Simulating Chronotropic Incompetence and Pacemakers

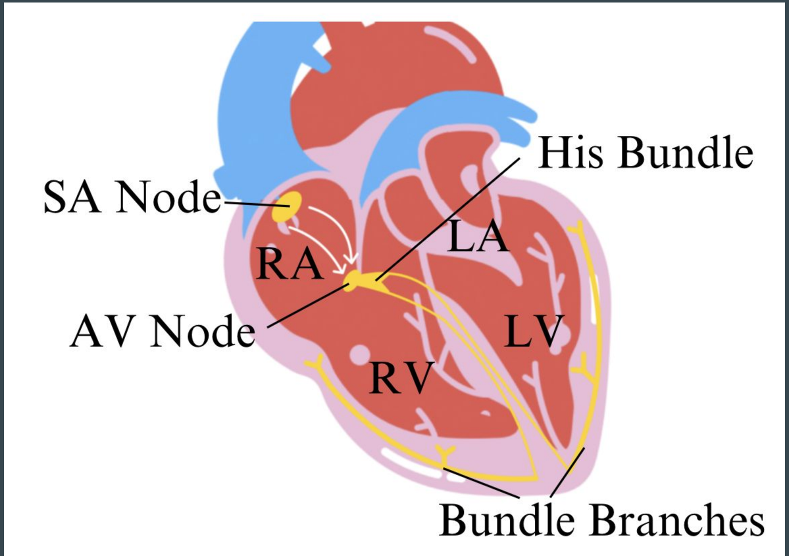
Jasmeet Bhatia, Rohil Deolalikar, Cole Wener, Andrew Zeller, Cameron Zeller

1 - Background

- Chronotropic Incompetence (CI)
- 30 - 50% of patients with HFpEF have this condition
- Caused by overloading of parasympathetic stimuli
- Failure of the SA Node
- Pacemakers act as an artificial SA

1 - Background

- Electrical physiology of the heart
- SA Node → Atria Contraction
- AV Node → Ventricular Contraction
- CI and pacemaker interactions can be simulated as a biosystem

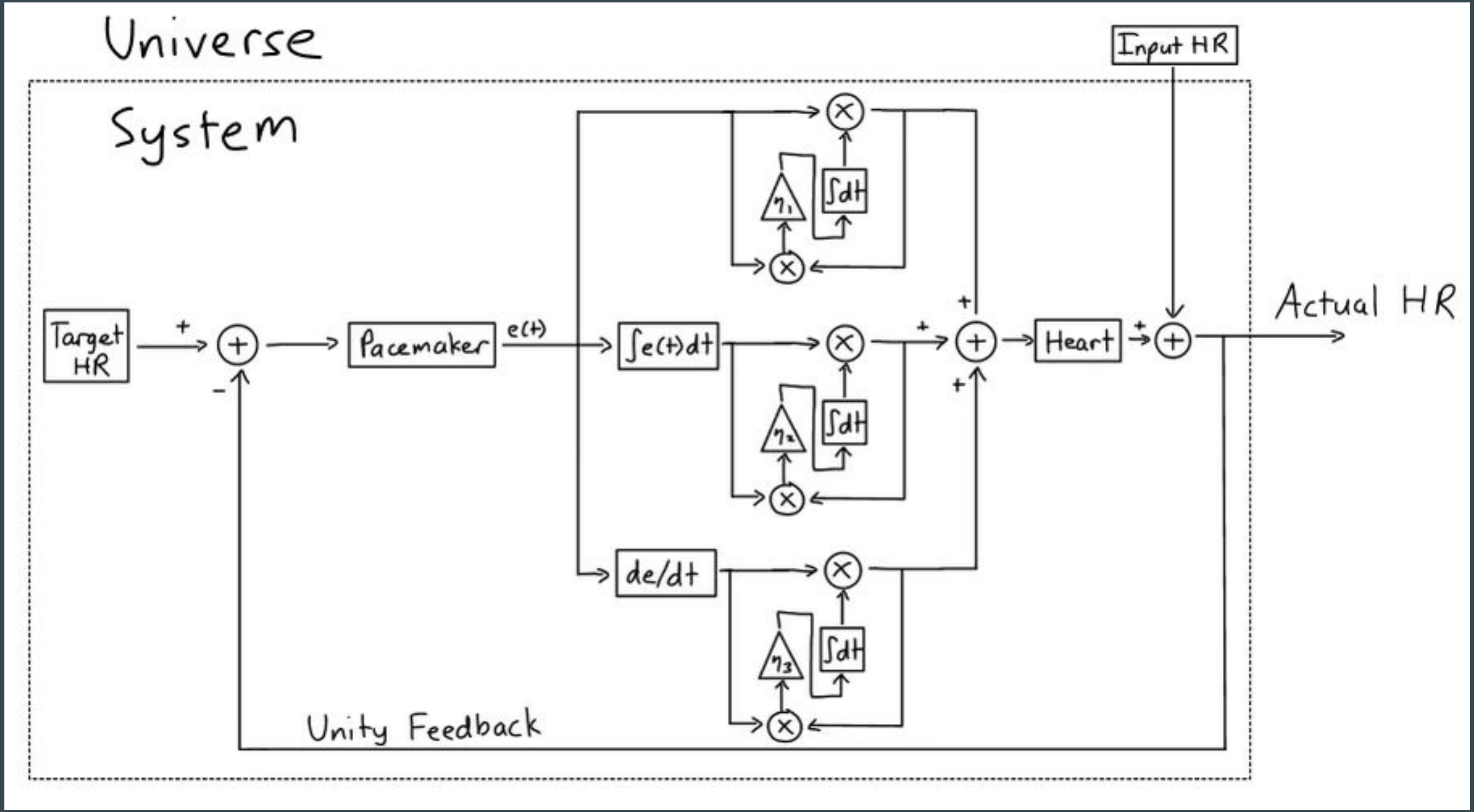


1 - Background

- Our goal for this project was to create an adaptable PID controller that updates itself as time goes on.

- Our Kp, Ki, and Kd values will update to become better over time, using internal feedback loops and delta rule

2 - Time domain block diagram



2 - Time domain equations

$R(t) = \text{target HR}$

$Y(t) = \text{actual HR}$

$I(t) = \text{input HR}$

$$e(t) = R(t) - Y(t)$$

$$G_p(t) = 8e^{-8t}$$

$$G_h(t) = \frac{169}{20}\left(\delta(t) - e^{-20t}\right)$$

$$u(t) = K_p e(t) + K_I \int e(t)\,dt + K_D \frac{de}{dt}$$

$$Y(t) = I(t) + u(t) * G_p(t) * G_H(t) * e(t)$$

$$\begin{cases} K_p = \int \eta_1 e(t) u_1(t)\,dt \\ \quad u_1(t) = K_p(t) e(t) \\ K_I = \int \eta_2 \int e(t)dt \cdot u_2(t)\,dt \\ \quad u_2(t) = K_I(t) \int e(t)\,dt \\ K_D = \int \eta_3 \frac{de}{dt} \cdot u_3(t)\,dt \\ \quad u_3(t) = K_D(t) \cdot \frac{de}{dt} \end{cases}$$

$$u(t) = u_1(t) + u_2(t) + u_3(t)$$

3 - Performance goals and operational constraints

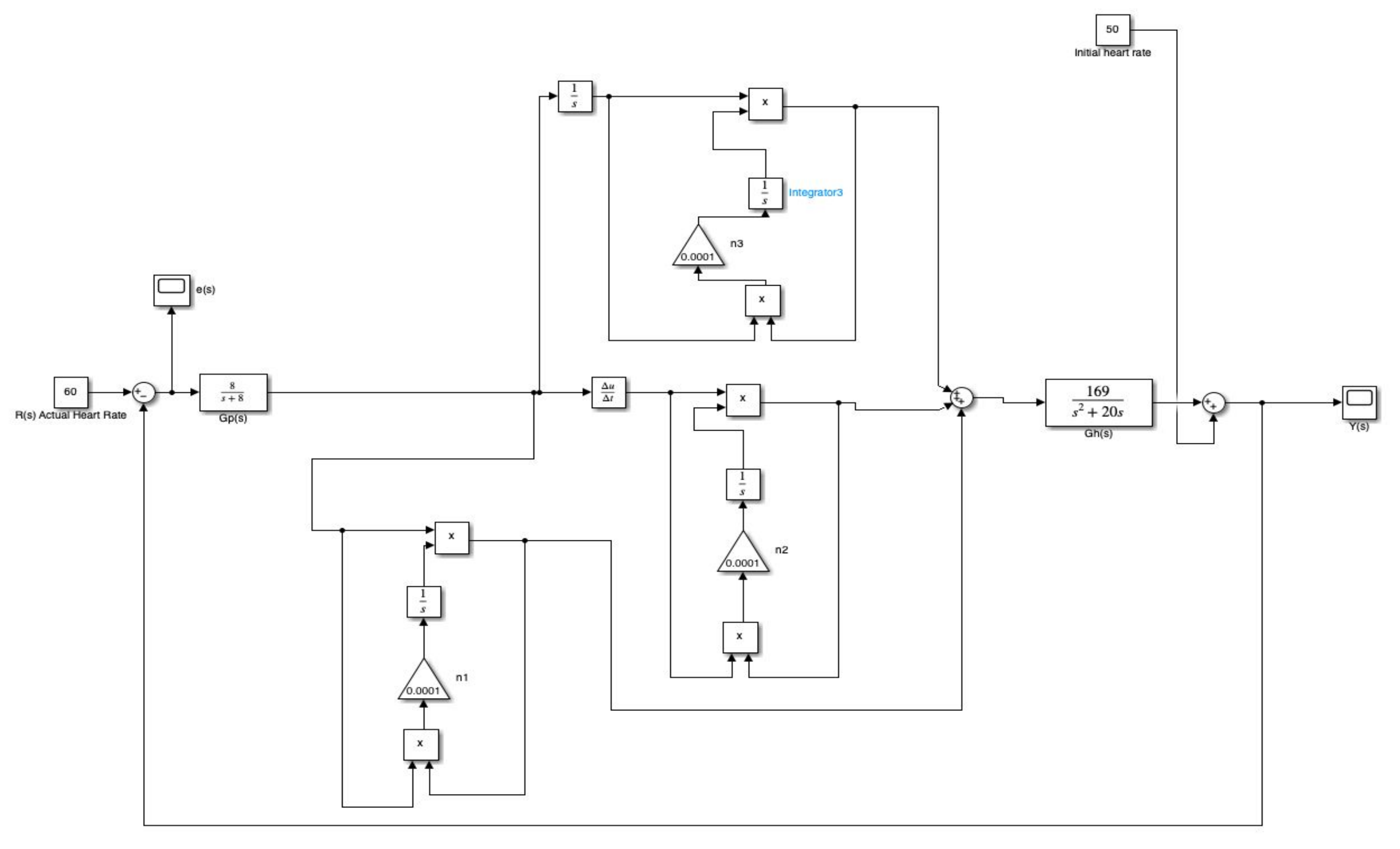
Performance Goals

- Add stability and control to a nonlinear system such as the heart
- Update PID parameters (Kp, Ki, Kd) to provide a more rapid response to increase stability
  - Method uses Delta Rule, MIT rule, Adaptive Correction Factor
- Model cardiac pacemaker and for disease such as Chronotropic Incompetence

Constraints

- System is more sensitive and can be fragile with too high of a learning rate
- Simplified block diagram that does not include nervous system inputs as a variable in block diagram for CI.

4 - Simulink model/Laplace block diagram



5 - Laplace equations

$U_{PID}(s) = K_p + \frac{1}{s} K_i + s K_d$ with $K_p = \frac{1}{s} \eta_1 e(s) u(s)$, $K_i = \frac{1}{s} \eta_2 \left(\frac{1}{s} e(s)\right) u_2(s)$, $K_d = \frac{1}{s} \eta_3 (s e(s)) u_3(s)$ (updated through Delta Rule)

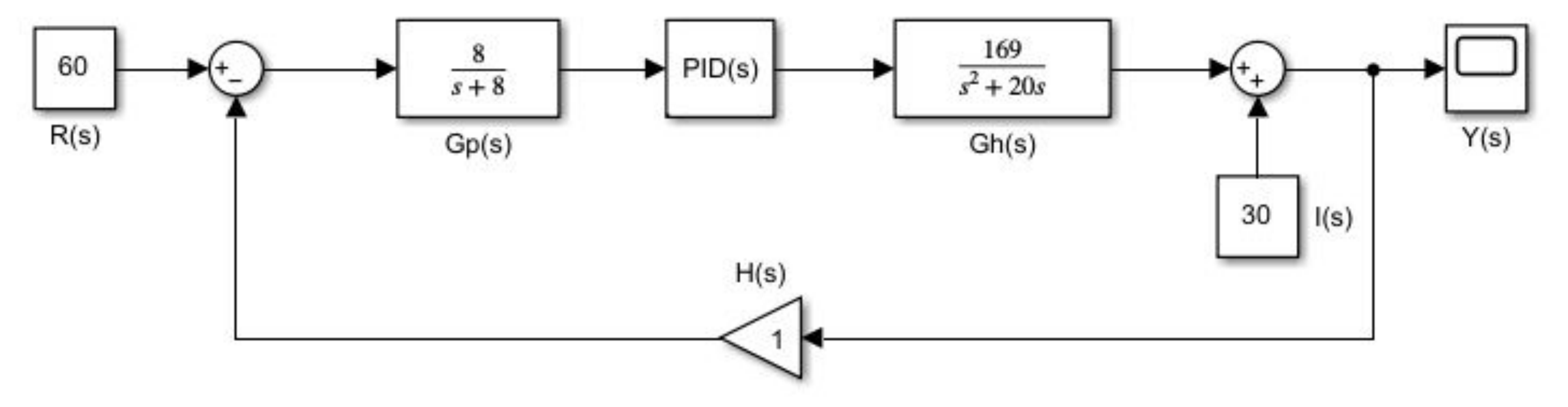
$= \frac{1}{s} \eta_1 e(s) u(s) \oplus \frac{1}{s^3} \eta_2 e(s) u_2(s) \oplus s \eta_3 (e(s)) u_3(s)$ where $u_1(s) = K_p e(s)$ $u_2(s) = \frac{K_i e(s)}{s}$ $u_3 = s \cdot K_d e(s)$

$$OL(s) = \left[\frac{8}{s+8}\right] \cdot \left[\frac{s^3 \eta_1 e^2 K_p + \eta_2 e^2 K_i + s^6 \eta_3 e^2 K_d}{s^4}\right] \cdot \left[\frac{169}{s^2 + 20s}\right]$$

$K_p K_d K_i$ $$OL(s) = \frac{.7152 s^6 + 4.056 s^3 + 5.753}{s^7 + 28 s^6 + 160 s^5}$$

& $K_p K_d$ $$OL(s) = \frac{.7152 s^3 + 4.056}{s^4 + 28 s^3 + 160 s^2}$$

6 - Simplified block diagram



6 - Laplace transfer function

$$Y(s) = I(s) + e(s)\,G_p(s)\,u(s)\,G_H(s)$$

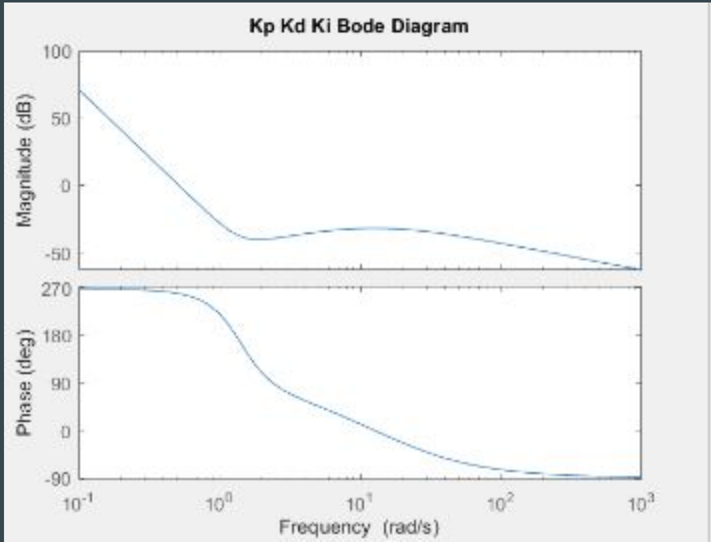
$$e(s) = R(s) - Y(s)$$

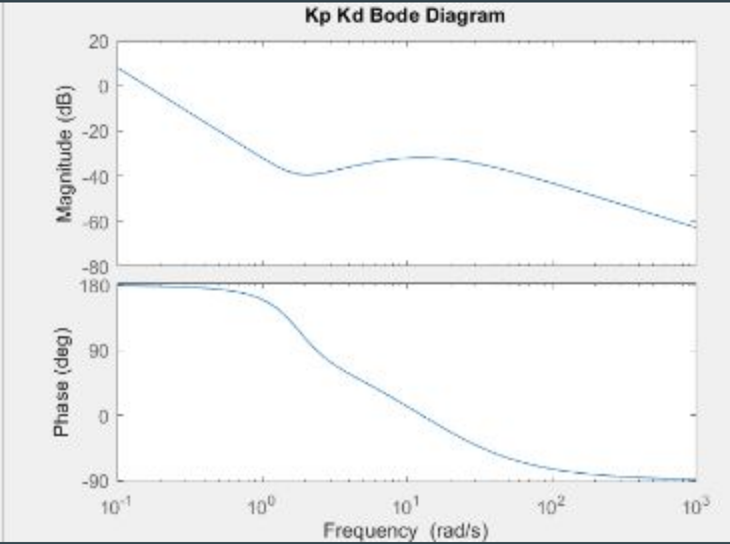
$$G_p(s) = \frac{8}{s+8} \qquad u(s) = K_p + K_I \frac{1}{s} + K_D s$$

$$G_H(s) = \frac{169}{s^2+20s}$$

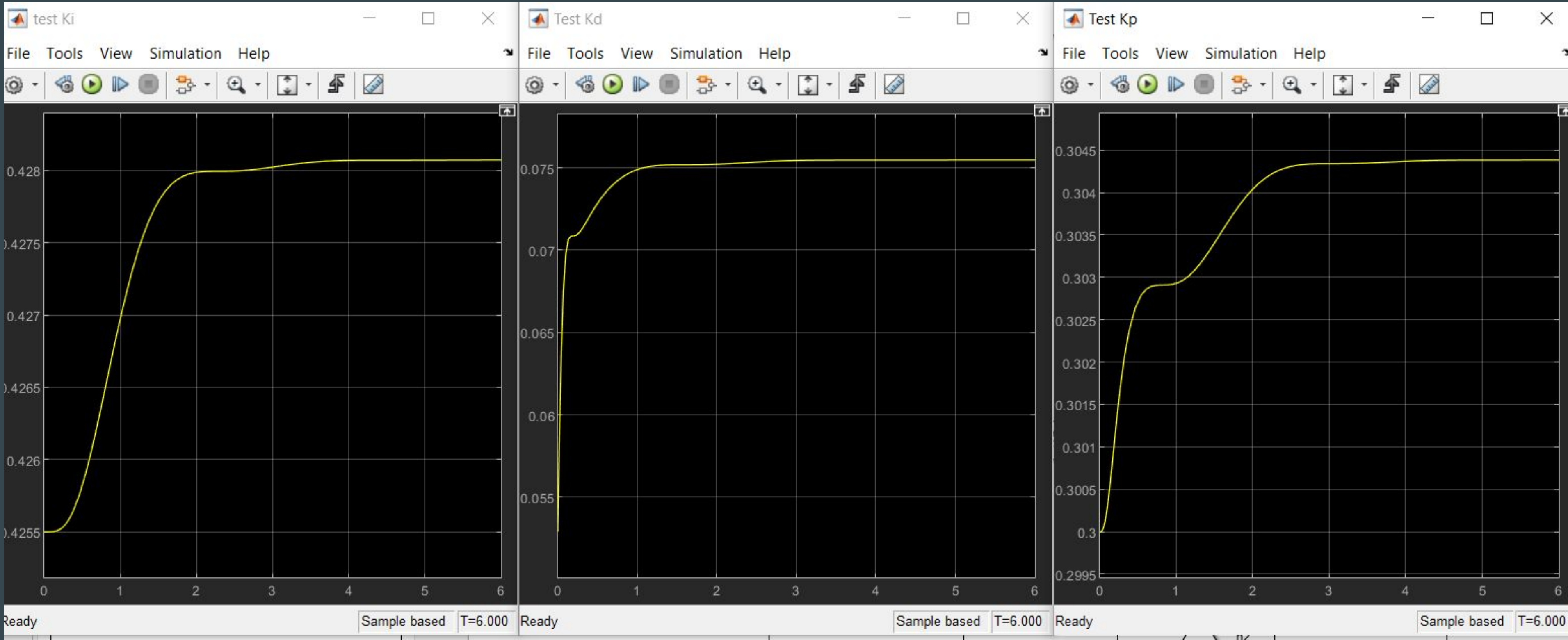
7 - Sensitivity Analysis

To detect stability of the overall system, 2 Bode plots were created. In figure(), we see the bode plot of the PD controller. The PD controller has zeros at S=-20,S=-8,and S=0 signifying that the controller is stable for all poles. This implies that the controllers response to perturbations or disturbances will settle over time. When just looking at KP, KD parameters, we get a gain margin of 0 dB, and a phase margin of -1.638 degrees which do not indicate high stability. The second bode plot looks at when we have a PID controller. Looking at figure (), we see a gain margin of 67 dB and a phase margin of 79 degrees. These values suggest a more robust stability for the system.





A look at the updating KP KD KI parameter values



KP KD KI

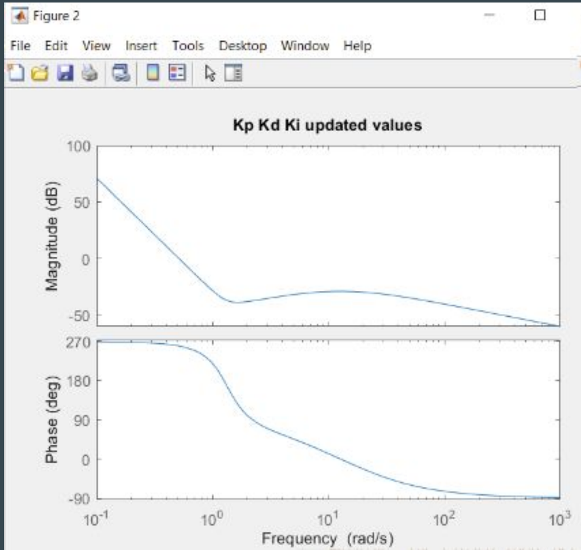
Gain=60.57 dB

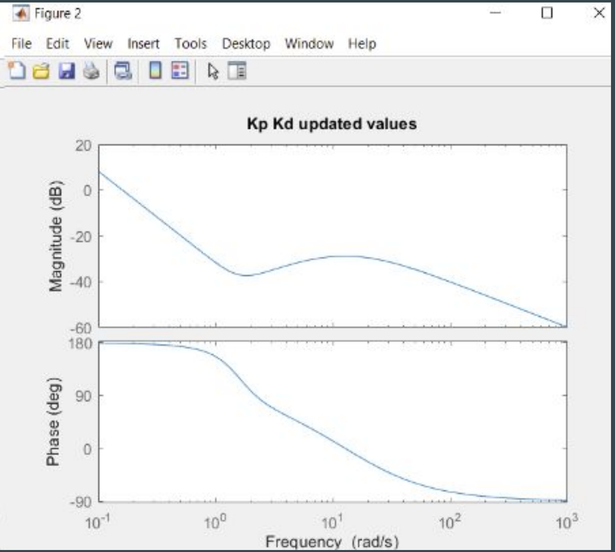
Phase Margin= 79.27

KP KD

Gain=0 dB

Phase Margin= -1.633





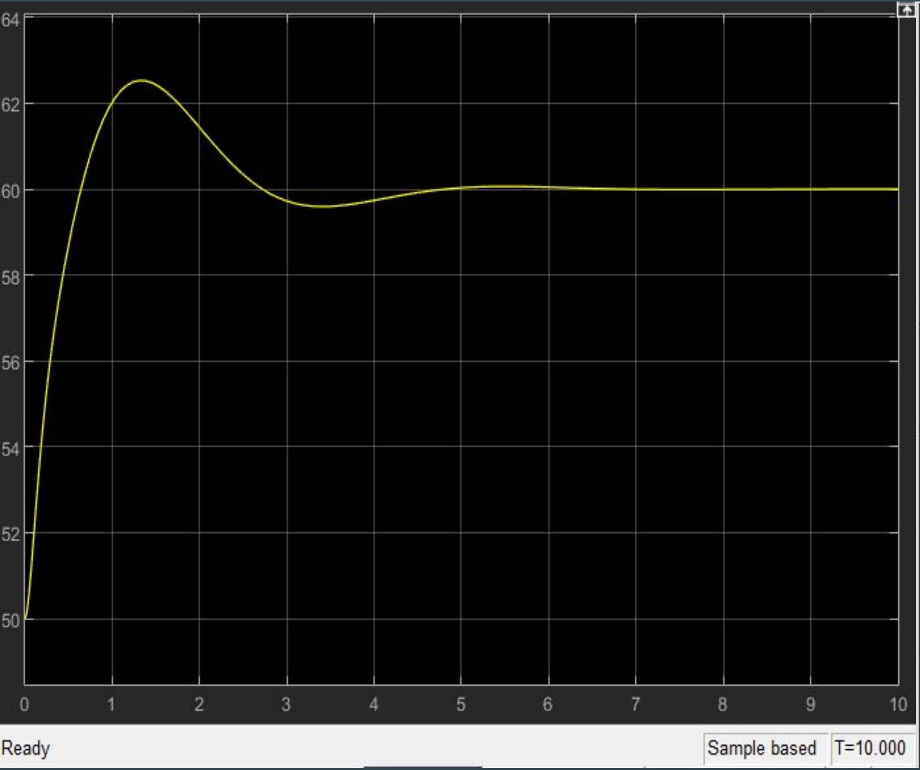
8

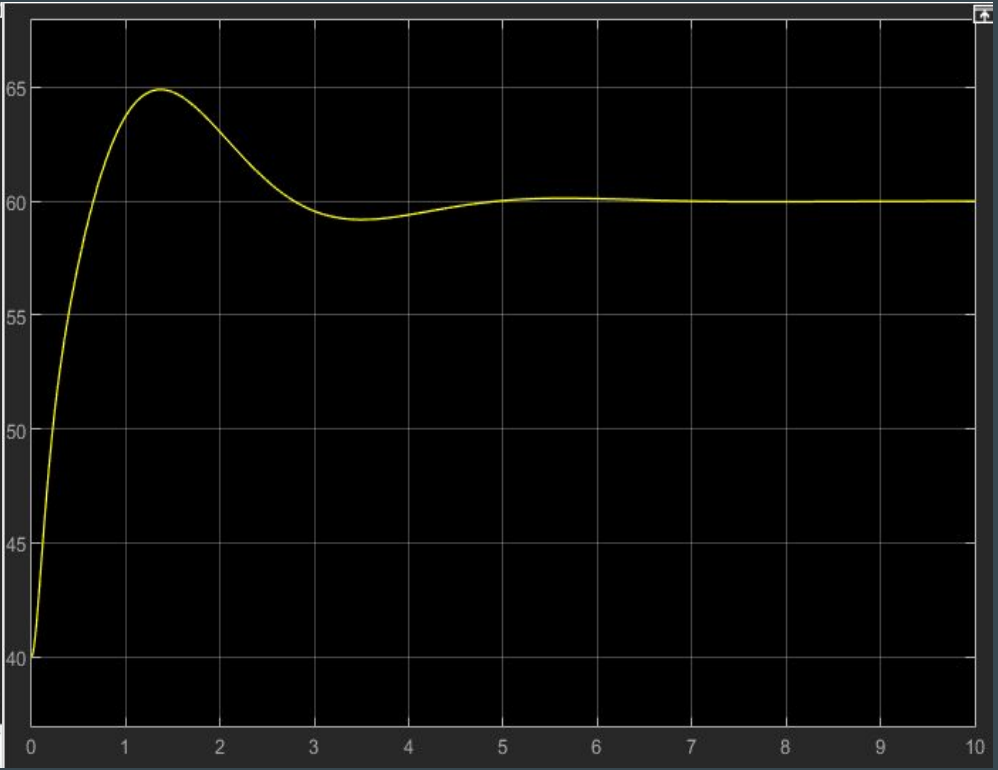
Some errors include our system being extremely similar to the regular PID controls system box in simulink. This could be due to the fact that we are running ideal parameters for Kp,Kd,Ki as found through the Ziegler Nichols Method, and our learning rates are very low. Low learning rates imply the model is making small adjustments as to reduce loss. Low rates can also cause more stability but decrease the convergence of the training process.

Additionally we see similar settle times in our graphs which can be beneficial to showing our how quickly our method works, but also adds potentially dangerous over and undershoots to get to the settle time just as quick when looking at different initial inputs. Wrong initially tuned learning rates can alter our output drastically as well.

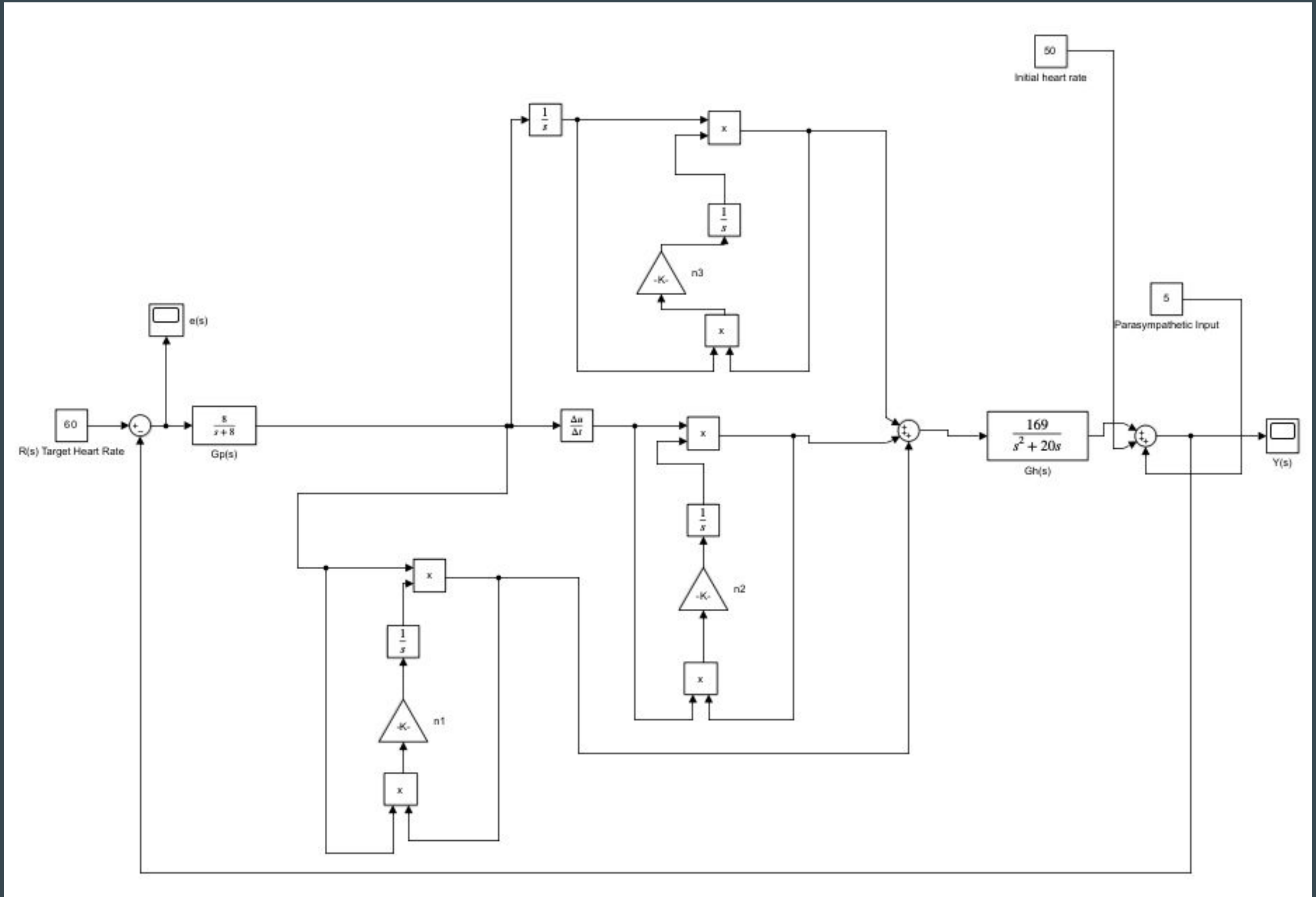
Scope of Regular Actual Heart Rate on ideal PID parameters and tuned learning rates

Scope of CI actual Heart Rate with ideal PID parameters and higher learning rates





10



50
Initial heart rate
5
Parasympathetic Input
60
R(s) Target Heart Rate
e(s)
Gp(s)
Gh(s)
Y(s)
n1
n2
n3
-K-
x

11

Pros

- Better understanding of the mechanisms
- Broad view CI conditions
- Starting platform for understanding more specific CI conditions

Cons

- Does not properly incorporate the parasympathetic nervous system (Scalar)
- Extrapolation is limited

Thank you!