NON-LINEAR SYSTEM CONTROL

1. Introduction

In practice, all physical processes exhibit some non-linear behaviour. Furthermore, when a process shows strong non-linear behaviour, a linear model may be inadequate; a non-linear model will be more realistic. Unfortunately, this improved closeness to reality is attained at a cost: the convenience and simplicity offered by the linear model is sacrificed. Since many processes exhibit only mildly non-linear dynamic behaviour, linear models can reasonably approximate them. However, even for such systems, the behaviour over a wide range of operating conditions will be noticeably non-linear, and no single linear model is able to represent such behaviour adequately.

The issue of analysing the dynamic behaviour of processes represented by non-linear models is the main concern in this topic; this is motivated by the fact that situations do arise when it is undesirable to neglect the inherent nonlinearities of a process. The objective here is to provide an introduction of how we can expect to carry out dynamic analysis and control when the process models are non-linear.

The nature of non-linear problems is that they are not as easily amenable to the general complete treatment which are possible with linear systems. To keep non-linear dynamic analysis on a realistic and practical level usually involves trade-offs between accuracy and simplicity.

There are two basic properties that characterize the behavior of a linear system:

**Superposition:**

The response of a process to input $I_1$ is $R_1$ and the response to $I_2$ is $R_2$, then according to the principle of superposition, the response to $I_1 + I_2$ is $R_1 + R_2$ if the system is linear.

In general, this principle states that the response of a linear system to a sum of $N$ inputs is the same as the sum of the responses to the individual inputs.

As a result of this principle, for example, the response of a linear system to step change of magnitude A is exactly the same as A times the system's unit step response.

**Independence of Dynamic Response Character and Process Conditions**

The dynamic character of a linear process response to an input change is independent of the specific operating conditions. I.e, identical input changes implemented at different operating steady-state conditions will give rise to output changes of identical magnitude and dynamic character.

The principle of superposition also means that from the same initial steady-state conditions, the output change observed in response to a certain positive input change will be a perfect mirror image of the output observed in response to a negative input change of the same magnitude.

According to this property, the step response of a linear system is the same regardless of the actual initial value of the output or input variable, or, the direction of the input change, the "step down" giving rise to a perfect mirror image of the "step up" response.

A nonlinear system does not exhibit any of these properties:
Non-linear Process Control

- the response to a sum of inputs is not equal to a sum of the individual responses;
- the response to a step change of magnitude $A$ is not equal to $A$ times the unit step response;
- the magnitude and dynamic character of the step response are dependent on the initial steady-state operating conditions;

In particular, the ultimate response of the nonlinear system to a sinusoidal input function is not a perfect sinusoidal output function.

The model equations that will represent the dynamic behavior of such systems adequately must necessarily contain nonlinear terms; and immediately, the convenience of being able to completely characterize the system's dynamic behavior by the linear transfer function is lost.

1.1 Methods of Dynamic Analysis of Nonlinear Systems

Since the process models are almost always nonlinear, the following major questions we wish to ask ourselves are:

How do we analyze the dynamic behavior of processes when they are represented by nonlinear models? Is there a technique for analyzing the dynamic behavior of nonlinear systems that can be generalized to cover a wide variety of such systems?

The answers to these questions form the basis of the material covered in this Section

1. The rigorous analytical approach
2. Numerical analysis
3. Approximate linearization
4. Exact linearization by variable transformation

Each of these method is briefly discussed below:

**Rigorous Analytical Techniques**

It is well known that very few nonlinear problems can be solved formally, primarily because there is no general theory for the analytical closed-form solution of nonlinear equations. Nevertheless, there are a few analytical method for analyzing the dynamic behavior of nonlinear systems; for example, classical method and the techniques of Liapunov, Hopf, Golubitsky, etc. These techniques will allow one to characterize qualitatively the dynamics of nonlinear systems.

However, these approaches find limited use in process control practice; thus a detailed discussion of their salient features has been excluded from the context. Perhaps, with further research, the analytical treatment of nonlinear systems will evolve to a state where it will be more useful for process control.

**Numerical Solutions and Computer Simulations**

The first step in the analysis of a system dynamic behavior is the solution of the modeling equation; and the models, are usually in the form of differential equations (ordinary or partial). For linear systems, analytical solutions are possible. For nonlinear system, instead of having an analytical expression that represents the behavior of the process output over time, we make use of numerical values for the process output at specific point of time.
Note that the recursive calculations required by this procedure are best carried out with the aid of a digital computer. There are many special subroutines available for the numerical integration of differential equations, so that details of this procedure are usually handled by the software.

**Variable Transformation**

On the basis of the fact that linear systems are easier to analyze than nonlinear ones, it seems attractive to consider the following proposition:

*If a system is nonlinear in its original variables is it possible that it will be linear in some transformations of these original variables?*

If this were possible, the general strategy for dynamic analysis would then involve first carrying out this transformation, then performing dynamic analysis on the linear, transformed version, and finally transforming the results back to the original variables.

**Linearization**

In the contest of process dynamic and control, linearization is a term used in general for the process by which a nonlinear system is approximated by a linear process model.

The most popular technique for obtaining the linear approximation is based on Taylor series expansions of the nonlinear aspects of the process model.

**Summary:** The first two approaches are clearly of limited control applications. The technique of numerical analysis, even though straightforward, provides limited information because it is usually impossible to infer anything of a general form from the numerical response of a process system. It provides specific *numerical* answers to specific problems; it is not possible to obtain general solutions in terms of arbitrary parameters and unspecified inputs in order to understand the process behavior in a more general fashion, as was the case with linear systems.

The question now is: can we ever achieve this desirable objective of being able to analyze, in a general fashion, the behavior of an arbitrary nonlinear system, in response to unspecified inputs? The answer is *yes*: but at a cost.

Observe that the nonlinear process model can be *linearized* around a particular steady-state value, and with this linearized approximation of the true nonlinear model, we can obtain fairly general results regarding the process behavior, but these will only be approximate, never completely representing true behavior accurately.

### 1.2 Nonlinear Control

The objective here is to examine the main aspects of the rapidly developing subject of nonlinear process control, and to provide a summary of the various alternative techniques available for designing control systems, including those that explicitly recognize the nonlinearities of the process. Control system may be carried out for nonlinear systems along the lines indicated by any of the following four schemes.

**Local Linearization**

This involves linearizing the modeling equations around a steady-state operating condition and applying linear control systems design results. It is obvious that the controller performance will deteriorate as the process moves further away from the steady state around which the model was linearized, but quite often, adequate controllers can be designed this way.
Local Linearization with Adaptation
This scheme seeks to improve on Scheme 1 by recognizing the presence of nonlinearities and consequently providing the controller a means for systematically adapting whenever the adequacy of the linear approximation becomes questionable.

Exact Linearization by Variable Transformations
This is a scheme by which a system model that is nonlinear in its original variables is converted into one that is exactly linear in a different set of variables by the use of appropriate nonlinear transformations; controller design may then be carried out for the transformed system with greater facility since it is now linear.

"Special Purpose" Procedures
These are usually custom made for specific processes, or specific types of nonlinearities. Some of the factors that typically determine which of these approach one should adopt are now summarized below:

- The specific nature of the nonlinear control problem at hand; the nature of the process, and the control system performance objectives.
- The amount of time available for carrying out the design.
- The type of hardware available for implementing the controller.
- The availability and quality of the process model.

2. Linearization
While there may be an extensive understanding of the behavior of non-linear processes, satisfactory methods for their control are still evolving. The prevalent approach to date has been to use a model of the process linearised about a steady-state operating point to design a linear controller such as the classical PID algorithm. This is only an approximate solution.

The General Linearization Problem
Consider the general nonlinear process model:

\[
\frac{dx}{dt} = f(x,u) \\
y = h(x)
\]

(2.1)

where \( f(*,\ast) \) is an arbitrary nonlinear function of the two variables, \( x \), the process state variable, and \( u \) the process input; \( h(*) \) is another nonlinear function relating the process output, \( y \) to the process state variable \( x \).

The linearized approximation of this very general nonlinear model (2.1) may now be obtained by carrying out a Taylor series expansion of the nonlinear functions around the point \((x_s,u_s)\). This gives:
\[
\frac{dx}{dt} = f(x_s, u_s) + \left( \frac{\partial f}{\partial x}(x_s, u_s) \right) (x - x_s) + \left( \frac{\partial f}{\partial u}(x_s, u_s) \right) (u - u_s) + \text{higher order terms}
\]
\[
y = h(x_s) + \left( \frac{\partial h}{\partial x}(x_s) \right) (x - x_s) + \text{higher order terms}
\]

Ignoring the higher order terms now gives the linear approximation:
\[
\frac{dx}{dt} = f(x_s, u_s) + a(x_s, u_s)(x - x_s) + b(x_s, u_s)(u - u_s)
\]
\[
y = h(x_s) + c(x_s)(x - x_s)
\]

where
\[
a(x_s, u_s) = \left( \frac{\partial f}{\partial x}(x_s, u_s) \right)
\]
\[
b(x_s, u_s) = \left( \frac{\partial f}{\partial u}(x_s, u_s) \right)
\]
\[
c(x_s) = \left( \frac{\partial h}{\partial x}(x_s) \right)
\]

It is customary to express the equation in terms of deviation variables:
\[
\ddot{x} = (x - x_s)
\]
\[
\ddot{u} = (u - u_s)
\]
\[
\ddot{y} = (y - y_s) = y - h(x_s)
\]

If addition to this, the linearization point \((x_s, u_s)\) is chosen to be a steady-state operating condition, then observe from the definition of a steady state that both \(dx_s/dt\) and \(f(x_s, u_s)\) will be zero. (2.2) then becomes
\[
\frac{d\ddot{x}}{dt} = a\ddot{x} + b\ddot{u}
\]
\[
\ddot{y} = c\ddot{x}
\]

where for simplicity, the arguments have been dropped from \(a, b,\) and \(c\). A transform-domain transfer function model may now be obtained by the usual procedure; the result is:
\[
\ddot{y}(s) = \left[ \frac{c(x_s)b(x_s, u_s)}{s - a(x_s, u_s)} \right] \ddot{u}(s)
\]

With the transfer function as indicated in the square brackets. This transfer function should provide an approximate linear model valid in a region close to \((x_s, u_s)\)

The principles involved in obtaining approximate linear models by linearization may now be summarized as follows:
Identify the functions responsible for the nonlinearity in the system model.

Expand the nonlinear function as a Taylor series around a steady state, and truncate after the first-order term.

Reintroduce the linearized function into the model; simplify, and express the resulting model in terms of deviation variables.

The application of these principles to specific problems is usually much simpler can be demonstrated by examples.

**Example:** linearization of a nonlinear model involving a nonlinear function of a single variable.

Consider a Liquid Level System shown as in Fig. 2.1.

![Fig. 2.1 Liquid Level System](image)

Material balance equations to Fig. 2.1 yield

\[
A \frac{dh}{dt} = F_i - F
\]  
(2.5)

But

\[
F = c\sqrt{h}
\]  
(2.6)

where \(c\) is flow resistance. Combining (2.5) and (2.6), we have

\[
A \frac{dh}{dt} = F_i - c\sqrt{h}
\]  
(2.7)

This is a nonlinear equation. To linearise this equation, one can use Taylor’s series. That is, around \(h = h_s\),

\[
f(h) = \sqrt{h} = h^{\frac{1}{2}} \approx (h_s)^{\frac{1}{2}} + \frac{1}{2} (h_s)^{-\frac{1}{2}} (h - h_s) + \text{higher order terms}
\]  
(2.8)

The approximation is shown as in Fig. 2.2
Substitute (2.8) into (2.7) and noting that the steady state flow

\[ F_{is} = c(h_s^{1/2}) \]

we have

\[ \tau \frac{dy}{dt} = Ku - y \] (2.9)

where

\[ \tau = A \frac{2(h_s^{1/2})}{c}; \quad u = F_i - F_{is}; \quad y = h - h_s \]

From (2.9), we get

\[ Y(s) = \frac{K}{\tau s + 1} U(s) \] (2.10)

The linearised system is given in Fig. 2.3.

Fig. 2.3 Linearised Liquid level system (about \( h = h_s \))

Linear controller can now be applied to the linearised system of Fig. 2.3.

Fig. 2.4 shows a feedback loop where a Proportional and Integral (PI) controller controls the linearised liquid level system of Fig. 2.3. Notice that the linearised system is an approximation since it is derived for a particular level \( h = h_s \). If the level changes, \( K \) and \( \tau \) will change with it as well.
\[
\frac{c}{r} = \frac{gg_c}{1 + gg_c}
\]

where

\[
g = \frac{K}{1 + s\tau}; \quad g_c = K_c(1 + \frac{1}{\tau_i s})
\]

The characteristic equation can be put as

\[
s^2 + s\left(1 + KK_c\right) + KK_c = 0
\]

Assuming the closed loop poles to be at the locations \(-2\pm 2j\), which correspond to the roots of the characteristic equation, for a linearised system with \(K = 2.38\) and \(\tau = 0.59\), we have \(K_c = 0.58\) and \(\tau_i = 0.29\).

**Example:**
Linearization of a nonlinear model involving a nonlinear function of two variables.

The dynamic behavior of the liquid level \(h\) in the conical storage tank system shown in Figure 2.5 can be shown to be represented below, where now the cross section area of the tank is given by:

\[
A = \pi r^2 = \pi \left(\frac{R h}{H}\right)^2
\]

so that the tank model becomes:

\[
\frac{dh}{dt} = \frac{\alpha F_i}{h^2} - \beta h^{-3/2}
\]  \hspace{1cm} (2.12)

where \(\alpha\) and \(\beta\) are parameters defined by:

\[
\alpha = \frac{1}{\pi} \left(\frac{H}{R}\right)^2, \quad \beta = c\alpha
\]

Here, \(F_i\) is, again, the inlet flowrate, the manipulated variable. Obtain an approximate linear model for this system.
Fig. 2.5 Conical storage tank system

Solution

The process model (2.12) has two types of nonlinear functions: \( F_i h^{-2} \), a product of two functions, and \( h^{-3/2} \). We shall have to linearize each of these functions separately around the steady state \((h_s, F_{is})\).

The linearization of \( f(h, F_i) = F_i h^{-2} \) proceeds as follows:

\[
f(h, F_i) = f(h_s, F_{is}) + \left( \frac{\partial f}{\partial h}(h_s, F_{is}) \right)(h - h_s) + \left( \frac{\partial f}{\partial F_i}(h_s, F_{is}) \right)(F_i - F_{is}) + \text{higher order terms}
\]

whereupon carrying out the indicated operations now gives:

\[
f(h, F_i) = f(h_s, F_{is}) - 2F_{is}h_s^3(h - h_s) + h_s^{-2}(F_i - F_{is}) \tag{2.13}
\]

if we ignore the higher order terms.

The steps involved in linearizing the second nonlinear term are no different from those illustrated in the previous example; the result is:

\[
h^{-3/2} = (h_s)^{-3/2} - \frac{3}{2} \left( \frac{\alpha}{\beta} \right)(h - h_s) \tag{2.14}
\]

We may now introduce these expressions in place of the corresponding nonlinear terms in (2.12). Recalling that under steady-state conditions \( \alpha F_{is} = \beta h_s^{1/2} \), and introducing the deviation variables \( y = (h - h_s) \) and \( u = (F_i - F_{is}) \), the approximate linear model is obtained upon further simplification as:

\[
\tau \frac{dy}{dt} + y = Ku \tag{2.15}
\]

where the steady-state gain, and time constant associated with this approximate linear model, are given by:

\[
K = \frac{2\alpha}{\beta} h_s^{1/2} = \frac{2}{c} h_s^{1/2}
\]

and

\[
\tau = \frac{2}{\beta} h_s^{5/2}
\]

If we desire an approximate transform-domain transfer function model, Laplace transformation again gives (2.10). Note that the approximate linear model for the conical tank has the same process gain, \( K \), as for the cylindrical tank; but the time constant is a much stronger function of the liquid level \( h_s \).

Some important points to note about the results of these two illustrative examples are the following:
In each case, the approximate transfer function model is of the first-order process, with the
time constants, and steady-state gain values dependent on the specific steady state
around which the system model was linearized.

Using such approximate transfer function models will give approximate results which are
good in a small neighborhood around the initial steady state; farther away from this
steady state, the accuracy of the approximate results becomes poorer.

Using such approximate models, we are able to get a general (even if not 100% accurate)
idea about the speed and magnitude of response to expect from any \textit{arbitrary} nonlinear
system. Numerical analysis can give more accurate information, but only about the
specific response to a specific input, starting from a specific operating condition for a
specific set of parameters.

Let us conclude with a comparison of the responses of the nonlinear model and an approximate
linear model for the conical tank of the Example. Suppose that $\alpha = 2$, $\beta = 1$, $c = 0.5$ so that the
nonlinear model is:

$$\frac{dh}{dt} = \frac{2F_i}{h^2} - h^{-3/2}$$

while the approximate linear model has the transfer function

$$Y(s) = \frac{4h_{ss}^{1/2}}{2h_{ss}^{5/2}s + 1}U(s)$$

which in the time domain is

$$\frac{dh}{dt} = \frac{1}{2h_{ss}^{5/2}} \{ h_s - h(t) + 4h_{ss}^{1/2} [F_i - F_{i\alpha}] \}$$

The step response (up and down) for these models is shown in Figure 2.6 for steps of two
different magnitudes. The response to a sine wave input of two different amplitudes is shown in
Figure 2.7.

For small magnitude step inputs and small amplitude sine waves, the linear model is a
good approximation of the nonlinear model;

For the large amplitude inputs the linear model predictions deviate somewhat from the
nonlinear model behavior.

The distinct differences in nonlinear response for a step up and step down.
Fig. 2.6 Response of models (2.15)/(2.12) for step inputs up/down $\Delta F_i = \pm 0.05$ and $\Delta F_i = \pm 0.15$

Fig. 2.7 Response of models (2.15)/(2.12) to sine wave inputs amplitudes $A = 0.05, 0.15$.

However, in some situations this may be inadequate (e.g. for the control of highly non-linear processes or batch systems) so the development of nonlinear controllers has featured prominently in process control in the last decade. In the following Sections, we will give an overview of some of the more popular non-linear methodologies that have been proposed for use within the process control industries including, Adaptive Control approaches; Globally Linearising Control (GLC) and Generic Model Control (GMC).

### 3. Adaptive Controls

A controller designed for the systems in the level control example in last section using the approximate transfer function model will function acceptably as long as the level, $h$, is maintained at, or close to, the steady-state value $h_s$. Under these conditions $(h - h_s)$ will be small enough to make the linear approximation adequate. However, if the level is required to change over a wide range, the farther away from $h_s$ the level gets, the poorer the linear approximation.
Observe also from the approximate transfer function model in (2.10) that the apparent steady-state gain and time constant are dependent on the steady state around which the linearization was carried out. This implies that in going from one steady state to the other, the approximate process parameters will change.

The main problem with applying the classical approach under these circumstances is that it ignores the fact that the characteristics of the approximate model must change as the process moves away from \( h_s \) if the approximate model is to remain reasonably accurate. The immediate implication is that any controller designed on the basis of this changing approximate model must also have its parameters adjusted if it is to remain effective.

In the adaptive control scheme, the controller parameters are adjusted (in an automatic fashion) to keep up with the changes in the process characteristics. We know intuitively that, if properly designed, this scheme will be a significant improvement over the classical scheme.

There are various types of adaptive control schemes, differing mainly in the way the controller parameters are adjusted. The three most popular schemes are: scheduled adaptive control, model reference adaptive control, and self-tuning controllers. We shall discuss each of these in the following subsections.

3.1 Scheduled Adaptive Control

A scheduled adaptive control scheme is one in which, as a result of a priori knowledge and easy quantification of what is responsible for the changes in the process characteristics, the commensurate changes required in the controller parameters are programmed (or scheduled) ahead of time. This type of adaptive control, sometimes referred to as gain scheduling, is illustrated by the block diagram in Fig. 3.1, which is an improvement over that of Fig.2.3.

![Scheduled Adaptive Controller](image)

Continuing with the example of liquid level system, Fig. 2.3 represents control of linearised approximation of the original non-linear system. The approximation is in the sense that the process transfer function \( g \) only holds good for a particular level of the liquid in the tank. If the level \( h \) changes drastically, the transfer function \( g \) may fail to represent the original system even approximately. Then there is a need to change the transfer function \( g \) to account for the change in level \( h \). One way of doing this is through Scheduled adaptive controller shown in Fig. 3.1.

From (2.10), it is seen that both \( K \) and \( \tau \) are affected by the liquid level \( h \). The effect on \( K \) is more prominent since it affects the system gain. The effect on the time constant is less apparent and not so prominent. Thus we only compensate for the effect of liquid level \( h \) on \( K \). This is done in a way that the overall system gain \( KK_c \) is constant. Notice that, from (2.10),

\[
K = \frac{2(h_s)^{\frac{2}{3}}}{c}
\]
So, to keep \( KK_c = K_0 \) (a constant),

\[ K_c = \frac{K_0}{K} = \frac{K_0 c}{2(h)^2} \]

This means that the controller gain \( K_c \) has to be adjusted inversely as the square root of the liquid level \( h \). This is what will be done by the block represented as “Parameter adjustment” in Fig. 3.1.

More generally, if \( K = K(t) \), that is, the process gain changes with time, then to keep the product \( KK_c \) constant it is necessary to adjust \( K_c \) inversely with respect to \( K(t) \), i.e.,

\[ K_c \propto 1/ K(t) \]

\( K(t) \) itself can be determined periodically by inserting a signal \( u \) into the process and monitoring the process output \( y \).

In practice, it is often possible to find measured variables that correlate well with changes in process dynamics. These variables can be used to change the controller parameters, using a pre-calculated schedule. The controller parameters are computed off-line for several operating conditions and stored in memory. Gain scheduling can also be based on nonlinear transformations such that the transformed system does not depend on the operating conditions.

It is difficult to give general rules. Each case must be treated individually. The key question is to determine the auxiliary variables to be used as scheduling variables. It is necessary to have a good insight into the dynamics of the process if gain scheduling is going to be used. The controller can be automatically tuned at different operating points and the resulting tuning parameters can be saved and a schedule created.

**Example:** Fig. 3.2 shows different valve characteristic, if the scheduling variable in a level control system is the opening of a control valve, then a schedule may look as in Table 3.1:

![Fig. 3.2 Different valve characteristic](image)

**Table 3.1 Scheduled control for Valve**

<table>
<thead>
<tr>
<th>Valve opening</th>
<th>( Kp )</th>
<th>( Ti )</th>
<th>( Td )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00-0.15</td>
<td>1.7</td>
<td>95</td>
<td>23</td>
</tr>
<tr>
<td>0.15-0.22</td>
<td>2</td>
<td>89</td>
<td>22</td>
</tr>
<tr>
<td>0.22-0.35</td>
<td>2.9</td>
<td>82</td>
<td>21</td>
</tr>
<tr>
<td>0.35-1</td>
<td>4.4</td>
<td>68</td>
<td>17</td>
</tr>
</tbody>
</table>

**Example:** The nonlinearity of the valve is assumed to be:
Let \( f^{-1} \) be an approximation of the inverse of the valve characteristic. To compensate for the non-linearity, the output of the controller fed through this function before it is applied to the valve. This gives the relation:

\[
v = f(u) = f(f^{-1}(c))
\]

where \( c \) is the output of the PI controller.

Assume that \( f(u) = u^4 \) is approximated by two lines as shown in Fig. 3.4. One from \((0,0)\) to \((1.3,3)\) and the other from \((1.3, 3)\) to \((2,16)\).

Then we have:

\[
\hat{f}^{-1}(c) = \begin{cases} 
0.433c & 0 \leq c \leq 3 \\
0.0538c + 1.139 & 3 \leq c \leq 16 
\end{cases}
\]

Simulation Results
3.2 Model Reference Adaptive Controller (MRAC)

The key component of the MRAC scheme is the reference model that consists of a reasonable closed-loop model of how the process should respond to a set-point change. This could be as simple as a reference trajectory, or it could be a more detailed closed-loop model. The reference model output is compared with the actual process output and the observed error $e_m$ is used to drive some adaptation scheme to cause the controller parameters to be adjusted so as to reduce $e_m$ to zero. The adaptation scheme could be some control parameter optimization algorithm that reduces the integral squared value of $e_m$ or some other procedure. This is an adaptive control technique where the performance specifications are given in terms of a model. The model represents the ideal response of the process to a command signal. The controller has two loops:

- The *inner loop*, which is an ordinary feedback loop consisting of the process and the controller.
- The *outer loop*, which adjust the controller parameters in such a way that the error $e = y - y_m$ is small (not trivial)
Fig. 3.7 Model Reference Adaptive Control Scheme

**Approaches:** Gradient approach

Tracking error:

\[ e = y - y_m \]  (3.1a)

Introduce the cost function \( J \):

\[ J(\theta) = \frac{1}{2} e^2 \]  (3.1b)

Where \( \theta \) is a vector of controller parameters. Change the parameters in the direction of the negative gradient of \( e^2 \)

\[ \frac{d\theta}{dt} = -\gamma \frac{dJ}{dt} = -\gamma e \frac{de}{dt} \]  (3.1c)

where \( \frac{de}{d\theta} \) is called the sensitivity derivative. It indicates how the error is influenced by the adjustable parameters \( \theta \).

**Example: MRAS of a first order system**

Process:

\[ \frac{dy}{dt} = -ay + bu \]

Model:

\[ \frac{dy_m}{dt} = -a_m y_m + b_m u_c \]

Controller:

\[ u = \theta u_c - \theta_2 y \]

Closed loop system:
\[
\frac{dy}{dt} = -ay + bu = -ay + b(\theta u_c - \theta y) = -(a + b\theta) y + b\theta u_c
\]

Ideal controller parameters for perfect model-following:
\[
\theta_1^0 = \frac{b_m}{b}; \quad \theta_2^0 = \frac{a_m - a}{b}
\]

Derivation of adaptive law
Error:
\[
e = y - y_m
\]
where
\[
y = \frac{b\theta_1}{s + a + b\theta_2} u_c
\]

Sensitivity derivatives:
\[
\begin{align*}
\frac{\partial e}{\partial \theta_1} &= \frac{b}{s + a + b\theta_2} u_c \\
\frac{\partial e}{\partial \theta_2} &= -\frac{b^2\theta_1}{(s + a + b\theta_2)} u_c = -\frac{b}{s + a + b\theta_2} y
\end{align*}
\]

Approximate
\[
(s + a + b\theta_2) \approx (s + a_m)
\]
then the MIT rule \(d\theta/dt = -\gamma' e \partial e/\partial \theta\), may be written as follows
\[
\begin{align*}
\frac{d\theta_1}{dt} &= -\gamma' \left( \frac{b}{s + a_m} u_c \right) e \Rightarrow \theta_1 = \left( \frac{-\gamma}{s} \right) \left( \frac{a_m}{s + a_m} u_c \right) e \\
\frac{d\theta_2}{dt} &= \gamma' \left( \frac{b}{s + a_m} y \right) e \Rightarrow \theta_2 = \left( \frac{\gamma}{s} \right) \left( \frac{a_m}{s + a_m} y \right) e
\end{align*}
\]
where \(\gamma = \gamma' b / a_m\)

Block diagram
Fig. 3.8 Model Reference Adaptive Control for the example

Simulation: $a = 1, b = 0.5, a_m = b_m = 2, u_c$ : square wave with period 20 s and values 1 and 0.
3.3 Self-tuning Adaptive Controller

Self-tuning adaptive controllers such as the self-tuning regulator differ from the model-reference adaptive controller in basic principle. The self-tuning controller, illustrated by the block diagram in Figure 3.8, makes use of the process input and output to estimate recursively, on-line, the parameters of an approximate process model. Thus as the actual non-linear process changes operating region or changes with time, an approximate linear model is continuously updated with new parameters. The updated model is then used in a prespecified control system design procedure to generate updated controller parameters. The controller could be a PID controller or some of the model-based controllers.

![Fig. 3.8. Self-Tuning Control Scheme](image)

Since the model estimated determines the effectiveness of the controller, the most essential feature of the self-tuning controller is reliable and robust model identification. This requires a good parameter estimation algorithm and procedures for ensuring adequate dynamic experimental design.

**Example:** Process is assumed to have the transfer function of the form, say,

\[
g(s) = \frac{Ke^{-\alpha s}}{\tau s + 1}
\]

Let the controller be a PI controller of the form

\[
g_c = K_c (1 + \frac{1}{\tau_i s})
\]  \hspace{1cm} (3.2)

The controller parameters are typically set according to a tuning formula, such as (Minimum ITAE tuning rule for set-point tracking),

\[
K_c = \frac{0.556 \tau}{K (\frac{\alpha}{\tau})^{0.916}}
\]

\[
\frac{1}{\tau_i} = \frac{1.03 - 0.165(\frac{\alpha}{\tau})}{\tau}
\]  \hspace{1cm} (3.3)

Notice that the controller parameters \(K_c\) and \(\tau_i\) are adjusted in terms of the process parameters \(K\), \(\alpha\), and \(\tau\). Since the process model parameters are continuously changing as the process operating (load) conditions are changed, the parameter estimator will periodically update the
parameters $K$, $\alpha$, and $\tau$. Based on this updated parameter values, the controller settings $K_c$ and $\tau_I$ are recalculated using (3.3). The new controller parameters are then adjusted on the controller as shown in Fig. 3.8.

3.4 Comparison of Adaptive Controllers

There is a clear differentiation in the philosophy behind each of the three adaptive controllers discussed above. Let us use an analogy to illustrate. When parents send a child off to college, they usually try to provide them with “adaptive controllers” to help them deal with the unknown situations they will counter out on their own. We can easily recognize three types of the adaptive controllers in use:

- **Scheduled Adaptive Control** is used by parents who use a list of “if this occurs, do this” instances. This means they assume they can anticipate and enumerate all the situations that can occur and specify programmed remedies. No thinking or decision making is expected of their children. Thus scheduled adaptive control will work when all possibilities are known in advance.

- **Model-Reference Adaptive Control** is employed by parents when they have a good role model (such as an older sibling) available. They only tell their child to emulate the behavior of the role model; i.e., “do whatever they do in the same situation”. In this case there is some limited decision making on the part of the child, but it is constrained to deciding how best to emulate their role model. So long as such emulation is possible, then MRAC works well.

- **Self-tuning Adaptive Control** is the strategy recommended by parents, have a little more confidence in the thinking ability of their children. While they do not recommend complete independence of action, they ask that the child carefully evaluate the situation based on the data at hand (model building) and then based on their judgment of the situation, take action based on fixed principles.

Self-tuning adaptive control allows the most freedom to the controller and thus provides the broadest range of adaptation to unknown situations; however, it is also the most dangerous, potentially unstable, form of adaptive control because poor model identification (poor assessment of situations) can lead to unstable systems and process runaway.

All these types of adaptive controllers are commercially available and have enjoyed a certain level of industrial acceptance. By their very nature, adaptive controllers can be applied to any system for which the process parameters vary with time; they are also applicable to other systems for which these parameter variations are not due to nonlinearities per se, but to other physical phenomena such as fouling in heat exchangers, catalyst decay in catalytic reactors, etc.

4. Variable Transformations

The idea of variable transformation can be exploited for nonlinear controller design in the following manner:

- Obtain an equivalent linear system by variable transformation.
- Design a linear controller for the transformed system, in terms of the new variables.
Implement the controller in terms of the original variables of the nonlinear system (such a controller will be nonlinear).

Several different methods have been reported in the literature for carrying out such variable transformations for non-linear systems and utilizing the result for controller design. The following is a simple illustration

### 4.1 Global Liberalization

Let the process be described by

\[ \frac{dx}{dt} = f(x,u) \]  

We can always separate the RHS of (4.1) into two terms where the first term depends only on \( x \) as in (4.2):

\[ f(x,u) = c_1 f_1(x) + c_2 f_2(x,u) \]  

Consider a new variable \( z(x) \).

\[ \frac{dz}{dt} = \frac{dz}{dx} \frac{dx}{dt} \]  

That is, using (4.3), (4.2) can be written as

\[ \frac{dz}{dt} = c_1 \frac{dz}{dx} f_1(x) + c_2 \frac{dz}{dx} f_2(x,u) \]  

If \( z(x) \) is chosen such that

\[ \frac{dz}{dx} = \frac{z}{f_1(x)}; \quad f_1(x) \neq 0 \]  

then (4.4) becomes,

\[ \frac{dz}{dt} = c_1 z + \frac{c_2 z f_2(x,u)}{f_1(x)} \]  

That is,

\[ \frac{dz}{dt} = c_1 z + c_2 v(x,u) \]  

where

\[ v(x,u) = \frac{z f_2(x,u)}{f_1(x)} \]

Notice that in (4.6), \( v \) is the new input and (4.6) is linear in the new variable \( z \) (because coefficient of \( z \) in the first term is a constant \( c_1 \)). Transformation of old input variable \( u \) into new input variable \( v \) is given by (4.7). One can also express, using (4.7), \( u \) in terms of \( v \) as

\[ u = q(x,v) \]  

for some function \( q \).
To find the function \( z(x) \), integrate (4.5) to get

\[
z = g(x) = e^{\int \frac{dx}{f_1(x)}}
\]

Alternatively, one can define, in place of (4.5), \( z(x) \) as

\[
\frac{dz}{dx} = \frac{1}{f_1(x)}; \quad f_1(x) \neq 0 \tag{4.9}
\]

Using (4.9), (4.2) can be expressed as

\[
\frac{dz}{dt} = c_1 + c_2 \frac{f_2(x,u)}{f_1(x)} \tag{4.10}
\]

(4.10) can be put as

\[
\frac{dz}{dt} = c_1 + c_2 v(x,u) \tag{4.11}
\]

(4.11) is linear in the new variable \( z \) and \( v \) is the new input. As before, new input is expressed as

\[
v = \frac{f_2(x,u)}{f_1(x)}
\]

Or,

\[
u = q'(x,v) \tag{4.12}
\]

for some function \( q' \). Integrating (4.9), \( z(x) \) is obtained as

\[
z = g'(x) = \int \frac{dx}{f_1(x)} \tag{4.13}
\]
Fig. 4.1. Global Linearized Control

**Example:** Application to level Control in a Cylindrical Tank (4.2) can be written as

\[ \frac{dx}{dt} = c_1 x^2 + c_2 u \]

where \( c_1 = -\frac{c}{A} \) and \( c_2 = \frac{1}{A} \). Consider a change of variable \( z(\alpha) \). Then,

\[ \frac{dz}{dt} = \frac{dz}{dx} \frac{dx}{dt} \]

Substituting (4.13) into (4.3),

\[ \frac{dz}{dt} = c_1 \frac{1}{x^2} + c_2 u \frac{1}{x^2} \]

Put

\[ \frac{dz}{dx} = \frac{1}{x^2}, \quad x \neq 0 \] (4.15)

Then (4.14) becomes

\[ \frac{dz}{dt} = c_1 + c_2 v \] (4.16)

where

\[ v(x, u) = \frac{u}{x^2} \]

Integrating (4.15),

\[ z = 2x^2 \] (4.17)

\( u \) can also be expressed as
Since (4.16) is linear, one can use a fixed controller, such as, PID controller to control the transformed system.

We now wish to apply these results to a specific flow system cylindrical pipe for which $A = 22.062 \text{ cm}^2$ (I.D. is 5.3 cm) and, from previous experimentation, it is known that the dependence of the outflow rate on the liquid level in the pipe is given by:

$$F_{out} = A \times 0.1176 \frac{h^{1/2}}{cm^3/s}$$

The control system is implemented as follows:

- Obtain the actual measurement $x$ (in this case $h$) from the process.
- Calculate $z$ from (4.17); i.e., take the square root, and multiply by $2$.
- From the same expression, given the desired value for $x$, obtain the desired value for $z$; i.e., $Z_d = 2\sqrt{x_d}$.
- The PI controller (tuned for the linear, transformed system (4.11)) receives $z$ and $z_d$ and prescribes $v(t)$, the control action required for the linear pseudo-system.
- The actual flowrate implemented on the real, nonlinear flow system is obtained from (4.18); a control law nonlinear in the original process variable $x$.

It is interesting to note that the transformation controller bases the prescribed inflowrate on a product of the control action for the linear pseudo-system and the square root of the liquid level in the tank. The physical implications of this are particularly important: the nonlinearity in the flow system is caused by the fact that the outflow is proportional to the square root of the liquid level in the tank; the transformation controller is seen to take advantage of this fact by adopting the indicated strategy. The conventional PI control strategy will base the inlet flowrate on the actual liquid level in the tank; the transformation controller bases its action on the product of the output of a linear controller, and the square root of the liquid level.

Using controller parameters $K_c = 90$ and $K_i = K_c/\tau_i = 0.2$ (for the PI controller tuned for the linear pseudo-system), the response of the example flow system to a set-point change in the liquid level from 40 to 75 cm, operating under the nonlinear transformation controller, is shown in the dashed line in Figure 4.2.

A conventional PI controller using the same controller parameters resulted in an almost unstable system upon reducing the controller parameters to $K_c = 9$ and $K_i = K_c/\tau_i = 1$ the system response to the same set-point change under conventional PI control (i.e., decisions on the inlet flowrate are taken using direct level measurements) is shown in the solid line. As we would expect, the transformation controller performs better than the conventional PI controller.
Application of the nonlinear transformation strategy is restricted by the considerations:

- The nonlinear transformations are not always easy to find (for some general approaches).
- The success of the controller design is dependent on having a good linear process model—something that is unusual in practice.

**Conclusion:** It is not a universally applicable control system design procedure. In general, the conceptual basis for GLC lies in explicitly relating the system output to its manipulated input in terms of an ordinary differential equation. The output is successively differentiated with respect to time until the resulting derivative is a linear function of the input signal. Differential geometric concepts are then applied in developing the control law.

## 5. Generic Model Control

Generic Model Control (GMC) uses a model of the process in formulating the control law. The design framework is similar to other model based approaches such as Dahlin’s algorithm and IMC. However, rather than adopting a classical approach of comparing the trajectory of the process output against a desired trajectory, GMC defines the performance objective in terms of the time derivatives of the process output, i.e. minimising the difference between the desired derivative of the process output and the actual derivative.

Generally the following control affine, non-linear state space description is used,

\[
\dot{x} = f(x) + g(x)u \\
y = h(x)
\]  

(5.1)

where \(f(x)\) and \(g(x)\) are vector fields, i.e. they are vector valued functions of a vector and \(h(x)\) is a scalar field, i.e. a scalar valued function of a vector, \(x\). In terms of a general system of equations this can be expressed as:

\[
\frac{dx_1}{dt} = f_1(x_1, \ldots, x_n) + g_1(x_1, \ldots, x_n)u \\
\vdots \\
\frac{dx_n}{dt} = f_n(x_1, \ldots, x_n) + g_n(x_1, \ldots, x_n)u \\
y = h(x_1, \ldots, x_n)
\]
The model is termed control affine or, control linear, because the manipulated input appears linearly - it simply multiplies $g(x)$.

Choose the following desired trajectory:

$$\dot{y}_d = \alpha_1(y_d - y) + \alpha_2 \int (y_d - y) \, dt \quad (5.2)$$

where $\alpha_1$ and $\alpha_2$ are design constants (to be specified) and $y_d$ is the setpoint. This is the form of a PI controller.

In order to design a controller so that the system follows the trajectory defined by equation (5.2) as closely as possible, the following performance index is specified:

$$J = \int_0^t \dot{e}^2 \, dt = \int (\dot{y}_d - \dot{y})^2 \, dt \quad (5.3)$$

i.e. minimise the error squared over a specified time horizon. In order to obtain $\dot{y}$ from equation (5.1) the chain rule must be used,

$$\frac{dy}{dt} = \frac{dy}{dx} \frac{dx}{dt} \quad (5.4)$$

**Note:** $\frac{dy}{dx}$ is a 'row' vector while $\frac{dx}{dt}$ is a column vector given by,

$$\frac{dx}{dt} = \dot{x} = f(x) + g(x)u \quad (5.5)$$

$$y = h(x)$$

therefore $\frac{dh}{dx}$ is also a 'row' vector and,

$$\frac{dy}{dt} = \frac{dh}{dx} \left[ f(x) + g(x)u \right]$$

$$= \frac{dh}{dx} f(x) + \frac{dh}{dx} g(x)u \quad (5.6)$$

Using equation (5.6) the performance index, equation (5.3), is minimised when $\dot{e} = 0$, i.e.

$$\alpha_1(y_d - y) + \alpha_2 \int (y_d - y) \, dt - \left[ \frac{dh}{dx} f(x) + \frac{dh}{dx} g(x)u \right] = 0 \quad (5.7)$$

$$u(t) = \frac{1}{\frac{dh}{dx} g(x)} \left[ \alpha_1(y_d - y) + \alpha_2 \int (y_d - y) \, dt - \frac{dh}{dx} f(x) \right] \quad (5.7)$$

This control law has two desirable properties:

- It utilises a process model.
- any inaccuracies in the process model (i.e. plant model mismatch) will be compensated for by the integral term within the control law.
**Example:** Consider the following continuous stirred tank exothermic reactor as shown in Fig. 5.1,

![Figure 5.1. A Continuous Stirred Tank Reactor (CSTR)](image)

The inlet concentration of the reactant and product are $A_i$ and $R_i$ (mol/l) respectively while the outlet concentration is $A_o$ and $R_o$ (mol/l). The inlet temperature is $T_i$ (K) while the outlet temperature is $T_o$ (K). The concentrations of both reactant and product in the CSTR are taken to be process states, along with the reactor outlet temperature. The dynamic behaviour of these states is described by three nonlinear-coupled differential equations:

\[
\begin{align*}
\frac{dA_o}{dt} &= \frac{A_i - A_o}{\tau} - k_1A_o + k_2R_o \\
\frac{dR_o}{dt} &= \frac{R_i - R_o}{\tau} + k_1A_o - k_2R_o \\
\frac{dT_o}{dt} &= -\Delta H_R (k_1A_o - k_2R_o) + \frac{(T_i - T_o)}{\tau}
\end{align*}
\]

where the reaction rates $k_1$ and $k_2$ depend on the reactor temperature via the Arrhenius relationship

\[k_i = c_i \exp\left(\frac{-E_i}{RT_o}\right), \quad k_2 = c_2 \exp\left(\frac{-E_2}{RT_o}\right)\]

and $E_1$ (Cal), $E_2$ (Cal), $R$ (Cal/mol k), $A_i$ (mol/l), $R$ (kg/l), $C_p$ (Cal/kg K), $\tau$(s), = time constant of the vessel and $\Delta H_R$ (Cal/mol) are normal system parameters. The control objective is to regulate reactor temperature ($T_o$) using inlet feed temperature ($T_i$).

This process is evidently highly non-linear due to the inter-relationships of the states, and particularly the exponential dependence of each state on the reactor temperature. The differential equations can be re-arranged into the generic vector representation of the process with the following vector and scalar fields:

**State Vector:** \[x = (A_o, R_o, T_o)^T\]

**f vector fields:** \[f(x) = (f_1(x), f_2(x), f_3(x))^T\]
Non-linear Process Control

\[ f_1(x) = \frac{A_i - A_o}{\tau} - k_i A_0 + k_2 R_o \]
\[ f_2(x) = \frac{R_i - R_o}{\tau} + k_i A_0 - k_2 R_o \]
\[ f_3(x) = \frac{-\Delta H_R}{\rho c_p}(k_i A_0 - k_2 R_o) - \frac{T_o}{\tau} \]

**g vector fields:** \( g(x) = (g_1(x), g_2(x), g_3(x))^T = \begin{pmatrix} 0 & 0 & 1/\tau \end{pmatrix}^T, \quad u = T_i \)

**h function:** With the CSTR, the \( h \) scalar field describing the output function is simply the state itself, i.e. \( y = h(x) = T_o \), therefore

\[ \frac{dy}{dt} = \frac{dx_3}{dt} \Rightarrow \dot{y} = f_3(x) + g_3(x)u \quad (5.8) \]

In other words, the GMC controller is given by,

\[ \alpha_1(y_d - y) + \alpha_2 \int (y_d - y) dt - \left[ \frac{-\Delta H_R}{\rho c_p}(k_i A_0 - k_2 R_o) - \frac{T_o}{\tau} + \frac{T_i}{\tau} \right] = 0 \quad (5.9) \]

and solving for the manipulated input, yields,

\[ T_i = -\tau \left\{ \alpha_1 (y_d - y) + \alpha_2 \int (y_d - y) dt - \left[ \frac{-\Delta H_R}{\rho c_p}(k_i A_0 - k_2 R_o) - \frac{T_o}{\tau} \right] \right\} \quad (5.10) \]

Taking Laplace transforms of equation (5.2) gives:

\[ \frac{y(s)}{y_d(s)} = \frac{\alpha_1 s + 1}{\alpha_2 s^2 + \frac{\alpha_1}{\alpha_2} s + 1} \quad (5.11) \]

This ‘ideal’ closed loop response is not the same as a classical 2nd order system. However, it is possible to plot normalised response of the system versus normalised time with the damping ratio of a second order system as a parameter. This allows the selection of the ‘shape’ of response as well as an appropriate closed loop time constant. This then allows the specification of \( \alpha_1 \) and \( \alpha_2 \).