

GEORGIA INSTITUTE OF TECHNOLOGY
School of Electrical and Computer Engineering

EE2823

Project No. 1

Discrete-Time Simulation of Continuous-Time Systems

Date Assigned: April 9, 1999

Date Due: April 23, 1999

1 Simulation

Many times, it is useful to *simulate* the behaviour of a continuous-time system by using digital computation. In such cases, it is necessary to deal with sampled signals and discrete-time systems that are derived from the continuous-time system. In this project you will study a simple example of how this can be done.

Many continuous-time systems are described by differential equations. Examples include electrical networks, biological processes, and mechanical systems. In such cases, the input and output are related by a linear, constant-coefficient, differential equation of the general form

$$\sum_{k=0}^N a_k y^{(k)}(t) = \sum_{k=0}^M b_k x^{(k)}(t), \quad (1)$$

where $y^{(k)}(t) = \frac{dy^k(t)}{dt^k}$. This equation is assumed to be satisfied for every value of t . Thus, if we sample $x(t)$ and $y(t)$ and all the required derivatives at times nT , we can obtain a relation involving the samples of the derivatives as in

$$\sum_{k=0}^N a_k y^{(k)}(nT) = \sum_{k=0}^M b_k x^{(k)}(nT). \quad (2)$$

In general, we would not know any of the derivatives of either $x(t)$ or $y(t)$. Nevertheless, Eq. (2) is a useful starting point for our discrete simulation.

In order to obtain a difference equation that we can solve for $y(nT)$, we need to express all the sampled derivatives in terms of $y(nT)$ and $x(nT)$. One simple approach is to recall that the derivative of a function at a given time is just the slope of the function at that time. For example we could approximate $y^{(1)}(nT)$ by the difference between $y(nT)$ and its neighboring sample divided by the time increment T . If we pick the sample just before time nT , we get

$$y^{(1)}(nT) = \frac{y(nT) - y(nT - T)}{T}. \quad (3)$$

Equation 3 is called a *first backward difference* approximation of the derivative. An alternative approximation is the *first forward difference* approximation

$$y^{(1)}(nT) = \frac{y(nT + T) - y(nT)}{T}. \quad (4)$$

Now since the derivatives are related as $y^{(k)}(t) = \frac{dy^{(k-1)}(t)}{dt}$, with $y^{(0)}(t) = y(t)$, we can successively apply either Eq. (3) or Eq. (4) to obtain all the approximate sampled derivatives in Eq. (2) in terms of either $y(nT)$ or $x(nT)$. The following example illustrates how this is done.

EXAMPLE Suppose that the system is described by the differential equation

$$y^{(2)}(t) + y^{(1)}(t) + 1.25y(t) = x(t).$$

The sampled version of this differential equation would be

$$y^{(2)}(nT) + y^{(1)}(nT) + 1.25y(nT) = x(nT), \quad (5)$$

which shows that we need to express both $y^{(2)}(nT)$ and $y^{(1)}(nT)$ in terms of $y(nT)$. Using the backward difference formula in Eq. (3) we get

$$y^{(1)}(nT) = \frac{y(nT) - y(nT - T)}{T}, \quad (6)$$

which gives us $y^{(1)}(nT)$ in terms of $y(nT)$. Now using the backward difference formula to obtain $y^{(2)}(nT)$ in terms of $y^{(1)}(nT)$ we obtain

$$y^{(2)}(nT) = \frac{y^{(1)}(nT) - y^{(1)}(nT - T)}{T}. \quad (7)$$

Now we can substitute Eq. (6) into Eq. (7) to obtain

$$y^{(2)}(nT) = \frac{y(nT) - 2y(nT - T) + y(nT - 2T)}{T^2}. \quad (8)$$

Finally, we substitute Eq. (7) and Eq. (8) into Eq. (5) and collect terms to obtain the difference equation

$$\begin{aligned} y(nT) &= \left(\frac{2 + T}{1 + T + 1.25T^2} \right) y(nT - T) - \left(\frac{1}{1 + T + 1.25T^2} \right) y(nT - 2T) \\ &\quad + \left(\frac{T^2}{1 + T + 1.25T^2} \right) x(nT). \end{aligned} \quad (9)$$

Now at last we have obtained a difference equation in the form that we are familiar with from EE2200.

Problem 1: We can implement the difference equation in Eq. (9) using MATLAB's `filter(b,a,x)` function. In using this function, we would need to sample the input $x(t)$ to obtain the vector \mathbf{x} , and we would need to obtain the vectors of coefficients \mathbf{b} and \mathbf{a} . Write a MATLAB program to compute the unit step response of the system of the above example; i.e., assume an input $x(t) = u(t)$. Your program should take the sampling period T as an input variable, and it should compute samples of the output over the time interval $0 \leq nT \leq 20$. You will have to compute the vectors \mathbf{a} and \mathbf{b} and compute the sampled input vector \mathbf{x} . (Since the unit step is infinitely long, you will have to decide how much of it you need.)

Problem 2: Run your program for values of $T = 1, 0.1,$ and $0.01,$ each time computing the output at sample times in the interval $0 \leq nT \leq 20.$ Plot all three simulations on the same graph and hand in the plot with your report. What can you say about the effect of the sampling parameter T ? Which value of T do you think gives a simulation that is closest to the theoretically correct answer?

Problem 3: Make a new version of your program in Problem 1 for the input signal $x(t) = u(t) - u(t - 10).$ Again evaluate the output over the interval $0 \leq nT \leq 20.$ Test your program for the value of T determined to give the best results in Problem 1. Plot the output as a function of time. Using the principle of superposition, explain why the plot looks as it does.

2 Simulation of a Power Supply Circuit

Figure 1 depicts a circuit for creating a DC voltage from an AC voltage. The diode bridge implements a full-wave rectifier, and the RC circuit provides lowpass filtering to produce a constant output.

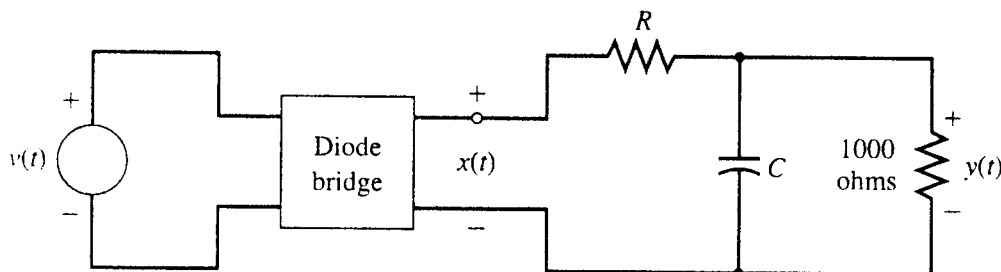


Figure 1: Diode bridge power Supply circuit.

The block diagram shown in Figure 2 represents the operations performed by the different parts of the circuit. The purpose of the lowpass filter is to remove most of the high frequencies

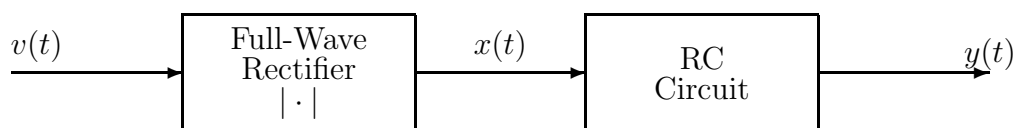


Figure 2: Block diagram representation of power supply.

in the output of the rectifier, leaving an output that is nearly constant with time (called "DC"). The full-wave rectifier is defined by the equation

$$x(t) = |v(t)| \quad (10)$$

where $v(t)$ is the input and $x(t)$ is the output of the full-wave rectifier. The output of the rectifier, $x(t)$, is the input to the RC circuit and $y(t)$ is the output of that circuit. They are related by the differential equation

$$y^{(1)}(t) + \left(\frac{1000 + R}{1000RC}\right)y(t) = \frac{1}{RC}x(t) \quad (11)$$

In a real power supply, the signal $v(t)$ would be the powerline voltage, which would be represented mathematically as $v(t) = 120\sqrt{2}\cos(120\pi t)$.

Problem 4: Using the first backward difference approximation for the derivative, determine a difference equation for simulation of the RC circuit. Your answer should be in terms of R and C .

Problem 5: Write a new MATLAB program to simulate the entire system in Figures 1 and 2. Your program should again use first backward difference approximation to the derivative and should take the sampling period T , the resistance R , and the capacitance C as input parameters. Evaluate the output over ten periods of the power line voltage $v(t)$. Build plotting into your program – plot the input $v(nT)$, the rectified signal $x(nT)$, and the output $y(t)$ over the time interval of ten periods of $v(t)$.

Problem 6: Test your simulation for the conditions $R = 20610$ ohms and $C = 0.000003$ farads. Try different values of T to determine the best approximation. Hint: What is the maximum value of T that we could use to sample the input sinusoid to avoid aliasing? Why should T be much smaller than this?

Problem 7: Using the value of T determined to work best in Problem 6, experiment with keeping $R = 20610$ and varying C . How should C be changed to make the output more like a constant (smaller ripples)? What value of C is necessary to achieve less than 1 volt of peak-to-peak ripple about the constant output value? What is the approximate DC output voltage of this power supply?

Problem Optional: Use your simulation to determine R and C such that the DC output voltage is 10 volts with less than 1 volt of peak-to-peak ripple.

3 Report

Prepare a concise report on this project. Organize it by giving answers to each of the seven problems posed above. In some cases you will simply need to give brief answers to any explicit questions that were asked. In all cases, give listings of MATLAB functions and scripts that you use to solve the problem. Also include (possibly hand annotated) hard copies of the graphs to refer to in your discussion.