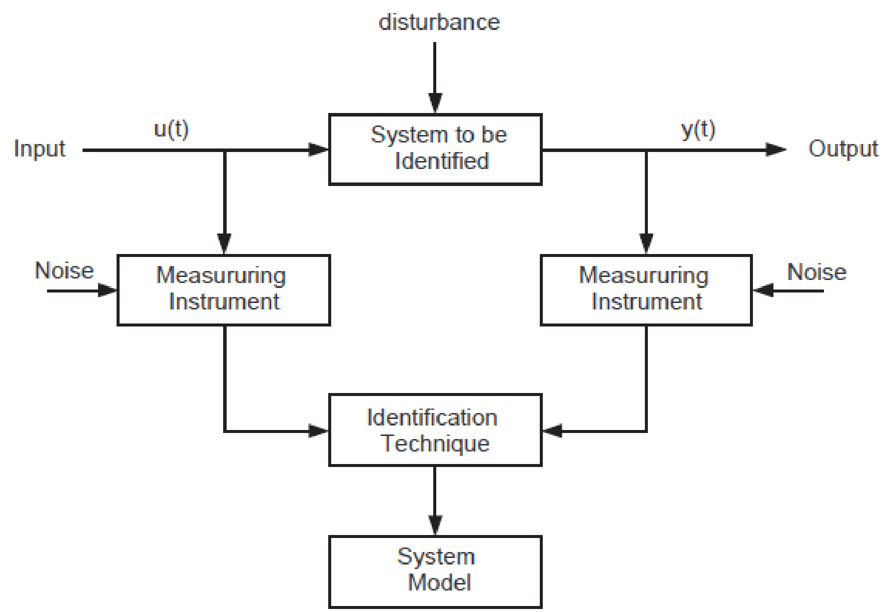
Process Dynamics & Control

Process Identification

What is Identification?

Identification is the process of constructing a empirical model of a (dynamical) system from observations of its inputs and outputs. In a dynamical system, the output at any instant depends on its history not just present input, i.e. it has some form of memory (storage).



Classification of identification

Based on the degree of a priori knowledge of the system.

Black Box:

- This means we know nothing about the basic characteristics of the system.
- Extremely difficult to solve.
- Usually some kind of assumptions have to be made before any meaningful solution can be attempted.

Grey Box:

- In this case, some basic characteristics of the system are known (ie. linearity, bandwidth, structure).
- However, order of the dynamic equation or values of the associated coefficients may be unknown.

Process Identification Procedure

The experiment design objective is to make choices such that the collected data is maximally informative, i.e.

- types of signals,
- amplitudes,
- what to measure
- input/outputs

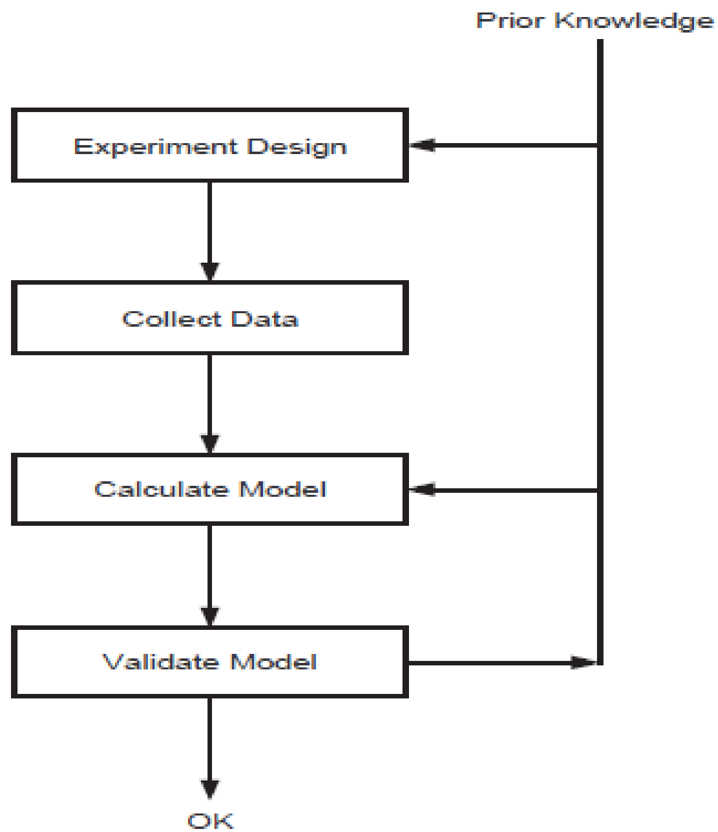
In the model calculation block, decisions need to be made on:

- what type of model,
- model structure,
- parameter estimation technique

Process Identification Procedure

To validate the model, the following questions need to be asked:

- How does the model output relate to the data observed?
- Is it good enough for specific purpose?



Types of Test Signals

The input signal used in an identification experiment can have a significant influence on the resulting parameter estimates.

Typical types of input signal often used in practice include:

- Impulse function (approximate)
- Step function
- Pseudo-random binary sequence
- Sinusoids

The signal we consider in this course is step function.

Types of Test Signals

Step function

- From the response of a process to a step input a number of practical parameters can be obtained, i.e. dead time, time constant, etc.
- It is easy to apply.
- It is very sensitive to noise. (Usually requires a rather large amplitude to obtain reasonable results)
- Generally, it can only give a basic model.
- Steady state gain is easily found, in the absence of drift, from the initial and final values of the step response.
- User choices to be considered:
  - Amplitude
  - Duration

Design of Identification Experiments

There are a number of factors to consider before performing an identification experiment.

- What form of test signal should be used, i.e. step, doublet, prbs?
    - This depends on the quality of the model you require. In most cases, for PID control, step tests are adequate.
- What size should the amplitude be? There a quite a number of factors to consider here.
    - There may be constraints on how much variation can be tolerated in the input and/or output. (economic, safety, actuator limits, etc).
    - One reason for a large amplitude is that the effects of noise become less (signal to noise ratio is larger).
    - For systems with known nonlinearities it is best to keep the amplitude small as generally you are interested in a model around a particular operating point.

Design of Identification Experiments

- What sampling frequency should be used?
  - Typically, the sampling frequency should be chosen as 10 - 20× that of the test signal frequency. (5× should be considered as the absolute minimum).
  - For a step test, you want to capture at least 5 samples per the time constant of the process.
- What should the frequencies of multi-step test signal be?
  - The frequencies of these test signals should be in the frequency region of interest and should be chosen as a multiple of the sampling frequency and of each other. This will minimise errors.
  - When collecting data from a multi-step test one should wait until the transients have decayed significantly.

Transfer Function from Process Data

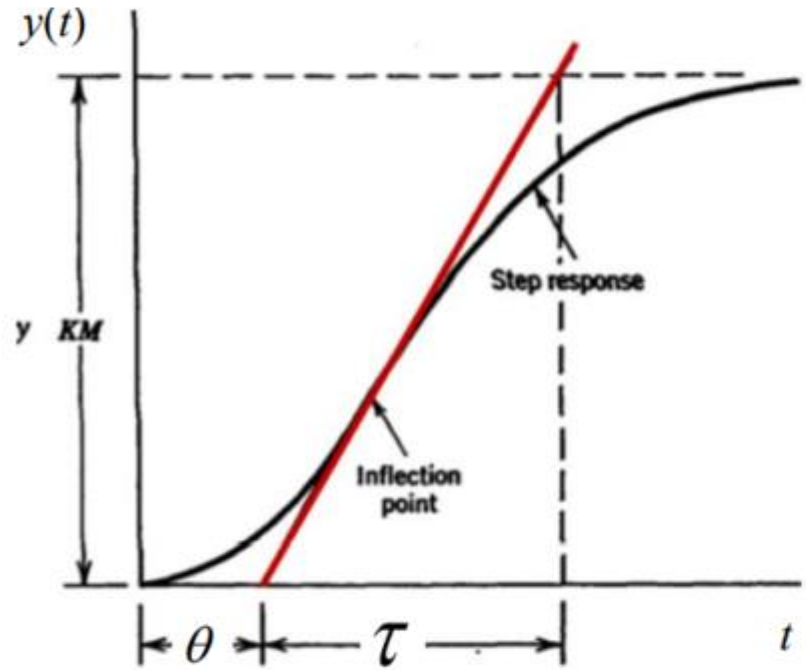
- Normally, chemical process transfer functions are approximated by First Order with Dead Time (FODT) model, $g(s) = \frac{Ke^{-\theta s}}{\tau s+1}$

- Four classical methods (using step response data)
  - Ziegler-Nichols method
  - Smith's method
  - Sundaresan and Krishnaswamy method
  - Nishikawa's method

Transfer function from Process Data

Ziegler-Nichols method (1942) consists of applying a tangent line to the curve at an inflection point, to determine the system's gain, time, and delay constants, as shown in Figure.

The inflection point is defined where the curve changes direction and the derivative is equal to zero.

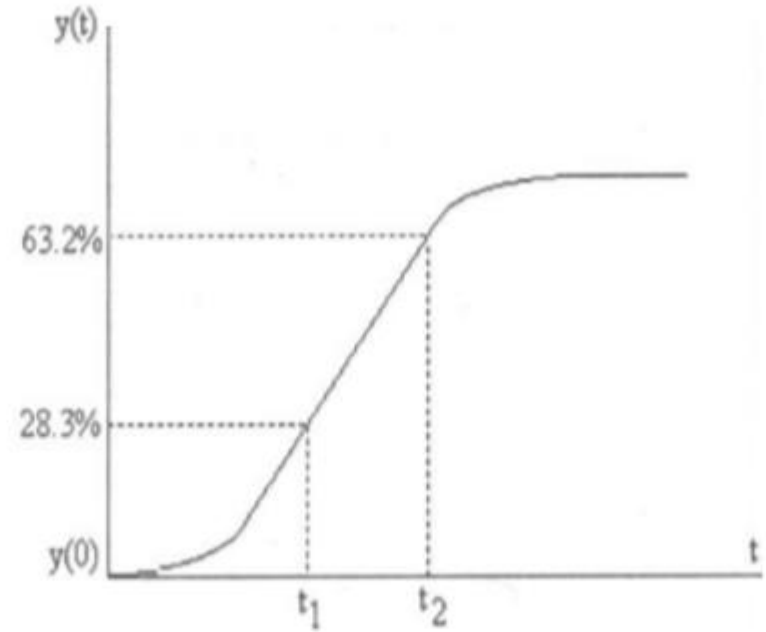


Drawback:

Difficult to find inflection Point and to draw tangent.

Transfer function from Process Data

Smith's Method (1972)proposed that the values of θ and τ be selected in such a way that the model and the real responses coincide in two points that present a high rate of variation.



The two time points are at 28.3% 63.2 % of change in output respectively.

So from figure, after finding t_1 and t_2, we can calculate

$K = \frac{\Delta output}{\Delta input}$ $\tau = \frac{3}{2}(t_2 - t_1)$ and $\theta = t_2 - \tau$

Transfer function from Process Data

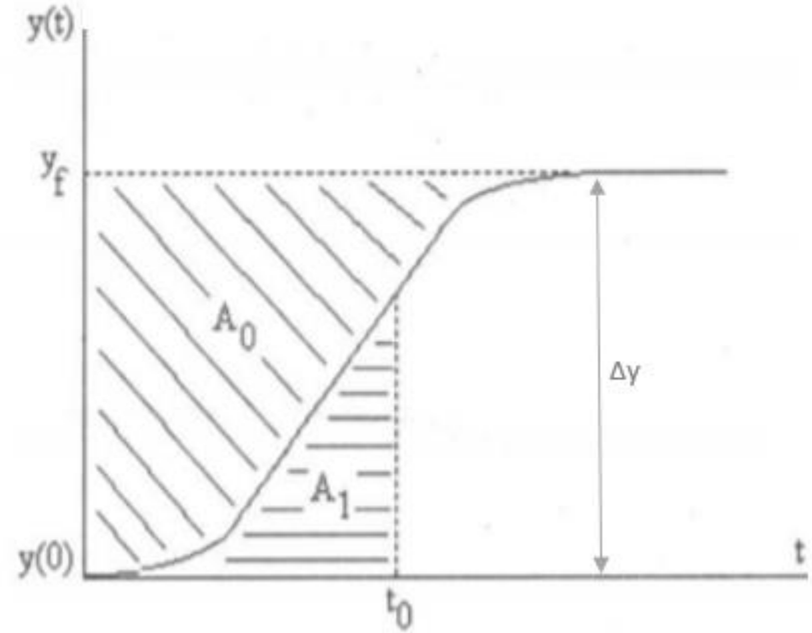
Sundaresan and Krishnaswamy Method:

This is similar to smith's method but the two time points t_1 and t_2 are based on 35.3% and 85.3% of change in output response respectively. So, model parameters are calculated as,

$$K = \frac{\Delta output}{\Delta input}\ ;\ \tau = \frac{2}{3}(t_2 - t_1);\ \theta = 1.3t_1 - 0.29t_2$$

Transfer function from Process Data

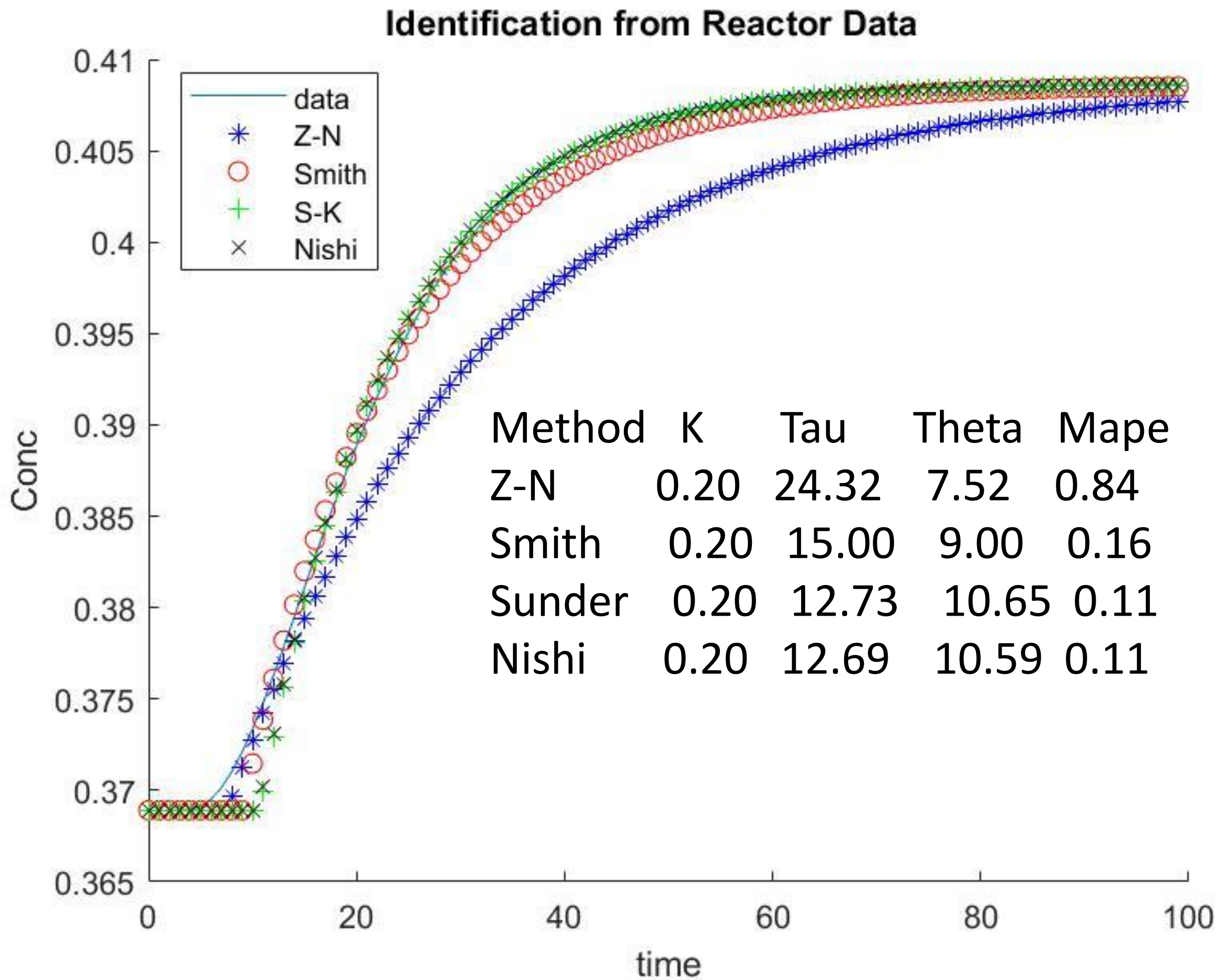
Nishikawa (2007) method determine the values of the constants using the calculation of the curve areas as shown in Figure.



$$A_0 = \int_0^\infty \big(\Delta y(\infty) - \Delta y(t)\big)dt$$

$A_1 = \int_0^{t_0} \Delta y(t)dt$ where, $t_0 = \frac{A_0}{\Delta y(\infty)}$

$\tau = \frac{A_1}{0.368\,\Delta y(\infty)}$; $\theta = t_0 - \tau$; $K = \frac{\Delta y(\infty)}{\Delta c(\infty)}$


Identification from Reactor Data
data
Z-N
Smith
S-K
Nishi
Conc
time
0.41
0.405
0.4
0.395
0.39
0.385
0.38
0.375
0.37
0.365
0
20
40
60
80
100
Method K Tau Theta Mape
Z-N 0.20 24.32 7.52 0.84
Smith 0.20 15.00 9.00 0.16
Sunder 0.20 12.73 10.65 0.11
Nishi 0.20 12.69 10.59 0.11

Transfer function from Process Data

Method of Moments method

- can be used to determine model parameters from the output response generated using arbitrary input function.

- is based on the definition of the Laplace transform of the impulse response *g(t)* of a system, which is its transfer function $\mathrm{G(s)} = \int_0^\infty e^{-st} g(t)\, dt$

As the n-th-order moment of a function f(x) is defined by

$$M_n(f) = \int_0^\infty x^n f(x)\, dx$$

Transfer function from Process Data

It can be noticed that the first two derivatives of G(s) w.r.t s $\mathrm{G}'(\mathrm{s}) = -\int_0^\infty te^{-st}g(t)\,dt$ & $\mathrm{G}''(\mathrm{s}) = \int_0^\infty t^2e^{-st}g(t)\,dt$ are related to the moments of the impulse response function by $G(0) = \int_0^\infty g(t)\,dt$; $G'(0) = -\int_0^\infty t\,g(t)\,dt$; and $G''(0) = \int_0^\infty t^2g(t)\,dt$;

- Thus G(O), -G'(O), G"(O) are respectively the zero-, first- and second-order moments of the impulse response *g(t)*.
- Note that the above three integrals can be calculated by using the measured output response.

Transfer function from Process Data

- Note that in this method, it is possible to use any type of input. We have, in general

$$Y(s) = G(s)\ U(s)\ ;\ Y'(s) = G'(s)\ U(s) + G(s)\ U'(s)$$

$$Y''(s) = G''(s)\ U(s) + G(s)\ U''(s) + 2\ G'(s)\ U'(s)$$

- from which the following equations are deduced

$$Y(0) = G(0)\ U(0)\ ;\ Y'(0) = G'(0)\ U(0) + G(0)\ U'(0)$$

$$Y''(0) = G''(0)\ U(0) + G(0)\ U''(0) + 2\ G'(0)\ U'(0)$$

These quantities can be calculated by the following equations:

$U(0) = \int_0^\infty u(t)dt\,;\ \ U'(0) = -\int_0^\infty t\,u(t)dt\,;$

$U''(0) = \int_0^\infty t^2 u(t)\,dt\,;\ \ Y(0) = \int_0^\infty y(t)\,dt;$

$Y'(0) = -\int_0^\infty t\,y(t)\,dt;$ and $Y''(0) = \int_0^\infty t^2 y(t)\,dt;$

Transfer function from Process Data

FODT model: $G(s) = \frac{K\,e^{-\theta s}}{\tau s+1}$

So, $G(0) = K = \frac{Y(0)}{U(0)}$

$$G'(0) = -K(\tau + \theta) = \frac{Y'(0) - G(0)U'(0)}{U(0)}$$

$$G''(0)=K(2\tau^2+2\tau\theta+\theta^2) = \frac{Y''(0)-G(0)U''(0)-2G'(0)U'(0)}{U(0)}$$

Solving above equations K, τ, θ can be obtained.

SODT model from step response

- $G(s) = \frac{Ke^{-\theta s}}{\tau^2 s^2 + 2\xi\tau s + 1}$ with $\xi > 0.707$

Method (Rangaiah & Krishnaswamy)

1. Calculate the process steady-state gain from the magnitude of the step input and that of the corresponding response;
2. Find t_1, t_2 and t_3 from the response data corresponding to 14%, 55%, and 91% of the actual response;

SODT model from step response

3. Calculate $\alpha, \beta\ and\ \xi$ using the following equations

$$\alpha = \frac{t_3 - t_2}{t_2 - t_1}$$

$$\beta = \ln\left(\frac{\alpha}{2.485 - \alpha}\right)$$

$$\xi^2 = 0.50906 + 0.51743\beta - 0.076284\beta^2 + 0.041363\beta^3 - 0.0049224\beta^4 + 0.00021234\beta^5$$

for $1.2323 < \alpha < 2.485$

SODT model from step response

4. Calculate τ and θ using the following equations

$$\frac{t_2 - t_1}{\tau} = 0.85818 - 0.62907\xi + 1.2897\,\xi^2 - 0.36859\,\xi^3 + 0.038891\,\xi^4$$

$$\frac{t_2 - \theta}{\tau} = 1.392 - 0.52536\,\xi + 1.2991\,\xi^2 - 0.36014\,\xi^3 + 0.037605\,\xi^4$$

Non-linear Regression

This method can be used for any form of linear or non-linear models.

For example, process having transfer function of one zero 2 pole system

$$\frac{y(s)}{u(s)} = G(s) = \frac{K(\tau_3 s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)}$$

For Step input of magnitude M in u gives,

$$y(t) = KM\left(1 + \frac{\tau_3 - \tau_1}{\tau_1 - \tau_2} e^{-\frac{t}{\tau_1}} + \frac{\tau_3 - \tau_2}{\tau_2 - \tau_1} e^{-\frac{t}{\tau_2}}\right)$$

The above equation can be regressed to get the model parameters $K, \tau_1, \tau_2, \tau_3$

Matlab Implementation

Step response data:

t	0	1	2	3	4	5	6	7	8	9	10
u	1	1	1	1	1	1	1	1	1	1	1
y	0	0.058	0.217	0.36	0.488	0.6	0.692	0.772	0.833	0.888	0.925

Use command
mdl = procest(data, Type)
Where, data should stored in iddata format.
Type ='P2DUZ' [2 poles, delay, U underdamped, Z zero]

```
>> u=[0;0;0;u];  y=[0;0;0;y]; % for step changes
>> z = iddata(y, u, 1);  % y and u should be column vectors
>> mdl= procest(z,'P2DUZ');
```

'procest' can work for step as well as arbitrary input changes.