# Assignment 1

## SC42101: Networked and Distributed Control Systems

Richard Adam [redacted]


Delft University of Technology

TUDelft

# Introduction

The purpose of this assignment is to analyze and control a networked control system(NCS) with delays $\tau$ and sampling time h. The schematic for the NCS can be seen in Figure 1:

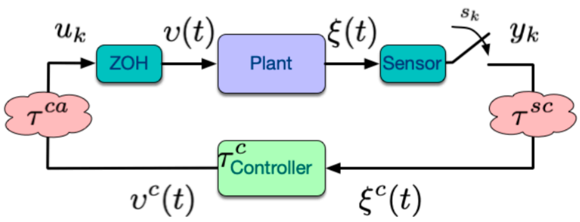



**Figure 1:** Diagram of a Networked Control System

The plant dynamics represent a linear time-invariant system of the form:

$$\dot{\xi}(t) = A\xi(t) + B\nu(t)$$

where $\nu$ is a piecewise constant signal applied in a sample-and-hold fashion. A and B are given by:

$$A = \begin{bmatrix} 0 & -1.5 \\ 2.2 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

# Question 1: Pole Placement and Sampled-Data Stability

## 1.0: Controller Design

Using the control law $\nu(t) = -K\xi(t)$, the poles are placed at $-2 \pm j$ by plugging in $K = \begin{bmatrix} K_1 & K_2 \end{bmatrix}$ to the equation $\xi(t) = (A - BK)\xi(t)$ and ensuring that the resulting system has poles at the desired location. Ultimately, the K matrix is found to be $K = \begin{bmatrix} 3 & -0.5909 \end{bmatrix}$.

## 1.1: Discretization and Stability Analysis

Assuming a constant sampling time h and $\tau = 0$, the discrete-time system $x_{k+1} = F(h)x_k + G(h)u_k$ can be explicitly derived. One can start with the solution to the continuous-time system:

$$\xi(t) = e^{A(t-t_0)}\xi(t_0) + \int_{t_0}^{t} e^{A(t-s)}B\nu(s)ds \tag{1}$$

We can define the following:

$$x_{k+1} = \xi(s_{k+1})$$

$$\nu(t) = u_k \quad for \quad t\epsilon[s_k, s_{k+1}]$$

Plugging these in and changing the integration variables gives the following equation:

$$x_{k+1} = e^{Ah}x_k + (\int_{0}^{h} e^{As}Bds)u_k \tag{2}$$

So one can define:

$$F(h) = e^{Ah} \tag{3}$$

and

$$G(h) = \int_0^h e^{Ah} B ds \tag{4}$$

Finding $e^{Ah}$ is relatively simple, as the eigenvalues of A are $-0.5 \pm 1.7464j$, so Equation 5, describing the exponential of a matrix with complex eigenvalues $a \pm bj$, can be used:

$$e^{Ah} = e^{ah}(cos(bh)I + \frac{sin(bh)}{b}(A - aI)) \tag{5}$$

which results in:

$$F(h) = e^{-0.5h} \begin{bmatrix} cos(1.746h) + 0.286sin(1.746h) & -0.859sin(1.746h) \\ 1.26sin(1.746h) & cos(1.746h) - 0.286sin(1.746h) \end{bmatrix} \tag{6}$$

To find $G(h)$, one can use the property:

$$e^{\tilde{A}h} = \begin{bmatrix} F(h) & G(h) \\ 0 & 1 \end{bmatrix} \tag{7}$$

where:

$$\tilde{A} = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} \tag{8}$$

This was solved with MATLAB's symbolic math toolbox and simplified to the following:

$$G(h) = \begin{bmatrix} 0.303 - e^{-0.5h}(0.303cos(1.746h) - 0.486sin(1.746h)) \\ e^{-0.5h}(0.667cos(1.746h) + 0.1908sin(1.746h)) \end{bmatrix} \tag{9}$$

The full derivation can be seen in the handwritten notes in Appendix A. Now that the analytical expression has been found for the discrete-time system. A stability analysis can be conducted by checking the spectral radius of $(F(h) - G(h)K)$ for a range of sampling times between zero and ten seconds. Ten seconds was chosen because the largest settling time of the system to a step input is 8.7 seconds, so ten seconds allows delays spanning the transient response and some steady-state response. The resulting spectral radius is graphed in Figure 2:

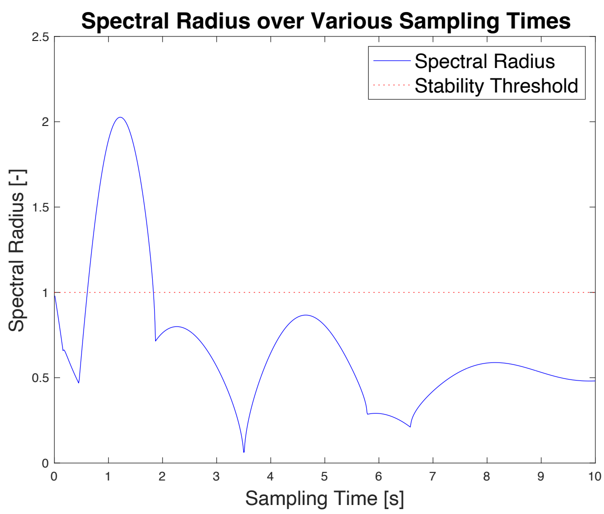



**Figure 2:** Spectral Radius of Controlled System for Sampling Time of 0-10 Seconds

The spectral radius begins in the stable region below 1, but crosses into the unstable region at about h=0.6. It then crosses back into the stable region around h=1.9s, where it remains fluctuating but stable until h=10s. It is important to note that, though the system will remain globally exponentially stable at sampling times higher than 2 seconds, the inter-sample behavior may not be ideal for all applications.

# Question 2: Modeling and Controlling NCS with Delays

## 2.1: Modeling with Delay

Alongside sampling time, delay is added to the system between the sensor, controller, and zero-order hold. The sum of these delays is designated as $\tau \ \epsilon \ [0, h)$. In order to derive the analytical expression for F and G, one can start with the solution to the continuous-time equation, given by Equation 1.

Knowing that $\tau$ lies between 0 and h allows us to define:

$$x_{k+1} = \xi(s_{k+1})$$

$$x_k = \xi(s_k)$$

$$\nu(t) = u_{k-1} \quad for \quad t\epsilon[s_k, s_k + \tau]$$

$$gi\nu(t) = u_k \quad for \quad t\epsilon[s_k + \tau, s_{k+1}]$$

which can then be plugged into the integral to find the equation, after the integral variables have been changed:

$$x_{k+1} = e^{Ah}x_k + \int_{h-\tau}^{h} e^{Ah}Bdsu_{k-1} + \int_{0}^{h-\tau} e^{Ah}Bdsu_k \tag{10}$$

which can be rewritten as:

$$x_{k+1} = F_x(h)x_k + F_u(h, \tau)u_{k-1} + G_u(h, \tau)u_k \tag{11}$$

This can be simplified by using the extended state $x_k^e = [x_k^T, \ u_{k-1}]^T$, which results in:

$$x_{k+1} = F(h, \tau)x_k^e + G(h, \tau)u_k \tag{12}$$

where

$$F = \begin{bmatrix} F_x(h) & F_u(h, \tau) \\ 0 & 0 \end{bmatrix} \text{ and } G = \begin{bmatrix} G(h, \tau) \\ 1 \end{bmatrix}$$

These equations can be found by individually evaluating $F_x(h), F_u(h, \tau)$, and $G(h, \tau)$. The values of $F_x(h), F_u(h, \tau)$, and $G(h, \tau)$ were found by first solving for the expression inside the integral using MATLAB's symbolic math toolbox. The integrals were solved element-wise and simplified in order to obtain exact expressions. The full derivation can be seen in Appendix A. After simplification, the final analytical expressions for $F_x$, $F_u$, and $G_u$ are:

$$F_x = e^{-0.5h} \begin{bmatrix} cos(f*h) + 0.2863sin(f*h) & -0.859sin(f*h) \\ 1.26sin(f*h) & cos(f*h) - 0.286sin(f*h) \end{bmatrix}$$

$$F_u = \begin{bmatrix} e^{-0.5h}(0.486sin(f*h) - 0.303cos(f*h)) - e^{-0.5(h-\tau)}(0.486sin(f*(h-\tau)) - 0.303cos(f*(h-\tau))) \\ e^{-0.5h}(-0.191sin(f*h) - 0.667cos(f*h)) - e^{-0.5(h-\tau)}(-0.191sin(f*(h-\tau)) - 0.667cos(f*(h-\tau))) \end{bmatrix}$$

$$G_1 = \begin{bmatrix} e^{-0.5(h-\tau)}(0.486sin(f*(h-\tau)) - 0.303cos(f*(h-\tau))) + 0.303 \\ e^{-0.5(h-\tau)}(-0.191sin(f*(h-\tau)) - 0.667cos(f*(h-\tau))) + 0.667 \end{bmatrix}$$

With $f = 1.746$. The closed-loop stability of these can be checked over a range of h and $\tau$ by analyzing the spectral radius of $(F - GK_1^e)$ where $K_1^e = [K, 0]$ to compensate for the new sizes of F and G. The closed loop spectral radius was calculated over a range of h and $\tau$ from zero to ten seconds, ensuring that $\tau$ never crosses h. The spectral radius for all the combinations can be seen in Figure 3:

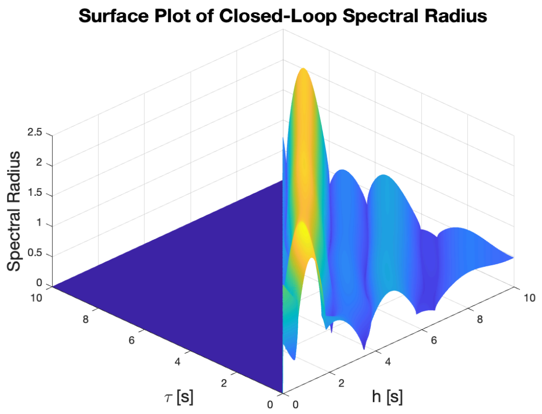



**Figure 3:** Isometric View of Spectral Radius Surface Plot

For clarity, a heatmap was generated where all unstable values are black, which can be seen in Figure 4:

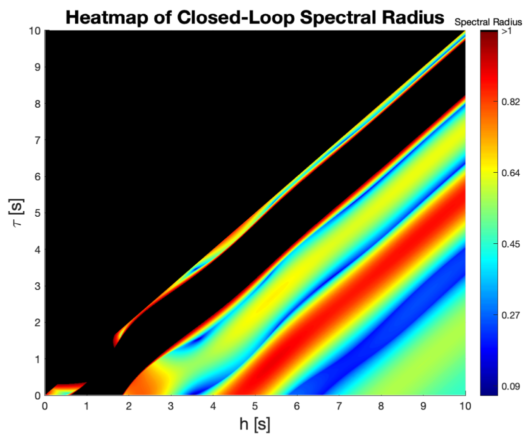



**Figure 4:** Heatmap of Closed-Loop Spectral Radius with Respect to h and $\tau$

To examine the stability of the smaller delays and sampling times, which are more realistic for modern systems, a plot was generated in Figure 5 showing the spectral radius for sampling times and delays between 0 and 3 seconds:

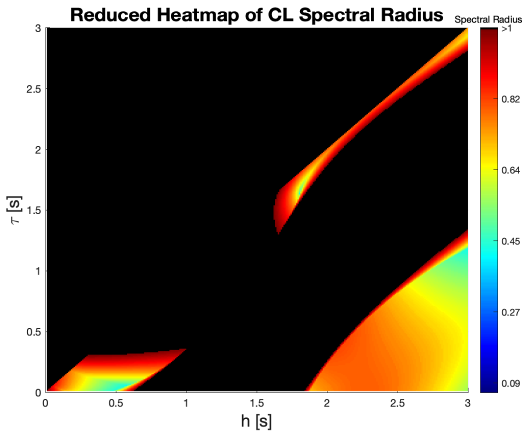



**Figure 5:** Reduced Heatmap of Closed-Loop Spectral Radius with Respect to h and $\tau$ to Show Finer Details

Clearly, the system starts stable for all delays, but around h=-0.2s, high delays cause instability of the system. After a diagonal patch running at about $\tau$=h-1, the system becomes stable again around h=2s.

## 2.2: Improved Controller Design

For controller design, only the case where h=0.4s was examined, as it is stable if $\tau$=0. The controller design focused on expanding the range of stable $\tau$ values while keeping sampling time constant, as well as incorporating the extended state $u_{k-1}$ via the final element of $K^e$. The final controller is $K_2^e = [2, -0.5, 0.25]$. A comparison of the old and new controller over a range of $\tau$ can be seen in Figure 6:

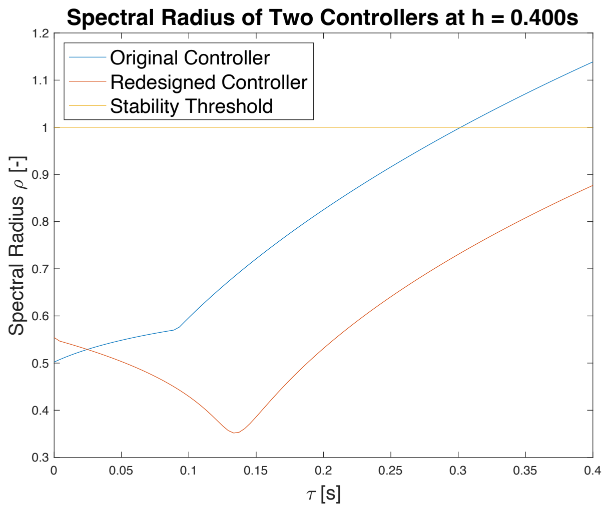



**Figure 6:** Comparison of Stability Range for $K_1^e$ and $K_2^e$

Clearly, $K_2^e$ retains stability for all delays from 0 to 0.4s, whereas $K_1^e$ destabilizes around $\tau \geq 0.28s$.

# Question 3: Extended State Controller with Delayed Inputs

Now, the controller will be changed to an extended-state controller defined as $u_{k-1} = -\bar{K}x_{k-1} - \bar{K}_2 x_{k-2}$. $\bar{K}$ is the same controller as the previous section, and $\bar{K}_2$ is found by using the `place()` function in MATLAB to place the roots of the closed-loop system, with the A and B defined in the introduction, at -1 and -3. After the pole-placement procedure, $K_2$ is found to be $[3.0, \ -1.5]$.

## 3.1: Extended State-Space Model

To solve for the new extended system, recall Equation 11:

$$x_{k+1} = F_x(h)x_k + F_u(h,\tau)u_{k-1} + G_u(h,\tau)u_k$$

Plugging in the controller equation results in the new equation:

$$x_{k+1} = F_x(h)x_k + F_u(h,\tau)(-\bar{K}x_{k-1} - \bar{K}_2x_{k-2}) + G_u(h,\tau)(-\bar{K}x_k - \bar{K}_2x_{k-1}) \tag{13}$$

Which can be simplified to:

$$x_{k+1} = (F_x(h) - G\bar{K})x_k + (-F_u(h,\tau)\bar{K} - G(h,\tau)\bar{K}_2)x_{k-1} + (-F_u(h,\tau)\bar{K}_2)x_{k-2} \tag{14}$$

The extended state is defined as $x_k^e = [x_k^T, x_{k-1}^T, x_{k-2}^T]^T$. The new system can be defined as:

$$x_{k+1}^e = \begin{bmatrix} x_{k+1} \\ x_k \\ x_{k-1} \end{bmatrix} = \begin{bmatrix} (F_x(h) - G\bar{K}) & (-F_u(h,\tau)\bar{K} - G(h,\tau)\bar{K}_2) & (-F_u(h,\tau)\bar{K}_2) \\ I & 0 & 0 \\ 0 & I & 0 \end{bmatrix} * \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \end{bmatrix}$$

Where $F_x(h), F_u(h,\tau)$ and $G_u(h,\tau)$ the final equations are the same as in section 2.

## 3.2: Stability Regions and Comparison

With the new controller, the same process for 2.1 was repeated to find a heatmap of spectral radius for a range of sampling times h and delays $\tau$. The heatmap can be seen in Figure 7:

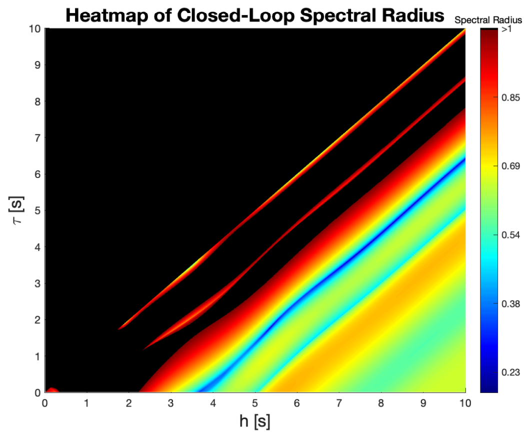



**Figure 7:** Heatmap of Closed-Loop Spectral Radius with Respect to h and $\tau$ for extended controller. All unstable values are black.

Comparing Figures 4 and 7, it is clear that the general structure is similar, with diagonal strips of interchanging stable and unstable regions. The extended controller has a much more restricted section of stability at lower delays and sampling times, though, which is problematic for applications where inter-sample behavior has to be strictly constrained over short periods of time.

## 3.3: Controller Redesign for $\tau = h/2$

In order to address this issue, a new $\bar{K}_2$ is optimized to maximize the range of stable sampling times if it is assumed that $\tau = \frac{h}{2}$. This optimization was done by a simple iteration over the space of $\bar{K}_2$ elements. The optimal controller is $\bar{K}_2 = [0.4,\ 0.6]$, which is stable over 86.3% of the sampling times tested. The plot of spectral radius over the range of sampling times with the new controller can be seen in Figure 8:

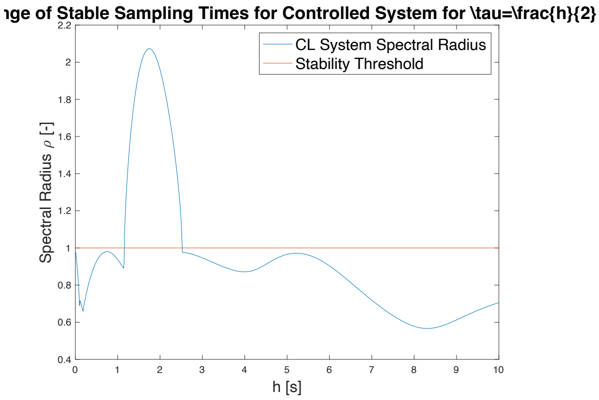



**Figure 8:** Spectral Radius of System with New Extended Controller

# Question 4: Varying Sampling Interval and LMIs for Stability

## 4.1: General LMI Conditions

Generally, for a system with a closed-loop form of $x_{k+1} = A(h)x_k$ where $A(h) = (F(h) - G(h)K)$, the set of LMIs defining the stability of this system, with no knowledge of the h, would be:

1. There exists some $P = P^T \succ 0$ such that
2. $A^T PA - P \preceq -Q$
3. For some: $Q = Q^T \succ 0$

These must hold for all h's since we have no knowledge of h at all. In the case where, for h1, a controller K1 is used and for h2 a controller K2 is used, we can define:

$$A_1 = (F(h_1) - G(h_1)K_1)$$

$$A_2 = (F(h_2) - G(h_2)K_2)$$

The new set of LMIs to verify the stability of the system is:

1. There exists some: $P = P^T \succ 0$ such that
2. $A_1^T PA_1 - P \preceq -Q$ and
3. $A_2^T PA_2 - P \preceq -Q$
4. For some: $Q = Q^T \succ 0$

Because both the switching order and value of $h_1$ and $h_2$ are unknown.

## 4.2: Simplification with Restricted Sequences

This situation can be significantly simplified for repeating sequences. For the sequence $(h_1h_2)^\omega$, the system repeats after two timesteps, and the equation for the system in two timesteps is:

$$x_{k+2} = A_1(h_1)A_2(h2)x_k$$

where $A_1 = (F(h_1) - G(h_1)K_1)$ and $A_2 = (F(h_2) - G(h_2)K_2)$. Determining the stability of this two-timestep system is sufficient to characterize the stability, so a simple analysis can be conducted by calculating the spectral radius of the two-step system for varying $h_1$ and $h_2$ values. The plot of spectral radius for varying $h_1$ and $h_2$ values can be seen in Figure 9.

It is also possible, instead of evaluating the spectral radius condition, to evaluate the LMIs:

1. There exists some $P = P^T \succ 0$ such that:
2. $(A_1(h_1)A_2(h_2))^T P(A_1(h_1)A_2(h_2)) - P \preceq -Q$

3. For some $Q = Q^T \succ 0$

For the case where the delays are given by $(h_1)^\omega$, the solution is even simpler, as this represents an infinitely repeating sequence of the same sampling time, which is the same as what was checked in section 1. The spectral analysis can be done on the system $x_{k+1} = Ax_k$ where $A = (F(h_1) - G(h_1)K_1)$. Stability could also be evaluated using the LMIs:

1. There exists some $P = P^T \succ 0$ such that:
2. $(A(h_1)^T PA(h_1)) - P \preceq -Q$
3. For some $Q = Q^T \succ 0$

## 4.3: Stability Analysis of Infinite Sampling Time Repetition

For the case where the sequence is given by $(h_1 h_2)^\omega$ the spectral radius analysis can be seen in Figure 9:

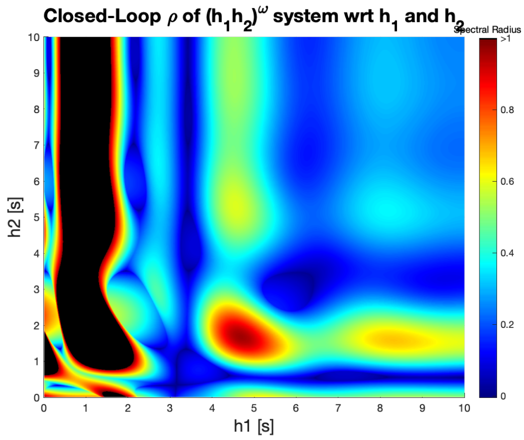



**Figure 9:** Spectral Radius of Various $h_1$, $h_2$ combinations for $(h_1 h_2)^\omega$. All unstable values are black.

For the case where the sequence is given by $(h_1)^\omega$ the spectral radius analysis can be seen in Figure 10:

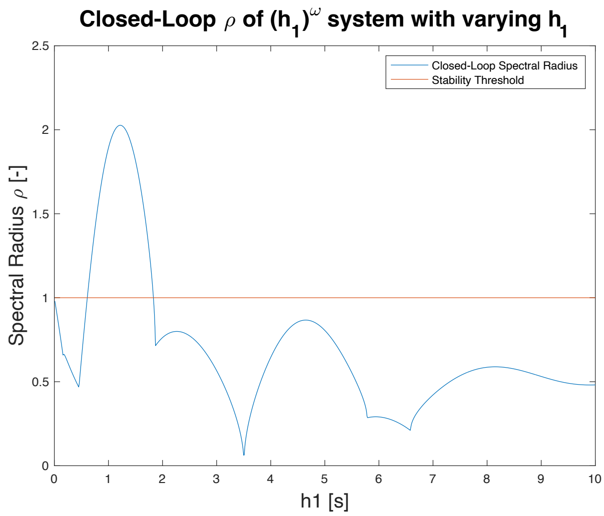
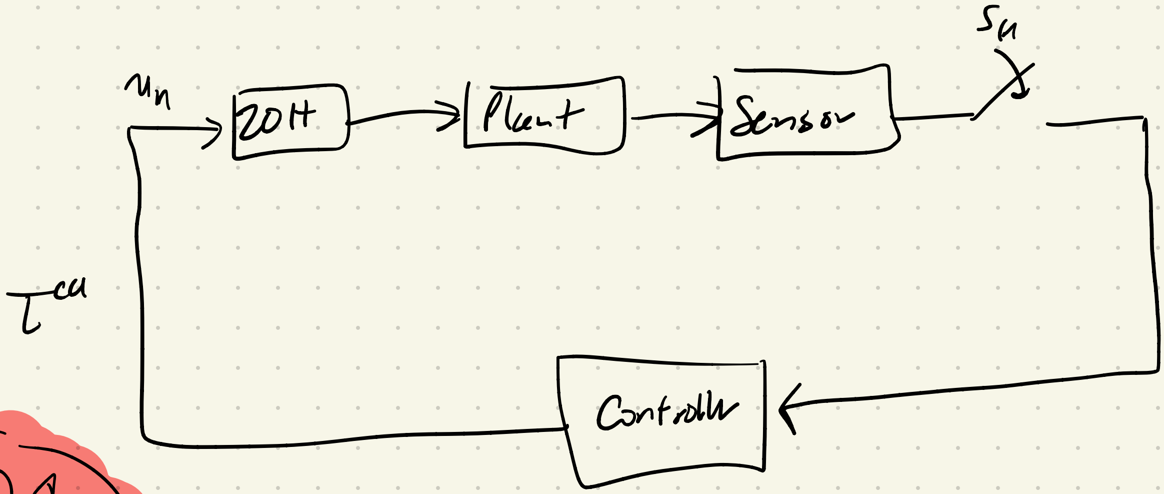



**Figure 10:** Spectral Radius of Various $h_1$ values for $(h_1)^\omega$

# Appendix A: Derivation Details




# Q1

$$A = \begin{bmatrix} 0 & -1.5 \\ 2.2 & -1.0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad K = \begin{bmatrix} k_1 & k_2 \end{bmatrix}$$

$$\dot{\xi}(t) = A\,\xi(t) + B\,\nu(t) \qquad \nu(t) = K\,\xi(t)$$

$$\dot{\xi} = (A - KB)\,\xi$$

$$A - BK = \begin{bmatrix} 0 & -1.5 \\ 2.2 & -1.0 \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \end{bmatrix}\begin{bmatrix} k_1 & k_2 \end{bmatrix} = \begin{bmatrix} 0 - k_1 & -1.5 - k_2 \\ 2.2 & -1.0 \end{bmatrix}$$

$$= \begin{bmatrix} -k_1 & -1.5 - k_2 \\ 2.2 & -1.0 \end{bmatrix}$$

get characteristic eqn

$$\left| \begin{bmatrix} -k_1 - \lambda & -1.5 - k_2 \\ 2.2 & -1 - \lambda \end{bmatrix} \right| = (-k_1 - \lambda)(-1 - \lambda) - (2.2(-1.5 - k_2))$$

$$= \lambda^2 + k_1\lambda + \lambda + 2.2 \cdot 1.5 + 2.2\,k_2 + k_1$$

$$= \lambda^2 + (k_1+1)\lambda + (2.2k_2 + 3.3 + k_1)$$

want eigenvalues @ $-2 \pm 1j$ : $\lambda^2 + 4\lambda + 5$

$k_1 + 1 = 4 \quad k_1 = 3$

$2.2k_2 + 3.3 + k_1 = 5$

$$k_2 = \frac{5 - 3.3 - 3}{2.2}$$

Getting $F(h)$ & $G(h)$

$F(h) = e^{Ah} \quad A = \begin{bmatrix} 0 & -1.5 \\ 2.2 & -1.0 \end{bmatrix}$

$$\boxed{\begin{matrix} \text{eigenvalue } -0.5 \pm 1.7464j \\ a = -0.5 \quad b = 1.7464 \end{matrix}}$$

Complex Eigenvalue Form of Matrix Exponential (From The U)

For eigenvalues $\lambda_1 = a + bi$

$$e^{At} = e^{at}\left(\cos bt\, I + \frac{\sin bt}{b}(A - aI)\right)$$

$$e^{Ah} = e^{-0.5h}\left(\begin{pmatrix} \cos(1.7464h) & 0 \\ 0 & \cos(1.7464h) \end{pmatrix} + \frac{\sin(1.7464h)}{1.7464h}\left(\begin{bmatrix} 0 & -1.5 \\ 2.2 & -1 \end{bmatrix} - \begin{bmatrix} -0.5 & 0 \\ 0 & -0.5 \end{bmatrix}\right)\right)$$

$$= \text{''} \quad \text{''}\left(\begin{bmatrix} \cos(1.7464h) & 0 \\ 0 & \cos(1.7464h) \end{bmatrix} + \frac{\sin(1.7464h)}{1.7464}\left(\begin{bmatrix} 0.5 & -1.5 \\ 2.2 & -0.5 \end{bmatrix}\right)\right)$$

$$= e^{-0.5h}\left(\begin{bmatrix} \cos(1.7464h) & 0 \\ 0 & \cos(1.7464h) \end{bmatrix} + \begin{bmatrix} \frac{0.5\sin(1.7464h)}{1.7464} & \frac{-1.5\sin(1.7464h)}{1.7464} \\ \frac{2.2\sin(1.7464h)}{1.7464} & \frac{-0.5\sin(1.7464h)}{1.7464} \end{bmatrix}\right)$$

say $\alpha = 1.7464h$

$$= e^{-0.5h}\begin{bmatrix} \cos(\alpha) + \frac{0.5}{1.7464}\sin\alpha & -\frac{1.5}{1.7464}\sin(\alpha) \\ \frac{2.2\sin(\alpha)}{1.7464} & \cos(\alpha) - \frac{0.5}{1.7464}\sin(\alpha) \end{bmatrix} =$$

$$F(h) = e^{-0.5h}\begin{bmatrix} \cos(1.7464h) + 0.2863\sin(1.7464h) & -0.859\sin(1.7464h) \\ 1.26\sin(1.7464h) & \cos(1.7464h) - 0.2863\sin(1.7464h) \end{bmatrix}$$

Double Checked w/ MATLAB c2d ✓

$$G(h) = \int_0^h e^{As} B\,ds \qquad \text{using} \quad \tilde{A} = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \quad e^{\tilde{A}h} = \begin{bmatrix} F(h) & G(h) \\ 0 & 1 \end{bmatrix}$$

F(h) works properly ✓

Output of Matlab Symbolic Math

Matlab expm function

$$G(h) = \Big[\, 0.303 - e^{(t\cdot(-0.5+1.7464i))}(0.1515+0.2429i) - e^{(-0.5-1.7464i)t}(0.1515-0.2429i)$$

Lets give Simplification a Shot:

$$\Big(0.1515e^{-0.5t}e^{1.7464it} + 0.2429ie^{-0.5t}e^{1.7464it} + 0.1515e^{-0.5t}e^{-1.7464it} - 0.2429ie^{-0.5t}e^{i1.7464it}$$

$$0.1515e^{-0.5t}\left(e^{1.7464it} + e^{-1.7464it}\right) + .2429ie^{-0.5t}\left(e^{1.7464it} - e^{-1.7464it}\right)$$

$$0.1515e^{-0.5t}\left(2\cos(1.7464t)\right) + .2429ie^{-0.5t}\left(2i\sin(1.7464it)\right)$$

$$0.303e^{-0.5t}\cos(1.7464t) - 0.4858e^{-0.5t}\sin(1.7464it)$$

$$G_1 = 0.303 - \left(0.303e^{-0.5t}\cos(1.7464t) - 0.4858e^{-0.5t}\sin(1.7464it)\right) ✓$$

$$G_2 = 0.667 - \left(e^{(t\cdot(-0.5+1.7464i))}(0.3333-0.0954i) + e^{(t\cdot(-0.5-1.7464i))}(0.3333+0.0954i)\right)$$

$$0.333e^{-0.5t}e^{1.7464it} - 0.0954ie^{-0.5t}e^{1.7464it} + 0.333e^{-0.5t}e^{-1.7464it}$$
$$+ 0.0954ie^{-0.5t}e^{-1.7464it})$$

$$0.333e^{-0.5t}\left(e^{1.7464it} + e^{-1.7464it}\right) + 0.0954ie^{-0.5t}\left(e^{-1.7464it} - e^{1.7464it}\right)$$

$$= 0.333e^{-0.5t}(2\cos(1.7464t) + 0.0954ie^{-0.5t}(-2i\sin(1.7464t))$$

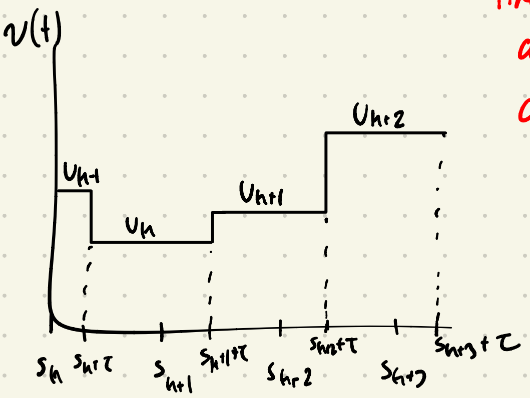
$$G_2 = e^{-0.5t}(0.6667\cos(1.7464t) + 0.1908\sin(1.7464t))$$

---

## Q2 Now looking @ NCS w/ delays

delay: $\tau = \sum \tau_i$

Look @ Graph of inputs




time delay means that actuator inputs are delayed,

From interval $s_k \to s_{k+1}$

$s_k \to s_k + \tau : v(t) = u_{k-1}$

$s_k + \tau \to s_{k+1} : v(t) = u_k$

Recall From last Part

Break into two pieces:

$$x_{k+1} = \xi(s_{k+1}) = e^{Ah}x_k + \int_{s_k}^{s_{k+1}} e^{A(s_{k+1}-s)}Bv(s)\,ds$$

$$x_{k+1} = e^{Ah}x_k + \left(\int_{s_k}^{s_k+\tau} e^{A(s_{k+1}-s)}B\,ds\right)u_{k-1} + \left(\int_{s_k+\tau}^{s_{k+1}} e^{A(s_{k+1}-s)}B\,ds\right)u_k$$

Change variables on integrals

$$x_{k+1} = e^{Ah}x_k + \left(\int_{h-\tau}^{h} e^{As}B\,ds\right)u_{k-1} + \left(\int_{0}^{h-\tau} e^{As}B\,ds\right)u_k$$

$$F_x(h) = \underbrace{e^{Ah}}_{\text{F is the Same as before}} \qquad F_u(h,\tau) = \int_{h-\tau}^{h} e^{As} B\,ds \qquad G_1(h,\tau) = \int_0^{h-\tau} e^{As} B\,ds$$

how tf to do $\int e^{As} B\,ds$

$$F_x(h) = e^{-0.5h} \begin{bmatrix} \cos(1.7464h) + 0.2863\sin(1.7464h) & -0.859\sin(1.7464h) \\ 1.26\sin(1.7464h) & \cos(1.7464h) - 0.2863\sin(1.7464h) \end{bmatrix}$$

$$F_u(h) = \int_{h-\tau}^{h} F_x(h) B\,ds$$

$$\int_{h-\tau}^{h} e^{-0.5s} \begin{bmatrix} \cos(1.7464s) + 0.2863\sin(1.7464s) & -0.859\sin(1.7464s) \\ 1.26\sin(1.7464s) & \cos(1.7464s) - 0.2863\sin(1.7464s) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} ds$$

$$= \int_{h-\tau}^{h} e^{-0.5s} \begin{bmatrix} \cos(1.7464s) + 0.2863\sin(1.7464s) \\ 1.26\sin(1.7464s) \end{bmatrix} ds$$

$\Downarrow$ wolfram Alpha

keep in mind for $G_1$

$$= e^{-0.5s} \begin{bmatrix} 0.486\sin(1.7464s) - 0.303\cos(1.7464s) \\ -0.191\sin(1.7464s) - 0.667\cos(1.7464s) \end{bmatrix} \Bigg|_{h-\tau}^{h}$$

$$= e^{-0.5h} \begin{bmatrix} 0.486\sin(1.7464h) - 0.303\cos(1.7464h) \\ -0.191\sin(1.7464h) - 0.667\cos(1.7464h) \end{bmatrix} - e^{-0.5(h-\tau)} \begin{bmatrix} 0.486\sin(1.7464h-\tau) - 0.303\cos(1.7464h) \\ -0.191\sin(1.7464(h-\tau)) - 0.667\cos(1.7464h) \end{bmatrix}$$

$$
= \underbrace{\begin{bmatrix} e^{-0.5h}\left(0.486\sin(1.7464h) - 0.303\cos(1.7464h)\right) - e^{-0.5(h-\tau)}\left[0.486\sin(1.7464(h-\tau)) - 0.303c(1.7464(h-\tau))\right) \\ e^{-0.5h}\left(-0.191\sin(1.7464h) - 0.667\cos(1.7464h) - e^{-0.5(h-\tau)}\left(-0.191\sin(1.7464(h-\tau)) - 0.667\cos(1.7464h)\right)\right. \end{bmatrix}}_{F_u(h,\tau)}
$$

$$
G_1(t) = \int_0^h e^{-0.5s} \begin{bmatrix} \cos(1.7464s) + 0.2863\sin(1.7464s) \\ 1.26\sin(1.7464s) \end{bmatrix} ds
$$

$$
= e^{-0.5s} \begin{bmatrix} 0.486\sin(1.7464s) - 0.303\cos(1.7464s) \\ -0.191\sin(1.7464s) - 0.667\cos(1.7464s) \end{bmatrix} \Bigg|_0^{h-\tau}
$$

$$
= e^{-0.5(h-\tau)} \begin{bmatrix} 0.486\sin(1.7464h-\tau) - 0.303\cos(1.7464h-\tau) \\ -0.191\sin(1.7464h-\tau) - 0.667\cos(1.7464h-\tau) \end{bmatrix} - 1 \begin{bmatrix} -0.303 \\ -0.667 \end{bmatrix}
$$

$$
= \underbrace{\begin{bmatrix} e^{-0.5(h-\tau)}\left(0.486\sin(1.7464h-\tau) - 0.303\cos(1.7464h-\tau) + 0.303\right. \\ e^{-0.5(h-\tau)}\left(-0.191\sin(1.7464h-\tau) - 0.667\cos(1.7464h-\tau) + 0.667\right. \end{bmatrix}}_{G_1(h,\tau)}
$$

Define the extended state

$$x_h^e = \begin{bmatrix} x_h \\ u_{h-1} \end{bmatrix} \qquad F(h,\tau) = \begin{bmatrix} F_x(h) & F_u(h,\tau) \\ 0 & 0 \end{bmatrix}$$

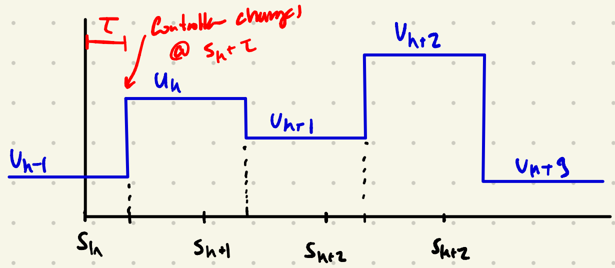
$$G(h,\tau) = \begin{bmatrix} G_1(h,\tau) \\ I \end{bmatrix}$$

---

## Q3

Poles now @ $-1$ & $-3$

$t = s_{h-1} + \tau \in [s_{h-1}, s_h)$




$$u_{h-1} = -\bar{k}\, x(s_{h-1}) - \bar{k}_2\, x(s_{h-2})$$

$$u_h = -k\, x(s_h) - \bar{k}_2\,(x(s_{h-1}))$$

$u_{h-1} =$

Derive System Dynamics

Say we start w/ CT System

$$\dot{x}(t) = Ax + Bu$$

Solution

$$\xi(t) = e^{A(t-t_0)}\xi(t_0) + \int_0^t e^{A(t-s)} B\nu(s)\, ds$$

@ $s_{h+1}$

$$x_{h+1} = \xi(s_{h+1})$$

Due to time delay

$$\nu(t) = u_{h-1} \quad \text{for } t \in [s_h, s_h+\tau)$$

$$\nu(t) = u_h \quad \text{for } t \in [s_h+\tau, s_{h+1})$$

$$x_{h+1} = e^{Ah} x_h + \int_{s_h}^{s_{h+1}} e^{A(s_{h+1}-s)} B u\, ds$$

Split the Integral

$$= e^{Ah} x_h + \int_{s_h}^{s_h+\tau} e^{A(s_{h+1}-s)} B u_{h-1}\, ds + \int_{s_h+\tau}^{s_{h+1}} e^{A(s_{h+1}-s)} B u_h\, ds$$

Take $u_s$ out of integrals

$$x_{k+1} = e^{Ah}x_k + \left(\int_{s_k}^{s_k+\tau} e^{A(s_{k+1}-s)}B\,ds\right)u_{k-1} + \left(\int_{s_k+\tau}^{s_{k+1}} e^{A(s_{k+1}-s)}B\,ds\right)u_k$$

$$= e^{Ah}x_k + \left(\int_{h-\tau}^{h} e^{Ah}B\,ds\right)u_{k-1} + \left(\int_0^{h-\tau} e^{Ah}B\,ds\right)u_k \Big\} \text{ Expressions in terms of } A, B, h, \tau$$

now plug in $u_{k-1} = -\bar{K}_1 x_{k-1} - K_2 x_{k-2}$

$u_k = -\bar{K}_1 x_k - \bar{K}_2 x_{k-1}$

$$x_{k+1} = e^{Ah}x_k - \left(\int_{h-\tau}^{h} e^{Ah}B\,ds\right)\bar{K}_1 x_{k-1} - \left(\int_{h-\tau}^{h} e^{Ah}B\,ds\right)\bar{K}_2 x_{k-2} - \left(\int_0^{h-\tau} e^{Ah}B\,ds\right)\bar{K}_1 x_k - \left(\int_0^{h-\tau} e^{Ah}B\,ds\right)\bar{K}_2 x_{k-1}$$

$$= F_x(h)x_k - F_u(h,\tau)\bar{K}_1 x_{k-1} - F_u(h,\tau)\bar{K}_2 x_{k-2} - G(h,\tau)\bar{K}_1 x_k - G(h,\tau)\bar{K}_2 x_{k-1}$$

$$= \left(F_x(h) - G(h,\tau)\bar{K}_1\right)x_k + \left(-F_u(h,\tau)\bar{K}_1 - G(h,\tau)\bar{K}_2\right)x_{k-1} + \left(-F_u(h,\tau)\bar{K}_2\right)x_{k-2}$$

Use extended state:

$$x_k^e = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \end{bmatrix}$$

$$x_{k+1}^e = \begin{bmatrix} x_{k+1} \\ x_k \\ x_{k-1} \end{bmatrix} = \overbrace{\begin{bmatrix} F_x(h) - G(h,\tau)K_1 & -F_u(h,\tau)K_1 - G(h,\tau)K_2 & -F_u(h,\tau)\bar{K}_2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}}^{\text{Closed-loop } \tilde{A}(h,\tau)} \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \end{bmatrix}$$

---

④ LMIs

① ✓

Origin of system is Globally Exponentially Stable if there is some $P = P^T \succ 0$ such that $(F(h) - G(h)K)^T P (F(h) - G(h)K) \prec -Q$ for some $Q = Q^T \succ 0$ for <u>all h</u>

I.E. if $A = (F(h) - G(h)K)$

b.) Say $A(h_i) = (F(h_i) - G(h_i)K_i)$

$A(h_1) = (F(h_1) - G(h_1)K_1)$

$A(h_2) = (F(h_2) - G(h_2)K_2)$

Case 1 $(h_1 h_2)^\omega$

If you know the sequence repeats every 2 steps, we can analyze the system:

$A_{sys} = A(h_1)A(h_2)$

$$x(k+2) = A(h_1)\,A(h_2)\,x(k)$$
$$= (F(h_1) - G(h_1)K_1)(F(h_2) - G(h_2)K_2)\,x(k)$$

At this point, I can just check the spectral radius, right?

Case 2 $(h_1)^\omega$

- this is just the same as previous conditions analyzed, so a simple spectral radius analysis w/o needing LMIs

$$x_{k+1} = A(h_1)x_k$$

Iterate through $h_1$s to find stability regions.

c.)