1 Modelling Dynamical Systems

The behavior of the system is best characterized in terms of state and its evolution over time. Before we proceed, we informally set up some terminology which will enable us to characterize behaviors of dynamical systems and design controllers for variety of systems. The basic entities which describe the behavior of the dynamical system are :

- **X** set of states of the system and the environment.
- **Y** set of outputs. Information available to the controller, since the information about the entire state is often not available to the controller.
- **U** set of control actions.

The domains from which these entities can come from depends on what type of behavior we are trying to capture. For example on case of mobile robot the state $\mathbf{X} = (x, y, \theta) \in \Re \times \Re \times S^1$, while if the subject of our control is washing machine, the part of the state can be the water in the washing machine which can be $\mathbf{X} = \{hot, warm, cold\}$.

Example Consider your new digital camera. Defined this system in terms of its state, input and output, i.e. specify the domains of each.

1.1 System State Equations

From the control theoretic standpoint we distinguish two entities: the subject of our control, which is in control literature often also referred to as *plant* and the *controller*.

The behavior of the system is described by time trajectories $\mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t)$. We will focus for the moment on the trajectories, obeying laws, which can be described in terms of differential equations :

$$\dot{\mathbf{x}} = f(\mathbf{x}(t), \mathbf{u}(t)) \quad (1)$$
$$\mathbf{y} = g(\mathbf{x}(t)) \quad (2)$$

or in terms of difference equations:

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t)) \quad (3)$$
$$\mathbf{y}(t+1) = g(\mathbf{x}(t)) \quad (4)$$

For a special class of linear systems, the system state equations have the following form:

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y} &= C\mathbf{x}(t) \end{aligned} \quad (5)$$

where A, B, C are matrices of appropriate dimensions. For linear systems, there is an ample of techniques which give us guidelines how to characterize system's performance, stability, controllability, observability and how to design control laws which are optimal with respect to some chosen objective.

The set of states, inputs and output is finite the trajectories of the system can be for example described by finite state machine, with inputs and outputs. The goal of control is then do design a control policy, which specifies what control actions should be done in every possible situation. In the most general setting the control policy can be viewed as a mapping

$$\pi : H_x \rightarrow U \quad \text{or} \quad \pi : H_y \rightarrow U$$

from state or output histories H_x, H_y to control actions.

Point Mass Here we demonstrate a simple instance of such system state equations for a point mass system and how to go from between dynamic equations and system state equations. Consider a point mass in ideal environment with no friction under influence of external forces F_{ext}. The dynamic equations of this system are fully characterized by Newton's second law:

$$m\ddot{x} = F_{ext}$$

The behavior of the system is at each instance characterized by it's position and its velocity. Hence the state of the system $\mathbf{x} = [x, \dot{x}]^T$. The system state equation, which captures the evolution of the system's state over time can then be described as differential equation

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} \mathbf{u}(t)$$

1.2 Control strategies

Lets have a look at some examples and different components of the control law. In the following examples we will assume that the output of the system $\mathbf{y}$ is directly the state $\mathbf{x}$. Consider mass-spring-damper mechanism, first in the absence of any external forces

$$m\ddot{x} + k_s x + k_d \dot{x} = 0$$

In the homework we had a chance to observe how the behavior of the state $\mathbf{x} = [x, \dot{x}]^T$ depends on the choice of constants k_s, k_d and initial conditions. Hence the open loop dynamics of the system is

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -\frac{k_s}{m} & -\frac{k_d}{m} \end{bmatrix} \mathbf{x}(t)$$

and the control input in this case is zero. Suppose now that we will apply some external force of the following form $F_{ext} = -k_p x - k_v \dot{x}$ which is proportional to the current position and current velocity of the mass. This would yield following dynamics equations

$$m\ddot{x} + k_s x + k_d \dot{x} = -k_p x - k_v \dot{x} \tag{6}$$

$$m\ddot{x} + (k_s + k_p)x + (k_d + k_v)\dot{x} = 0 \tag{7}$$

Note that the second question describes the system of the same type, but by adding the external force terms, we effectively changed the coefficients of the system and hence change the system's behavior.

Example For mass-spring-damper system above, suggest the formula for F_{ext} such that the dynamic equations of the closed-loop system will have the following have the following dynamics, will behave as a simple point mass

$$\ddot{x} = F$$

We did this example in class (F_{ext} will be some function on F).

Example Consider again a simple point mass system (with no damping and no friction).

$$\ddot{x} = F_{ext}$$

We would like the point mass follow particular trajectory which was computed ahead of time $\ddot{x}_d, \dot{x}_d, x_d$. Suppose we first apply external force

$$F_{ext} = \ddot{x}_d$$

If we simply use this control law the dynamics of the system would be

$$\ddot{x} = \ddot{x}_d$$

In case we would like to compensate for the possible initial errors in $x, \dot{x}$ let's consider the following control law

$$F_{ext} = \ddot{x}_d - k_v\dot{e} - k_p e$$

with $e = \theta - \theta_d$. This would yield the following system dynamics

$$\ddot{x} = \ddot{x}_d - k_v\dot{e} - k_p e \quad (8)$$

$$\ddot{e} + k_v\dot{e} + k_p e = 0 \quad (9)$$

This derivation can vary depending on how is the error defined. The above equations now described the error dynamics. We can now investigate the behavior of the error as a function of time and choose the constants k_p, k_v appropriately to yield the desired performance. In the context of robotics this control law is also referred to as *computed torque* law. The $\ddot{x}_d$ part of the control law is also referred to as *feed-forward* term and it would be sufficient if our model is perfect.

Proportional Derivative Control Even simplest control law which we can apply is to

$$F_{ext} = -k_v\dot{e} - k_p e$$

Compared to previous case this control law has no feed-forward term. In practice, when the objective is to track very complex trajectories it is quite hard to achieve without the feed-forward term. Furthermore proportional derivative control law leaves some steady state error. In order to compensate for steady state errors additional term *integral term* can be added to the system.

$$F_{ext} = -(k_v\dot{e} + k_p e + k_i \int e dt)$$

- What is difference between closed-loop an open-loop system ?
- What is the role of feed-back in the control system ?

- What is the role of feed-forward term in the control system ?

Note that the names of the terms actually correlate with the way the arrows are drawn in the system.

- The proportional control law at each instance of time responds to the current error in position. How fast should it respond is specified by constant k_p which is called proportional gain. If the gain is too high the system can overshoot and eventually lead to oscillations. Damping can be used to prevent the oscillations.
- The derivative control is used to correct momentum of the system depending how far away it is from the goal. It is proportional to the derivative of position (or error).
- The integral control provides another improvement to the control law since it integrates the steady state errors over time to compensate for errors.