ISSN 2229-5518

# TUNING OF DECENTRALISED PI (PID) CONTROLLERS FOR TITO PROCESS

**P.Sujatha*  Dr.M.Bharathi,**Dr.C. selvakumar*****

**HOD, EIE Dept, Bharath University, Chennai-73
***HOD, ICE Dept, St. Joseph's College of Engineering , Chennai-119
*AP, ECE Dept,VRS College Of Engineering And Technology,Villuppuram-607107

**Abstract**

This paper presents one of the simplest method to tune decentralised PI (PID) controllers for two-input and two-output (TITO) processes. The TITO process was decoupled through a decoupler matrix. To handle loop interactions a model reduction method with suitable FOPDT and SOPDT model for each element of the resulting diagonal process through fitting the nyquist plots at particular points. Simulation examples of MIMO systems are given to demonstrate the effectiveness and accuracy of the proposed algorithm. Compared to other tuning methods it has less number of tuning parameters and more over uses only less number of controllers, therefore it is one of the cost effective method to control the process.

**Key words:** TITO,MIMO system, PID

## 1.Introduction

The PID controller is the most popular controller used in process control, because of its remarkable effectiveness and simplicity of implementation. Although significant developments have been made in advanced control theory, according to the literature, more than 95% of industrial controllers are still PID, mostly PI controllers. PI (PID)

control is sufficient for a large number of control processes, particularly when dominant process dynamics are of first (second) order and there design requirements are not too rigorous. Although this controller has only three parameters, it is not easy to find their optimal values without a systematic procedure. As a result PI (PID) tuning methods are extremely desirable due to their wide spread use.

Generally, most industrial processes are multi variable systems. When interactions in different channels of the process are modest, a diagonal PID controller is often adequate. Two-input two-output (TITO) systems are one of the most prevalent categories of multi variable systems. So an approach for square systems is to use a decoupler plus a decentralized PID controller. One great advantage of this method is that it allows the use of single-input single-output (SISO) controller design methods. One of the simplest methods to tune decentralised PI (PID) controllers for TITO process is proposed in this paper.

The TITO process was decoupled through a decoupler matrix that allows for more flexibility in choosing the transfer functions of the decoupled apparent process. A model reduction method with suitable FOPDT and SOPDT model for each element of the resulting diagonal process through fitting the nyquist plots at particular points is implemented to handle loop interactions. Simulation examples OF MIMO systems are incorporated to validate the usefulness of the presented algorithm. Compared to other tuning methods it has less number of tuning parameters and more over uses only less number of controllers, therefore it is one of the cost effective method to control the process

.Many PI or PIDcontrollers have been proposed (Chien and Fruehauf1990; Tyreus and Luyben, 1992).

Juan et.al.,(2012) proposed a generalized formulation of simplified decoupling to $n \times n$ *processes* that allows for different configurations depending on the decoupler elements set to unity.

Zhuo et.al., (2011) proposed the design of a multi-loop PI controller to achieve the desired gain and phase margins for two-input and two-output (TITO) processes .

An extension of the inverted decoupling approach that allows for more flexibility in choosing the transfer functions of the decoupled apparent process (Juan et.al.,2011).

The idea of an effective open-loop transfer function (EOTF) is first introduced to decompose a multi-loop control system into a set of equivalent independent single loops.( Truong et.al., 2010)

Branislav and Miroslav (2010) designed multivariable controller based on ideal decoupler *D*(*s*) and PID controller optimization under constraints on the robustness and sensitivity to measurement noise

Nordfeldta and Haddlund(2006) proposed controller consists of a decoupler and a diagonal PID controller.

*Control Engineering Practice*, *Volume 14, Issue 9*, *September 2006*, *Pages 1069-1080*
Saeed Tavakoli, Ian Griffin, Peter J. Fleming

,

Tavakoli et.al(2006) presented a decentralised PI (PID) tuning method for two-input two-output processes based on dimensional analysis

Wang et.al.,(2006) considered auto-tuning of simple lead-lag decoupler plus decentralized

PI/PID controllers for effective control of two-input and two-output (TITO) processes.

Maghade and Patre (2012) proposed a decentralized PI/PID controller design method based on gain and phase margin specifications for two-input–two-output (TITO) interactive processes

*Automatica*, *Volume 31, Issue 7, July 1995*, *Pages 1001-1010*

Z.J. Palmor, Y. Halevi, N. Krasney

Palmor et.al., (1995) present an algorithm for automatic tuning of decentralized PID control for two-input two-output (TITO) plants that fully extends the single-loop relay auto-tuner to the multiloop case.

Rajapandiyan and Chidambaram(2012) proposed the closed-loop identification of two-input–two-output (TITO) second-order plus time delay (SOPTD) transfer function models of multivariable systems is presented based on optimization method using the combined step-up and step-down responses.

## 2. Structure of PI controller design

Most of the industrial process have multivariable control variable that are common properties for the models of industrial processes to have significant uncertainties, strong interaction and non-minimum phase behaviour so it is important for control engineer, chemical engineer to understand the non-idealities of industrial processes. The structure of MIMO control system is shown in figure1.

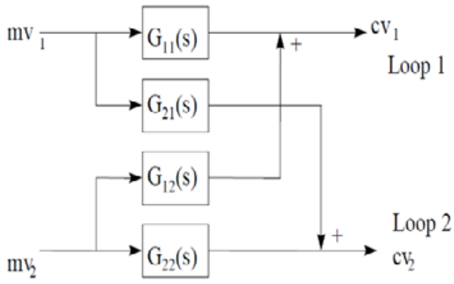


Fig1. (2×2) Multivariable Model Structure

Here from fig 1,$G_{11}(s)$ is a symbol used to represent the forward path dynamics between $mv_1$ and $cv_1$, while $G_{22}(s)$ describes how $cv_2$ responds after a change in $mv_2$. The interaction effects are modelled using transfer functions $G_{21}(s)$ and $G_{12}(s)$. $G_{21}(s)$ describes how $cv_2$ changes with respect to a change in $mv_1$ while $G_{21}(s)$ describes how $cv_1$ changes with respect to a change in $mv_2$.

### 2.1 Interaction

Interaction is undesirable in a MIMO system. This is true for setpoint disturbances.While changing setpoint of one loop the other loop should not be affected and if the loop do not interact, each individual loop can be tuned by itself and the whole system should be stable if each individual loop is stable. Unfortunately, due to this interactions design of an effective control system for multivariable process is difficult.

ISSN 2229-5518

When interactions are significant the multi loop PID design most often fails to give acceptable responses. In other words, adjusting control parameters of one loop affects the performance of another, sometimes to the extent of destabilising the entire system. So one approach for square system is to use a decoupler plus a decentralised PID controller.

One great advantage of this method is that it allows the use of single-input single-output(SISO) controller design methods.

## 2.2 Decentralised controller

Decentralized PID controller is one of the most common control scheme for interacting multi-input multi-output (MIMO) plants in chemical and processing industries. The main reason for this is its relatively simple structure, which is to understand and to implement. With decentralized techniques, from figure.2, a multivariable system inputs and output variables is treated as n monovariable systems. The number of tuning parameters is 3n where n is the number of inputs and outputs. While in full matrix PID control there are $3n^2$ parameters. In case of actuator or sensor failure, it is relatively easy to stabilize manually because only one loop is directly affected by the failure. Despite its simple structure, decentralized PID control has long record of satisfactory performance.

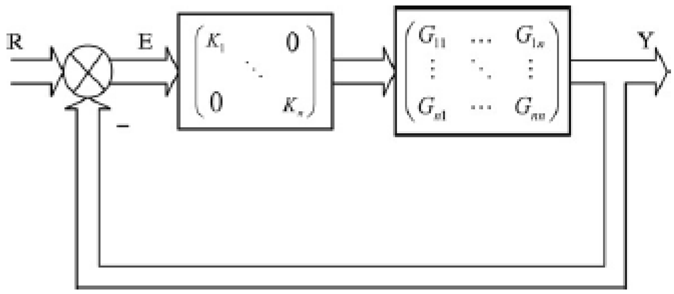


Fig 2.Decentralized control of MIMO process

## 2.3 Decoupler design

The design of decoupled control system with decoupler matrix can be done combining a diagonal controller $K_d(s)$ with a block compensator D(s). With this configuration the controller see the process as a set of n completely independent process or with interaction minimized.

The objective in decoupling is to compensate for the effect of interactions brought about by cross coupling of the process variables.

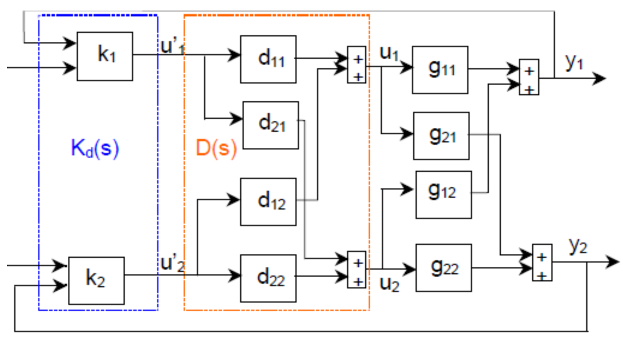


Figure 4. General 2×2 system with decouplers and single-loop controllers

Let the TITO process be

$$G(s) = \begin{pmatrix} g_{11}(s)e^{-\tau_{11}s} & g_{12}(s)e^{-\tau_{12}s} \\ g_{21}(s)e^{-\tau_{21}s} & g_{22}(s)e^{-\tau_{22}s} \end{pmatrix} \quad (4)$$

It supposed that off-diagonal elements of G(s) have no RHP poles. Another assumption is that diagonal elements of G(s) are no RHP zeros

From fig 4,

$$D(s) = \begin{pmatrix} v_1(s) & d_{12}(s)v_2(s) \\ d_{21}(s)v_1(s) & v_2(s) \end{pmatrix} \quad (5)$$

Where:

$$v_1(s) = \begin{cases} 1, & \tau_{21} \geq \tau_{22}, \\ e^{(\tau_{21}-\tau_{22})s}, & \tau_{21} < \tau_{22} \end{cases}$$

$$v_2(s) = \begin{cases} 1, & \tau_{12} \geq \tau_{11} \\ e^{(\tau_{12}-\tau_{11})s} & \tau_{12} < \tau_{11} \end{cases}$$

$$d_{12}(s) = -\frac{g_{12}(s)}{g_{11}(s)} e^{-(\tau_{12}-\tau_{11})s}$$

$$d_{21}(s) = -\frac{g_{21}(s)}{g_{22}(s)} e^{-(\tau_{21}-\tau_{22})s}$$

(6)

$d_{11}(s)=1$ and $d_{22}(s)=1$

$Q(s)= G(s)D(s) = diag\{ q_1(s),q_2(s) \}$ (7)

Q(s) is to be controlled through a decentralised PI(PID) controller. In other words, $q_1(s)$ and $q_2(s)$, which are two SISO plants, are controlled through $k_1(s)$ and $k_2(s)$ respectively, where as each leading diagonal element of decentralised control matrix is a PI(PID) controller.It is worth noting that interactions are zero unless additional large poles are added to the decoupler to make it proper .

## 3.0 Model reduction

In this section a first (second) order plus dead time model is determined for each decoupled process. This approximation is done to determine the tuning parameter values from the optimum tunung formula

### 3.1 FOPDT Approximation Model

Approximation of higher order processes by lower order process plus dead time model is a common practice. Although a FOPDT model does not capture all the features of higher order process, it often reasonably describes the process gain, overall time constant and effective dead time of such a process.

In order to find a approximate FOPDT model for h(s), three unknown parameters namely $k_p$, $\tau_d$ and T should be determined.

The first order system equation is given as

$$l(s) = \frac{k_p e^{-\tau_d s}}{Ts+1} \quad (8)$$

ISSN 2229-5518

$$k_p = h(0) \quad \{9\}$$

The steady state gain and gain margin are same for higher order process and FOPDT model are same.

Hence,

$$l(0) = h(0)$$
$$|l(j\omega_c)| = |h(j\omega_c)|$$
$$\angle\{l(j\omega_c)\} = \angle\{h(j\omega_c)\}$$

Where the cross over frequency, $\omega_c$, of the original system is determined using

$$\angle h(j\omega_c) = \pi$$

As a result, the parameters of FOPDT model can be calculated.

$$k_p = h(0)$$

By equating h $(j\omega_c)$ and the values of $l(j\omega_c)$ can be given as;

$$|h(j\omega_c)| = \frac{k_p}{\sqrt{1+(T\omega_c)^2}}$$
$$|h(j\omega_c)| = \frac{h(0)}{\sqrt{1+(T\omega_c)^2}}$$

$$(T\omega_c)^2 = \left(\frac{h(0)}{|h(j\omega_c)|}\right)^2 - 1$$
$$(T\omega_c) = \sqrt{\left(\frac{h(0)}{|h(j\omega_c)|}\right)^2 - 1}$$
$$T = \frac{\sqrt{\left(\frac{h(0)}{|h(j\omega_c)|}\right)^2 - 1}}{\omega_c} \quad (10)$$

Hence this equation gives the time constant.

For finding $\tau_d$ ,

We are equating phase of $h(j\omega_c)$,

$$-\tau_d\omega_c - \tan^{-1}(T\omega_c) = \angle h(j\omega_c)$$

Where

$$\angle h(j\omega_c) = -\pi$$
$$-\tau_d\omega_c - \tan^{-1}(T\omega_c) = -\pi$$
$$-\tau_d\omega_c = -\pi + \tan^{-1}(T\omega_c)$$
$$\tau_d\omega_c = \pi - \tan^{-1}(T\omega_c)$$
$$\tau_d = \frac{\pi - \tan^{-1}(T\omega_c)}{\omega_c} \quad (11)$$

**3.2 SOPDT Approximation Model:**

Although a large number of industrial processes can be fairly accurately modelled using FOPDT transfer function, if a process has an oscillatory step response, FOPDT model cannot model the process well. In this case a more accurate model of the process can be obtained using SOPDT model in

$$\tilde{l}(s) = \frac{k_p \omega_n^2 e^{-\tau_d s}}{s^2 + 2\xi\omega_n s + \omega_n^2}, 0 < \xi \le 1$$

(12)

Therefore four unknown parameters should be determined they are $k_p$,$\tau_d$,$\omega_n$and$\xi$. To determine the four unknowns four real equations are needed and can be constructed by fitting the process gain h(s) at to nonzero frequency points.

$$\tilde{l}(j\omega_b) = h(j\omega_b)$$
$$\tilde{l}(j\omega_c) = h(j\omega_c)$$

In this method, we pick two points s= j$\omega_c$ and s= j$\omega_b$,

where

$$\angle h(j\omega_b) = -\frac{\pi}{2}$$
$$\angle h(j\omega_c) = -\pi$$

Where $\omega_b$ is determined by equating h(j$\omega_b$) and $\tilde{l}(j\omega_b)$

$$h(j\omega_b) = \frac{k_p\omega_n^2\left(\cos(\omega_b\tau_d) - j\sin(\omega_b\tau_d)\right)}{-\omega_b^2 + 2j\xi\omega_n\omega_b + \omega_n^2}$$
$$-j\left|h(j\omega_b)\right| = \frac{k_p\omega_n^2\left(\cos(\omega_b\tau_d) - j\sin(\omega_b\tau_d)\right)}{\left(\omega_n^2 - \omega_b^2\right) + 2j\xi\omega_n\omega_b}$$
$$\left|h(j\omega_b)\right| = \frac{k_p\omega_n^2\left(\cos(\omega_b\tau_d) - j\sin(\omega_b\tau_d)\right)}{-j\left(\omega_n^2 - \omega_b^2\right) - 2\xi\omega_n\omega_b}$$

By equating the real part

$$\left|h(j\omega_b)\right| = \frac{k_p\omega_n^2\cos(\omega_b\tau_d)}{2\xi\omega_n\omega_b}$$
$$k_p\omega_n\cos(\omega_b\tau_d) = 2\xi\omega_b\left|h(j\omega_b)\right|$$

(13)

By equating the imaginary part

$$\left|h(j\omega_b)\right| = \frac{-k_p\omega_n^2\sin(\omega_b\tau_d)}{-\left(\omega_n^2 - \omega_b^2\right)}$$
$$k_p\omega_n^2\sin(\omega_b\tau_d) = \left(\omega_n^2 - \omega_b^2\right)\left|h(j\omega_b)\right|$$

(14)

Where $\omega_c$ is determined using

$$h(j\omega_c) = \left|h(j\omega_c)\right|(-1 - j(0))$$
$$h(j\omega_c) = -\left|h(j\omega_c)\right|$$

(15)

Now by equating

$$h(j\omega_c)\, and\, \tilde{l}(j\omega_c)$$

$$h(j\omega_c) = \frac{k_p\omega_n^2\left(\cos(\omega_c\tau_d) - j\sin(\omega_c\tau_d)\right)}{-\omega_c^2 + 2j\xi\omega_n\omega_c + \omega_n^2}$$

$$\left|h(j\omega_b)\right| = \frac{k_p\omega_n^2\left(\cos(\omega_b\tau_d) - j\sin(\omega_b\tau_d)\right)}{-\left(\omega_n^2 - \omega_b^2\right) - 2j\xi\omega_n\omega_b}$$

By equating the real part

$$\left|h(j\omega_c)\right| = \frac{k_p\omega_n^2\left(\cos\left(\omega_c\tau_d\right)\right)}{-\left(\omega_n^2-\omega_c^2\right)}$$

$$k_p\omega_n^2\left(\cos\left(\omega_c\tau_d\right)\right) = -\left(\omega_n^2-\omega_c^2\right)\left|h(j\omega_c)\right|$$

(16)

By equating the imaginary part

$$\left|h(j\omega_c)\right| = \frac{-k_p\omega_n^2\left(\sin\left(\omega_c\tau_d\right)\right)}{-2j\xi\omega_n\omega_c}$$

$$k_p\omega_n\left(\sin\left(\omega_c\tau_d\right)\right) = 2j\xi\omega_c\left|h(j\omega_c)\right|$$

(17)

Thus by equating (13) and (17)

$$\frac{k_p\omega_n\cos\left(\omega_b\tau_d\right)}{k_p\omega_n\sin\left(\omega_c\tau_d\right)} = \frac{2\xi\omega_b\left|h(j\omega_b)\right|}{2\xi\omega_c\left|h(j\omega_c)\right|}$$

$$\frac{\cos\left(j\omega_b\tau_d\right)}{\sin\left(j\omega_c\tau_d\right)} = \frac{\omega_b\left|h(j\omega_b)\right|}{\omega_c\left|h(j\omega_c)\right|}$$

$$-\left|h(j\omega_c)\right| = \frac{k_p\omega_n^2\left(\cos\left(\omega_c\tau_d\right)-j\sin\left(\omega_c\tau_d\right)\right)}{\left(\omega_n^2-\omega_c^2\right)+2j\xi\omega_n\omega_c}$$

(18)

By merging Eq. (14) and (16)

$$\frac{k_p\omega_n^2\sin\left(\omega_b\tau_d\right)}{k_p\omega_n^2\cos\left(\omega_c\tau_d\right)} = \frac{\left(\omega_n^2-\omega_b^2\right)\left|h(j\omega_b)\right|}{-\left(\omega_n^2-\omega_c^2\right)\left|h(j\omega_c)\right|}$$

$$\frac{\omega_n^2-\omega_b^2}{\omega_n^2-\omega_c^2} = -\frac{\sin\left(\omega_b\tau_d\right)\left|h(j\omega_c)\right|}{\cos\left(\omega_c\tau_d\right)\left|h(j\omega_b)\right|} \quad 19$$

Eq. (19) gives the value of $\omega_n$ ,then $k_p$ and $\xi$ are determinedfromtheEq(13)and(14) respectively.

ISSN 2229-5518

### 4.0 OPTIMAL TUNING FORMULA

### 4.1 Optimal PI controller for FOPDT process

In order to propose a set of PI tuning formulae for FOPDT model, the PI parameters should be defined based on model parameters,

$$k_c = f_1\left(k_p, \tau_d, T\right),$$
$$T_i = f_2\left(k_p, \tau_d, T\right).$$

Functions $f_1$ and $f_2$ should be determined so that a set of performance criteria is optimised. The optimisation problem may also be include some constrains. Clearly it is very difficult to determine these functions because each parameter of the control is a function of three parameter of the model. To simplify the procedure of determining these functions, non-dimensional tuning(NDT) method for tuning PI controllers for FOPDT process is used. The objective function was to minimize the integral of absolute error(IAE) for a step change in the setpoint. Since the load disturbanceresponses may be poor if the ratio of the time delay to time constant is too small, say less than one-ninth, the integral time should be modified for such processes which are referred to as lag dominant processes. In addition, robustness is a key issue in control systems. Gain and phase margins are often used aas measures of robustness. Considering GM≥3 and PM≥60 as the robustness constrains. Optimal PI tuning formulae resulting from the NDT method capable of coping with FOPDT lag dominant and integrating processes.

The tuning formulae is given by

$$k_c k_p = \frac{T}{2\tau_d} + \frac{1}{14}$$

(20)

$$\frac{T_i}{T} = \min\left(1 + \frac{1}{7}\frac{\tau_d}{T}, 9\frac{\tau_d}{T}\right).$$

(21)

### 4.2 Optimal PID Controller for SOPDT Process

Considering a step change in the setpoint, the NDT formulae for a SOPDT process minimising the IAE and satisfying the GM and PM constraints are shown,

$$k_c = \frac{\xi}{k_p \tau_d \omega_n},$$

$$T_i = \frac{2\xi}{\omega_n},\ T_d = \frac{1}{2\xi\omega_n}.$$

(22)

It is well known that despite good robustness and good setpoint responses, load disturbance responses may be poor if the cancelled poles are shown in comparison with dominant poles.

## 5.0 SIMULATION RESULTS

Three simulation examples are now considered to demonstrate the closed-loop performances of decentralized PID designed with the proposed method.

### 5.1 EXAMPLE 1

The wood-berry binary distillation column plant is a multivariable system with strong interaction and significant time delays, it is described by the following process transfer matrix:

$$G_1(s)=\begin{pmatrix}\dfrac{12.8e^{-s}}{16.7s+1} & \dfrac{-18.9e^{-3s}}{21.0s+1}\\ \dfrac{6.6e^{-7s}}{10.9s+1} & \dfrac{-19.4e^{-3s}}{14.4s+1}\end{pmatrix}$$

$$v_1(s)=1$$
$$v_2(s)=1$$

From Eq.(6)

$$d_{12}(s)=-\frac{\dfrac{-18.9e^{-3s}}{21.0s+1}}{\dfrac{12.8e^{-s}}{16.7s+1}}$$

$$d_{12}(s)=\frac{1.477(16.7s+1)e^{-2s}}{21s+1}$$

$$d_{21}(s)=-\frac{\dfrac{6.6e^{-7s}}{10.9s+1}}{\dfrac{-19.4e^{-3s}}{14.4s+1}}$$

$$d_{21}(s)=\frac{0.34(14.4s+1)e^{-4s}}{10.9s+1}$$

By substituting all values in Eq.(5)

The decoupler matrix is given as

$$D(s)=\begin{pmatrix}1 & \dfrac{1.477(16.7s+1)e^{-2s}}{21s+1}\\ \dfrac{0.34(14.4s+1)e^{-4s}}{10.9s+1} & 1\end{pmatrix}$$

The resulting diagonal system is

$$Q(s)=diag\{q_1(s),q_2(s)\},$$

$$q(s)=G_1(s)D(s)$$

$$q_1(s)=\frac{12.8e^{-s}}{16.7s+1}-\frac{6.43(14.4s+1)e^{-7s}}{(10.9s+1)(21.0s+1)}$$

$$q_2(s)=\frac{-19.4e^{-3s}}{14.4s+1}+\frac{9.745(16.7s+1)e^{-9s}}{(10.s+1)(21.0s+1)}$$

In order to determine a PI controller using the NDT formulae, $q_1(s)$ and $q_2(s)$ should be expressed as FOPDT processes. It is determined by finding $\omega_c$ from the nyquist plot of the original system.

ISSN 2229-5518

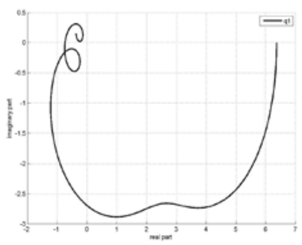

Fig 6. Nyquist plot of the original system(example 1)

From the fig 6,

$$\omega_c = 1.58$$
$$\left|h(j\omega_c)\right| = -0.740$$

$$\left|h(0)\right| = 6.37$$

By substituting all values in Eq.(9),(10) and(11)

$$k_p = 6.37$$

$$T = \frac{\sqrt{\left(\frac{6.37}{-0.740}\right)^2 - 1}}{1.58} = \frac{\sqrt{74.099-1}}{1.58} = 5.41$$

$$\tau_d = \frac{\pi - \tan^{-1}(5.411 \times 1.58)}{1.58} = \frac{\pi - \tan^{-1}(}{1.58}$$
$$\tau_d = 1.065$$

By substituting all the parameters in the Eq.(8)

$$l_1(s) = \frac{6.37e^{-1.065s}}{5.411s+1}$$

Similarly from the Nyquist plot of $q_2$(s),the values are given by,

$$k_p = -9.655, \quad T = 4.684, \quad \tau_d = 2.157$$

Therefore by substituting the parameters in Eq.(8)

$$l_2(s) = \frac{-9.655e^{-2.157s}}{4.684s+1}$$

From fig 7 & 8 shows the nyquist plots of original system and FOPDT models process.

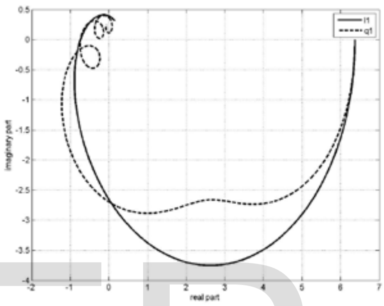

Fig 7. Nyquist plots of $q_1$(s) and $l_1$(s)

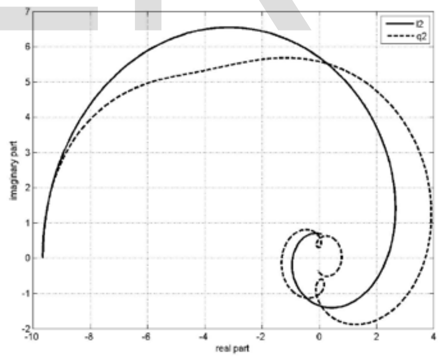

Fig 8.Nyquist plots of $q_2$(s) and $l_2$(s)

Optimal PI controller for FOPDT process:

By substituting values in Eq.(20),

$$k_{c_1} = \frac{1}{6.37} \times \left( \frac{5.411}{2 \times 1.065} + \frac{1}{14} \right)$$

$$k_{c_1} = \frac{1}{6.37} \times (2.54037 + 0.0714)$$

$$k_{c_1} = \frac{1}{6.37} \times (2.6117)$$

$$k_{c_1} = 0.41$$

by substituting all values in Eq.(21)

$$T_{i_1} = 5.411 \times \min\left(1 + \frac{1}{7} \times \frac{1.065}{5.411}, 9\frac{1.065}{5.411}\right)$$

$$T_{i_1} = 5.411 \times \min(1.028, 1.771)$$

$$T_{i_1} = 5.411 \times 1.028$$

$$T_{i_1} = 5.56$$

$$k_{i_1} = 0.41 \times \frac{1}{5.56}$$

$$k_{i_1} = 0.074$$

Simillarly,

$$k_{c_2} = -0.12$$
$$T_{i_2} = -0.024$$

The NDT controller is given as

$$K_{NDT} = \begin{pmatrix} 0.41 + \frac{0.074}{s} & 0 \\ 0 & -0.12 - \frac{0.024}{s} \end{pmatrix}$$

The output response is given as

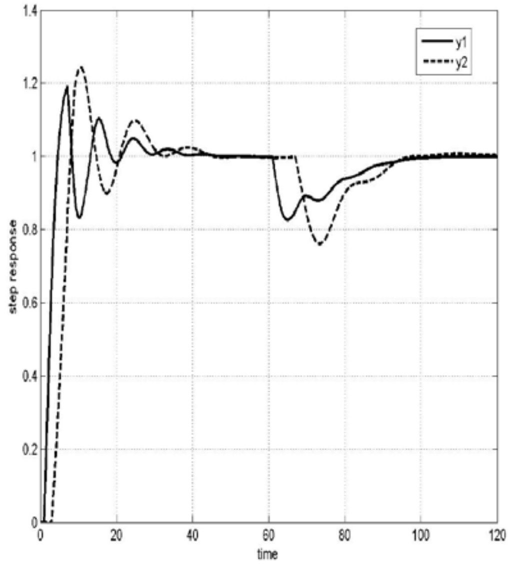


Fig 9.Response of example 1(WOOD-BERRY distillation column)

The BLT and MV method does not use any decoupling strategy. NDT and wang method use the same decouplers but, the advantage is that NDT use only four tuning parameters whereas wang method uses six tuning parameters.

**5.2 EXAMPLE** 2

Process control is an essential part of desalination industry that requires for operation at the optimum operating conditions, an increase in life time of the plant and reduction of unit product cost.

The Alatiqi subsystem transfer matrix is given as

$$G_2(s)=\begin{pmatrix} \dfrac{-0.51e^{-7.5s}}{(32s+1)^2(2s+1)} & \dfrac{-1.68e^{-2s}}{(28s+1)^2(2s+1)} \\ \dfrac{-1.25e^{-2.8s}}{(43.6s+1)(9s+1)} & \dfrac{4.78e^{-1.15s}}{(48s+1)(5s+1)} \end{pmatrix}$$

From Eq.(5)

The decoupler is given as

$$D(s)=\begin{pmatrix} 1 & \dfrac{3.294(32s+1)^2}{(28s+1)^2} \\ \dfrac{0.262(48s+1)(5s+1)e^{-1.65s}}{(43.6s+1)(9s+1)} & e^{-5.5s} \end{pmatrix}$$

Where,

$$v_1(s)=1$$

$$v_2(s)=e^{-5.5s}$$

$$d_{12}(s)=-\frac{\dfrac{-1.68e^{-2s}}{(28s+1)^2(2s+1)}}{\dfrac{-0.51e^{-7.5s}}{(32s+1)^2(2s+1)}}$$

$$d_{12}(s)=\frac{3.294(32s+1)^2e^{5.5}}{(28s+1)^2}$$

$$d_{21}(s)=-\frac{\dfrac{-1.25e^{-2.8s}}{(43.6s+1)(9s+1)}}{\dfrac{4.78e^{-1.15s}}{(48s+1)(5s+1)}}$$

$$d_{21}(s)=\frac{0.262(48s+1)(5s+1)e^{-1.65s}}{(43.6s+1)(9s+1)}$$

The diagonal system is given as

$$q(s)=G_2(s)D(s)$$

The diagonal elements of Q(s) are as follows

$$q_1(s)=\frac{-0.51e^{-7.5s}}{(32s+1)^2(2s+1)}+\frac{0.439(48s+1)(5s+1)e^{-3.65s}}{(28s+1)^2(43.6s+1)(9s+1)(2s+1)}$$

$$q_2(s)=\frac{4.78e^{-6.65s}}{(48s+1)(5s+1)}+\frac{4.118(32s+1)^2e^{-2.8s}}{(28s+1)^2(43.6s+1)(9s+1)}$$



In order to determine a PI controller using the NDT formulae, $q_1(s)$ and$q_2(s)$ should be expressed as FOPDT processes. It is determined by finding $\omega_c$ from the nyquist plot of the original system.

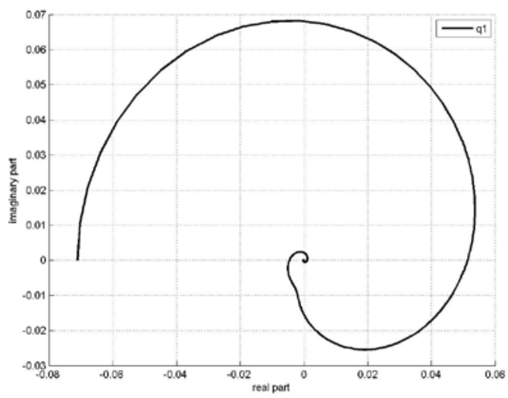


Fig 10.Nyquist plot of the original system.(example 2)

ISSN 2229-5518

From the fig 10,

$$\omega_c = 0.26$$

$$\left|h(j\omega_c)\right| = -0.0045$$

$$\left|h(0)\right| = -0.071$$

By substituting values in Eq.(9), (10) and (11),

$$k_p = -0.071$$

$$T = 58.761$$

$$\tau_d = 77.24$$

By substituting these values in Eq.(9)

$$l_1(s) = \frac{-0.071e^{-77.24s}}{58.761s+1}$$

Similarly

$$l_2(s) = \frac{0.662e^{-61.981s}}{32.133s+1}$$

Fig 11 & 12 shows the Nyquist plots of original system and FOPDT model of the process

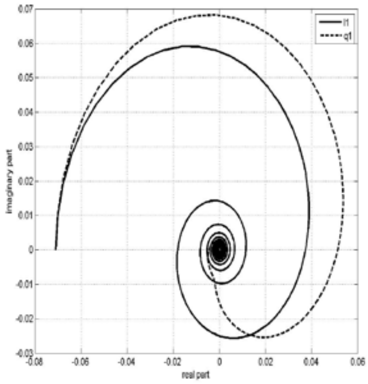


Fig 11.Nyquist plots of $q_1(s)$ and $l_1(s)$

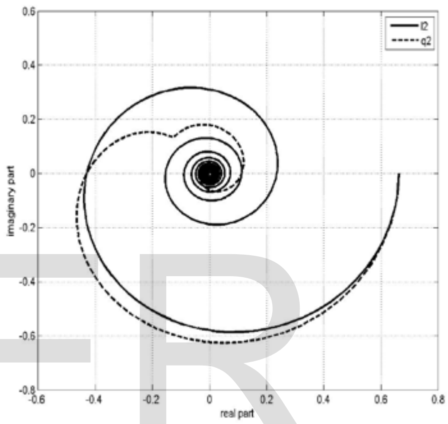


Fig 12.Nyquist plots of $q_2(s)$ and $l_2(s)$



Optimal PI controller for FOPDT process:

By substituting values in Eq.(20),

$$k_{c_1} = \frac{1}{-0.071} \times \left(\frac{58.76}{2\times 77.24} + \frac{1}{14}\right)$$

$$k_{c_1} = \frac{1}{-0.071} \times (0.3803 + 0.0714)$$

$$k_{c_1} = \frac{1}{-0.071} \times (0.45180)$$

$$k_{c_1} = -6.393$$

By substituting values in Eq.(21),

$$T_{i_1} = 58.76 \times \min\left(1 + \frac{1}{7} \times \frac{77.24}{58.76}, 9\frac{77.24}{58.76}\right)$$

$$T_{i_1} = 5.411 \times \min(1.1877, 11.83)$$

$$T_{i_1} = 5.411 \times 1.1877$$

$$T_{i_1} = 6.427$$

$$k_{i_1} = -6.393 \times \frac{1}{6.427}$$

$$k_{i_1} = -0.092$$

Similarly,

$$k_{c_2} = 0.499$$

$$T_{i_2} = 0.012$$

The NDT controller is given as

$$K_{NDT} = \begin{pmatrix} -6.393 + \frac{0.092}{s} & 0 \\ 0 & 0.499 + \frac{0.012}{s} \end{pmatrix}$$

The response is given as

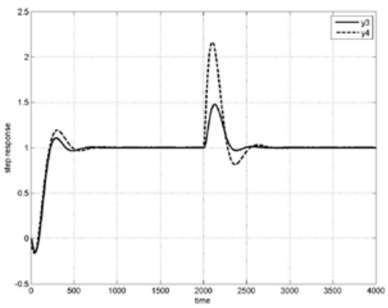


fig 13

Fig 13.Response of example 2(Alatiqi subsystem)

**5.3 EXAMPLE 3**

Let the process be:

$$G_3(s) = \begin{pmatrix} \frac{0.5s}{(0.1s+1)^2(0.2s+1)^2} & \frac{-1}{(0.1s+1)(0.2s+1)^2} \\ \frac{1}{(0.1s+1)(0.2s+1)^2} & \frac{2.4}{(0.1s+1)(0.2s+1)^2(0.5s+1)} \end{pmatrix}$$

Due to large interactions in this process, the performance of the decentralised PID controller based on the critical point is not satisfactory and so is not considered for comparison

Using Eq.(5), the decoupler is given by:

$$D(s) = \begin{pmatrix} 1 & 2(0.1s+1) \\ \frac{-5}{12}(0.5s+1) & 1 \end{pmatrix}$$

Where:

$$v_1(s) = 1$$
$$v_2(s) = 1$$

$$d_{21}(s) = -\frac{\frac{1}{(0.1s+1)(0.2s+1)^2}}{\frac{2.4}{(0.1s+1)(0.2s+1)^2(0.5s+1)}} = \frac{-5}{12}(0.5s+1)$$

$$d_{12}(s) = -\frac{\frac{-1}{(0.1s+1)(0.2s+1)^2}}{\frac{0.5s}{(0.1s+1)^2(0.2s+1)^2}} = 2(0.1s+1)$$



Using additional poles with small time constants, a practical decoupler is given by:

$$\hat{D}(s) = \begin{pmatrix} 1 & 2\frac{0.1s+1}{0.01s+1} \\ \frac{-5}{12}\frac{(0.5s+1)}{0.05s+1} & 1 \end{pmatrix}$$

The diagonal elements of

$$Q(s) = G(s)\hat{D}(s)$$

Are given as

$$q_1(s) = \frac{1}{(0.1s+1)(0.2s+1)^2}\left(\frac{0.5}{0.1s+1} + \frac{\frac{5}{12}(0.5s+1)}{0.05s+1}\right)$$

$$q_2(s) = \frac{1}{(0.2s+1)^2}\left(\frac{2.4}{(0.1s+1)(0.5s+1)} + \frac{2}{0.01s+1}\right)$$

In order to determine a PI controller using the NDT formulae, $q_1(s)$ and $q_2(s)$ should be expressed as FOPDT processes. It is determined by finding $\omega_c$ from the nyquist plot of the original system.

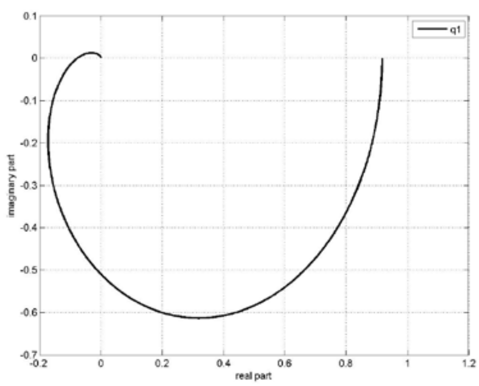


Fig 14.Nyquist plot of the original system.(example 3)

From the Fig 14,

$$\omega_c = 19.89$$

$$|h(j\omega_c)| = -0.0777$$

$$|h(0)| = 0.917$$

By substituting values in Eq(9) (10) and (11)

$$k_p = -0.0777$$

$$T = 0.591$$

$$\tau_d = 0.083$$

By substituting all values in Eq.(8)

The FOPDT model for $q_1(s)$ is given by:

$$l_1(s) = \frac{0.917e^{-0.083s}}{0.591s+1}$$

Similarly

The FOPDT model for $q_2(s)$ is given by:

$$l_2(s) = \frac{4.4e^{-0.252s}}{3.003s+1}$$

SOPDT model reduction:

Since FOPDT model cannot model the process well. In this case, a more accurate model of the process can be obtained using the SOPDT model.

ISSN 2229-5518

It is given as

$$\tilde{l}(s)=\frac{k_p\,\omega_n^2\,e^{-\tau_d s}}{s^2+2\xi\,\omega_n\,s+\omega_n^2},\quad 0<\xi\le 1$$

$\omega_b=5.572$ and $\omega_c=19.89$

$$\left|h(j\omega_b)\right|=0.510155$$
$$\left|h(j\omega_c)\right|=19.89$$

By substituting values in Eq.(18)

$$\frac{\cos(5.572\times\tau_d)}{\sin(19.89\times\tau_d)}=\frac{5.572\times 0.510155}{19.89\times 0.0777}$$

$$\cos(5.572\times\tau_d)=\frac{0.510155}{19.89}$$
$$\cos(5.572\times\tau_d)=0.0256$$

and $\sin(19.89\times\tau_d)=\frac{0.0777}{5.572}$

$$\sin(19.89\times\tau_d)=0.139$$

$$\tau_d=0.029$$

From Eq.(19), substituting the values

$$\frac{\omega_n^2-5.572^2}{\omega_n^2-19.89^2}=-\frac{\sin(5.572\times 0.029)\times 0.0777}{\cos(19.89\times 0.029)\times 0.510155}$$

$$\frac{\omega_n^2-31.047}{\omega_n^2-395.6121}=-\frac{0.002820\times 0.0777}{0.999\times 0.510155}$$

$$\omega_n^2-31.047=(\omega_n^2-395.6121)(-0.0004299)$$

$$\omega_n^2=41.229$$

Then $k_p$ and $\xi$are determined from Eqs.(14) and (13), respectively

$$k_p\times 41.29\times\cos(19.89\times 0.029)=(41.229-31.047)\times 0.077$$

$$k_p=0.7925$$

$$0.7925\times 41.229\times\cos(5.572\times 0.029)=2\times\xi\times 0.510155$$

$$\xi=0.88374$$

Thus by substituting all four unknown values in Eq.(12)

$$\tilde{l}_1(s)=\frac{32.674e^{-0.029s}}{s^2+11.349s+41.229}$$

Similarly SOPDT model for $q_2$ is

$$\tilde{l}_2(s)=\frac{46.601e^{-0.008s}}{s^2+7.838s+9.443}$$

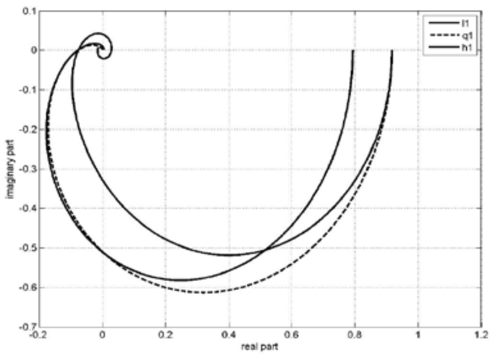


Fig 15.The Nyquist plots of $q_1(s)$ and its appxoximation models(example 3)

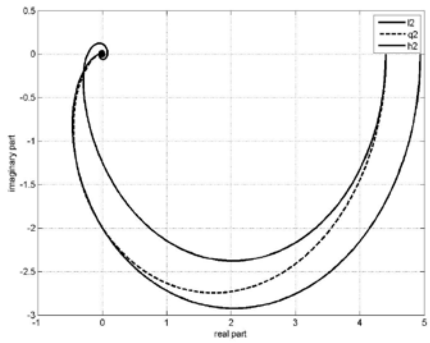

Fig 16.The Nyquist plots of $q_2$(s) and its approximation models (example 3)

OPTIMAL TUNING:

Optimal PI controller for FOPDT process

$$k_{c_1} = \frac{0.917 \times 0.591}{2 \times 0.083} + \frac{1}{14}$$

$$k_{c_1} = 3.946$$

and

$$T_{i_1} = 0.591 \times \left(\min\left(1.02006, 1.2695\right)\right) = 0.591 \times 1.02006$$

$$T_{i_1} = 0.6028$$

$$k_{i_1} = 3.946 \times \frac{1}{0.6028}$$

$$k_{i_1} = 6.551$$

Similarly,

$$k_{c_2} = 1.368$$

$$k_{i_2} = 0.451$$

The NDT-PI controller is given by

$$K_{NDT-PI} = \begin{pmatrix} 3.946 + \frac{6.551}{s} & 0 \\ 0 & 1.368 + \frac{0.451}{s} \end{pmatrix}$$

The response of it is given as

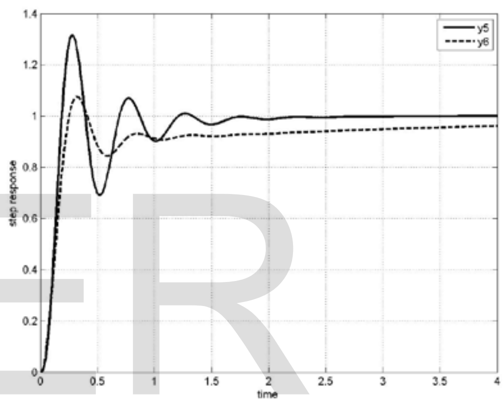

Fig 17. Response of example 3(with PI controller)

Optimal PID controller for SOPDT process:

By substituting all values in Eq.(22)

$$k_{c_1} = \frac{0.88374}{0.7925 \times 0.029 \times 6.42098}$$

$$k_{c_1} = \frac{0.88374}{0.14757}$$

$$k_{c_1} = 6.088$$

$$T_{i_1} = \frac{2 \times 0.88374}{6.42098}$$

$$T_{i_1} = 0.275266$$

$$k_{i_1} = 6.088 \times \frac{1}{0.275266}$$

$$k_{i_2} = 22.11$$

$$T_{d_1} = \frac{1}{2 \times 0.88374 \times 6.42098}$$

$$T_{d_1} = \frac{1}{11.34895}$$

$$T_{d_1} = 0.088113$$

$$k_{d_1} = 6.088 \times 0.088113$$

$$k_{d_1} = 0.536$$

The NDT-PID controller is given by

$$K_{NDT-PID} = \begin{pmatrix} 6.088 + \frac{22.113}{s} + 0.536s & 0 \\ 0 & 10.529 + \frac{12.686}{s} + 1.344s \end{pmatrix}$$

The response is given as

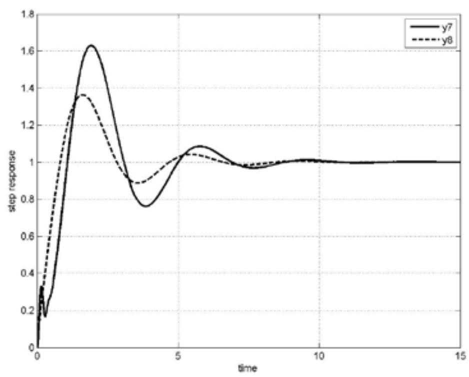


Fig 18. Response of example 3(with PID controller)

Due to adding poles with small time constants to off-diagonal elements of the decoupler, the NDT controllers have non-zero interactions. Clearly the NDT-PID controllers gives the best response interms of set point regulation and load disturbance rejection.

## 6. CONCLUTION

This method is one of the simplest method to tune decentralised PI(PID) controllers for TITO process. The TITO process was decoupled through a decoupler matrix. A model reduction method was done to find the suitable FOPDT(SOPDT) model for each element of the resulting diagonal process through fitting the nyquist plots at particular points. The performance of the NDT technique was investigated through examples. Compared to other methods it has less number of tuning parameters and more over uses only less number of controllers, therefore it is one of the cost effective method to control the process

## REFERENCE

1. Luyben, W.L. (1986). "Simple method for tuning for SISO controllers in multivariable systems".
2. SaeedTavakoli, Ian Griffin, Peter . Fleming (2006). "Tuning of decentralised PI(PID) controllers for TITO process".
3. Tavakoli, S., Griffin, I., & Fleming, P.J.(2003).

"Optimal tuning of PI controlles for first order plus dead time/long dead time models using dimension analysis".

4. Wang, Q.G.,Zou,B.,Lee,T.H., &Bi, Q.(1997)."Auto-tuning of multi-variable PID controllers from decentralized relay feedback".

5. Wang, Q.G., Lee, T.H., Fung, H. W., Bi, Q.,&Zhang, Y.(1999). "PID tuning for improved performance."

6. Wood, R. K., & Berry, M.W. (1973). "Terminal composition control of a binary distillation column. Chemical Engineering Science".

www.Waset.com

www.archivesofcontrolscience.com