1. Differential equations with piecewise constant inputs

Working through this question will help you understand better differential equations with inputs and the sampling of a continuous-time system of differential equations into a discrete-time view.

(a) Consider the scalar system

\[
\frac{d}{dt} x(t) = \lambda x(t) + u(t). \tag{1}
\]

Suppose that our \( u(t) \) of interest is \emph{constructed} to be piecewise constant over durations of width \( \Delta \). In other words:

\[
u(t) = u[i] \text{ if } t \in [i\Delta, (i+1)\Delta) \tag{2}\]

Given that we start at \( x(i\Delta) = x_d[i] \), where do we end up at \( x_d[i + 1] = x((i+1)\Delta) \)?

**Answer:**

Our differential equation takes the form,

\[
\frac{d}{dt} x(t) = \lambda x(t) + u(i\Delta) \tag{3}
\]

where \( u(i\Delta) \) is a constant value of some input function \( u(t) \) at time \( t = i\Delta \).

First we solve the differential equation by guessing

\[
x(t) = \alpha e^{\lambda (t-i\Delta)} + \beta
\]

This gives,

\[
\frac{d}{dt} x(t) = \lambda \alpha e^{\lambda (t-i\Delta)}
\]

We know that this should equal to the right hand side of (3), so we get,

\[
\lambda \alpha e^{\lambda (t-i\Delta)} = \lambda x(t) + u(i\Delta) = \lambda (\alpha e^{\lambda (t-i\Delta)} + \beta) + u(i\Delta)
\]

\[
\Rightarrow \lambda \alpha e^{\lambda (t-i\Delta)} = \lambda \alpha e^{\lambda (t-i\Delta)} + \lambda \beta + u(i\Delta)
\]

Now using \( u(i\Delta) = u[i] \), we get,

\[
\beta = \frac{-u[i]}{\lambda}
\]

Further, since \( x(i\Delta) = x_d[i] \), we get,

\[
x_d[i] = x(i\Delta) = \alpha e^{\lambda (i\Delta - i\Delta)} + \beta = \alpha + \beta
\]
And using, $\beta = -\frac{u[i]}{\lambda}$ we get,

$$x_d[i] = \alpha + \frac{-u[i]}{\lambda}$$

$$\implies \alpha = x_d[i] + \frac{u[i]}{\lambda}$$

So, we get that,

$$x(t) = (x_d[i] + \frac{u[i]}{\lambda})e^{\lambda(t-i\Delta)} - \frac{u[i]}{\lambda}$$

$$\implies x(i\Delta) = x_d[i]e^{\lambda(t-i\Delta)} + (\frac{e^{\lambda(t-i\Delta)} - 1}{\lambda})u[i]$$

Thus,

$$x_d[i+1] = x((i+1)\Delta) = x_d[i]e^{\lambda\Delta} + (\frac{e^{\lambda\Delta} - 1}{\lambda})u[i]$$

(b) Suppose that $x_d[0] = x_0$. **Unroll the implicit recursion you derived in the previous part to write $x_d[i+1]$ as a sum that involves $x_0$ and the $u[j]$ for $j = 0, 1, \ldots, i$.**

For this part, feel free to just consider the discrete-time system in a simpler form

$$x_d[i+1] = ax_d[i] + bu[i]$$

and you don’t need to worry about what $a$ and $b$ actually are in terms of $\lambda$ and $\Delta$.

**Answer:** Let’s look at the pattern starting with $x_d[1]$, given that $x_d[i+1] = ax_d[i] + bu[i]$,

$$x_d[1] = ax_d[0] + bu[0]$$


$$\implies x_d[2] = a(ax_d[0] + bu[0]) + bu[1] = a^2(x_d[0]) + bu[1]a + bu[1]$$


$$\implies x_d[3] = a^3x_d[0] + bu[0]a^2 + u[1]a + u[2])$$

So, given this pattern, if we guess,

$$x_d[i] = a^ix_d[0] + b\left(\sum_{j=0}^{i-1} u[j]a^{i-j}\right)$$

(5)

Then, let’s see what we get for $x_d[i+1]$,

$$x_d[i+1] = ax_d[i] + bu[i] = a(a^ix_d[0] + b\left(\sum_{j=0}^{i-1} u[j]a^{i-j}\right)) + bu[i]$$

$$\implies x_d[i+1] = a^{i+1}x_d[0] + b\left(\sum_{j=0}^{i-1} u[j]a^{i-j}\right) + u[i] = a^{i+1}x_d[0] + b\left(\sum_{j=0}^{i-1} u[j]a^{i-j}\right)$$

This satisfies (5), for i+1 and hence our guess was correct!

This turns out to be a proof by induction, with base case $x_d[1] = ax_d[0] + bu[0]$. Going from $i$ to $(i+1)$ is the inductive step. This is how we transform a recursively found pattern into a rigorous proof!
(c) **For a given time \( t \) in real time, what is the \( i \) that corresponds to it?** Meaning that if we think of the computer viewing the system as perceiving a constant state for durations \([i\Delta,(i+1)\Delta)\), which is the discrete time index for the \( x_d[i] \) that corresponds to \( t \) in real time.

**Answer:**

\[
i = \left\lfloor \frac{t}{\Delta} \right\rfloor
\]

is the discrete time index \( i \) that corresponds to the time \( t \) in real time, because it is the only \( i \) satisfying \( t \in [i\Delta,(i+1)\Delta) \)

Note: \( \lfloor x \rfloor \) is the largest integer smaller than \( x \).

(d) Now, we are going to turn this around. Suppose that the \( u[i] \) is actually a sample of a desired control input \( u_c(t) \) in continuous time. Namely that \( u[i] = u_c(i\Delta) \). Recall what the \( a \) and \( b \) mean in \( [\text{1}] \) and approximate \( x(t) \) if we apply this piecewise constant control \( u(t) \) to the system \( [\text{1}] \). You can assume that \( \Delta \) is small enough that \( x(t) \) can’t change too much over an interval of length \( \Delta \).

**Answer:**

Using the result derived in part (b), we get,

\[
x(t) \approx x(a) = \sum_{j=0}^{\lfloor \frac{t}{\Delta} \rfloor} a \frac{\Delta}{\lambda} u[j] + b \sum_{j=0}^{\lfloor \frac{t}{\Delta} \rfloor} u[j] \frac{\Delta}{\lambda}
\]

Now, since \( u[j] = u_c(j\Delta) \) and from part (a), we have that, \( a = e^{\lambda\Delta} \) and \( b = (\frac{e^{\lambda\Delta} - 1}{\lambda}) \) we get,

\[
x(t) \approx (e^{\lambda\Delta}) \frac{\Delta}{\lambda} x_d[0] + \sum_{j=0}^{\lfloor \frac{t}{\Delta} \rfloor} (e^{\lambda\Delta} - 1) \frac{\Delta}{\lambda} u_c(j\Delta)
\]

(e) **Draw a picture of what is going on and then further approximate the previous expression by considering \( n = \lfloor \frac{t}{\Delta} \rfloor \approx \frac{t}{\Delta} \) where needed and treating \( \Delta \approx \frac{t}{n} \).** This is a meaningful approximation when we think about \( n \) large enough.

**Answer:** Plugging in \( n = \lfloor \frac{t}{\Delta} \rfloor \approx \frac{t}{\Delta} \) into the result of part (d), we get,

\[
x(t) \approx (e^{\lambda\Delta}) x_d[0] + \sum_{j=0}^{\lfloor \frac{t}{\Delta} \rfloor} (e^{\lambda\Delta} - 1) \frac{\Delta}{\lambda} u_c(j\Delta)
\]

(f) **Take the limit of \( \Delta \to 0 \) by taking the limit \( n \to \infty \). What is the expression you get for \( x(t) \)?**

*(HINT: Remember your definition of definite integrals as limits of Riemann sums in calculus.)*

**Answer:**

Since we are allowed to assume \( \Delta \approx \frac{t}{n} \) from part (e), we get,

\[
\lim_{\Delta \to 0} x(t) = \lim_{n \to \infty} x(t) = \lim_{\Delta \to 0} (e^{\lambda\Delta}) x_d[0] + \lim_{\Delta \to 0} (e^{\lambda\Delta} - 1) \sum_{j=0}^{\lfloor \frac{t}{\Delta} \rfloor} u_c(j\Delta)
\]

Now, a first-order approximation gives us \( \lim_{\Delta \to 0} (\frac{e^{\lambda\Delta} - 1}{\lambda}) = \frac{t}{\Delta} \) and further, \( \lim_{n \to \infty} (e^{-\frac{t}{n}}) = 1 \), so we get,
\[ \lim_{\Delta \to 0} x(t) = e^{\lambda t} x_d[0] + \lim_{n \to \infty} e^{\lambda t} \sum_{j=0}^{n-1} u_c(t j/n) \left( e^{-\lambda t j/n} \right) \]

This looks like a Riemann sum, so plugging in \( \tau = t j/n \) which would imply that \( d\tau = \lim_{n \to \infty} t/n \) we get,

\[ \lim_{n \to \infty} x(t) = e^{\lambda t} x_d[0] + e^{\lambda t} \int_0^t u_c(\tau) e^{-\lambda \tau} d\tau \]

(g) Stepping back. Suppose we have a system of differential equations with an input that we express in vector form:

\[
\frac{d}{dt} \bar{x}_c(t) = Ax_c(t) + bu(t) \tag{6}
\]

where \( \bar{x}_c(t) \) is \( n \)-dimensional.

Suppose further that the matrix \( A \) is invertible and has distinct eigenvalues \( \lambda_1, \lambda_2, \ldots, \lambda_n \). With corresponding eigenvectors \( \bar{v}_1, \bar{v}_2, \ldots, \bar{v}_n \). Collect the eigenvectors together into a matrix \( V = [\bar{v}_1, \bar{v}_2, \ldots, \bar{v}_n] \).

If we apply a piecewise constant control input \( u(t) \) as in (2), and sample the system \( \bar{x}_d[i] = \bar{x}_c(i\Delta) \), what are the corresponding \( A_d \) and \( \bar{b}_d \) in:

\[
\bar{x}_d[i+1] = A_d \bar{x}_d[i] + \bar{b}_d u[i]. \tag{7}
\]

**Answer:** First, we change coordinates so that \( \bar{x}_c(t) = V \bar{x}(t) \) and \( \bar{x}(t) = V^{-1} \bar{x}_c(t) \).

We have,

\[
(\bar{x}_d[i+1])[j] = (e^{\lambda_j \Delta})(\bar{x}_d[i])[j] + \left( \begin{array}{c} \frac{e^{\lambda_j \Delta} - 1}{\lambda_j} \\ \vdots \\ \frac{e^{\lambda_j \Delta} - 1}{\lambda_n} \end{array} \right) (V^{-1} \bar{b}[j])(u[i])
\]

\[
\bar{x}_d[i+1] = \left( \begin{array}{cccc} e^{\lambda_1 \Delta} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & e^{\lambda_n \Delta} \end{array} \right) \bar{x}_d[i] + \left( \begin{array}{cccc} \frac{e^{\lambda_1 \Delta} - 1}{\lambda_1} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \frac{e^{\lambda_n \Delta} - 1}{\lambda_n} \end{array} \right) V^{-1} \bar{b} u[i]
\]

Now we define the following notations,

\[
E_{\Delta \Lambda} = \left( \begin{array}{cccc} e^{\lambda_1 \Delta} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & e^{\lambda_n \Delta} \end{array} \right)
\]

\[
\Lambda^{-1} = \left( \begin{array}{cccc} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\lambda_n} \end{array} \right)
\]

So,

\[
x_d[i+1] = V \bar{x}_d[i+1] = (V E_{\Delta \Lambda} V^{-1}) x_d[i] + (V \Lambda^{-1}(E_{\Delta \Lambda} - I)V^{-1} \bar{b}) u[i]
\]

Hence,
\[ A_d = (VE_{\Lambda\Lambda}V^{-1}) \]

and

\[ \tilde{b}_d = (V\Lambda^{-1}(E_{\Lambda\Lambda} - I)V^{-1}\tilde{b}) \]

(h) Leveraging what you learned in the previous part, explain why if the input \( u(t) = e^{st} \) is applied in (6), then any of the state variables within \( \tilde{x}_c(t) \) can be expressed as a sum \( \sum_{k=1}^{n} (\alpha_k(s)e^{\lambda_k t} + \frac{\beta_k}{s-\lambda_k}e^{st}) \) where the \( \beta_k \) do not depend on \( s \) or the initial condition even though the \( \alpha_k(s) \) might depend on both \( s \) and the initial condition \( \tilde{x}_c(0) \).

Answer:

In the diagonal basis, we have \( \tilde{b}_d = \begin{bmatrix} \tilde{b}_{d_1} \\ \tilde{b}_{d_2} \\ \vdots \\ \tilde{b}_{d_n} \end{bmatrix} = \Lambda^{-1}(E_{\Lambda\Lambda} - I)V^{-1}\tilde{b} \). Hence,

\[ \tilde{x}_d[i + 1] = E_{\Lambda\Lambda}\tilde{x}_d[i] + \tilde{b}_d u[i] \]

\[ \Rightarrow (\tilde{x}_d[i + 1])[j] = e^{\alpha_j}(\tilde{x}_d[i])[j] + \tilde{b}_j e^{\lambda_j} \]

But, we know the solution to the above system:

\[ (\tilde{x}(t))[j] = e^{\alpha_j}(\tilde{x}_d[0])[j] + e^{\alpha_j} \int_0^t \tilde{b}_j e^{\lambda_j} e^{-\lambda_j \tau} d\tau \]

\[ = e^{\alpha_j}(\tilde{x}_d[0])[j] + e^{\alpha_j} \left( \frac{\tilde{b}_j}{s-\lambda_j} e^{(s-\lambda_j) \tau} \right) \bigg|_0^t \]

\[ = e^{\alpha_j} \left( (\tilde{x}_d[0])[j] - \tilde{b}_j \right) + \frac{\tilde{b}_j}{s-\lambda_j} e^{st} \]

From above, we find the solution in our diagonal coordinate system, but we still need to convert back to our original coordinate system,

\[ \tilde{x}_c = V\tilde{x} = \begin{bmatrix} \tilde{v}_1 \\ \tilde{v}_2 \\ \vdots \\ \tilde{v}_n \end{bmatrix} \tilde{x} = \begin{bmatrix} v_{11} & v_{21} & \cdots & v_{n1} \\ v_{21} & v_{22} & \cdots & v_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{1n} & v_{2n} & \cdots & v_{nn} \end{bmatrix} \tilde{x} \]

\[ \Rightarrow (\tilde{x}_c(t))[j] = \sum_{k=1}^{n} v_{jk} \left( (\tilde{x}_d[0])[k] - \tilde{b}_k \right) e^{\lambda_k t} + v_{jk} \frac{\tilde{b}_k}{s-\lambda_k} e^{st} \]

\[ : (\tilde{x}_c(t))[j] = \sum_{k=1}^{n} (\alpha_k(s)e^{\lambda_k t} + \frac{\beta_k}{s-\lambda_k}e^{st}) \]

where, \( \alpha_k(s) = v_{jk} \left( (\tilde{x}_d[0])[k] - \tilde{b}_k \right) \) and \( \beta_k = v_{jk}\tilde{b}_k \)

Contributors: