On the problem of control and observation: a general overview

Krishna Kumar Busawon*

Abstract

This note is dedicated to undergraduate students who wish to enroll in the control engineering program. Some basic notions encountered in control engineering are discussed. The main difficulties encountered in the design and implementation of controllers are explained in simple terms without entering deeply into the mathematical details. Finally, some conclusions are given.

1. INTRODUCTION

The fundamental objective of control engineering, as a science discipline, is to *control* a *dynamical system.* The common meaning of the verb *to control* is *to verify, to inspect* and *to master.* However, in control engineering, *to control a system* is understood more in the sense of *mastering a system* even though the inspection and monitoring of a system is also part of the discipline. By the term *dynamical* we refer to something which is evolving with time. Finally, we must give a precise meaning to the word *system* in the context of control engineering . Indeed, the word *system* has a very broad meaning in everyday life. Normally, by this word one would understand an abstract set of things which are interconnected in some way or another; for example we talk about solar systems, meteorogical systems, political systems, physical systems etc.. Evidently, such a broad meaning would not be appropriate in the context of control engineering since a precise mathematical analysis is required for its study.

A first definition of a *dynamical control system* in the context of control engineering would be the following :

Definition. A dynamical control system is a system whose *behaviour* can be modified by some *external actions.*

For example, the meteorological system is not a control system since we cannot stop the rain from falling or the sun from shining. On the other hand, a car is a control system since we can make a car accelerate, decelerate or even stop, whenever we want to. More precisely, in this case, we would talk of *controllable* systems.

The *external actions* are known as *inputs,* commonly denoted by the function *u(t)*. They are responsible for changing the behaviour of a system. The inputs maybe measurable or nonmeasureable. Non measureable inputs are usually known as perturbations or disturbance. Somehow or other disturbances are always present in a system. In this note we shall not discuss the aspects of disturbances even though, in the majority of cases, a control design makes sense because of the presence of disturbances. The measurable inputs are in fact the only degree of freedom that we, as control engineers or technicians, have in order to influence the behaviour of a system. They are signals provided by the actuators.

Normally, we study the behaviour of a system via some signal or function which characterises the system. If the signal is measured then it is referred to as an *output* of the system and is commonly denoted by the function *y(t).* These are signals coming from the sensors. For example, the behaviour of the temperature of a room is observed via a thermometer, or the pulses of a patient's heartbeat is studied via the signal issued by a cardiogram.

* Universidad Autónoma de Nuevo León, FIME-UANL

From the above definition, one would intuitively understand that a system possesses a unique behaviour at a time. For example, a car cannot accelerate and decelerate at the same time. Mathematically speaking, a system can possess two or more behaviours at the same time. These are systems where bifurcations are present and are chaotic in nature. However, the great majority of systems in the industry possess the propriety of presenting one behaviour at a time. We shall see later what this property means in mathematical terms. But first of all, before coming to this point, we should know how to represent a system mathematically.

2. REPRESENTATION OF A SYSTEM

To study the behaviour of a system correctly, it is necessary to give a representation or a model of the system.

From the above definition a schematical representation of a system would be as follows:

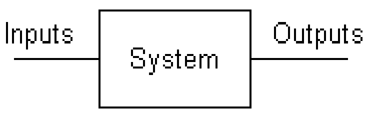

Fig.1. System

However, to study the system in a more mathematical way, control engineers usually adopt different points of views depending on the nature of the system.

Input-output point of view

One point of view would be to view a system as a *function S* which to the function *u(t)* will associate another function *y(t)*; i.e. $S: u(t) \mapsto y(t)=S(u(t))$. This described in the figure 2 below.

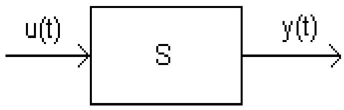

Fig.2. Input-ouput representation

This point of view is known as the input-output point of view and the relation *y(t)=S(u(t))* is known as the input-output relation. Therefore, roughly speaking, a system, from this point of view, is a function of functions. It should be noted that *u(t)* are not arbitrary functions. They are usually bounded functions and they take the value zero for negative times. These kinds of functions are known as *causal* functions. In simple terms this reflect the fact that an action cannot take place in negative times. It takes place only at the instant it starts acting on the system. For example, a car will not start before we turn the key or the temperature in a room will not rise before we turn the heater on.

It is well-known that for linear systems the function *y(t)* is given by

$$y(t)=\int_0^t h(v)u(t-v)dv.$$

In the case of linear systems, we can also work in the frequency domain instead of the time domain and the above input-output relation is given by the well-known transfer function

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

where *Y(s), H(s), U(s)* are the respective Laplace transforms of the functions *y(t), h(t)* and *u(t).*

State-space point of view

Another point of view would be to assume that the system is characterised by a time-dependent vector *x(t)* known as the *state* of the system. The state of a system is in fact the minimum number of variables that are needed to characterise a system completely. For example, a moving particle is completely characterised by its position and velocity. Any additional variable would necessarily be a function of the position and velocity, and would constitute a redundant information on the system.

The state of a system is a vector which evolves in time and is therefore characterised by a differential equation of the form

$$(\Sigma): \quad \frac{dx}{dt} = f(x, u), \qquad y = h(x, u)$$

where $x=(x_1, x_2, ..., x_n) \in R^n$; $u=(u_1, u_2, ..., u_m) \in R^m$ and $y = (y_1, y_2, ..., y_p) \in R^p$.

This means that we have n state variables which characterise the system, m inputs acting on the system and p variables which are measured from the system. The number n is known as the *dimension* of the system. The outputs $y_1, y_2, ..., y_p$ are supposed to be independent of each other. This representation is illustrated in the figure below:

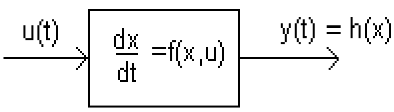

Fig 3. State-space representation

The number p is usually less than or equal to n. In the case where p is strictly less than n, this means that only part of the state variable can be measured. We shall see later that this has important implications regarding the implementation of a control law.

If the system is linear then the function $f(x, u)=Ax+Bu$ and $y=Cx+Du$, where A is a $n \times n$ matrix, B is a $n \times m$ matrix, C is a $p \times n$ matrix and D is a $p \times m$ matrix.

Knowledge-based representations

The above two point of views are based upon a mathematical description of the system. However, when the system is of a very large dimension or is very complicated and its dynamics not well-known, it is not always easy to provide an adequate mathematical model of the system. In such a case, we can give a representation of the system based upon the qualitative knowledge that we have on the system. Representation via artificial intelligence, expert systems, neural networks, fuzzy logic, all fall in this category.

In this note, we shall only study the state-space representation.

It is important to note that all of these different representations are equivalent. The choice of a particular representation is basically motivated by two factors :

i) the nature of the system ; i.e. whether it is linear, nonlinear or difficult to model etc ..

ii) the degree of complexity brought by a particular representation ; i.e. whether one representation is simpler than another.

Example

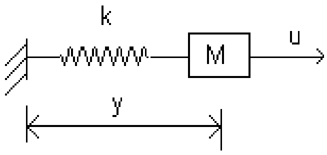

Fig. 4 : Spring mass system

Consider the above spring mass system. Here, u is the force pulling the mass M and y is the distance between the center of mass and the point where the spring is fixed. The quantity y can be measured easily and is considered as an output of the system. By Newton's laws of motion, we have

$$m\frac{d^2y}{dt^2} + k\frac{dy}{dt} - u=0$$

This in fact an input-output relation ship.

To obatain a state-space representation we set $x_1=y$ and $x_2=\frac{dy}{dt}$. Then, a simple computation shows that

$$\frac{dx_1}{dt} = x_2$$

$$\frac{dx_2}{dt} = -\frac{k}{m}x_1 + \frac{u}{m}$$

State space analysis

This particular representation of a system is the most commonly used in control engineering nowadays and is normally termed *modern control engineering*. This is mainly because there is no restriction on the nature of inputs applied to the system for its analysis. Recall that classically a linear system is studied in the frequency domain using Bode or Nyquist plots. For such analysis the inputs are resticted to either a step function, a ramp or a sinusoidal function. However, in the state space representation such restrictions are not necessary. Another reason for the popularity of state space representation is that complicated control laws can easily be implemented due to the advent of computers. This was not possible several decades ago.

The study of a system given in state space form requires some knowledge of linear algebra if the system is linear, or differential geometry if the system is nonlinear and obviously some knowledge of diffrential calculus.

As we have mentioned before, the state is the quantity which characterises a system. Therefore, we shall first start by studying the solution of system (Σ).

Suppose that time $t=t_0$, the initial value of the state is $x(t_0)$. Then, by a solution or trajectory of system (Σ), we mean any function $x_u(t\,;\,t_0, x_0)$ which satisfies the above differential equation with the condition that $x_u(t_0\,;\,t_0, x_0) = x(t_0)$. In fact, by abuse of language, the function $x_u(t\,;\,t_0, x_0)$ is simply denoted by $x(t)$.

The way that this solution evolves in time characterises the behaviour of the system. That is the solution may be increasing or decreasing with time or may be constant etc. Roughly speaking, if the solution is unbounded then we say that the system is not stable otherwise it is stable.

Example 1.

Consider the system given by

$$\frac{dx}{dt} = x+u$$

Assume that at time $t_0=0$ the value of the state is $x(0)=x_0$. For $u=k$ where k is a constant, the solution is given by : $x_u(t\,;\,t_0, x_0) := x(t) = e^t(k+x_0)-k$. In particular, for $u=0$, the solution is given by $x(t) = e^t x_0$. Since the solution tends to infinity when t goes to infinity, the system is unstable.

Notice that the solution of a system depends on its initial condition and on the input applied to the system. This means that if we change the expression of $u(t)$ then the solution will also change. In particular, if the solution is unbounded, we can try to replace $u(t)$ by another function such that the solution becomes bounded. This is the fundamental purpose of control engineering.

Therefore, designing a control law for a system means designing a function, which is possibly a function of the state or the output of the system, such that the solution of the system behaves in a desired manner.

If the control law is a function of the state, $u(t) := a(x)$, then we call the control a *state feedback* control law. On the other hand, if the control law is a function of the output only, $u(t) := a(y)$, then we call the control law an *output feedback* control law.

Example 2.

Consider again the above system:

$$(S1): \qquad \frac{dx}{dt} = x+u$$

If we replace u by $u = a(x) = -2x$. The the system becomes

$$(S2): \qquad \frac{dx}{dt} = -x$$

and the solution is given by : $x(t) = e^{-t}x_0$. It is easy to see that the solution now goes to zero when t goes to infinity. Therefore, system (S2) is stable. In the control engineering jargon, system (S1) is said to be in *open-loop* whereas system (S2) is said to be in *closed-loop*.

It is important to notice that in control engineering the functions $a(x)$ and $a(y)$ is denoted by $u(x)$ and $u(y)$ respectively ; i.e. $u(t) := u(x)$ or $u(t) := u(y)$. This *does not* mean that t is replaced by x or y in the function $u(t)$. It instead means that the *function* $u(t)$ is replaced by the *function* $u(x)$. This is an abuse of language which is very misleading for many students.

Let us now come back to the definition of 'system' that we gave in the introduction. We said that a system in control engineering should not have two or more solutions for one initial condition. We might then ask under which condition can we guarantee that system (Σ) will have a unique solution for a given initial condition. It is well-known that if the function f in system (Σ) is locally Lipschitzian, then there exists a unique solution for system (Σ) for a given initial condition. A local Lipschitz function is generally a continuous function and does not presents any jumps. For example, the sign function is not a local Lipschitz function.

There are two important implications for the existence and the unicity of the differential equation (Σ). First, we have seen above that controlling a system means controlling its trajectory or solution. Now, if we have two trajectories for one initial condition and we do not know in which one of them the system is evolving, then it would be difficult to control the system. We might be controlling the wrong trajectory! Secondly, if the system satisfies the condition of existence and unicity of solution, then the trajectories issued from two different initial conditions will never intersect.

Difficulties in control design

We shall now talk about some difficulties which are encountered in the process of designing a control law for a system. We shall discuss the problem of how to design a control law and the particular techniques which exist for designing a control law. We shall rather point out the main difficulties which one would encounter in control design independently of the technique used for designing the control law.

The first difficulty is related to the possibly of designing the control law. It is always possible to design a controller for a system. We must first check if the system is controllable ; if not, we cannot design a control law. Roughly speaking, the controllability of a system is the property which determines whether we can modify the behaviour of a system. Another difficulty is that the majority of systems in nature are nonlinear and the mathematical tools for analysing a nonlinear system is not totally established up to now. In particular, there does not exist a direct analytical method to solve a nonlinear differential equation. Fortunately enough, there exists methods to study the behaviour of a system without calculating the solution of a system (Lyapunov method).

Implementation problem

Once we have designed a control law we need to know if the latter can be implemented for application purposes. It usually happens, especially when we are have designed a state feedback control, that some state variables that are involved in the control law are not available for measurement. This is the case where *p* (the number of outputs) in equation (Σ) is strictly less than *n* (the number of state variables). In such a case, the control law cannot be implemented.

One solution would be to design additional (hardware) sensors. However, this is not always possible and may be very costly at times. For example the rotor flux or current in an induction motor cannot be easily measured. Similarly, it is very difficult to obtain the online concentration of certain components in some chemical reactions. Consequently, there has been some incentive to find other cheap methods to obtain the nonavailable state variables. In this respect, it is important to mention the new emerging micro-sensor technology. Another alternative consists in the design of observers. More specifically, an observer is an auxiliary dynamical system which uses the available measurements on the systems (inputs and outputs) in order to provide an estimate $\hat{x}$ of the state of the system. This is schematically represented in the figure below.

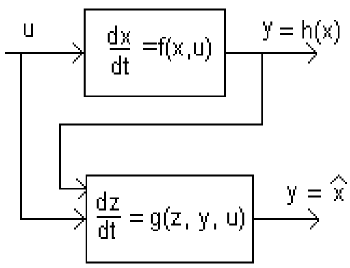

Fig.5. Observer

The dynamical nature of an observer means that the estimates of the state variable are provided on line. Basically, an observer is a software sensor. Consequently, the cost of realising an observer is relatively low.

From a mathematical point of view, observers and controllers designs are dual problems. Consequently, similar difficulties as for the controller design are encountered in the process of observer design.

CONCLUSIONS

In this note we have briefly presented some basis problems that are encountered in control engineering. We showed the main difficulties of control design and its implementation. We have also highlighted the different mathematical tools which are needed to analysize a control system depending on the representation chosen for the latter. It is hoped that this simple note will incite some interest to undergraduate students who wish to enroll in the subject.

REFERENCES

1. W. L. Brogan, *Modern control theory*, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1982.
2. C. T. Chen, *Introduction to linear system theory*, HRW series in Electrical Engineering, Electronics, and Systems, 1970.
3. P. A. Isermann, *Digital control systems*, Springer, Berlin New York, 1981.
4. K. Ogata, *Discrete-time control systems*, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1987.