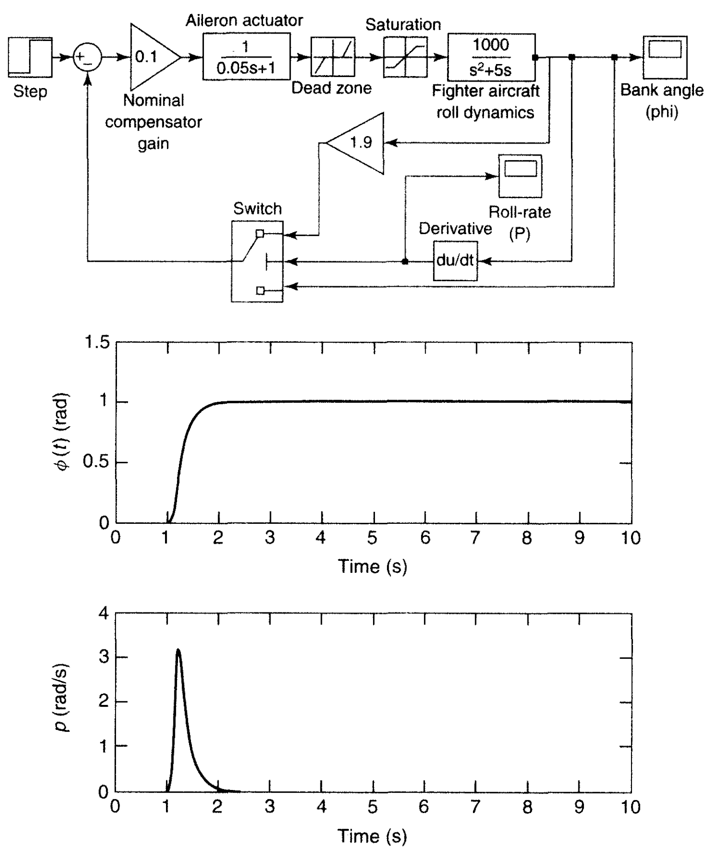

Figure 4.21 Simulation of a nonlinear control system for a fighter aircraft's roll dynamics using SIMULINK