Christopher Lum
lum@uw.edu

Lecture 01b
Overview of System Modeling and Control

The YouTube video entitled ‘AE511 Week01’ that covers this lecture is located at https://youtu.be/Lee9QEMmU7g

Outline

-Overview of Control Systems
- -System Modeling
- -Simulation
- -System Control

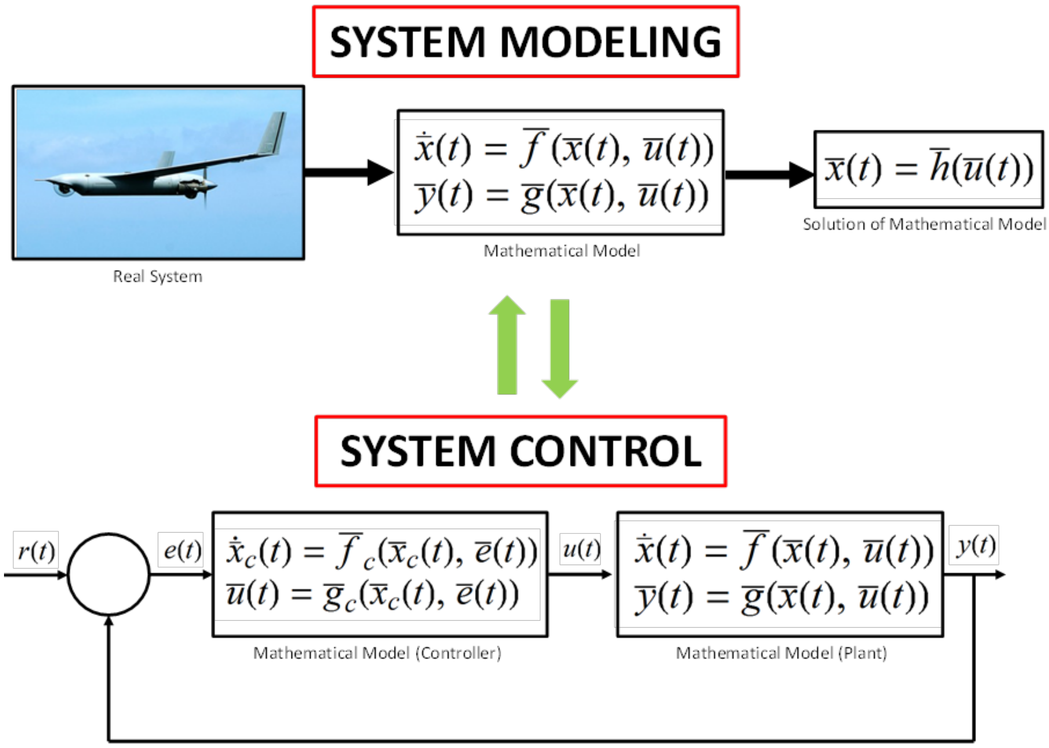
-Autonomous Systems Development

Roadmap for AE511

We will focused on 3 main topics during the AE511 class

1. System Modeling
2. Simulation
3. System Control

This can be represented graphically as shown below. Note that this explicitly shows system modeling and system control but simulation is an inherent part of both components.



System Modeling

Typically involves modeling the plant using some type of mathematical model. In general, when we say "model" we are referring to a mathematical abstraction of an actual system. This model undoubtedly neglects some portions of the actual system and is therefore an approximation. The goal is to create a model which accounts for the important behavior without becoming too complicated.

For example, when modeling an aircraft, it may be necessary to model the airspeed and how the elevator affects the orientation, but it may or may not be necessary to model the rotation of the earth.

There are many different types of models that one can use to model a system

<u>Linear Ordinary Differential Equations</u>

-Works well for simple systems

-Analysis and tools are well developed

Example: DC motor, LRC circuit

$$\dot{\overline{x}}(t) = A\,\overline{x}(t) + B\,\overline{u}(t)$$

<u>Nonlinear Ordinary Differential Equations</u>

-Can describe more complex systems

-Analysis and tools are well somewhat more involved

Example: DC motor over large operating conditions, single or double jointed pendulum

$\dot{\overline{x}}(t) = f(\overline{x}(t), \overline{u}(t))$ (explicit)

$\overline{0} = g\left(\dot{\overline{x}}(t), \overline{x}(t), \overline{u}(t)\right)$ (implicit)

<u>Partial Differential Equations</u>

-Difficult to study and simulate but most accurate

Examples: Fluid flow, multi-dimensional thermal conduction

$\frac{\partial}{\partial x} u(x, y) = 0$

<u>Difference Equations</u>

-Good for systems with discrete dynamics

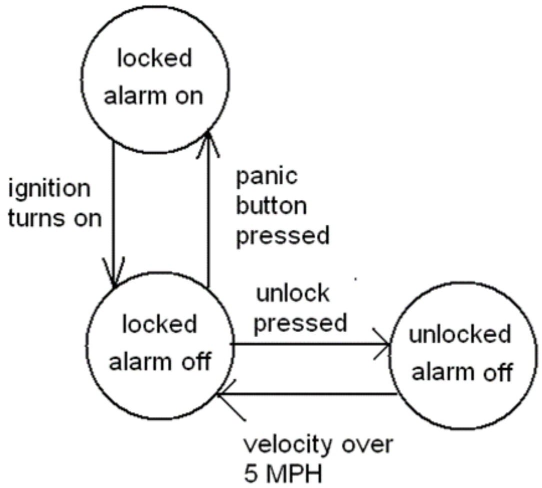
Examples: Digital control systems

$\overline{x}(k+1) = f(\overline{x}(k), \overline{u}(k))$

<u>Discrete Event Systems/Petri Nets</u>

-Good for systems with discrete dynamics and complicated transition between states

Examples: ATM interaction, car keyless entry system



Finite State Machine for Car Alarm/Lock System

<u>Rule Based Dynamics</u>

-Systems with complex interactions or behavior

-Systems which switch behavior based on certain inputs

Examples: animal behavior, social interaction, look up tables

```
//Dog Behavior
if(hungry) {
    action.push_back(findFood);
}
```

```
if(toyInSight) {
    action.push_back(grabToy);
}

if((mood == bored) && (isExpensive(nearestItem))) {
    destroyItem(nearestItem);
}
```

Empirical Models

-Systems which are experimentally fit with a particular model

Examples: fuel slosh fit to a first order dynamic model

And many more...

Simulation

Once we have a mathematical model of how the system operates, we can attempt to use it to predict how the system will behave given certain conditions and/or inputs.

For example, when modeling an aircraft, it may be necessary to model the airspeed and how the elevator affects the orientation, but it may or not be necessary to model the rotation of the earth.

Educated Guess

-Use previous experience or wisdom to predict behavior of system

Examples: put steak in front of dog, he will probably eat it.

Analytical Solution

-Solve mathematical model for closed form solution of system

Examples: solve linear ode via Laplace method and obtain expression for $x(t)$

Approximate Solution

-Original mathematical model is too complicated to analyze, so make simplifications to model until it is easy enough to work with

Examples: linearizing a non-linear system about an operating point

Numerical Simulation

-A type of approximate solution, use a computer to provide numerical approximation of solution to system

Examples: numerical solutions of linear and non-linear differential equations, multiple iterations of rule based or discrete event system.

Hardware-in-the-Loop or Software-in-the-Loop Simulation

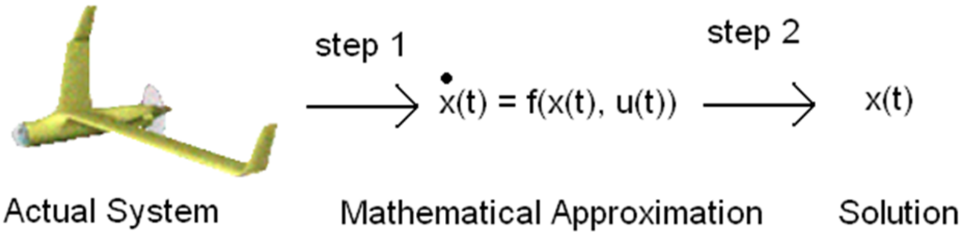
-Combine elements of pure numerical simulation with actual hardware.

Examples: hardware-in-the-loop simulator, A/D and D/A pipeline

And many more...

In the context of this class, the system modeling and simulation go one after another. In other words, for our class, we are mostly interested in modeling the system as an ODE and then using various methods to solve the system.

1. Take a dynamic system and model it as an ODE
2. Use various methods to solve this ODE.



In other words, given an ODE

$$\dot{x}(t) = f(x(t), u(t))$$

Find $x(t)$ which satisfies this equation for a given $u(t)$. For our class, we use simulation methods to obtain a solution for the system.

System Control

If the system does not behave like we would like it to, we can attempt to change the behavior of the overall system.

Modify System Parameters

-Conceivably, you could modify your plant to obtain different performance

-This is not typically considered control but in the loosest sense of the word, we could consider this a control strategy

Examples: put a bigger engine on aircraft, modify control surfaces

Continuous Inner Loop Controller

-Leave plant untouched but add controller to modify behavior

Examples: PID, full state feedback, lead/lag

Outer Loop Controller

-Perform higher level functions

Examples: Search algorithms, path planning, mission tasking

Look Up Tables

-hard to analyze but may cover a large range of operating conditions

Examples: gain scheduling

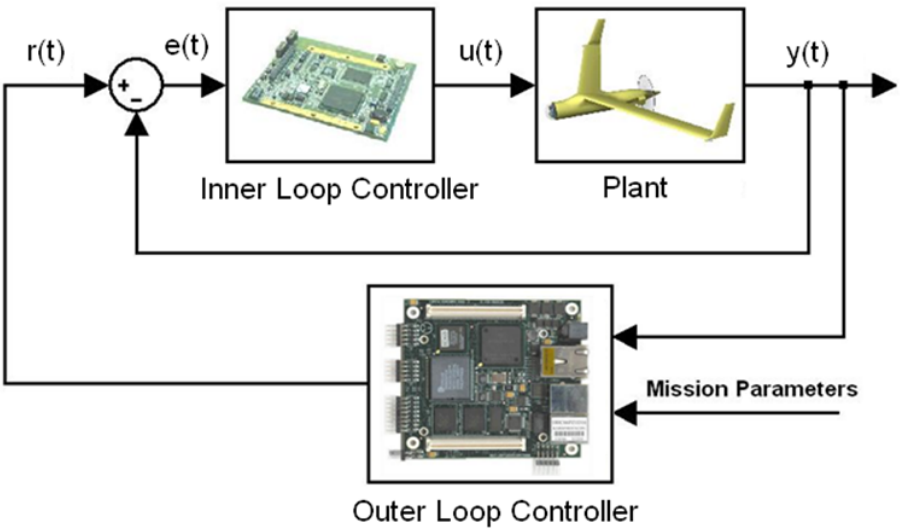
And many more...

Autonomous Systems Development

Consider an actual autonomous system and let us investigate some of the issues involved in its development.

<Go to slides>

The control architecture of a fully controlled, autonomous system might look something like



Where the signals are defined as

$y(t)$ =output of plant, to be measured by sensors

$u(t)$ =input to plant, input is generated by inner loop controller and then applied to actuators of the plant

$r(t)$ =reference signal that we would like the plant to track

$e(t)$ =error between desired output of plant, $r(t)$, and actual output, $y(t)$

The function of the inner loop controller is to take the error, $e(t)$, and generate a control signal, $u(t)$, which can then be applied to the plant. Typically, this inner loop controller is concerned with low level actions like

-signal tracking

-state stabilization

The outer loop controller is a higher level entity which can be in charge of more abstract functions. This is beyond the scope of this class. In this class, we will mostly focus on the plant and inner loop controller. At the end of the quarter, we may investigate some higher level control design (outer loop architectures)