The Theory of Control: A Brief Overview

Robin H. Pearce

Abstract

Control systems form a vital part of engineering, where anything that needs to be regulated or optimised can be done so with a technique that is encompassed by control theory. Since the 16th century, problems have been modelled as dynamical systems so that a mathematical control strategy could be devised and an equivalent physical strategy invented. Methods for controlling mechanical, analogue and digital systems have been invented and some are widely used. I will briefly introduce and explain many of the relevant topics considered to be part of control theory.

"Control Theory" is somewhat an umbrella term, used to describe a wide range of analytic techniques and styles which are applied to the control of a mathematical system. Much of this theory is linked to systems theory, since many of the problems that can be solved can also be modelled as a dynamical system. The methods involved range from classical open or closed loop control, up to more modern disturbance rejection and noise filtering, as well as stabilisation and fine-tuning.

Systems

Many of the systems can be modelled as a system of differential equations. Usually these are modelled in continuous time, however discrete time equations can also be useful. The basic system model in continuous time is

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), t) \\ y(t) &= h(x(t), u(t), t) \end{aligned}$$

where $x(t)$ is the state, $u(t)$ is the control input and $y(t)$ is the system output at time t. The easiest system to consider is one where x, y and u are all scalar. This is known as a SISO system, which stands for single input single output. Generally, these systems look at signal processing, particularly for systems which are both linear and time-invariant. These systems are known as LTI systems and are extremely useful in control and signal processing.

When considering the simplest, stateless LTI SISO system, we have that

$$y(t) = (u \star h)(t)$$

where $h(t)$ is the impulse response of the system. By taking the Laplace transform of both sides, we end up with

$$Y(s) = U(s)H(s)$$

where $Y(s)$, $U(s)$ and $H(s)$ are the laplace transforms of $y(t)$, $u(t)$ and $h(t)$ respectively. The function $H(s)$ is known as the transfer function of the system and can be written as

$$H(s) = \frac{Y(s)}{U(s)}$$

Closed loop system

One of the simplest and most useful control systems is a closed-loop feedback controller, similar to the one seen in Figure 1. In this system, the output is measured and sent back into the system to help regulate the response of the system. Using the same notation before, we can consider two controllers, $G_1(S)$ and $G_2(s)$, so that we can find the closed-loop transfer function $\tilde{H}(s)$, that is the function which satisfies

$$Y(s) = R(s)\tilde{H}(s)$$

where $R(s)$ is the laplace transform of the reference signal $r(t)$ and $U(s)$ is the laplace transform of the signal $u(t)$ before it enters the plant $H(s)$.

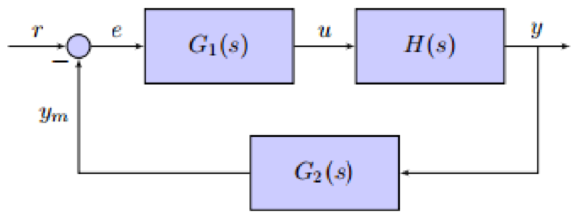

Figure 1: Closed-loop feedback controller

By starting with $Y(s) = U(s)H(s)$ and substituting back for $U(s)$, $G_1(s)$, $E(s)$, $R(s)$ and $G_2(s)$, we end up with the expression

$$Y(s) = R(s)\frac{G_1(s)H(s)}{1 + G_2(s)G_1(s)H(s)}$$

so we have an expression for $\tilde{H}(s)$, where $Y(s) = R(s)\tilde{H}(s)$. Quite often these systems are being controlled because some property of stability is desired.

Stability

There are two main types of stability in control systems: BIBO stability and internal stability. BIBO stands for bounded inupt bounded output, and concerns the systems ability to return bounded outputs when given bounded inputs. The second type of stability is often the most important, and can be determined using the Routh-Hurwitz criterion.

Internal stability refers to the ability of the system to return to a state of equilibrium after an arbitrary displacement from equilibrium. This is commonly associated with the location of the poles of the transfer function, $H(s)$. That is, if

$$H(s) = \frac{N(s)}{D(s)}$$

where $N(s)$ and $D(s)$ are polynomials, then the stability can be determined by the location of all s which satisfy $D(s) = 0$. By the Routh-Hurwitz criterion, the system will be stable if the real parts of all such s are negative. If we are given a particular system where $H(s)$ is unstable, then we can design a feedback controller like the one in Figure 1, such that $\tilde{H}(s)$ or $H_c(s)$ is stable by choosing $G_1(s)$ and $G_2(s)$. If

$$H(s) = \frac{N_H(s)}{D_H(s)}, G_1(s) = \frac{N_1(s)}{D_1(s)}, G_2(s) = \frac{N_2(s)}{D_2(s)},$$

then we can rewrite the closed loop transfer function as

$$H_c(s) = \frac{N_H N_1 D_2}{D_H D_1 D_2 + N_H N_1 N_2}.$$

This system will now be stable if the solutions to $D_H D_1 D_2 + N_H N_1 N_2 = 0$ are all in the left-hand plane, that is the real part of all the solutions are negative. The simplest set of controls for the closed loop system are to take $G_1(s) = K$ and $G_2(s) = 1$, so now the system is stable if the zeros of $1 + KH(s)$ are in the left-hand plane.

States

So far, the control methods considered are for systems where the state doesn't matter. In many real life examples this is not the case, so we need to consider methods which control the state of a system. Most often, systems will be either linear or easily approximated by a linear system. These linear systems take the form

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

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$. The matrices are all constant and the system is commonly referred to as an (A, B, C, D) system. The discrete time equivalent is just the same, except t is replaced by n and $\dot{x}(t)$ becomes $x(n+1)$.

For a linear autonomous system, $\dot{x}(t) = Ax(t) + g(t)$, solutions take the form $\Phi(t, t_0) = e^{A(t-t_0)}$, where $e^{A(t-t_0)}$ is the matrix exponential. The laplace transform for e^{At} is $(sI - A)^{-1}$ which is known as the resolvant. For the full (A, B, C, D) system, the same computations lead to

$$H(s) = C(sI - A)^{-1}B + D$$

Of course, e^{At} can often be difficult to calculate for non-diagonal systems, so we can consider a similarity transform $P\tilde{x} = x$, where $\det(P) \neq 0$. The system is now a $(P^{-1}AP, PB, CP, D)$ system, where $P^{-1}AP$ is diagonal and e^{At} is in Jordan normal form.

Controllability

For any (A, B, C, D) system, we can find all states that we can reach through controlling the system. A state x_d is considered reachable or controllable-from-the-origin if there exists a control input u that takes the state $x(t)$ from the zero state to x_d in a finite amount of time.

The converse definition is that a state x_s is called controllable if there exists a control input u that takes the state from x_s to the zero state in a finite amount of time. Any set of states can also be called reachable or controllable if every state in the set is reachable or controllable. If all states are reachable or controllable, then the system is a reachable or controllable system.

To determine if a system is controllable, we can calculate the controllability matrix from the matrices A and B, defined to be

$$\mathrm{con}(A, B) = [B, AB, A^2B, ..., A^{n-1}B]$$

where the system is controllable if and only if the controllability matrix has full rank. If the matrix does not have full rank, we can still work out the controllable subspace. If $\Phi(t)$ is nonsingular, then reachability implies controllability and vice versa. This concept is important if we are trying to ensure stability of a system by controlling the state to the origin.

The linear state feedback law is one way of ensuring stability in a system when the pair (A, B) is controllable. If we set

$$u(t) = Fx(t) + r(t)$$

we get an augmented system which only depends on $x(t)$ and $r(t)$. Also, if (A, B) is controllable, then it is possible to choose the eigenvalues of the state transition matrix to be what we want by making the right choice of F.

Observability

A system is observable if we can determine the initial condition for the state given the inputs and outputs over a finite time interval. The observability of a system can be found from the matrices A and C using a similar method to controllability, where the observability matrix is given by

$$\mathrm{obs}(A, C) = [C^T, A^TC^T, (A^2)^TC^T, ..., (A^{n-1})^TC^T]^T$$

The condition for observability of a system is similar to that for controllability; that is, the observability matrix has full rank if and only if the system is observable. One interesting result is that there is a duality between controllability and observability. When given a standard (A, B, C, D) system, we can define the dual system as

$$\begin{aligned} \dot{x}(t) &= A^Tx(t) + C^Tu(t) \\ y(t) &= B^Tx(t) + D^Tu(t) \end{aligned}$$

The result is that the dual system is controllable if and only if the original system is observable and the original system is controllable if and only if the dual system is observable. In real life, not all systems are observable, so it is useful to design an observer to estimate the state of the system and make corrections based on this estimation.

Observers

When a system is not observable, it is possible to create a state estimator and use that to control the system instead. The only problem remaining is to make sure that the estimate starts and remains accurate. The *Luenberger Observer* is given by the system on the next page.

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K(y - \hat{y})$$
$$\hat{y}(t) = C\hat{x}(t) + Du(t)$$

This augmented system is now a $(A - KC, [B - KD, K], C, [D, 0])$ system whose input is now $[u^T, y^T]$, that is the input and output of the original system. All that remains is to ensure that the distance between the estimate and the actual state is stable. If we define the error between the state and the estimate to be

$$e(t) = x(t) - \hat{x}(t)$$

then it turns out that the estimation error actually behaves like an autonomous system, where

$$\dot{e}(t) = (A - KC)e(t)$$

Now if we choose K such that the real parts of the eigenvalues of $(A - KC)$ are all strictly negative, the estimate will be asymptotically stable, meaning the estimation error will disappear as $t \to \infty$. We can also combine this with a linear state feedback similar to the one on the last page to control the estimated system to the desired state while the observer controls the estimate to the actual state.

Linear Quadratic Regulator

For the Linear Quadratic Regulator (LQR), we will consider the system

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

where A and B are constant and the system is controllable. Now, we will consider a cost for controlling the system, given by a cost functional J. The integrand of J is quadratic in x and u and is of the form

$$J(u) = \int_0^T \left(x^T Q x + u^T R u\right) dt + x(T)^T Q_f x(T)$$

where $Q = Q^T \geq 0$, $Q_f = Q_f^T > 0$ and $R = R^T > 0$. This is necessary since Q represents the cost of being at a particular state and R is the cost of implementing a control. If either of these matrices were negative it would represent being paid for implementing a control or being at a state.

This representation of LQR is looking for a control $u(t)$ which will regulate the system at the origin such that the cost is minimal. Here, x can be anything we want to control to be zero, for instance, it could be the error between the state and a state estimator such as the one presented in the last section. This is often used for trajectory tracking where we need to control the system to another non-zero point, where we can choose to make the cost proportional to the distance between the estimate and the state.

The optimal control for this system is a linear state feedback control law, similar to the ones presented previously. In this case,

$$u(t) = (-R^{-1} B^T P(t))x(t)$$

where $P(t)$ solves the riccati differential equation

$$-\dot{P}(t) = A^T P(t) + P(t)A - P(t)BR^{-1}B^T P(t) + Q$$

This equation is specified "backward in time", that is we start with a final condition and work backwards to the initial condition. This leads us into the topic of Model Predictive Control.

Model Predictive Control

Also called "receding horizon control", Model Predictive Control, or MPC for short, works by solving a problem over a short planning horizon, taking the first step and then recalculating the optimal control over the new planning horizon. This method is not optimal, however it is usually less computationally taxing and also allows for small unexpected disturbances without throwing out the whole control scheme.

MPC is not guaranteed to be stable, however there are different methods for applying the MPC so that the resulting system will be stable. One of these methods is to enforce an end-point for the system to drive towards. This usually makes the system stable and uses the ideas found in dynamic programming.

Dynamic Programming

There are a number of methods that fall into the category of dynamic programming, such as Dijkstra's shortest path algorithm, which finds the shortest or cheapest path through a network. The main result from dynamic programming that applies to Control Theory is the *Principle of Optimality*, which states that if the optimal path between points a and b passes through a third point c, then the optimal path from c to b is the same as the section of the optimal path from a to b starting from c.

This principle is used when computing the minimal cost of a functional similar to that of the LQR on the previous page. By applying the ideas of dynamic programming to a continuous system, we can derive the Hamilton-Jacobi-Bellman equation for finding optimal solutions to LQR. The HJB equation is

$$J_t^*((x,t) + \min_u[g(x,u,t) + J_x^* a(x,u,t)] = 0$$

where J^* is the minimal cost functional. All of the techniques presented thus far are for deterministic systems with no noise, however stochasticity is present in almost all physical systems, so it is important to consider techniques for dealing with uncertainty.

Linear Quadratic Gaussian and Kalman Filtering

Many real world systems are imperfect and have inherent uncertainty in the measurements. We can simulate this by considering a modification to the standard linear system where we add some noise terms to the state and the output. The system now becomes

$$\dot{x}(t) = Ax(t) + Bu(t) + \xi_x(t)$$
$$y(t) = Cx(t) + Du(t) + \xi_y(t)$$

where ξ is a random process, that is a sequence of random variables. In the continuous time case, this is similar to Brownian motion and other methods for solving stochastic differential equations are required, making the discrete time case much simpler.

The Kalman Filtering Algorithm works similarly to the Luenberger Observer that was seen on the previous page, however this time the matrix K depends on time. The estimator for a simple case where $B = D = 0$ this time is

$$\hat{x}(n+1) = A\hat{x}(n) + K(n)(y(n+1) - CA\hat{x}(n))$$

This is the Linear Minimum Mean Squared Error (LMMSE) estimator for the state $x(n)$. If the noise terms are gaussian, then the LMMSE is also the optimal MSE estimator. This means that the Kalman Filtering Algorithm will give the most accurate estimates of the state when the measurements contain gaussian noise.

Non-linear Control

All of the systems considered so far have been linear, which makes their analysis much easier, however many real world systems are non-linear and sometimes a linear approximation is not sufficient. For non-linear systems, many of the concepts such as stability, controllability and observability still mean the same, however the equivalent results are more dependent on the particular model.

The biggest difference between a linear and non-linear system is the location and number of critical points. Linear systems only have one critical point at the origin, making stability an important topic, however non-linear systems can have multiple critical points which make the dynamics even more difficult to control. It is also possible that non-linear systems will exhibit properties such as periodic orbits, bifurcations and chaos, as well as finite escape time.

Conclusion

There are many topics and methods that fall into the category of Control Theory, but what they all have in common is that they are designed to solve real world problems. Many of these are particularly useful for engineers designing just about anything that moves. The topics in Control Theory are important for many reasons, most of all because they keep our world stable.