# MATH4406 - Control Theory
# Course Summary

Thomas Plevnik
41755433

**Abstract**

In a broad sense, control theory is the behavioural study of dynamical systems with inputs. Control theory does not truly belong to either mathematics or engineering exclusively, but rather, as an amalgam of various components of both fields. Whilst it may exist in various forms and guises, the underlying principle remains largely the same: The systems considered have both an input, and an output, and a *controller* manipulates the inputs to a system to obtain the desired effect on the output of the system. These systems are generally modelled with respect to time, as they are representative of real physical problems, and are therefore commonly represented by differential equations.

# Overview

For centuries, the theory of control has been applied to some of the greatest technological problems faced by man, and has since provided results that have crafted and shaped the world in which we live. From the steam engine, to spacecraft, control theory has been molded and warped to suit the requirements, drawing from a rich background in mathematics and engineering.

Control theory is studied in a variety of contexts, each employing different tools and apparatus. One shared feature however, is the formulation of the system, or *plant*, itself. We have already stated that problems are presented as differential equations, so the basic system model is generally of the form

$$\dot{x}(t) = f\big(x(t), u(t), t\big) \tag{1}$$

$$y(t) = h\big(x(t), u(t), t\big) \tag{2}$$

where the state, which describes the system at any given time, is represented by $x \in \mathbb{R}^n$, the input by $u \in \mathbb{R}^m$, and the output by $y \in \mathbb{R}^k$. Alternately, a system may be modelled in discrete time by

$$x(n+1) = f\big(x(n), u(n), n\big) \tag{3}$$

$$y(n) = h\big(x(n), u(n), n\big) \tag{4}$$

Using these representations, we may now take a closer look at some specific applications [1].

# SISO Linear Systems

It was previously asserted that the systems under consideration have both an input and an output. A more refined definition in the current context would be to refer to systems as a mapping of an input signal to an output signal. A signal in this instance is essentially synonymous with a function, and as such, may be continuous time or discrete time. If the signals are scalar, then the system is termed *SISO*, which stands for *single input single output*. If the signals are vectors, then the system is termed *MIMO*, short for *multi input multi output*. The combinations SIMO and MISO are also valid.

There are several types of systems that are of interest, and some that are not, at least not for the purposes of this paper. However, the properties of *linearity* and *time-invariance* are of significant interest, as these systems are widely used in electrical engineering. For a system to be *linear*, an input of the form $\alpha_1 u_1(t) + \alpha_2 u_2(t)$ corresponds to an output of the form $\alpha_1 y_1(t) + \alpha_2 y_2(t)$, for $\alpha_i \in \mathbb{R}$. A system is *time-invariant* if for any time shift $\tau$, the output $y(t-\tau)$ corresponds to the input $u(t-\tau)$. Systems that satisfy these two properties are known as *Linear Time Invariant* systems (LTI). Furthermore, a general notion to consider is that of *BIBO stability*, where BIBO stands for bounded-input bounded-output. A system is BIBO stable if whenever the input $u$ satisfies $\|u\|_\infty < \infty$, the output $y$ satisfies $\|y\|_\infty < \infty$.

As we know, a system maps an input to an output. But what if there is a desired output to which the system did not map? Well, in order to avoid trivialities, we assume that the system behaves in such a way that $u(t) \neq y(t)$. This resulting output may be fed back into the system, or plant, to yield another output, distinct from the previous. This idea of *feedback* is key to controlling the system. However, given our current representation, this may be easier said than done. Using integral transforms, it can be shown that the output $y(t) = H(s)e^{st}$, where $H(s)$ is the Laplace transform of the impulse response, and is important enough to control to warrant a name, the *transfer function*.

The transfer function arises from the relation $y(t) = (u * h)(t)$, which denotes a convolution between the input and the impulse response, and using Laplace transforms, it can be shown that the output polynomial $Y(s) = U(s)H(s)$. Given that LTI systems are often described by ordinary differential equations, Laplace transforms become extremely useful, as these systems are reduced to simple algebraic problems. Note that the Z transform is used for discrete time systems.

The control element of these systems really comes into play with the introduction of the LTI controllers $G_1$ and $G_2$. The entire system relating the output $y$ to an input reference $r$ is then LTI, and may thus be represented as in Figure 1 [2].

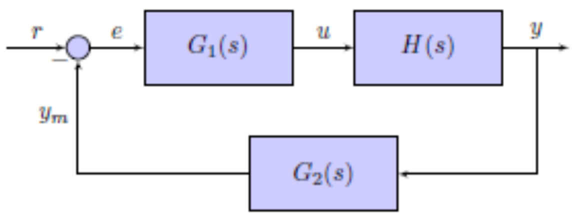


Figure 1: A plant $H(s)$ is controlled by the blocks $G_1(s)$ and $G_2(s)$.

# Classic Engineering Control

Consider now a system with plant $H(s)$ and control components $G_1(s)$ and $G_2(s)$ as before, with input signal $R(s)$ and disturbances $V(s)$ and $W(s)$. The system may be expressed in the form

$$Y = H\Big(W + G_1\big(R - G_2(Y + V)\big)\Big)$$

The *controlled plant* $H_c(s)$ is defined as

$$H_c := \frac{HG_1}{1 + HG_1G_2}$$

Using $H_c$, an alternate expression for $Y$ may be derived, and the error $E := R - Y$ may expressed purely in terms of the original system components above.

Designing the control components $G_1$ and $G_2$ leads to considering a number of other factors, which will be defined where necessary. Such factors include stability, robustness, regulation, tracking, and simplicity.

Stability generally exists in two basic forms; the ability of the system to be BIBO, and the ability of the system to return to equilibrium after an arbitrary displacement. These properties are generally distinct for non-linear and time-varying systems, but they are essentially equivalent for LTI systems. At this stage, a system is stable if the real part of the pole locations of a given transfer function are negative. The standard way to check for stability is to solve the polynomial $D(s) = 0$ for $H(s) = \frac{N(s)}{D(s)}$ and determine whether all solutions lie in the left hand complex plane. Should $H(s)$ be unstable, one should strive to design $G_1$ and $G_2$ such that $H_c$ is stable.

As in the previous section, feedback is still the key part of controlling systems. However, we introduce a widely used controller known as the *Proportional-Integral-Derivative* (PID) controller. The system is parameterised by $K_P$, $K_I$ and $K_D$, and the controller transfer function is given by

$$G_1(s) = K_P + \frac{1}{s}K_I + sK_D \quad \& \quad G_2(s) = 1$$

Based on the current rate of change, these values may be heuristically interpreted in terms of time: $P$ depends on the present error, $I$ on the accumulation of past errors, and $D$ is a prediction of future errors.

When considering a graphical method for controller design, the most intuitive approach is loop shaping. This essentially means adding poles and zeros and the right system gain to achieve the desired closed loop response, otherwise known as a *Bode plot*. They are logarithmic plots of frequency response, where gain and phase are displayed in separate plots.

The drawback of Bode plots is that they are unable to handle transfer functions with right-half plane singularities. However, this problem is circumvented by *Nyquist plots.*

Nyquist plots are useful for analysis of closed loop transfer functions, expressed as

$$H_c = \frac{N_H N_1 D_2}{D_H D_1 D_2 + N_H N_1 N_2}$$

where the rational polynomials $H$, $G_1$ and $G_2$ are expressed in terms of a numerator $N(s)$ and a denominator $D(s)$. The concepts arise from *Cauchy's principle of argument.* The key idea is that since the transfer function is complex, applying Cauchy's principle of argument to the open loop system transfer function will lead to information about the stability of the closed loop transfer function. Stability requires that the roots of the denominator lie in the LHP as they are poles of the closed loop system [2].

# State Space Description & Control of Linear Systems

This section investigates *state space representation*, which models a system as a set of input, output and state variables related by first-order equations. It will focus mainly on linear systems of the form

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

for $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$, with a discrete time analogue reminiscent of (3) and (4). For the SISO systems considered in the previous sections, $m = p = 1$. This section introduces the two *regularity conditions*, known as *controllability* and *observability.*

*Controllability* describes the ability of an external input to move the internal state of a system from any initial state to any other final state in a finite time interval. A state $x_s \in \mathbb{R}^n$ is *controllable* if there exists an input that transfers the state from $x_s$ to the zero state in some finite time. A state $x_d \in \mathbb{R}^n$ is *reachable* if the converse holds. If the entire system is controllable, then the pair $(A, B)$ is controllable, and for $k \in \mathbb{Z}$, the controllability matrix is defined to be

$$\text{con}_k(A, B) = \left[B, AB, A^2B, \ldots, A^{k-1}B\right] \in \mathbb{R}^{n\times mk}$$

In contrast, *observability* is a measure for how well internal states of a system can be inferred by knowledge of its external outputs. Observability is the dual of controllability, and as such, if the entire system is observable, then the pair $(A, C)$ is observable, with observability matrix

$$\text{obs}_k(A, C) = \left[C, CA, CA^2, \ldots, CA^{k-1}\right]^T \in \mathbb{R}^{pk\times n}$$

There exist a number of algebraic properties of these matrices, the details of which are unnecessary for the scope of this paper, with the exception that the specified matrices have full rank iff the state space model is continuous and time-invariant.

The *linear state feedback law* is denoted

$$\boldsymbol{u}(t) = F\boldsymbol{x}(t) + \boldsymbol{r}(t)$$

for some $F \in \mathbb{R}^{m\times n}$ and some external input vector $\boldsymbol{r}(t) \in \mathbb{R}^m$, known as the *reference.*

In general, $\boldsymbol{x}(t)$ is unobservable, so it's quite useful to design a system with state $\hat{\boldsymbol{x}}(t)$ so that $\hat{\boldsymbol{x}}$ is an estimate of $\boldsymbol{x}$. This is generally known as the *Luenberger observer.* For control purposes the output of the observer system is fed back to the input of both the observer and the plant through the gains matrix $K \in \mathbb{R}^{n\times p}$. The system is designed such that

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}}(t) &= A\hat{\boldsymbol{x}}(t) + B\boldsymbol{u}(t) + K\left(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)\right) \\ \hat{\boldsymbol{y}}(t) &= C\hat{\boldsymbol{x}}(t) + D\boldsymbol{u}(t) \end{aligned}$$

The input of the Luenberger observer system is the input of the original system together with the output of the system.

State feedback and observers are generally combined into a controlled system that has an observer for generating $\hat{\boldsymbol{x}}(t)$ and then uses it as an input to a *state feedback* controller [4]. Using the resulting input $\boldsymbol{u}(t) = F\hat{\boldsymbol{x}}(t) + \boldsymbol{r}(t)$ with the error $\boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)$, the resulting system may be expressed in the form

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{e}} \end{bmatrix} = \begin{bmatrix} A + BF & -BF \\ 0 & A - KC \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \boldsymbol{r}$$

$$\boldsymbol{y} = \begin{bmatrix} C + DF & -DF \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \end{bmatrix} + D\boldsymbol{r}$$

# Adaptive Control

When dealing with complex systems that have unpredictable parameter deviations and uncertainties, the fundamental requirement is a control law that adapts itself to a changing condition. This is known as *adaptive control*, and is illustrated in Figure 2. The

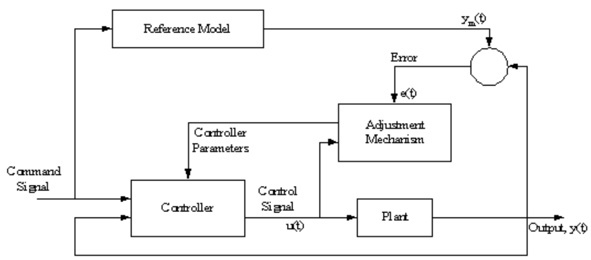


Figure 2: A more complicated system than before.

theory is inherently non-linear, and ultimately stems from Lyapunov stability theory. Parameter estimation is the foundation of adaptive control, where the most frequent methods of estimation are recursive least squares and gradient descent. These both provide update laws, laws which are used in real time to modify the estimates [3].

# Linear Quadratic Regulator & Model Predictive Control

The previous sections were largely influenced by engineering, and engineering related mathematics. This section and those that follow draw largely from pure mathematics, in particular, the theory of differential equations (as in the last section also) and the calculus of variations.

Optimal control is concerned with operating a dynamical system at minimum cost. The linear structures (1) and (3) are still used, and the cost is described by a quadratic functional, given by

$$\begin{aligned} J(\boldsymbol{u}) = {} & \boldsymbol{x}^T(t_f)Q(t_f)\boldsymbol{x}(t_f) \\ & + \int_{t_0}^{t_f} \left[\boldsymbol{x}^T(t)Q(t)\boldsymbol{x}(t) + \boldsymbol{u}^T(t)R(t)\boldsymbol{u}(t)\right]\, dt \end{aligned} \tag{6}$$

where the time horizon may be infinite, with real matrices where $Q(t) \succ 0$, $Q(t_f) \succeq 0$ and $R(t) \succeq 0$. The discrete analogue shares the same idea, but is expressed as a sum, rather than an integral. This cost may thought of as a sum or series of deviations of key measurements from their desired values, and the algorithm determines the controller settings that minimise the undesired deviations. This is generally known as the *linear quadratic regulator*.

It can be shown that the optimal control is a linear state feedback control law, and is expressed as

$$\boldsymbol{u}^*(t) = \left(-R^{-1}(t)B^T(t)P(t)\right)\boldsymbol{x}(t) \tag{7}$$

where the $n \times n$ matrix $P(t)$ solves the Riccati matrix differential equation

$$\begin{aligned} -\dot{P}(t) = {} & A^T(t)P(t) + P(t)A(t) \\ & - P(t)B(t)R^{-1}(t)B^T(t)P(t) + Q(t) \end{aligned} \tag{8}$$

In the case that $t_f = \infty$, $P(t)$ is replaced by the steady state solution $P$, and (8) becomes an algebraic equation of constant matrices with $\dot{P}(t_f) = 0$.

The theory of *model predictive control* has its basis in that of LQR, but operates on an iterative short time horizon $[t, t+T]$. An extemporaneous calculation is used to explore state trajectories that emanate from the current state and determine, through the Euler-Lagrange equations, a cost-minimising control strategy until time $t + T$ [5]. Only the first step of the control strategy is implemented, the calculations are repeated from this new state, and the process is repeated, as illustrated in Figure 3.

# Dynamic Programming

*Dynamic programming* was the USA's answer to the control problems faced by space exploration in the 1950s. It was developed by Richard Bellman and is a method which may be applied to computer programming as well as mathematical optimisation. In either case, it normally refers to simplifying a decision by breaking it down into a sequence of sub-decisions in a recursive manner.

Bellman's *Principle of Optimality* describes the method of this simplification as "an optimal policy

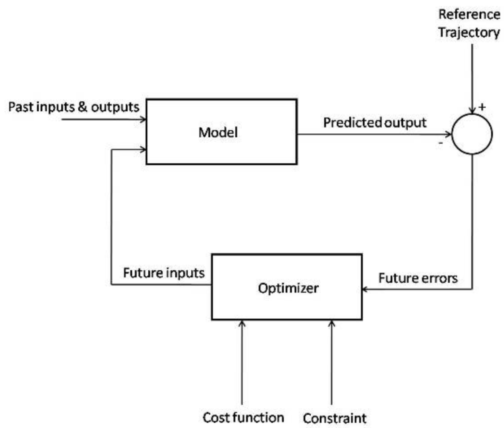


Figure 3: A block diagram representing the implementation of model predictive control.

with the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."

In practice, this is achieved by defining a sequence of value functions $V_i(y)$ representing the state at times $i = 1, \ldots, n$. At the final time $n$, $V_n(y)$ is the value obtained in state $y$, and the values at $V_i(y)$ for $i = n-1, n-2, \ldots, 2, 1$ are determined by working backwards, using the recursive relationship known as the *Bellman equation*. For all but the first time, $V_{i-1}(y)$ is calculated from $V_i(y)$ by maximising a function of the gain form decision $i-1$ and the function $V_i$ at the new state of the system if the decision is made. This operation yields $V_{i-1}$ for those states, since $V_i$ has already been calculated. The initial state of the system $V_1$ is the optimal solution.

The study of dynamic programming resulted in what is known as the *Hamilton-Jacobi-Bellman equation*, a partial differential equation central to optimal control theory. The process described by the state equation $\dot{\boldsymbol{x}}(t) = \boldsymbol{a}\big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\big)$ is to be controlled to minimise the performance measure

$$J(\boldsymbol{x}(t), \boldsymbol{u}(t), t) = h\big(\boldsymbol{x}(t_f), t_f\big) + \int_{t_0}^{t_f} g\big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\big)\, dt \tag{9}$$

Now setting $J^*(\boldsymbol{x}, t) = \min J(\boldsymbol{x}, t)$ for $t \in [t_0, t_f]$, subdividing the interval, and taking an appropriate Taylor series expansion, they HJB equation may be derived to be

$$0 = J_t^* + \mathcal{H}\Big(\boldsymbol{x}(t), \boldsymbol{u}^*\big(\boldsymbol{x}(t), J_x^*, t\big), J_x^*, t\Big) \tag{10}$$

where the *Hamiltonian* is defined to be

$$\mathcal{H}\big(\boldsymbol{x}(t), \boldsymbol{u}(t), J_x^*, t\big) = g\big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\big) + J_x^{*T}\big(\boldsymbol{x}(t), t\big)\dot{\boldsymbol{x}}(t) \tag{11}$$

and the Hamiltonian appearing in (10) is the minimum of (11) over $\boldsymbol{u}(t)$. The solution of the HJB equation yields a system of ordinary differential equations $K(t)$, which describe the nature of the optimal control function $\boldsymbol{u}^*(t)$ [6].

## Pontryagin's Minimum Principle

Whilst the USA had Bellman's Dynamic Programming in the space race, the USSR had *Pontryagin's Minimum Principle*, a technique developed by Lev Pontryagin for determining the best possible control of a dynamical system from one state to another, particularly in the presence of state or input constraints.

The goal here is to employ techniques from the calculus of variations to determine an admissible control $\boldsymbol{u}^*(t)$ that causes the system $\dot{\boldsymbol{x}}(t) = \boldsymbol{a}\big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\big)$ to follow an admissible trajectory $\boldsymbol{x}^*(t)$ that minimises the performance measure (9). Through the construction of a function called the *Hamiltonian*, given by

$$\mathcal{H}\big(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\psi}(t), t\big) = g\big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\big) + \boldsymbol{\psi}^T(t)\dot{\boldsymbol{x}}(t)$$

necessary conditions for optimality are derived. Here, $\boldsymbol{\psi}(t)$ are known as *costate equations*, which arise from constraints, and are derived from Lagrange multipliers used in the calculus of variations.

For an optimal control $\boldsymbol{u}^*(t)$, an optimal state trajectory $\boldsymbol{x}^*(t)$ and an optimal costate trajectory $\boldsymbol{\psi}^*(t)$, Pontryagin's minimum principle essentially states that

$$\mathcal{H}\big(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{\psi}^*(t), t\big) \leq \mathcal{H}\big(\boldsymbol{x}^*(t), \boldsymbol{u}(t), \boldsymbol{\psi}^*(t), t\big)$$

for $t \in [t_0, t_f]$ and all $u \in \mathcal{U}$, where $\mathcal{U}$ is the domain over which the Hamiltonian must be minimised. It may be shown that the three necessary conditions for optimality are

$$\dot{\boldsymbol{x}}^*(t) = \mathcal{H}_{\boldsymbol{\psi}} \quad \dot{\boldsymbol{\psi}}^*(t) = -\mathcal{H}_{\boldsymbol{x}} \quad 0 = \mathcal{H}_{\boldsymbol{u}}$$

for all $t \in [t_0, t_f]$. Ultimately, by substituting the optimal control in terms of the costates, the problem is reduced to a system of $n$ equations with $n$ unknowns. This allows the costate equations themselves to be determined, which leads to determining the optimal control function. The ultimate goal of PMP is the same as that of dynamic programming, which is determining the optimal control function $\boldsymbol{u}^*(t)$ [6].

## Linear Quadratic Regulator Revisited

Both dynamic programming and the calculus of variations may be used in deriving the LQR results stated before. Both methods achieve this by applying (5) and (9) to their respective Hamiltonian systems.

Dynamic programming achieves this by constructing a Hamiltonian of the form

$$\begin{aligned}\mathcal{G} = &\frac{1}{2}\boldsymbol{x}^T(t)Q(t)\boldsymbol{x}(t) + \frac{1}{2}\boldsymbol{u}^T(t)R(t)\boldsymbol{u}(t)\\ &+ J_x^{*T}\big(A(t)\boldsymbol{x}(t) + B(t)\boldsymbol{u}(t)\big)\end{aligned}$$

whereas the calculus of variations and PMP achieve this by constructing an alternate Hamiltonian of the form

$$\begin{aligned}\mathcal{H} = &\frac{1}{2}\boldsymbol{x}^T(t)Q(t)\boldsymbol{x}(t) + \frac{1}{2}\boldsymbol{u}^T(t)R(t)\boldsymbol{u}(t)\\ &+ \boldsymbol{\psi}^T(t)A(t)\boldsymbol{x}(t) + \boldsymbol{\psi}^T(t)B(t)\boldsymbol{u}(t)\end{aligned}$$

It can be shown that both of these approaches yield the derivation of both (7) and (8) [6].

## Conclusion

The different areas of control theory covered in this paper are but the tip of the iceberg. The topics of discussion were all developed prior to the 1970s, and since then, the theory of control has evolved divergently to encompass the fields of economics, computer science, and cryptographic systems amongst others. Despite the various approaches, the ultimate objective of control has been with varying degrees of success, whilst leaving room for improvement or refinement.

# Bibliography


[1] Albertos, P., Mareels, I. *Feedback and Control for Everyone*. Springer, 2010

[2] Franklin, G.F., Powell, J.D., Emami-Naeni, A. *Feedback Control of Dynamic Systems*. Pearson, 2010

[3] Astrom, K.J., Wittenmark, B. *Adaptive Control*. Addison Wesley, 2nd Edition, 1994

[4] Antsaklis, P.J., Michel, A.N. *A Linear Systems Primer*. Birkhäuser, 2007

[5] Friedland, B. *Control System Design: An Introduction to State-Space Methods*. McGraw-Hill, 1986

[6] Kirk, D.E. *Optimal Control Theory: An Introduction*. Dover Publications, 2004