Christopher Lum
lum@uw.edu

Lecture 05e
Bang/Bang Control

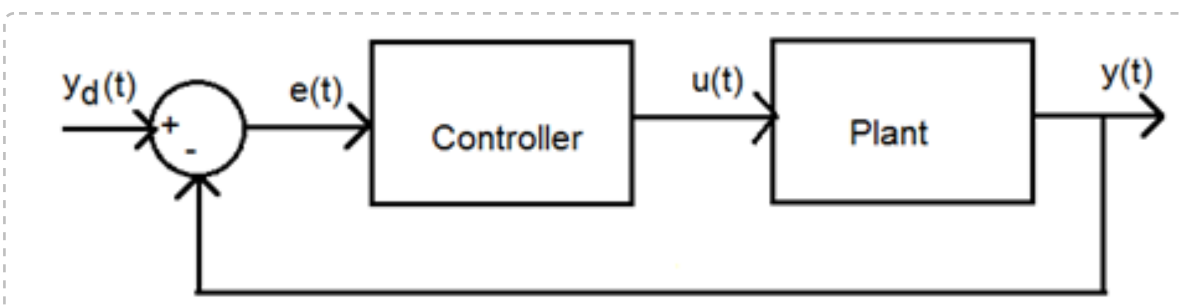
The YouTube video entitled 'Bang/Bang Control' that covers this lecture is located at https://youtu.be/T-T3Wn7dDNYo.

Outline

-Bang/Bang Control
-Results
  -Simulation
  -Real Hardware
-Summary

Bang/Bang Control

One of the main drawbacks of the lookup table controller is that it is an open loop scheme (a simple type of feedforward controller). If the model parameters are inaccurate or some aspects/dynamics of the system are not modeled, the performance will suffer (as evidenced by the significant steady state error). Therefore, we would like to implement a feedback control system which will ameliorate some of these issues.



One simple feedback controller is bang/bang controller. The simplest bang/bang controller operates by examining the error signal and turning on the control signal to a predefined limit when the sign of the error is a certain sign (for example positive) and off when the sign of the error is the opposite sign

(for example negative)

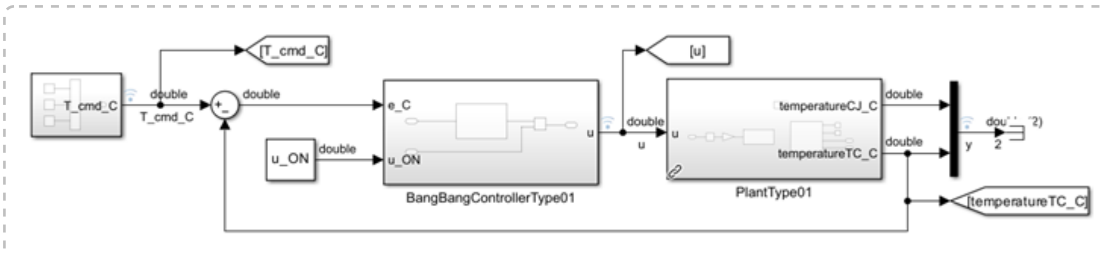
<GO TO CAR DRIVING EXAMPLE>

Let us return to the thermal system we examined in the video entitled ‘Open Loop Control Via Lookup Tables’ at https://youtu.be/Bq-wmKQZsgI.

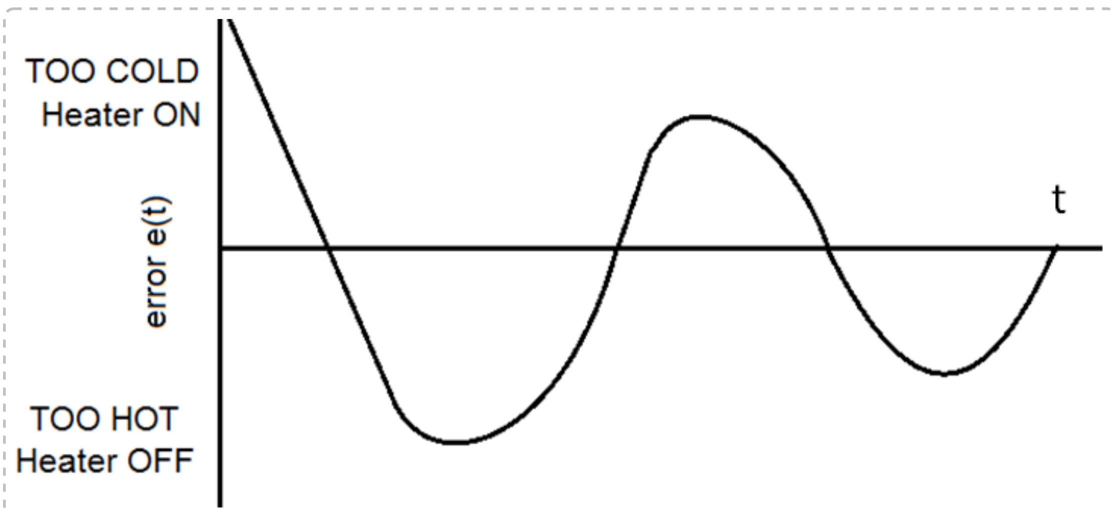
In the situation of our temperature control system situation, we define the error as

$$e(t) = T_{\text{cmd}}(t) - T(t)$$

In our case, the heater should turn on when the system is too cold (error is positive) and turn off when the system is too hot (error is negative). The overall Simulink model is shown below.



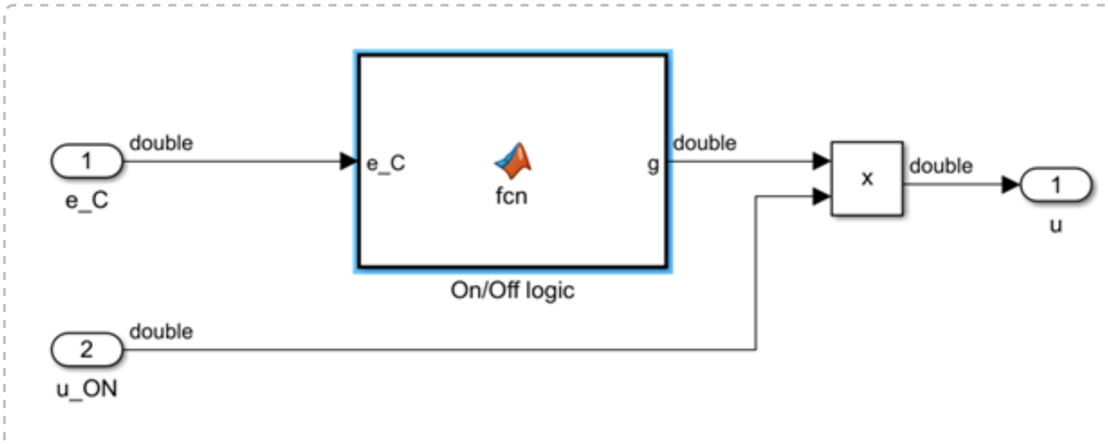
We can derive the logic for the bang/bang controller by looking at a plot of the error



So we see that in our case, the logic for our controller is

```
if(e>0)
    %heater on
else
    %heater off
end
```

We can implement this bit of Matlab code using an ‘MATLAB Function’. This block allows us to utilize a subset of the Matlab language directly in Simulink. The overall controller subsystem looks like



The contents of the ‘MATLAB function’ looks like

```
function g = fcn(e_C)


if(e_C > 0)
        %Too cold, turn heater on
        g = 1;
else
        %Too hot, turn heater off
        g = 0;
end
```

Some notes on Embedded Matlab Functions

-May need to run "mex -setup" to select compiler.

-May need to open the compiler/IDE once to make sure it loads settings before running your Simulink model.

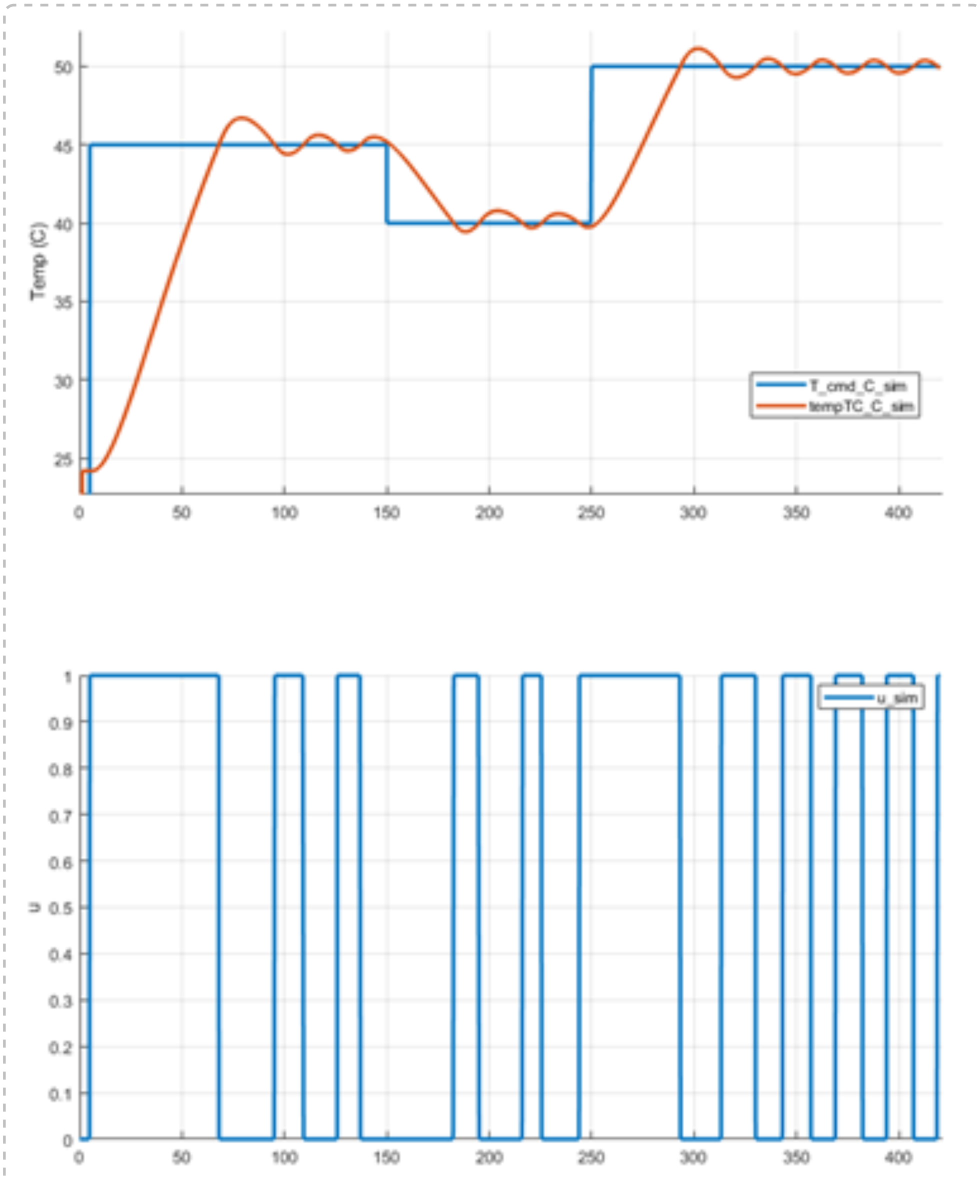
-‘MATLAB Functions’ only support a subset of the Matlab language, check documentation before making esoteric function calls.

-Must make sure all code-flow paths assign the output variable.

Results

Simulation

In simulation, this type of controller seems to work well as shown below

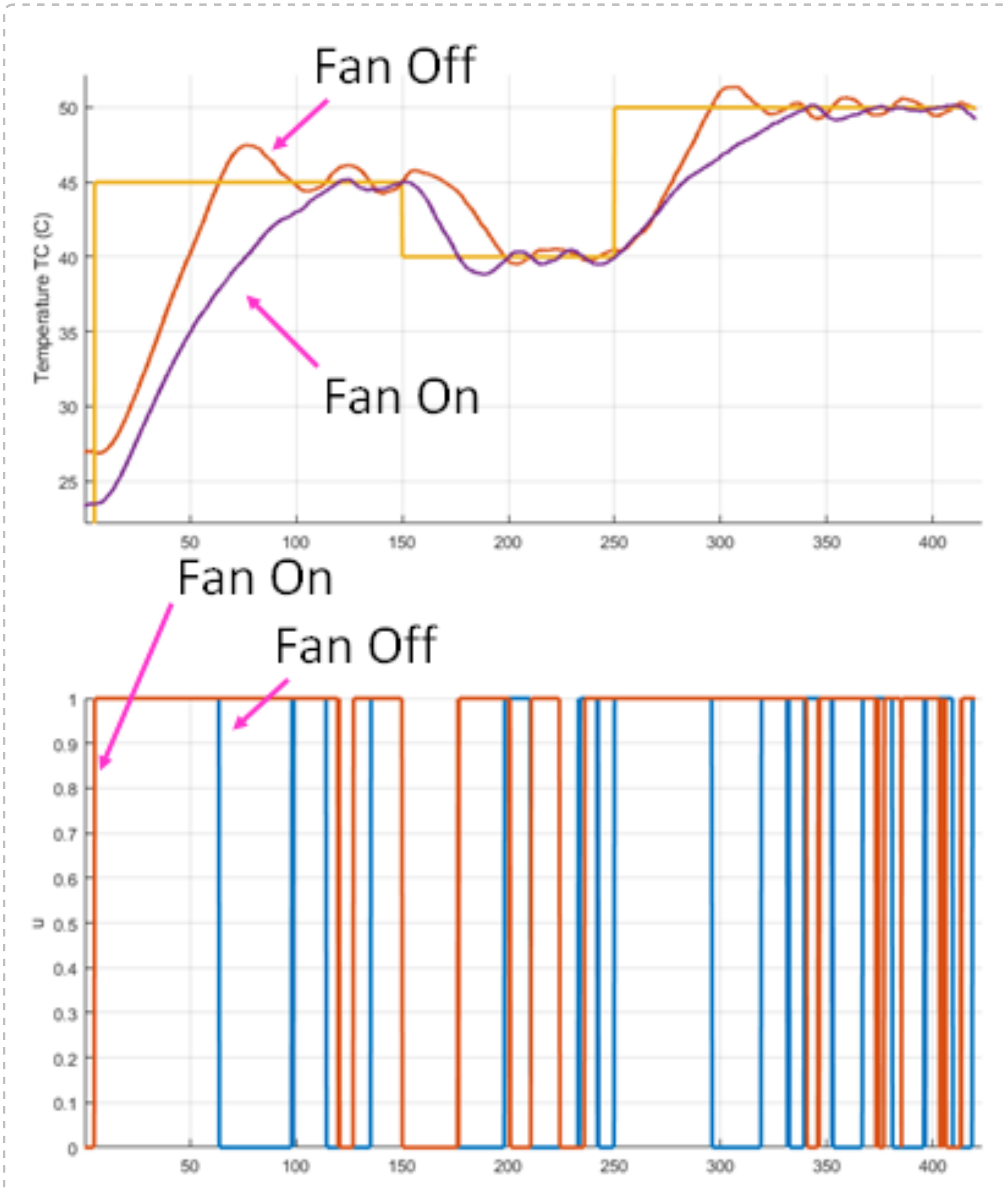


As can be seen, the system is able to track the desired temperature better than the open loop response. Although there is some fluctuations, the temperature reaches the desired temperature.

One drawback of this system is that it is never able to attain and hold the desired temperature, it is constantly fluctuating about the desired setpoint.

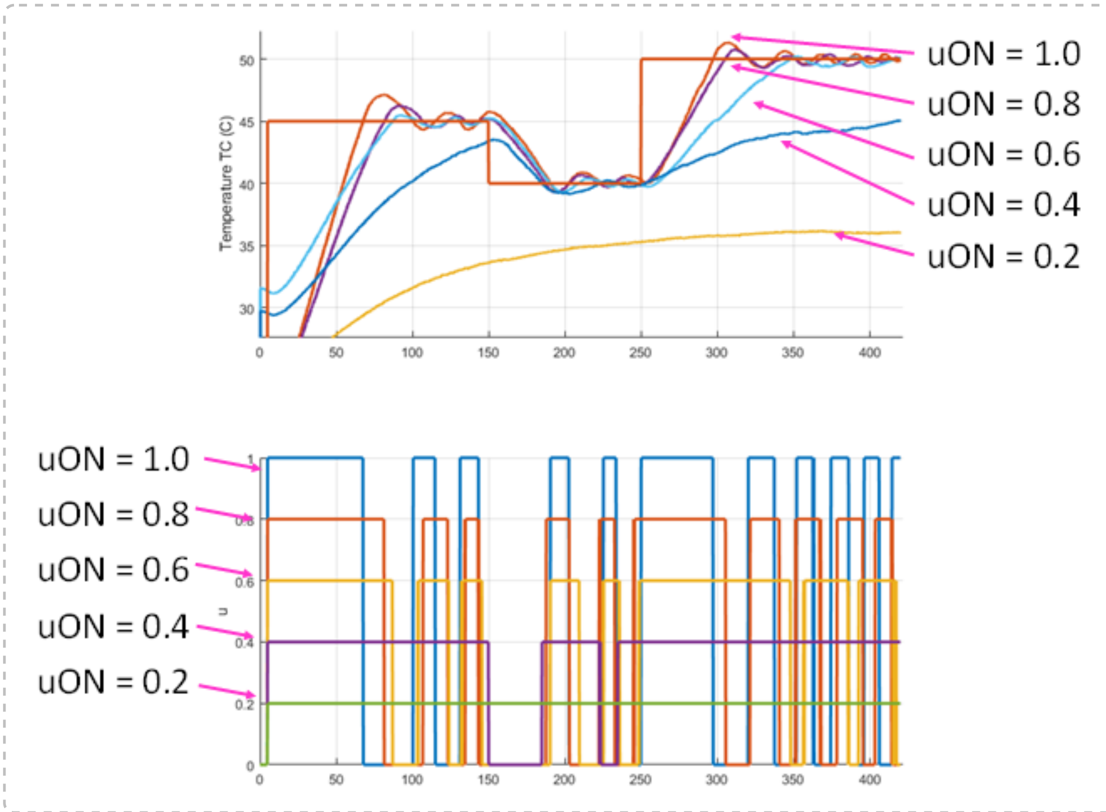
Real Hardware

The system can actually control a real system. Consider the real thermal system controlled by the Arduino. We run the bang/bang controller with environmental disturbances (ie the fan) off and on.

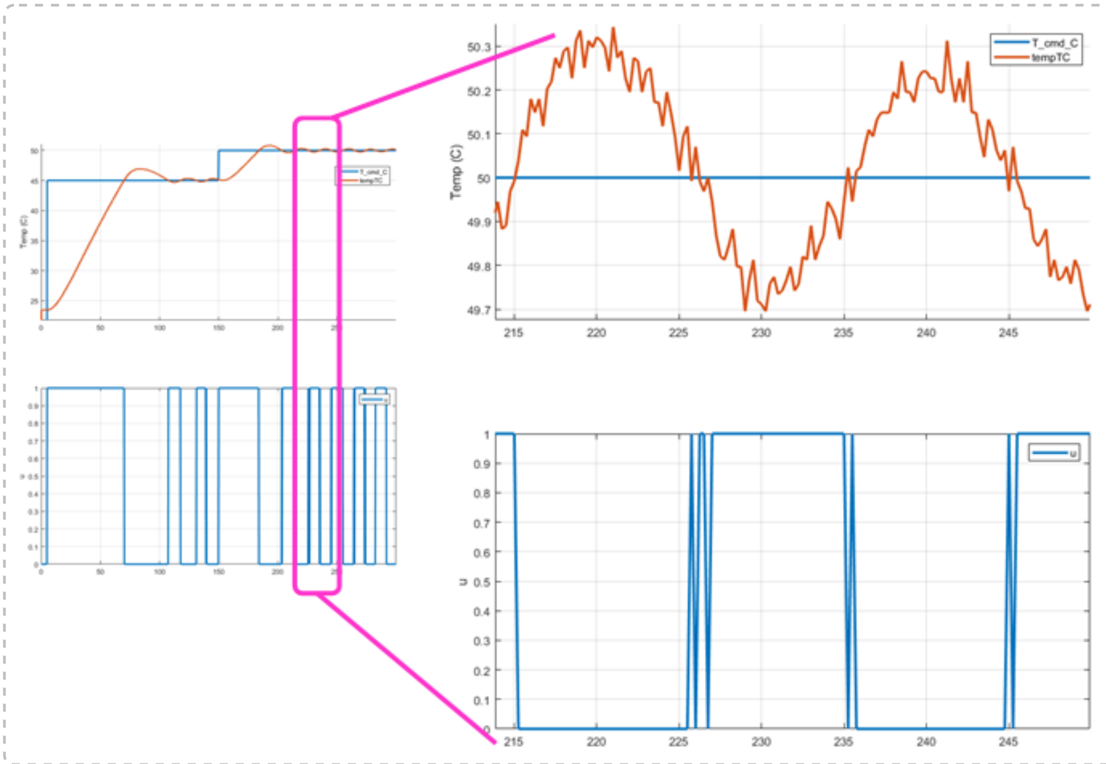


This functions significantly better than open loop control, especially when the disturbance/fan is on.

Note that we can vary u_{ON} and the system can still function as seen below



In addition the the steady state error, the bang/bang control scheme has other issues. In the real world, our sensor is noisy. In noisy situations, we can show that the bang/bang controller does not perform well. We examine a case as shown below.



Another major drawback can be seen by examining the control signal. Notice that the control signal "chatters" (switches rapidly between on/off). This can be detrimental to the system as it will rapidly wear out the actuator. This is mostly due to the presence of noise (recall that we used the unfiltered sensor signal for feedback). If we first filter this signal, we may be able to decrease the amount of chatter. Although this is a valid approach, it may not fix all the chatter so we investigate a modification to the algorithm in a future video.

Summary

Summary of Pros/Cons

As we have seen, bang/bang controllers have several advantages and disadvantages

PROS
-Simple.
-Only a simple sensor is required (needs to sense sign of error only).
-Requires very low computational power.
-Does not require advanced techniques to design controller.
-Mostly plant agnostic (the only parameter that is specific to the plant is the magnitude of u_{ON})

CONS
-Typically cannot hold a set point with zero steady state error.

-Typically has non-trivial over/undershoot.
-A noisy measurement signal typically leads to control/actuator chatter.

EXTRA

Example: Student Demo

We can investigate an example of this using humans to play the part of the bang/bang controller and plant.

Student A - controller
Student B - plant

-Draw a point on the white board showing where we would like to place Student B.
-Have Student B stand a certain distance away from the mark with their back to the board/mark.
-Student A commands student B to move either left of right. Student B must respond by moving at a constant rate in the commanded direction. The rate of movement can represent different levels of u_{ON} (walking represents a small u_{ON} whereas running represents a large u_{ON})

Example: Dog Demo

01_open_loop.3gp: Shows open loop control (blindly issuing a command)
02_bang_bang.3gp: Shows bang/bang control
03_closed_loop.3gp: Shows closed loop control (two people are on either sides of the line adapting their control input based on error)