

NOTICE: this is the author's version of a work that was accepted for publication in Journal of Process Control. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Journal of Process Control, Vol.24, (2014). DOI: 10.1016/j.jprocont.2014.03.001

Multi-Loop Design of Multi-Scale Controllers for Multivariable Processes

Jobrun Nandong^{*,1} and Zhuquan Zang²

¹ *Department of Chemical Engineering, Curtin University Sarawak, 98009 Miri, Sarawak, Malaysia*

² *Department of Electrical and Computer Engineering, Curtin University Sarawak, 98009 Miri, Sarawak, Malaysia*

*Corresponding author: Jobrun Nandong

Tel: +6085-443824; Fax: +6085-443838; Email: jobrun.n@curtin.edu.my

ABSTRACT

Based on the recently proposed (SISO) multi-scale control scheme, a new approach is introduced to design multi-loop controllers for multivariable processes. The basic feature of the multi-scale control scheme is to decompose a given plant into a sum of basic modes. To achieve good nominal control performance and performance robustness, a set of sub-controllers are designed based on the plant modes in such a way that they are mutually enhanced with each other so as to optimize the overall control objective. It is shown that the designed multi-scale controller is equivalent to a conventional PID controller augmented with a filter. The multi-scale control scheme offers a systematic approach to designing multi-loop PID controllers augmented with filters. Numerical studies show that the proposed multi-loop multi-scale controllers provide improved nominal performance and performance robustness over some well-established multi-loop PID controller schemes.

Keywords: Multivariable Process; Decentralized Control; Multi-Loop PID; Multi-Scale Control

1. Introduction

Most industrial processes are multivariable or multi-input and multi-output (MIMO) in nature where for many decades, the decentralized control architecture (multi-loop PID control) has been widely applied to these types of processes. The main reason that the multi-loop PID control has been preferred to full multivariable control is due to the fact that the multi-loop PID control system is relatively simple to design and implement [1]. However, the effectiveness of the multi-loop PID control in MIMO processes has often been limited by the presence of process interactions or control-loop interactions. The presence of process interactions in MIMO processes has been recognized as one of the main culprits responsible for poor multi-loop control performance. Besides the process interactions, the presence of deadtime (time delay) and inverse-response behaviors have also been recognized as important factors imposing limitation on control performance in process plants. For MIMO processes, a number of multi-loop PID control designs have been proposed over the last decades with the aim to achieve good control performance despite the limitation imposed by process interactions. One example is the independent design method proposed in [2]. This method is quite simple to apply but it has a disadvantage resulting from the negligence of how the other control-loops are designed. This negligence leads to poor control performance. Another example is the sequential design method proposed in [3]. Unlike the independent method, the sequential design method attempts to include the effect of closing subsequent loops into the design problem. Nevertheless, as the control performance can be highly dependent on how the design sequence is chosen, the sequential design method could also result in poor overall control performance [4].

Besides the independent and sequential design methods, another well-known multi-loop control design is based on the detuning approach. In the detuning approach, the performance of individual controllers is first tuned based on a single-loop controller design approach (e.g.,

Ziegler-Nichols tuning). The individual controllers are then detuned (reduced performance) once all the control loops are closed. A well-known detuning method is the Biggest Log-Modulus (BLT) tuning proposed in [5]. The BLT method is based on the Ziegler-Nichols tuning for each single-loop controller where a single detuning parameter is introduced to meet the stability criterion of the Biggest Log-Modulus. The BLT method is simple to use but it can lead to sluggish or oscillatory responses. In addition to the BLT method, Lee and Edgar [4] proposed another method via which the dominant poles can be shifted to some favorable locations. By shifting the poles to desirable locations, the multi-loop control performance can be improved and sluggish or responses can also be avoided. Another pole placement approach is the root trajectory method proposed by Zhang et al. [6].

It is interesting to note that some researchers have also proposed the application of Internal Model Control (IMC) to the multi-loop PID control design, i.e., IMC-PID design [7-9]. In the IMC-PID design of Vu et al. [9], the individual PID controller parameters are expressed in term of a single tuning parameter, i.e., based on the closed-loop time constant of each loop. Note that, the controllers designed based on the IMC approach may not be in the first place the same as PID controllers. Hence, in order to obtain the standard PID controllers via the IMC design, a controller reduction process is often required, e.g., in [9] the IMC-based controllers are reduced to PID controllers using Maclaurin series.

Chen and Seborg [10] proposed a method combining the idea of independent design and Nyquist stability analysis. This method consists of two steps: (1) identifying the stability region for PI controllers, and (2) selecting appropriate PI controller settings within this stability region. Advantageously, this method can guarantee the closed-loop stability. Lee et al. [11] proposed a method that combined the Nyquist array analysis with an iterative continuous cycling approach in order to design multi-loop PI controllers. Kaspar and Ray [17] proposed the application of chemometric approach, i.e., Principal Component Analysis

(PCA) and Partial Least Squares (PLS) to the design of multi-loop PI control. The advantages of the approach as part of the overall control system design include automatic decoupling and efficient loop pairing, as well as the natural ability to handle non-square system. Lakshminarayanan et al. [28] further extended the chemometric approach to the dynamic PLS case, where the reduced process model can be used to design multi-loop control system including feedforward controller. There are many other methods for multi-loop control designs; e.g., see [12-18].

In this paper, we introduce a new multi-loop controller design based on the recently proposed multi-scale control (MSC) scheme for SISO processes proposed by Nandong and Zang [19-20]. The basic principle of the MSC scheme is to decompose a given plant into a sum of few basic modes or factors each with distinct speed of responses - different time-scales. To achieve good nominal control performance and performance robustness it is vital that the required controller is designed in such a way that it can promote good cooperation among these different plant modes. It is interesting to point out that, the designed MSC controller is actually equivalent to a conventional PID controller augmented with a simple (often a first or second order) filter. In this respect, the MSC scheme provides a competitive alternative to PID controller design. We shall demonstrate the applicability and effectiveness of the MSC scheme to designing multi-loop PID control for multivariable processes.

The rest of this paper is organized as follows. In Section 2, a brief overview of the multi-scale control (MSC) scheme and the derivation of two PID controller tuning formulas are presented. In Section 3, a general procedure for the multi-loop MSC controller design and a simple algorithm based on the MSC-PID tuning formulas are provided. Section 4 presents some illustrative examples to compare the performances of the proposed multi-loop MSC controllers (equivalent PID controllers) with some of the existing multi-loop PID controller

designs including the centralized Model Predictive Control (MPC) strategy. Finally, some concluding remarks and future work are highlighted in Section 5.

2. Fundamental of Multi-Scale Control Scheme

2.1 Preliminary

In the multi-scale control (MSC) scheme proposed by Nandong and Zang [19-20], it is assumed that a given plant of interest can be decomposed (via partial fraction expansion) into a sum of basic modes or factors. These basic modes should be different in their speed of responses to a given input (manipulated variable) – the modes with multi-scale dynamics.

In general, for a single-input single-output (SISO) process the decomposition in the MSC scheme can be represented as

$$P(s) = M_0(s) + M_1(s) + M_2(s) + \dots + M_n(s) \quad (1)$$

where $P(s)$ denotes the plant and $M_i(s), \forall i \in \{0, 1, 2, \dots, n\}$ represent the plant modes, which could be either a first or second order transfer function with real coefficients. It is also assumed that $M_j(s)$ has a slower speed of response than $M_{j+1}(s), \forall j \in \{0, 1, \dots, n-1\}$. Note that, equation (1) implies that the plant P can be decomposed into a sum of $n+1$ basic modes.

The principle of the MSC scheme is to synthesize a controller that can enhance cooperation among the different plant modes, which is crucial to improve both nominal performance and performance robustness. To achieve this enhanced cooperation, the MSC scheme advocates the idea of using several individual sub-controllers where each sub-controller is tailored to control a specific plant mode. In theory, for a given plant that can be decomposed into a sum of $n+1$ basic modes, there will be $n+1$ number of separate sub-controllers required. In a practical application, however, a fewer number of sub-controllers might in fact be required than in the ideal case where this can be done by applying a model reduction process to a given high-order process based on which the controller is designed.

For example, a fourth-order process model will ideally require 4 sub-controllers. If the fourth-order model could be reduced to a second-order model, then the MSC scheme based on this reduced model will only require 2 sub-controllers.

2.2 Realization of the Multi-Scale Control Scheme

The details about the MSC scheme can be found in the recent papers by Nandong and Zang [19-20]. Here, we only present a brief overview of the scheme based on the 3-layer MSC structure. Fig. 1 shows the realization block diagram of the 3-layer MSC structure. It is assumed that a given plant $P(s)$ can be decomposed into a sum of 3 basic modes. Referring to Fig. 1, K_i is the sub-controller to control the mode $M_i(s)$, $\forall i \in \{0, 1, 2\}$; $W_j(s)$, $j = 1, 2$ the multi-scale predictor for the j^{th} – inner-loop; E , D , R , Y and U_i denote the signals for error, disturbance, setpoint, controlled variable and i^{th} – sub-controller output, respectively. Notice that, the 3-layer MSC scheme shown in Fig. 1 can be reduced to a single-loop block diagram as shown in Fig. 2.

The multi-scale predictor $W_j(s)$, $j = 1, 2$ is often chosen to be the inner mode

$$\mathcal{W}(s) = \begin{bmatrix} W_1(s) \\ W_2(s) \end{bmatrix} = \begin{bmatrix} \overline{M}_1(s) \\ \overline{M}_2(s) \end{bmatrix} \quad (2)$$

where $\overline{M}_i(s)$ denotes the nominal model for the mode $M_i(s)$.

Referring to Fig. 2, the innermost layer ($n = 2$) transfer function is written as follows

$$G_2(s) = \frac{U_2(s)}{U_1(s)} = \frac{K_2(s)}{1 + K_2(s)W_2(s)} \quad (3)$$

where the next inner-layer ($n = 1$) transfer function can be expressed as

$$G_1(s) = \frac{K_1(s)}{1 + K_1(s)W_1(s)G_2(s)} \quad (4)$$

The overall multi-scale controller K_{msc} can be obtained as follows

$$K_{msc}(s) = K_0(s)G_1(s)G_2(s) \quad (5)$$

Note that, the inner-layer transfer function (3) is useful for synthesizing the innermost sub-controller K_2 . To synthesize the remaining 2 sub-controllers, it will be helpful to first construct the augmented inner-layer transfer function given by

$$Q_1(s) = W_1(s)G_2(s) \quad (6)$$

and the augmented overall plant transfer function

$$P_c(s) = G_1(s)G_2(s)P(s) \quad (7)$$

It is often sufficient to choose the inner-layer sub-controller as a P-only controller to simplify the controller tuning. Meanwhile, the outermost sub-controller is chosen as a PI controller. For a complicated process, a more advanced controller can be chosen for the outermost sub-controller, e.g., a PID or LQG controller.

For the MSC scheme shown in Fig. 1, the closed-loop setpoint transfer function from R to Y is given by

$$H_{ry}(s) = \frac{K_{msc}(s)P(s)}{1 + K_{msc}(s)P(s)} \quad (8)$$

and that from the input disturbance D to Y as

$$H_{dy} = \frac{P(s)}{1 + K_{msc}(s)P(s)} \quad (9)$$

Notice that, the transfer functions (8) and (9) have similar characteristic equations. Thus, if the multi-scale controller $K_{msc}(s)$ is stable for the setpoint tracking, then it will also be stable for the input disturbance even if the plant is an open-loop unstable process. Significantly, this means that the MSC scheme does not suffer from internal instability problem.

2.3 Equivalent MSC-PID Controller Derivation

In the following, we shall demonstrate how to synthesize an equivalent PID controller based on the MSC scheme – derivation of the MSC-PID tuning formula. In the present work, we only derive 2 tuning formulas based on First-Order plus Deadtime (FOPDT) and Second-Order plus Deadtime (SOPDT) models. Furthermore, we assume that the 1/1 Padé formula is applied to approximate the deadtime; thus, the FOPDT model will lead to a 2-layer MSC scheme while the SOPDT model will result in a 3-layer MSC scheme.

Case 1 – 2-Layer MSC Scheme

Consider a process which can be represented by the FOPDT model

$$P(s) = \frac{K_m e^{-\theta_m s}}{\tau_m s + 1} \quad (10)$$

By applying the 1/1 Padé formula to approximate the delay component, an approximated second-order model is obtained

$$P(s) = \frac{K_m (1 - \tau_1 s)}{(\tau_m s + 1)(1 + \tau_1 s)} \quad (11)$$

where it is assumed that $\tau_1 = 0.5\theta < \tau_m$.

Upon the decomposition of equation (11) using partial fraction expansion

$$P(s) = \frac{k_0}{\tau_m s + 1} + \frac{k_1}{\tau_1 s + 1} \quad (12)$$

where the mode gains k_0 and k_1 can be calculated using

$$k_0 = \frac{K_m (\tau_m + \tau_1)}{\tau_m - \tau_1} \quad (13)$$

$$k_1 = \frac{2\tau_m K_m}{\tau_1 - \tau_m} \quad (14)$$

If $\tau_1 = 0.5\theta > \tau_m$ then the expressions for k_0 and k_1 will be reversed as given by

$$k_0 = \frac{2\tau_m K_m}{\tau_1 - \tau_m} \quad (15)$$

$$k_1 = \frac{K_m(\tau_m + \tau_1)}{\tau_m - \tau_1} \quad (16)$$

For $\tau_1 = 0.5\theta < \tau_m$, the multi-scale predictor is chosen to be the faster inner mode, i.e., $W(s) = k_1/(\tau_1 s + 1)$. If $\tau_1 = 0.5\theta > \tau_m$, the lag component will be the faster mode and the predictor becomes $W(s) = k_1/(\tau_m s + 1)$. From this point onward, we assume that $\tau_1 = 0.5\theta < \tau_m$ unless otherwise stated.

Assuming that the P-only controller with gain K_{1p} is used to control the inner mode, we can write the inner-layer closed-loop setpoint transfer function as

$$H_{r1}(s) = \frac{K_{1p}W(s)}{1 + K_{1p}W(s)} \quad (17)$$

After substituting $W(s) = k_1/(\tau_1 s + 1)$ into equation (17), and followed by simplification

$$H_{r1}(s) = \frac{K_{1p}^o}{\tau_{c1}s + 1} \quad (18)$$

where the overall gain and closed-loop time constant are given by

$$K_{1p}^o = \frac{K_{1p}}{1 + K_{1p}k_1} \quad (19)$$

$$\tau_{c1} = \frac{\tau_1}{1 + K_{1p}k_1} \quad (20)$$

Let us define the ratio of the open-loop to the closed-loop (OC) time constant as follows

$$\lambda_1 = \frac{\tau_1}{\tau_{c1}} \quad (21)$$

Significantly, the OC time-constant ratio defined in equation (21) indicates the closed-loop response speed in comparison with the open-loop case for a given inner-layer loop. The larger the value of λ_1 , the faster is the closed-loop response of the inner-layer loop. Hence, we can specify a desired value for λ_1 based on which the value of K_{1p} is determined. To

determine K_{1p} , let us substitute equation (20) into (21) where after simplification K_{1p} can be written as

$$K_{1p} = \frac{\lambda_1 - 1}{k_1} \quad (22)$$

Equation (22) can be used to obtain K_{1p} by first specifying the λ_1 .

Our next task is to determine the required setting for the outermost sub-controller. To do this, we assume that the PI controller is chosen to control the outermost mode, i.e.:

$$K_0(s) = K_{0p} \left(1 + \frac{1}{\tau_{I0} s} \right) \quad (23)$$

To determine K_{0p} , we can use the same approach as in the inner mode case, i.e., by specifying the OC time-constant ratio for the outermost mode as follows

$$K_{0p} = \left(\frac{\lambda_0 - 1}{k_0} \right) \Gamma_{k_1} \quad (24)$$

where Γ_{k_1} indicates the sign for the inner mode gain k_1 (this sign is included to ensure that a correct sign of the overall controller gain is obtained).

Meanwhile, the reset time is set based on a desired fraction γ of the open-loop time constant. We suggest

$$\tau_{I0} = \gamma \tau_m, \quad 0.1 \leq \gamma \leq 1.2 \quad (25)$$

For this 2-layer MSC scheme, the only inner-layer transfer function is given by

$$G_1(s) = \frac{K_{1p}^o (\tau_1 s + 1)}{\tau_{c1} s + 1} \quad (26)$$

The overall MSC controller can then be obtained as in (5), i.e., combining (23) and (26):

$$K_{msc}(s) = K_{0p} K_{1p} \left(1 + \frac{1}{\tau_{0I} s} \right) \left(\frac{\tau_1 s + 1}{\tau_{c1} s + 1} \right) \quad (27)$$

Notice that, equation (27) is similar to the classical form of PID controller, which can be easily converted into a PID controller augmented with a first order filter given as follows

$$K_{msc}(s) = K_c \left(1 + \frac{1}{\tau_I s} + \tau_D s \right) \left(\frac{1}{\tau_f s + 1} \right) \quad (28)$$

Upon comparison of equations (27) and (28), the PID controller parameters in terms of new tuning parameters $(\lambda_0, \lambda_1, \gamma)$ are

$$K_c = \left(\frac{(\lambda_0 - 1)(\lambda_1 - 1)}{\gamma \lambda_1} \right) \left(\frac{\gamma \tau_m + 0.5 \theta_m}{\tau_m + 0.5 \theta_m} \right) \left(\frac{(\tau_m - 0.5 \theta_m)^2}{\tau_m \theta_m K_m} \right) \quad (29)$$

$$\tau_I = \gamma \tau_m + 0.5 \theta_m \quad (30)$$

$$\tau_D = \frac{\gamma \theta_m \tau_m}{2 \gamma \tau_m + \theta_m} \quad (31)$$

$$\tau_f = \frac{\theta_m}{2 \lambda_1} \quad (32)$$

where the OC time-constant ratios $\lambda_0 > 1$ and $\lambda_1 > 1$.

Case 2: 3-Layer MSC Scheme

In this case, we consider a process given by the SOPDT model as

$$P(s) = \frac{K_m (\tau_z s + 1) e^{-\theta_m s}}{(\tau_0 s + 1)(\tau_1 s + 1)} \quad (33)$$

The application of the 1/1 Padé formula to equation (33) leads to

$$P(s) \frac{K_m (\tau_z s + 1)(1 - \tau_2 s)}{(\tau_0 s + 1)(\tau_1 s + 1)(1 + \tau_2 s)} \quad (34)$$

The fractional expansion of (34) yields a sum of 3 basic modes:

$$P(s) = \frac{k_0}{\tau_0 s + 1} + \frac{k_1}{\tau_1 s + 1} + \frac{k_2}{\tau_2 s + 1} \quad (35)$$

where the mode gains k_0 , k_1 and k_2 are given by

$$\begin{bmatrix} k_0 \\ k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} \frac{K_m(\tau_0 - \tau_z)(\tau_0 + \tau_2)}{(\tau_0 - \tau_1)(\tau_0 - \tau_2)} \\ \frac{K_m(\tau_1 - \tau_z)(\tau_1 + \tau_2)}{(\tau_1 - \tau_0)(\tau_1 - \tau_2)} \\ \frac{2\tau_2 K_m(\tau_2 - \tau_z)}{(\tau_2 - \tau_0)(\tau_2 - \tau_1)} \end{bmatrix} \quad (36)$$

The mode gains (36) are derived based on the assumption that $\tau_2 = 0.5\theta_m < \tau_1 < \tau_0$, i.e. half of the delay is less than the faster lag time τ_1 .

Similar to the first case, let us assume that the P-only controllers are used to control the first and second inner modes with gains K_{1p} and K_{2p} respectively; a PI controller is used to control the outermost mode. The sub-controller gains can be obtained in the same way as in the previous case 1:

$$\begin{bmatrix} K_{0p} \\ K_{1p} \\ K_{2p} \end{bmatrix} = \begin{bmatrix} \left(\frac{\lambda_0 - 1}{k_0} \right) \Gamma_{k_1 k_2} \\ \left(\frac{\lambda_1 - 1}{k_1} \right) \Gamma_{k_2} \\ \left(\frac{\lambda_2 - 1}{k_2} \right) \end{bmatrix} \quad (37)$$

where $\Gamma_{k_i k_j}$ denote the sign of $k_i \times k_j$.

For this 3-layer MSC scheme, the combined inner-layer transfer functions can be expressed in the form of

$$L(s) = G_1(s)G_2(s) = K_A \frac{(\tau_1 s + 1)(\tau_2 s + 1)}{a_2 s^2 + a_1 s + 1} \quad (38)$$

$$K_A = \frac{K_{1p} K_{2p}^o}{1 + K_{2p}^o K_{1p} k_1} \quad (39)$$

$$a_1 = \frac{\tau_1 + \tau_2(1 + K_{2p}^o K_{1p} k_1)}{1 + K_{2p}^o K_{1p} k_1} \quad (40)$$

$$a_2 = \frac{\tau_1 \tau_2}{1 + K_{2p}^o K_{1p} k_1} \quad (41)$$

Here, $\tau_{c2} = \tau_2 / (1 + K_{2p}k_2)$ and $K_{2p}^o = K_{2p} / (1 + K_{2p}k_2)$.

By combining (23) with (38) as in (5), the overall MSC controller can be expressed as

$$K_{msc}(s) = K_c \left(1 + \frac{1}{\tau_{0I} s} \right) (\tau_1 s + 1) G_f(s) \quad (42)$$

From (42), by taking the augmented filter as

$$G_f(s) = \frac{\tau_2 s + 1}{a_2 s^2 + a_1 s + 1} \quad (43)$$

We can now obtain the PID controller parameters using the following equations (44) – (46):

$$K_c = \left[\left(\frac{K_A (\gamma \tau_0 + \tau_1)}{\gamma \tau_0} \right) \left(\frac{\lambda_0 - 1}{k_0} \right) \right] \Gamma_{K_m} \quad (44)$$

$$\tau_I = \gamma \tau_0 + \tau_1 \quad (45)$$

$$\tau_D = \frac{\gamma \tau_0 \tau_1}{\gamma \tau_0 + \tau_1} \quad (46)$$

The equations (39) – (41) can be expressed in terms of the new tuning parameters as

$$K_A = \frac{1}{k_1} \left(1 + \frac{\lambda_2 k_2}{(\lambda_1 - 1)(\lambda_2 - 1)} \Gamma_{k_2} \right)^{-1} \quad (47)$$

$$a_1 = \tau_1 \left(1 + \frac{(\lambda_1 - 1)(\lambda_2 - 1)}{\lambda_2 k_2} \Gamma_{k_2} \right)^{-1} + \tau_2 \quad (48)$$

$$a_2 = \left(\frac{\tau_1 \tau_2}{\lambda_2} \right) \left(1 + \frac{(\lambda_1 - 1)(\lambda_2 - 1)}{\lambda_2 k_2} \Gamma_{k_2} \right)^{-1} \quad (49)$$

Note in this case, the overall MSC controller (42) is equivalent to a PID controller augmented with a second-order filter (43). The values for $k_i, i = 0, 1, 2$ are calculated via equation (36).

3. Multi-Loop MSC Controller Design Procedure

3.1. General Procedure

The following steps represent a general procedure for designing equivalent MSC-PID controllers for multivariable processes.

Step 1: Identify the suitable controller pairings. This can be done via RGA analysis for simplicity. After the controller pairings have been determined, arrange the transfer function matrices in a way that corresponds to the direct pairing case. Each MSC controller $K_{msc,i}(s)$ can be designed independently based on the diagonal transfer function $g_{ii}(s)$.

Step 2: Decompose $g_{ii}(s)$ into a sum of basic modes as illustrated below:

$$g_{ii}(s) = M_{i,0}(s) + M_{i,1}(s) + \cdots + M_{i,n}(s) \quad (50)$$

Here, for $M_{ij}(s)$ the first subscript i denotes the i^{th} control loop and the second subscript j denotes the j^{th} mode.

Step 3: Design the $K_{msc,i}(s)$ based on (50) (assuming 3-layer MSC scheme):

Step 3.1: Assume that a P-only controller is chosen, first design the innermost sub-controller $K_{i,2}$ based on the innermost mode $M_{i,2}(s)$. Obtain the sub-controller tuning value $K_{i,2}^*$ based on the minimum Integral Absolute Error (IAE) criterion via the Matlab SISO Design tool.

Step 3.2: The implemented sub-controller is $K_{i,2} = 0.5K_{i,2}^*$. Use this setting to construct the innermost layer transfer function $G_{i,2}(s)$ as in (3) and the augmented inner-layer transfer function $Q_{i,1}(s)$ as in (6).

Step 3.3: Assume a P-only controller is used, design the second inner-layer sub-controller $K_{i,1}$ based on the minimum IAE criterion to obtain a tuning value $K_{i,1}^*$.

Step 3.4: Set the sub-controller tuning $K_{i,1} = 0.5K_{i,1}^*$ and construct $G_{i,1}(s)$ as in (4).

Step 3.5: Construct the augmented overall transfer function $P_{i,c}(s)$ as in (7) and finally use this transfer function to design the outermost sub-controller $K_{i,0}(s)$. The overall MSC controller for the i^{th} –loop is obtained as in (5). The overall MSC controllers can be easily rearranged into the form of practical PID controllers augmented with filters; see [20].

Step 4: Refinement of MSC-PID controllers designed in Step 3.

Step 4.1: Upon closing all the control loops, it is often required to refine (probably detuning) the MSC controllers. We recommend refining the outermost sub-controllers first while keeping the inner-layer sub-controllers constant. As an illustration, let us assume that a PI controller is adopted for the outermost sub-controller $K_{i,0}$ where based on the minimum IAE criterion (or some other criteria) we can obtain via Step 3.5:

$$K_{i,0}^* = K_{c,i}^* (1 + 1/\tau_{I,i} s)$$

Next, we can retune the above controller as:

$$K_{i,0} = K_{i,0}^* / RF_{i,0}$$

In other words, the controller gain $K_{c,i}^*$ is retuned by a factor of $RF_{i,0}$; $RF_{i,0} > 1$ implies detuning of the multi-scale controller while $RF_{i,0} < 1$ implies increasing the controller tuning (increase the controller aggressiveness).

Step 4.2: After retuning all the outermost sub-controllers as in Step 4.1, it may still be necessary to further retune the inner-layer sub-controllers if certain target responses remain unachievable, e.g. overshoot of certain controlled variables are too high which implies that some of the innermost sub-controllers might be too aggressive.

Remark 1 - The P/PI/PID controllers used in the MSC scheme are designed using the Matlab SISO Design Tool (Matlab Control System Toolbox). The proposed design (i.e., tuning procedure) is rather general where other tuning procedures, e.g. IMC and LQG synthesis can also be applied.

3.2. Algorithm: Design via MSC-PID Tuning Formula

As an alternative to the general procedure given in the previous sub-section, in the following we propose an algorithm for the direct synthesis of PID controller based the MSC-PID formulas given in Sub-Section 2.3.

Step 1: For a given i^{th} –loop, specify the values for open-loop to closed loop (OC) time constant ratios λ_j ($j = 1,2$ based on the FOPDT model or $j = 1,2,3$ based on the SOPDT model) with an initial setting of $\gamma = 1$.

Step 2: Obtain the PID controller parameters using equations (29) to (32) if the diagonal transfer function $g_{ii}(s)$ is represented by the FOPDT model or using the equations (44) to (49) if $g_{ii}(s)$ is given as the SOPDT model.

Step 3: Check the gain margin (GM) and phase margin (PM) using either the Nyquist or Bode plot. The target value for GM should be approximately between 6 dB and 10 dB while the PM should be approximately between 40° and 65° . If GM and PM are outside these desired ranges, go back to Step 1 and reset λ_j .

Step 4: Once the values of GM and PM fall within the desired ranges, gradually reduce the value for γ until the desired shape of response (e.g., setpoint response) is obtained.

4. Illustrative Examples

The usefulness of the 2 proposed multi-loop control designs based on the MSC scheme is demonstrated using two 2x2 and one 3x3 multivariable processes reported in the open literature.

Example 1 - Wardle and Wood (WW) Column

The WW column has been used by some researchers in studying the performances of some multi-loop PID control schemes; e.g., see references [5-6, 16, 21-22]. The nominal model of the WW column [5] is given by:

$$\mathcal{G}(s) = \begin{bmatrix} \frac{0.126e^{-6s}}{60s + 1} & \frac{-0.101e^{-12s}}{(48s + 1)(45s + 1)} \\ \frac{0.094e^{-8s}}{38s + 1} & \frac{-0.12e^{-8s}}{35s + 1} \end{bmatrix} \quad (51)$$

The Bristol's relative gain array (RGA) based on model (51) is as follows:

$$RGA = \begin{bmatrix} 2.687 & -1.687 \\ -1.687 & 2.687 \end{bmatrix}$$

Based on the RGA values, the direct pairing should be used. Notice that the RGA diagonal elements are quite big (above 1) indicating significant interactions in the process.

Jung et al. [22] compared the performance of the multi-loop PID control designed based on their one-parameter method with the multi-loop control schemes designed based the BLT (Biggest Log-Modulus) and SAT (sequential autotuning) methods. They reported that a significant closed-loop performance improvement was achieved based on the one-parameter method over the BLT and SAT methods. In the present study we shall compare the performance of the multi-loop PID control based on the MSC scheme with that of the Jung et al. [22] and the centralized Model Predictive Control (MPC).

Decentralized MSC strategy

Here, we demonstrate the applicability and effectiveness of the general procedure given in Sub-Section 3.1 to the multi-loop PID control design. It is assumed that the time-delays are approximated using the first-order Padé formula. Following the plant decomposition, each of the diagonal transfer functions is decomposed into a sum of 2 basic modes given as follows

$$\begin{bmatrix} g_{11}(s) \\ g_{22}(s) \end{bmatrix} = \begin{bmatrix} \frac{0.139}{60s+1} - \frac{0.013}{3s+1} \\ \frac{-0.151}{35s+1} + \frac{0.031}{4s+1} \end{bmatrix} \quad (52)$$

Next, we choose the multi-scale predictors as

$$\begin{bmatrix} W_{1,1}(s) \\ W_{2,1}(s) \end{bmatrix} = \begin{bmatrix} -0.013/(3s+1) \\ 0.031/(4s+1) \end{bmatrix} \quad (53)$$

Following the procedure in Section 3.1, the finalized tuning values for the first MSC controller are given by

$$\begin{bmatrix} K_{1,0} \\ K_{1,1} \end{bmatrix} = \begin{bmatrix} -0.043(22s+1)/s \\ -47.2 \end{bmatrix} \quad (54)$$

and for the second MSC controller

$$\begin{bmatrix} K_{2,0} \\ K_{2,1} \end{bmatrix} = \begin{bmatrix} -0.049(16s+1)/s \\ 132 \end{bmatrix} \quad (55)$$

Note that, the retuning factors used for the inner-layers of control-loops 1 and 2 are $RF_{11} = 7.2$ and $RF_{22} = 1$, i.e., there is no need to retune the loop 2. In this example, only the inner sub-controller for the control-loop 1 is detuned to avoid sharp ‘‘spike’’ in Y_2 following a setpoint change in Y_1 .

From equation (54), the overall MSC controller for the control-loop 1 is given by

$$K_{msc,1}(s) = -0.043 \left(\frac{22s+1}{s} \right) \left(\frac{-29.2(3s+1)}{1.86s+1} \right) \quad (56)$$

Equation (56) can easily be rearranged in the form of a PID controller augmented with a first order filter

$$K_{msc,1}(s) = 31.4 \left(1 + \frac{1}{25s} + 2.64s \right) \left(\frac{1}{1.86s+1} \right) \quad (57)$$

We can obtain the second PID controller for the control-loop 2 from equation (55) using the same procedure as above. The equivalent PID controllers corresponding to the multi-scale controllers (54) and (55) are displayed in Table 1.

Decoupling MSC strategy

Next, we demonstrate the applicability and usefulness of the alternative MSC-PID tuning algorithm given in Sub-Section 3.2 to the multi-loop PID control design. In this case, we propose the use of 2 ideal decoupling controllers given as:

$$D_{12}(s) = K_d \left(\frac{6.1s+0.1}{272.2s^2+11.7s+0.13} \right) e^{-6s} \quad (58)$$

$$D_{21}(s) = K_d \left(\frac{3.3s+0.09}{4.6s+0.12} \right) \quad (59)$$

Notice that, we have introduced the moderating factor K_d in equations (58) and (59) in order to reduce the sensitivity of overall control performance when subject to modeling errors. The performance of an ideal decoupler is often too sensitive to modeling error. For this example, we choose $K_d = 0.8$. The PID controller tunings based on the algorithm given in the Sub-Section 3.2 is shown in Table 2. Additionally, for the purpose of comparison we also show the PID controller tunings used in the Jung et al. [22] in Table 3.

MPC strategy

For the linear MPC design, the following settings are used: the time interval $\Delta T = 2$, prediction horizon $N_p = 25$, control horizon $N_u = 2$, input rate weightage $R_f = [0.1 \ 0.1]$ and output weightage $Q_y = [45 \ 50]$. The MPC controller is designed via the Matlab MPC Toolbox.

The performances of the 4 different control strategies are compared based on sequential step changes in Y_1 and Y_2 at times $t = 5$ units and $t = 100$ units, respectively. Furthermore, the performance robustness in the presence of modeling errors in the model given by (51) are compared under 4 perturbed conditions defined as follows:

- i. Perturbed condition A: 30% time delay errors and 10% gain errors in all transfer function elements.
- ii. Perturbed condition B: 30% time delay errors and -10% gain errors in all transfer functions.
- iii. Perturbed condition C: 30% time delay errors, 10% gain errors in the diagonal transfer functions and -10% gain errors in the off-diagonal transfer functions.
- iv. Perturbed conditions D: 30% time delay errors, -10% gain errors in the diagonal transfer functions and 10% gain errors in the off-diagonal transfer functions.

The performance as measured in terms of the total Integral Absolute Error (IAE) for the 4 different control strategies under various conditions are shown in Table 4. Notice that from Table 4, under the nominal condition the difference in performances between the MPC and decoupling MSC is only about 5%. Meanwhile, the decentralized MSC provides nearly 150% performance improvement over the decentralized PID scheme of Jung et al. [22]. For all perturbed conditions based on the modeling errors of 30% in time delays and $\pm 10\%$ in process gains, the MPC is unstable while all other control strategies remain stable. On average, the decoupling MSC provides about 100% and 270% performance improvement over the decentralized MSC and Jung et al. schemes, respectively. Thus, we can conclude that by adding the decoupling controllers (58) and (59), the control performance can be increased significantly over the purely decentralized control structure. Also, we can conclude that the MSC-based PID control design produces improved performance over the Jung et al. PID control design for this multi-loop process.

Figs. 3 and 4 show the closed-loop responses at the nominal condition for the outputs and inputs respectively. As shown in Fig. 3 it is obvious that the decentralized MSC produces better performance than the Jung et al. decentralized PID control scheme; meanwhile, the MPC and decoupling MSC responses are quite closed at the nominal condition.

Example 2 - Wood and Berry (WB) Column

The Wood and Berry (WB) column [23] is perhaps the most widely cited two-input two-output example in the multi-loop control study; e.g., see [4, 6, 11, 23-24]. The model for WB distillation column is given by

$$\mathbf{G}(s) = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \quad (60)$$

The RGA for the WB column is given by

$$RGA = \begin{bmatrix} 2.009 & -1.009 \\ -1.009 & 2.009 \end{bmatrix}$$

which suggest that direct controller pairings should be used. The diagonal RGA values of the WB column is quite similar to that of the previous WW column – indicates that the process interaction is quite serious.

The general procedure in Section 3.1 is used to devise the multi-loop (decentralized) MSC controllers where the resulting equivalent PID controllers are shown in Table 1. Meanwhile, the algorithm in Section 3.2 is used to directly synthesize the PID controllers for the decoupling MSC strategy (the result is shown in Table 2). For the decoupling MSC strategy, the following 2 ideal decouplers are used

$$D_{12}(s) = K_d \left(\frac{315.6s + 18.9}{268.8s + 12.8} \right) e^{-2s} \quad (61)$$

$$D_{21}(s) = K_d \left(\frac{95.0s + 6.6}{211.5s + 19.4} \right) e^{-4s} \quad (62)$$

where the moderating factor is chosen to be $K_d = 0.8$. For comparison we also use the multi-loop PID control of Vu and Lee [23] where the controller tunings are shown in Table 3. Vu and Lee [23] reported that their multi-loop PID control design based on an independent method resulted in a better performance than the methods proposed by Lee et al. [13], Loh et al. [21] and Ho et al. [25]. Additionally, we also design a linear MPC controller for the WB column for comparison with the proposed decoupling MSC scheme. For the MPC design, the following settings are used: $\Delta T = 1$, $N_p = 30$, $N_u = 2$, $R_I = [0.15 \ 0.15]$ and $Q_y = [6.7 \ 6.7]$.

The performances of the 4 different control strategies are compared on the basis of sequential step changes in Y_1 and Y_2 at times $t = 5$ units and $t = 100$ units respectively. The performance robustness is compared under the 4 perturbed conditions similar to those defined in the WW column example; the only difference is that for this WB example, the perturbed conditions are based on 20% time delay errors and $\pm 10\%$ gain errors.

Table 5 shows the total IAE values for the different control strategies. It is worth highlighting that on average the decoupling MSC performs markedly better than the MPC while the decentralized MSC-PID performs better than the multi-loop PID control designed based on the method of Vu and Lee [23].

Fig. 6 shows the closed-loop responses at the nominal condition for the different control strategy. Again, notice that the responses for the decoupling MSC and MPC schemes are quite closed. The input movements at the nominal condition are shown in Fig. 7; the input movements for the Vu and Lee multi-loop PID control is not shown in the figure because of the huge spikes (10^{10} in magnitude) in the controlled outputs. Just like in the previous example, overall the decoupling MSC demonstrates better performance robustness than the centralized MPC strategy.

Example 3 - Ogunnaike and Ray (OR) column

The OR column [26] has been used by several researchers to test the performances of various multi-loop/multivariable control schemes, e.g., see [4, 16, 24]. The model for the 3x3 OR column is given as follows

$$G(s) = \begin{bmatrix} \frac{0.66e^{-2.6s}}{6.7s+1} & \frac{-0.61e^{-3.5s}}{8.64s+1} & \frac{-0.0049e^{-s}}{9.06s+1} \\ \frac{1.11e^{-6.5s}}{3.25s+1} & \frac{-2.36e^{-3s}}{5s+1} & \frac{-0.01e^{-1.2s}}{7.09s+1} \\ \frac{-34.68e^{-9.2s}}{8.15s+1} & \frac{46.2e^{-9.4s}}{10.9s+1} & \frac{0.87(11.61s+1)e^{-s}}{(3.89s+1)(18.8s+1)} \end{bmatrix} \quad (63)$$

The RGA for the OR column is

$$RGA = \begin{bmatrix} 2.008 & -0.722 & -0.286 \\ -0.646 & 1.825 & -0.179 \\ -0.362 & -0.103 & 1.465 \end{bmatrix}$$

Again, the RGA diagonal elements are quite large above 1, which indicate strong process interaction in the OR column. For the first and second control-loops, notice that g_{11} and g_{22}

are both decomposable into a sum of only 2 basic modes after the approximation of the delays using the first-order Padé formula. Meanwhile with respect to the third control-loop, the transfer function g_{33} is decomposed into 3 basic modes. Thus, for this third control-loop we may choose to use the 3-layer MSC scheme instead of only 2-layer MSC scheme. In the present work, for control-loop 3 we use the 3-layer MSC scheme as the transfer function g_{33} is quite complex in terms of its dynamic behaviors (with slow zero) compared to the transfer functions g_{11} and g_{22} .

The decentralized MSC control scheme is designed based on the general procedure given in Section 3.1 (see Table 1 for equivalent PID controller tunings). Additionally, we also develop a partial decoupling control strategy where the required PID controllers are tuned using the MSC-PID algorithm given in Sub-Section 3.2. The resulting PID controller tunings for this partial decoupling MSC strategy are given in Table 2. The 3 ideal decoupling controllers are given by

$$D_{12} = K_d \left(\frac{4.09s + 0.61}{5.70s + 0.66} \right) e^{-0.9s} \quad (64)$$

$$D_{21} = K_d \left(\frac{5.55s + 1.11}{7.67s + 2.36} \right) e^{-3.5s} \quad (65)$$

$$D_{31} = K_d \left(\frac{2536s^2 + 786.9s + 34.68}{82.32s^2 + 17.19s + 0.87} \right) e^{-8.28s} \quad (66)$$

where the moderating factor $K_d = 0.8$.

For this example, we also use the multi-loop PID control of Vu and Lee [24] and the centralized MPC scheme for comparison purposes. The PID controller tunings for the Vu and Lee scheme are shown in Table 3. Note that, for this example Vu and Lee [24] had shown that their multi-loop PID control performed better than the multi-loop PID control designed via the one-parameter method of Jung et al. [22] and the BLT [5]. The centralized MPC

controller is designed with the following settings: $\Delta T = 1$, $N_p = 24$, $N_u = 2$, $R_I = [0.1 \ 0.1 \ 0.1]$ and $Q_y = [100 \ 44 \ 10]$.

The nominal performance and performance robustness are evaluated on the basis of sequential step changes of magnitudes 1, 1, and 10 units in Y_1 , Y_2 and Y_3 respectively (at times $t = 5$ units, $t = 100$ units and $t = 200$ units respectively). The perturbed conditions are as defined in the previous examples based the modeling errors of 20% in time delays and $\pm 10\%$ in process gains. Table 6 illustrates the performance comparison measured in term of the total IAE values for the 4 different control strategies. Interestingly, on average the decentralized MSC scheme not only outperforms the decentralized PID control based on Vu and Lee [24] but also produces almost the same performance as the fully multivariable MPC controller. For this example, the decoupling MSC exhibits the best performance (i.e., lowest IAE) followed by the decentralized MSC, MPC and Vu and Lee schemes. The fact that the decoupling MSC exhibits better performance than the decentralized MSC implies that the process interaction has a significant influence on the control performance.

Fig. 8 demonstrates the closed-loop responses at the nominal condition while Fig. 9 shows the corresponding input movements. As displayed in Fig. 8, it is clear that the MSC-based PID control design provides large improvement over other schemes for the control-loop 3, i.e., Y_3 response. Fig. 9 shows that all control strategies do not encounter the problem of derivative or proportional kick (i.e. no huge spike in input movement).

5. Conclusions

We have presented a new approach to designing the multi-loop PID control based on the principle of the multi-scale control scheme of Nandong and Zang [19-20]. Two MSC-based procedures for the multi-loop PID control design have been proposed: (a) a general MSC procedure, and (b) an algorithm using MSC-PID tuning formulas. In the general procedure

(Section 3.1), the minimum IAE tuning criterion is applied sequentially to the outer and inner modes. Note that, different tuning rules can also be used in this general procedure, e.g., IMC, and Ziegler-Nichols. Moreover, for each mode different tuning rule can also be applied – different tuning rules can be mixed. In Sub-Section 3.2, a simple algorithm for the multi-loop PID control design has been proposed, which makes use of the MSC-PID tuning formulas derived in Sub-Section 2.3. Interestingly, the decoupling multi-loop MSC scheme has shown superior performance robustness over the centralized MPC strategy while the decentralized MSC scheme has demonstrated better performance robustness over the two well-known decentralized PID control schemes. It is worth highlighting that, for the 3x3 Ogunnaike and Ray column the decentralized MSC scheme outperformed the MPC scheme. Future work will focus on the extension of the proposed MSC-based design procedures to more complex (e.g., higher order transfer functions) and nonlinear systems.

References

- [1] S. Skogestad, M. Morari, Robust performance of decentralized control systems by independent designs, *Automatica* 25 (1989) 119-125.
- [2] M. Hovd, S. Skogestad, Improved independent design of robust decentralized controllers, *J. Process Contr.* 3 (1993) 43-51.
- [3] M. Hovd, M. Skogestad, Sequential design of decentralized controllers, *Automatica*, 30 (1993) 1601-1607.
- [4] J. Lee, T.F. Edgar, Multiloop PI/PID control system improvement via adjusting the dominant pole or the peak amplitude ratio, *Chem. Eng. Sci.* 61 (2006) 1658-1666.
- [5] W.L. Luyben, Simple method for tuning SISO controllers in multivariable systems, *Ind. Eng. Chem. Res.* 25 (1986) 654-660.
- [6] Y. Zhang, Q.G. Wang, K.J. Astrom, Dominant pole placement for multi-loop control systems, *Automatica* 38 (2002) 1213-1220.

- [7] C.G. Economou, M. Morari, Internal model control. 6. Multiloop design, *Ind. Eng. Chem. Process Des. Dev.* 25 (1986) 411-419.
- [8] J. Lee, J.Y. Choi, Design of multiloop PI controller, *J. KICChE* 31 (1993) 272-278.
- [9] T.N.L. Vu, J. Lee, M. Lee, Design of multi-loop PID controllers based on the generalized IMC-PID method with Mp criterion, *Int. J. Control Autom.* 5 (2007) 212-217.
- [10] D. Chen, D.E. Seborg, Design of decentralized PI control system based on Nyquist stability analysis, *J. Process Contr.* 13 (2003) 27-39.
- [11] J. Lee, W. Cho, T.F. Edgar, Multiloop PI controller tuning for interacting multivariable processes, *Computers Chem. Engng.* 22 (1998) 1711-1723.
- [12] T.N.L. Vu, M. Lee, Analytical design of multi-loop PI controllers for interactive multivariable processes, *J. Chem. Eng. Japan* 43 (2010) 196-208.
- [13] M. Lee, K. Lee, C. Kim, J. Lee, Analytical design of multiloop PID controllers for desired closed-loop responses, *AIChE* 50 (2004) 1631-1635.
- [14] T.N.L. Vu, M. Lee, Multi-loop PI controller design based on the direct synthesis for interacting multi-time delay processes, *ISA Transactions* 49 (2009) 79-86.
- [15] S.J. Shiu, S.H. Hwang, Sequential design method for multivariable decoupling and multiloop PID controllers, *Ind. Eng. Chem. Res.* 37 (1998) 107-119.
- [16] H.P. Huang, J.C. Jeng, C.H. Chiang, W. Pan, A direct method for multi-loop PI/PID controller design, *J. Process Contr.* 13 (2003) 769-786.
- [17] C. Rajapandiyam, M. Chidambaram, Controller design for MIMO processes based on simple decoupled equivalent transfer functions and simplified decoupler, *Ind. Eng. Chem. Res.* 51 (2012) 12398-12410.
- [18] S. Tavakoli, I. Griffin, P.J. Fleming, Tuning of decentralized PI (PID) controllers for TITO processes, *Control Eng. Pract.* 14 (2006) 1069-1080.

- [19] J. Nandong, Z. Zang, Novel multiscale control scheme for nonminimum-phase processes, *Ind. Eng. Chem. Res.* 52 (2013) 8248-8259.
- [20] J. Nandong, Z. Zang, High-performance multi-scale control scheme for stable, integrating and unstable time-delay processes, *J. Process Contr.* 23 (2013) 1333-1343.
- [21] A.P. Loh, C.C. Hang, C.X. Quek, V.U. Vasnani, Autotuning of multiloop proportional-integral controllers using relay feedback, *Ind. Eng. Chem. Res.* 32 (1993) 1102-1107.
- [22] J. Jung, J.Y. Choi, J. Lee, One-parameter method for a multi-loop control system design, *Ind. Eng. Chem. Res.* 38 (1999) 1580-1588.
- [23] R.K. Wood, M.W. Berry, Terminal composition control of binary distillation column, *Chem. Eng. Sci.* 28 (1973) 1707-1717.
- [24] T.N.L. Vu, M. Lee, Independent design of multi-loop PI/PID controllers for interacting multivariable processes, *J. Process Contr.* 20 (2010) 922-933.
- [25] W.H. Ho, T.H. Lee, O.P. Gan, Tuning of multi-loop PID controllers based on gain and phase margin specifications, *Ind. Eng. Chem. Res.* 36 (1997) 2231-2238.
- [26] B.A. Ogunnaike, J.P. Lemaire, M. Morari, W.H. Ray, Advanced multivariable control of a pilot plant distillation column, *AIChE* 29 (1983) 632-640.
- [27] M.H. Kaspar, W.H. Ray, Chemometric methods for process monitoring and high-performance controller design, *AIChE J.*, 38 (1992) 1593-1608.
- [28] S. Lakshminarayanan, S.L. Shah, K. Nandakumar, Modeling and control of multivariable processes: dynamic PLS approach, *AIChE J.*, 43 (1997) 2307-2322.

Table

Table 1. Equivalent MSC-PID controller tunings via the general procedure in Section 3.1

	PID Controller	Filter	Stability
WW Column			
Loop 1	$G_{c1} = 31.4 \left(1 + \frac{1}{25s} + 2.64s \right)$	$G_{f1} = \frac{1}{1.86s + 1}$	$GM = 12.3dB, PM = 50.9^\circ$
Loop 2	$G_{c2} = -25.4 \left(1 + \frac{1}{20s} + 0.05s \right)$	$G_{f2} = \frac{1}{0.785s + 1}$	$GM = 8.14dB, PM = 48^\circ$
WB Column			
Loop 1	$G_{c1} = 0.848 \left(1 + \frac{1}{13.5s} + 0.48s \right)$	$G_{f1} = \frac{1}{0.235s + 1}$	$GM = 7.65dB, PM = 60.4^\circ$
Loop 2	$G_{c2} = -0.079 \left(1 + \frac{1}{3.93s} + 0.031s \right)$	$G_{f2} = \frac{2.13s + 1}{0.195s + 1}$	$GM = 9.1dB, PM = 24.3^\circ$
OR Column			
Loop 1	$G_{c1} = 0.829 \left(1 + \frac{1}{2.9s} + 0.72s \right)$	$G_{f1} = \frac{1}{0.236s + 1}$	$GM = 17.3dB, 46.6^\circ$
Loop 2	$G_{c2} = -0.309 \left(1 + \frac{1}{3.3s} + 0.15s \right)$	$G_{f2} = \frac{1.5s + 1}{0.275s + 1}$	$GM = 10.6dB, PM = 58.9^\circ$
Loop 3	$G_{c3} = 2.03 \left(1 + \frac{1}{0.96s} + 0.24s \right)$	$G_{f3} = \frac{3.89s + 1}{(1.07s + 1)(0.065s + 1)}$	$GM = 5.74dB, PM = 33.1^\circ$

Table 2. PID controller tunings via the MSC tuning formulas in Section 3.2

	Setting	Stability	PID Controller	Filter
WW				
Loop 1	$\lambda_0 = 2, \lambda_1 = 3.5$ $\gamma = 0.9$	$GM = 7.8dB$ $PM = 64.7^\circ$	$G_{c1} = 51.4 \left(1 + \frac{1}{57s} + 2.84s \right)$	$G_{f1} = \frac{1}{0.86s + 1}$
Loop 2	$\lambda_0 = 2.1, \lambda_1 = 5$ $\gamma = 0.8$	$GM = 7.6dB$ $PM = 62^\circ$	$G_{c2} = -25.8 \left(1 + \frac{1}{32s} + 3.5s \right)$	$G_{f2} = \frac{1}{0.8s + 1}$
WB				
Loop 1	$\lambda_0 = 2, \lambda_1 = 2.5$ $\gamma = 0.7$	$GM = 8.8dB$ $PM = 64.2^\circ$	$G_{c1} = 0.746 \left(1 + \frac{1}{12.2s} + 0.48s \right)$	$G_{f1} = \frac{1}{0.2s + 1}$
Loop 2	$\lambda_0 = 2, \lambda_1 = 6$ $\gamma = 0.9$	$GM = 7.9dB$ $PM = 66.1^\circ$	$G_{c2} = -0.17 \left(1 + \frac{1}{14.5s} + 1.34s \right)$	$G_{f2} = \frac{1}{0.25s + 1}$
OR				
Loop 1	$\lambda_0 = 2.1, \lambda_1 = 4$ $\gamma = 0.8$	$GM = 10.3dB$ $PM = 65.2^\circ$	$G_{c1} = 2.18 \left(1 + \frac{1}{6.66s} + 1.05s \right)$	$G_{f1} = \frac{1}{0.325s + 1}$
Loop 2	$\lambda_0 = 2.4, \lambda_1 = 5$ $\gamma = 0.8$	$GM = 10.7dB$ $PM = 66.4^\circ$	$G_{c2} = -0.41 \left(1 + \frac{1}{5.5s} + 1.09s \right)$	$G_{f2} = \frac{1}{0.3s + 1}$
Loop 3	$\lambda_0 = 3.5, \lambda_1 = 1.4$ $\lambda_2 = 1.2, \gamma = 0.12$	$GM = 8.3dB$ $PM = 49.5^\circ$	$G_{c3} = 7.91 \left(1 + \frac{1}{6.15s} + 1.43s \right)$	$G_{f3} = \frac{0.5s + 1}{1.135s + 3.224s + 1}$

Table 3. PID controller tunings based on Jung et al. [22] and Vu & Lee [24]

	PID Controller	Filter	Stability
WW			
Loop 1 ^a	$G_{c1} = 37.83 \left(1 + \frac{1}{28.09s} + 33.02s \right)$	$G_{f1} = \frac{1}{39.01s + 1}$	$GM = 12.2dB, PM = 69.2^\circ$
Loop 2 ^a	$G_{c2} = -38.86 \left(1 + \frac{1}{30.62s} + 31.36s \right)$	$G_{f2} = \frac{1}{68.11s + 1}$	$GM = 10.6dB, PM = 61.9^\circ$
WB			
Loop 1 ^b	$G_{c1} = 0.66 \left(1 + \frac{1}{10.55s} + 0.02s \right)$	NA	$GM = 9.9dB, PM = 57.5^\circ$
Loop 2 ^b	$G_{c2} = -0.11 \left(1 + \frac{1}{7.54s} + 1.04s \right)$	NA	$GM = 12.6dB, PM = 52.3^\circ$
OR			
Loop 1 ^b	$G_{c1} = 2.25 \left(1 + \frac{1}{7.12s} + 2.58s \right)$	$G_{f1} = \frac{1}{4.37s + 1}$	$GM = 13.3dB, PM = 50.5^\circ$
Loop 2 ^b	$G_{c2} = -0.49 \left(1 + \frac{1}{6.44s} + 3.37s \right)$	$G_{f2} = \frac{1}{5.65s + 1}$	$GM = 13dB, PM = 51.5^\circ$
Loop 3 ^b	$G_{c3} = 4.83 \left(1 + \frac{1}{3.11s} + 10.16 \right)$	$G_{f3} = \frac{1}{5.74s + 1}$	$GM = 3.58dB, PM = 41.2^\circ$

^aJung et al. [22] and ^bVu & Lee [24].

Table 4. Total IAE values for 4 different control strategies (WW column)

Condition	MPC	Decoupling MSC	Decentralized MSC	Vu & Lee
Nominal	9.2	10.5	16.8	18.9
Perturbed A	14.6	13.5	21.6	23.2
Perturbed B	12.6	12.9	19.8	22.0
Perturbed C	14.7	12.7	17.0	19.0
Perturbed D	18.2	20.5	31.9	35.3
Average	17.3	14.0	21.4	23.7

Table 5. Total IAE values for 4 different control strategies (WB column)

Condition	MPC	Decoupling MSC	Decentralized MSC	Jung et al.
Nominal	34	36.8	69.5	172.9
Perturbed A	Unstable	47.1	104.9	170.2
Perturbed B	Unstable	45.9	97.5	187.6
Perturbed C	Unstable	41.7	86.2	175.1
Perturbed D	Unstable	63.6	121.7	175.8
Average	-	47.0	96.0	176.3

Table 6. Total IAE values for 4 different control strategies (OR column)

Conditions	MPC	Decoupling MSC	Decentralized MSC	Vu & Lee
Nominal	137.3	144.3	121.3	412.3
Perturbed A	191.4	177	216.6	438.5
Perturbed B	171.8	164.1	167.2	480.6
Perturbed C	188.1	146.9	156.7	422.2
Perturbed D	232.9	250.3	252.9	508.1
Average	184.3	176.5	182.9	452.3

Figure Captions

Fig. 1 Realization block diagram of the 3-layer multi-scale control (MSC) scheme

Fig. 2 Single-loop equivalence of the 3-layer multi-scale control (MSC) scheme

Fig. 3 Closed-loop responses under the nominal condition for the WW column

Fig. 4 Input movements under the nominal conditions for the WW column

Fig. 5 Input movements under the perturbed condition D for the WW column

Fig. 6 Closed-loop responses under the nominal condition for the WB column

Fig. 7 Input movements under the nominal condition for the WB column

Fig. 8 Closed-loop responses under the nominal conditions for the OR column

Fig. 9 Input movements under the nominal conditions for the OR column

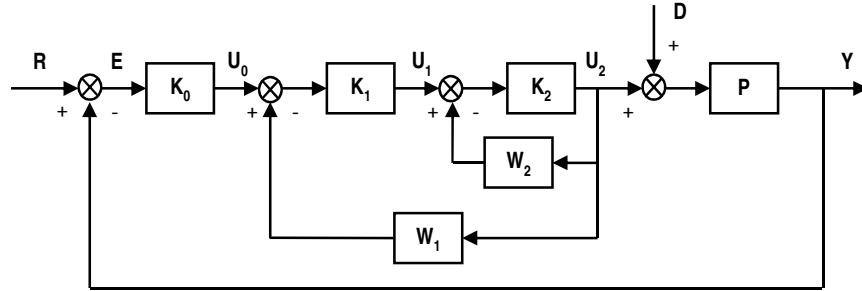


Figure 1

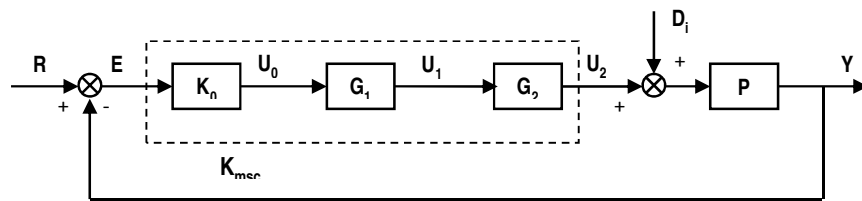


Figure 2

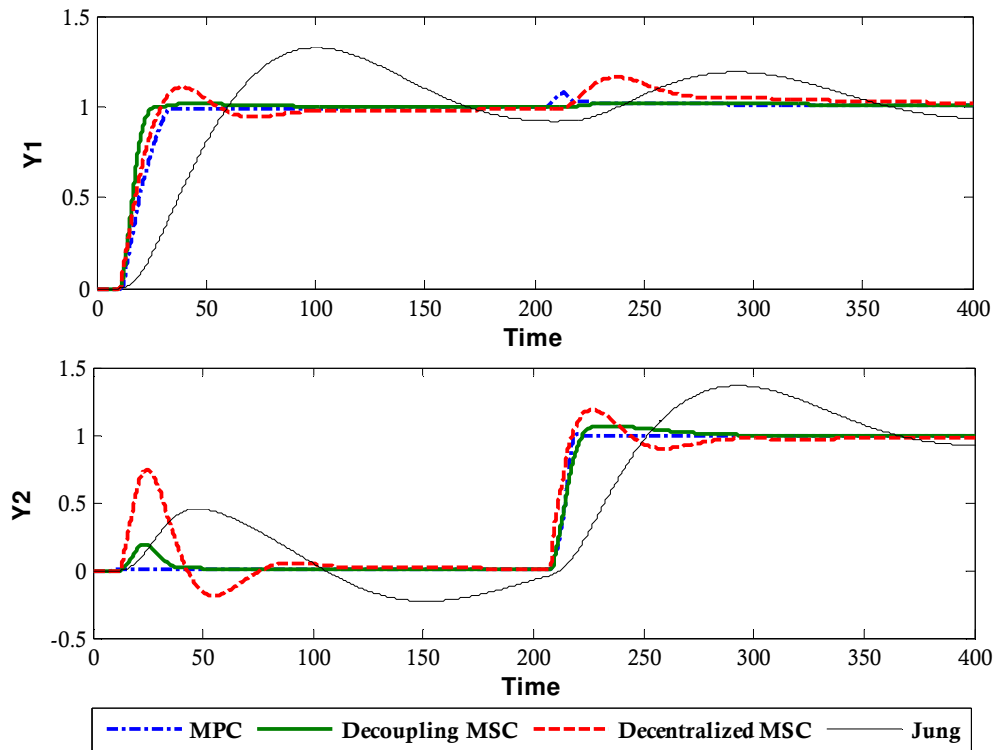


Figure 3

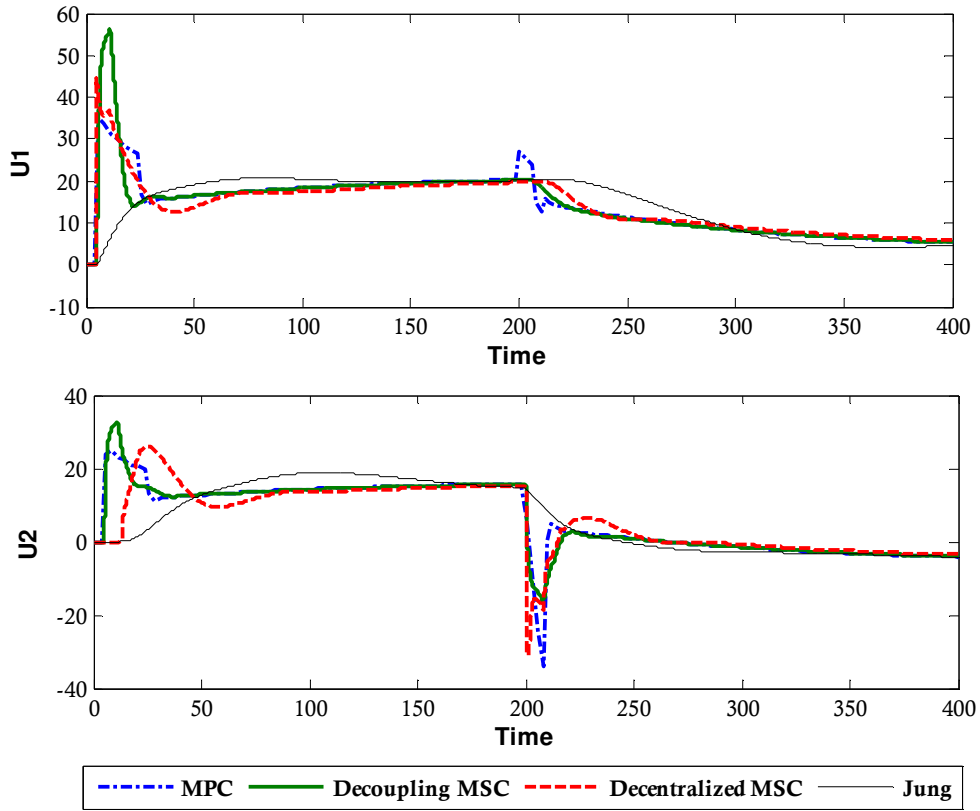


Figure 4

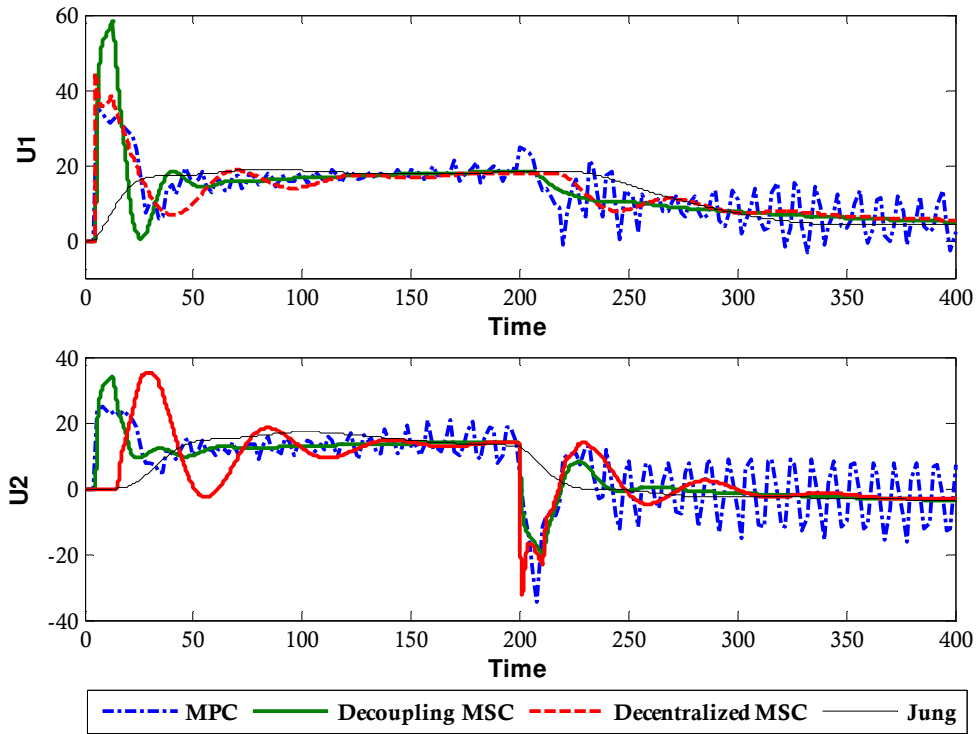


Figure 5

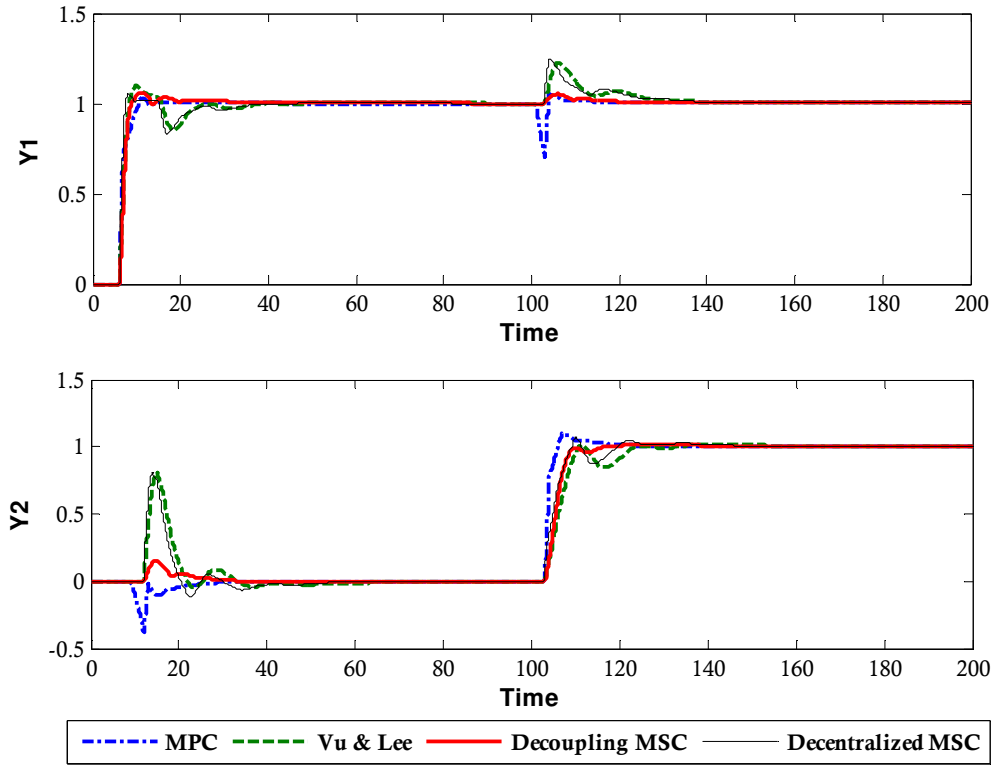


Figure 6

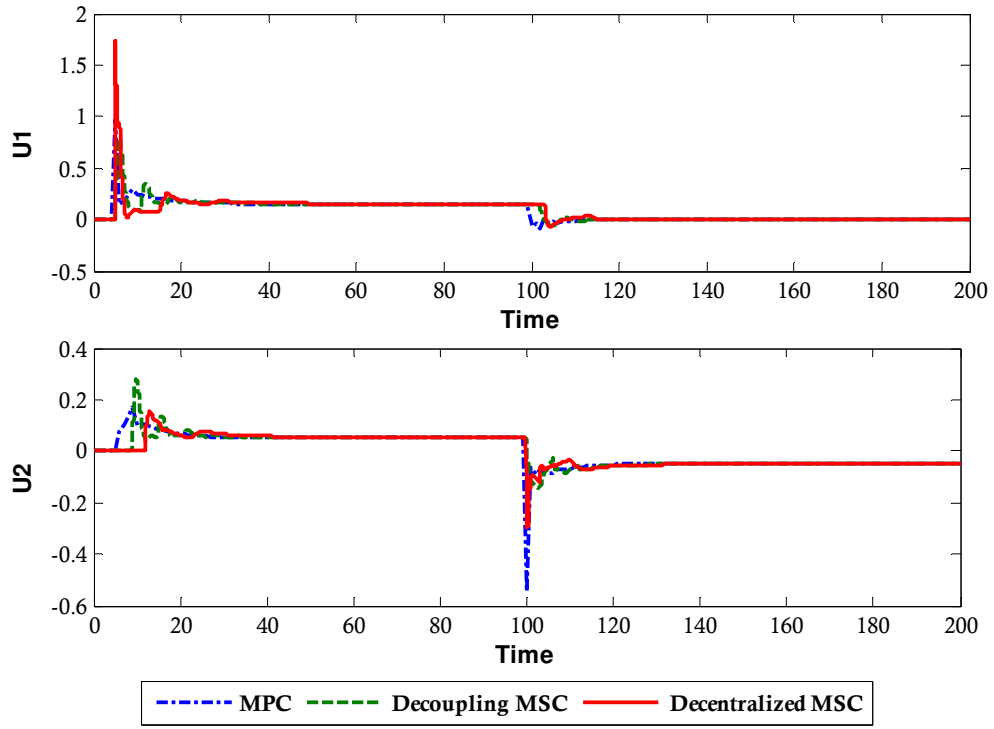


Figure 7

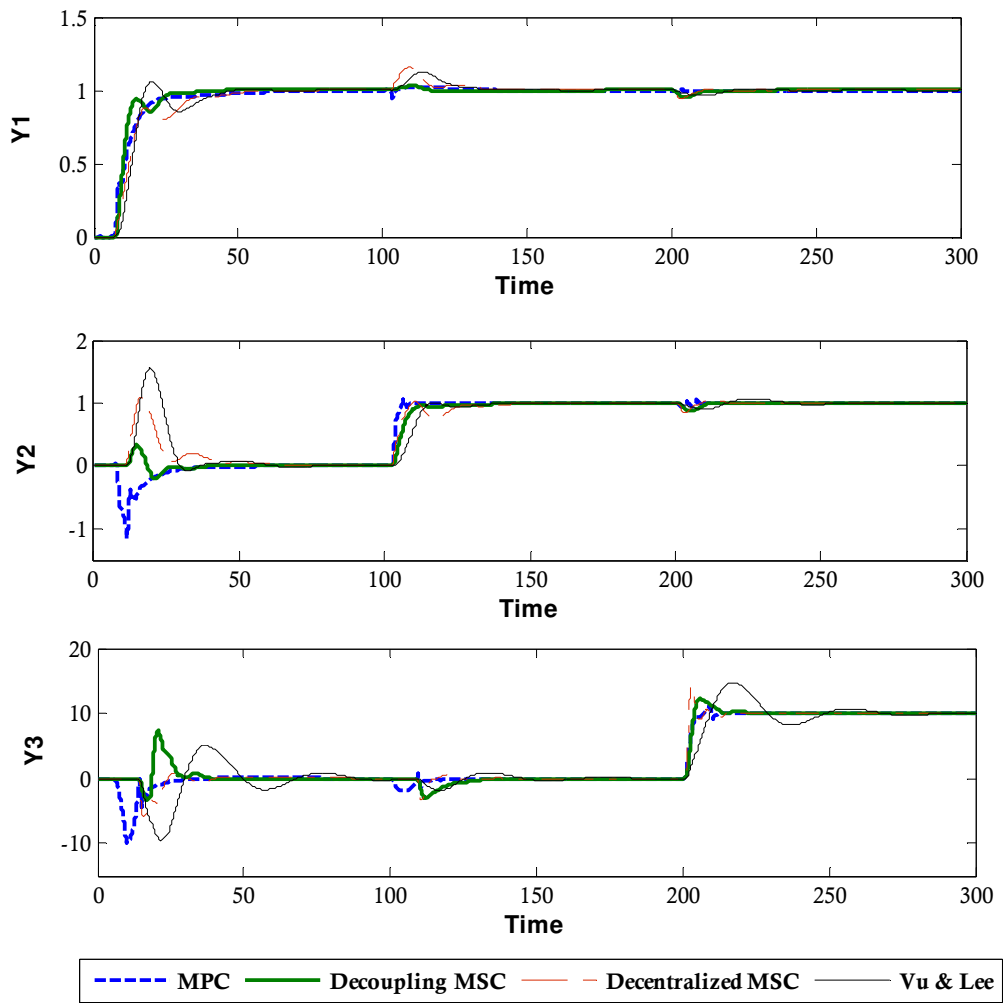


Figure 8

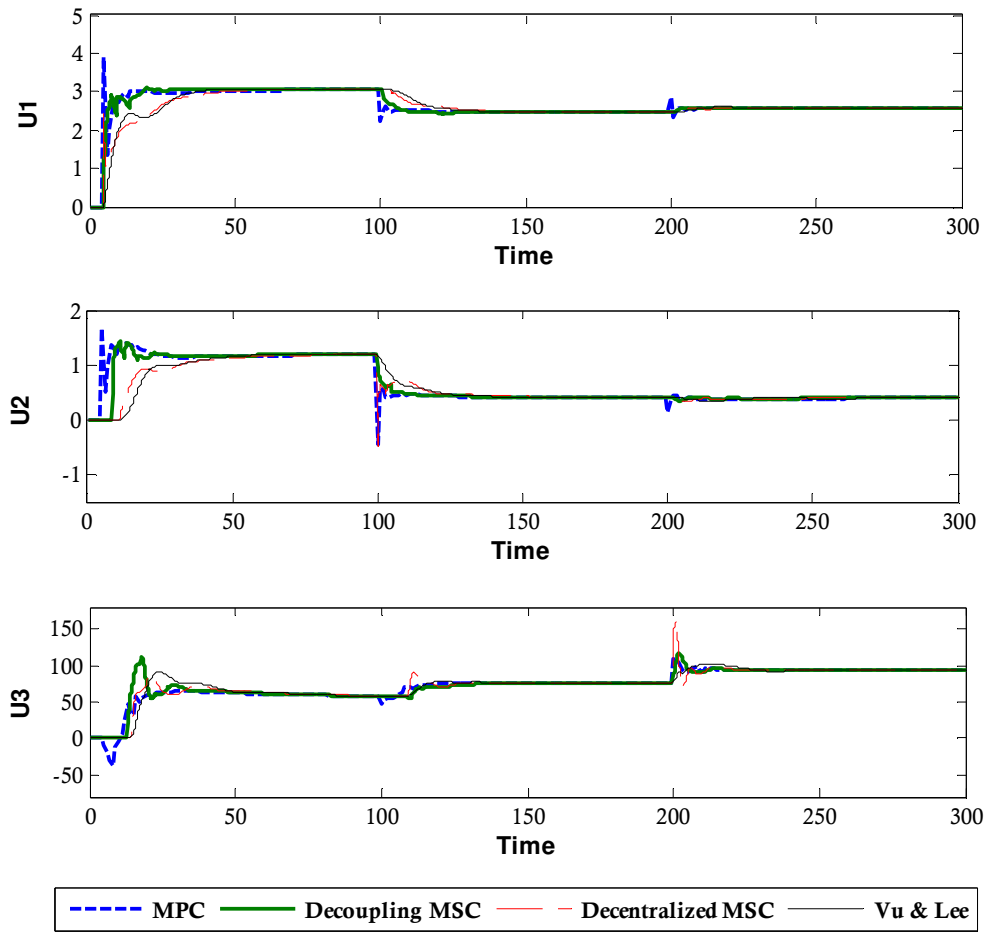


Figure 9