On Fractional-order PID Controllers

Emmanuel Edet*. Reza Katebi**

Department of Electronics and Electrical Engineering
University of Strathclyde
Glasgow, United Kingdom.

**UK (Tel: +447459184146; e-mail: emmanuel.edet@strath.ac.uk)*
*** UK (Tel: +441415484297; e-mail: m.r.katebi@ strath.ac.uk).*

Abstract: A new Fractional Order Proportional-Integral (FOPI) controller tuning method is proposed in this paper for process control systems. This is achieved by combining Biggest Log-modulus Tuning (BLT) method and Internal Model Control (IMC) method of designing conventional PID controllers to tuning FOPI controllers for multivariable processes. Unlike the conventional PID case, internal model control method is first used to design the FOPI controller and obtain preliminary values of controller parameters. This yields simple formulae for setting controller gains. Thereafter, the FOPI controller gains are adjusted using a single detuning factor (F) until a biggest log modulus of $2n$ dB is obtained where n is the number of loops. Extended simulation studies show that good compromise between performance and robustness can be achieved for multiloop process control applications with the proposed FOPI controller.

Keywords: Fractional Order Proportional-Integral Controller; Internal Model Control; Robustness

1. INTRODUCTION

This paper addresses the design and tuning of Fractional Order Proportional-Integral (FOPI) controller for multivariable process control systems. Multivariable system control is known to be more challenging to design when compared to scalar processes. This is primarily due to the presence of interactions and directionality in such systems. This limits the scope of application of most parametric model-based design algorithms to Single Input Single Output (SISO) applications (Huang, et al., 2003). Over the past decades, several methods of solving multivariable control issues have been proposed for conventional PID controllers (Loh, et al., 1993; Luyben, 1986). Niederlinski modified Ziegler-Nichol's tuning rule for MIMO processes by introducing a detuning factor to meet the stability and performance of the multi-loop control system. Luyben introduced the Biggest Log-modulus Tuning (BLT) method which is a frequency domain PID controller design method. It uses a detuning factor (F) iteratively to decouple an interactive MIMO system (Luyben, 1986). A detailed review of some multivariable PID design methods was published by Shiu and Hwang (Shiu & Hwang, 1998). One common limitation of these design methods is that all the algorithms are limited to conventional PID controllers and do not address fractional-order controllers.

The level of interaction in MIMO systems can be estimated using Relative Gain Array (RGA). This information is a useful guide in variable pairing for some form of multi-loop decoupled control. In MIMO system, the relative gain of *ij*th loop (λ_{ij}) is defined as the ratio of the gain of *ij*th loop when all other loops in the system are open to the gain of the same loop when all the other loops are closed.

RGA is generally computed as a function of frequency. It is the corresponding matrix of relative gains (G_{ij}) as given in (1).

$$\lambda_{ij} = \left[G\right]_{ij}\left[G^{-1}\right]_{ji} \tag{1}$$

A large RGA value indicates high level of interaction in a particular system. Similarly, small RGA signifies lower level of interaction between the associated variables. Physical relationship of variables are also given primary consideration during variable pairing before designing the multivariable controller. It is assumed in this work that parameters are effectively paired using similar techniques and each sub-transfer function of the model is open loop stable. Many processes in practice are found to be open loop stable. Relative success of these conventional PID control design methods for MIMO systems can be found in many publications (Jevtovića & Mataušek, 2010; Besta & Chidambaram, 2016).

Besta and Chidambaram (2016) modified Luyben's BLT method by using internal model control approach to design conventional PID controllers for two input two output systems. The authors implemented designed controllers using two configurations: centralised and decentralised (multi-loop) control structures. However, it was limited in scope to conventional PID controllers with integer order. In this paper, a multi-loop design approach is extended to controllers with fractional orders (FOPI controllers) and BLT tuning method is developed for tuning FOPI controller gains.

This paper is organised as follows. This section sets out the introduction and background problems of multivariable control. Section 2 reviews BLT method of tuning conventional PID controllers for multi-loop process control systems and Internal Model Control (IMC) design method for conventional PID controllers. In section 3, IMC method is extended to design FOPI controllers and the FOPI controller gains are analytically derived. Section 4 describes the tuning of derived IMC FOPI controller settings in order to meet a frequency domain based performance objective. In addition, robust stability analysis is addressed in this section. Section 5 presents simulation study of distillation column control. Performance of proposed controller is also addressed in section 5 while section 6 presents major conclusions of the paper.

2. BACKGROUND OF BLT TUNING METHOD

In the original BLT control design method, Ziegler-Nichols setting was used to obtain initial gains of the controller before final fine tuning (Luyben, 1986). Ultimate gains and ultimate periods of diagonal elements of the system's transfer function $G(s)$ were first determined experimentally as $k_{u,jj}$ and $\tau_{u,jj}$. Subsequently, a Ziegler-Nichols setting for each loop was calculated ($k_{c,jj}, \tau_{i,jj}$) and final fine tuning of the conventional PID controller was carried out. The BLT tuning method is summarised as follows:

Firstly, the *j-th* diagonal PI controller is given by (2) below:

$$C(s) = k_{c,j}(s)\left(1+\frac{1}{s\tau_{i,j}}\right) \quad (2)$$

where $k_{c,j}$ = controller gain; $\tau_{i,j}$ = integral time constant for *j*th PI controller. Thereafter, the frequency domain characteristic function (W) is defined where:

$$W(j\omega) = -I + \left|1+G(j\omega)C(j\omega)\right|;\ I = \text{identity matrix.}$$

The tuning factor F is initially chosen such that $2 < F < 5$. The detuning factor (F) is adjusted by defining a closed loop function L as follows:

$$L(\omega) = 20Log_{10}\left|\frac{W(\omega)}{I+W(\omega)}\right| \quad (3)$$

The factor F is further tuned to meet a specified sensitivity requirement. Final controller gains are obtained using F as follows:

$$k_{c,j} = \frac{k_{c,jj}}{F};\ \tau_{i,j} = F\times\tau_{i,jj}.$$

Immediate advantages of this method are simplicity and less computational load. One disadvantage is that it requires experimental determination of a process's critical frequency point. However, in the new method proposed in this paper, ultimate frequency point experiment is not required as the design method is not based on Ziegler-Nichol's PID tuning rule. FOPI controller is designed using internal model control method.

2.1 *Brief Review of Internal Model Control Method*

A simple method of IMC design commonly termed *SIMC* algorithm was developed for tuning conventional PID controllers by Skogestad (Skogestad & Grimholt, 2012). Here, controller parameters are derived to meet a desired closed loop set-point specification. It retains some features of the direct synthesis method. Consider a process $G(s)$ with First Order Plus Dead Time (FOPDT) characteristic:

$$G(s) = \frac{ke^{-Ls}}{\tau s+1}$$

where: L = time delay; τ = process time constant; k = system's steady state gain. SIMC method results in a conventional PI controller with gains defined as follows:

$$k_c = \frac{\tau}{k(\theta+\tau_f)};\ \tau_i = \min\{\tau, 4(\theta+\tau_f)\} \quad (4)$$

The filter's time constant τ_f is usually selected as a function of the system's time constant. This gives room for tuning using a small parameter α. i.e. $\tau_f = \alpha\tau$. α is sometimes chosen between 0.7 and 1.5.

If the model is a Second Order Plus Dead Time (SOPDT) system, PID controller type is obtained with gains defined as:

$$k_c = \frac{\tau_1}{k(\theta+\tau_f)};\tau_i = \min\{\tau_1, 4(\theta+\tau_f)\};\tau_D = \tau_2. \quad (5)$$

SOPTD model is of the form: $G(s) = \dfrac{ke^{-\theta s}}{(\tau_1 s+1)(\tau_2 s+1)}$.

These formulae are unsuitable for FOPI controllers and as such, new formulae are derived analytically in this paper for FOPI controller type. Recently, optimisation-based methods have been exploited to design optimal FOPID controllers using IMC framework but the authors developed it for a class of SISO system which exhibits fractional first order plus dead-time dynamic characteristic (Padula, et al., 2014). However, this paper addresses design and tuning of FOPI controller for multivariable process control systems.

3. PROPOSED DERIVATION OF FOPI CONTROLLER

Consider a SISO transfer function G(s):

$$G(s) = \frac{k_1 e^{-Ls}}{\tau_1 s+1} \quad (6)$$

where: L = time delay; τ_1 = process time constant; k_1 = system's steady state gain. Let the desired trajectory be denoted by D. Since set-point tracking is a primary design objective, the expected trajectory D can be expressed as shown in (7):

$$D = \frac{e^{-Ls}}{\tau_c s + 1} \tag{7}$$

It is clear that D is the desired closed loop set-point specification for the entire control system. If $C(s)$ represents the controller, it implies that:

$$D = \frac{C(s)G(s)}{1 + C(s)G(s)}. \tag{8}$$

The controller $C(s)$ is of the FOPI form given in (2) and can be re-written as shown below in (9):

$$C(s) = k_c\left(\frac{\tau_i s^{\mu} + 1}{\tau_i s^{\mu}}\right). \tag{9}$$

From (8): $C(s) = \dfrac{D}{G(s) - DG(s)}$;

$$C(s) = \frac{e^{-Ls}}{\tau_c s + 1} \div \left(\frac{k_1 e^{-Ls}}{\tau_1 s + 1} - \left(\frac{e^{-Ls}}{\tau_c s + 1}\right)\left(\frac{k_1 e^{-Ls}}{\tau_1 s + 1}\right)\right).$$

Substituting controller equation as given in (9):

$$k_c\left(\frac{\tau_i s^{\mu} + 1}{\tau_i s^{\mu}}\right) \equiv \frac{\tau_1 s + 1}{k_1(\tau_c s + 1) - k_1 + k_1 Ls} \tag{10}$$

$$k_c\left(\frac{\tau_i s^{\mu} + 1}{\tau_i s^{\mu}}\right) = \frac{\tau_1 s + 1}{k_1(\tau_c + L)s}$$

To simplify (10), put: $s = j\omega$.

Also, substitute the term: $j^{-\lambda} = \cos\frac{\lambda\pi}{2} - j\sin\frac{\lambda\pi}{2}$ in (10).

$$\Rightarrow k_c\left(1 + \frac{1}{\omega^{\mu}\tau_i}\left(\cos\frac{\mu\pi}{2} - j\sin\frac{\mu\pi}{2}\right)\right) \equiv \frac{1 + j\omega\tau_1}{j\omega k_1(\tau_c + L)} \tag{11}$$

Considering the right-hand side of (11) and rationalising it to remove complex operator from denominator:

$$k_c\left(1 + \frac{1}{\omega^{\mu}\tau_i}\left(\cos\frac{\mu\pi}{2} - j\sin\frac{\mu\pi}{2}\right)\right) = A - jB$$

where:

$$A = \frac{k_1\left(L - \omega^2\tau_1\tau_c\right)}{-\left(\omega^2 k_1^2 \tau_c^2 + 2\omega k_1^2 L\tau_c + k_1^2 L^2\right)}$$

$$B = \frac{\omega k_1 \tau_c + \omega k_1 L \tau_1}{\omega^2 k_1^2 \tau_c^2 + 2\omega k_1^2 L\tau_c + k_1^2 L^2}$$

Comparing real part yields:

$$k_c\left(1 + \frac{\cos\frac{\mu\pi}{2}}{\omega^{\mu}\tau_i}\right) = A \tag{12}$$

$$k_c = \frac{A}{1 + \omega^{\mu}\tau_i\cos\frac{\mu\pi}{2}} \tag{13}$$

Integral gain (and by extension integral time) can be obtained by comparing imaginary part:

$$-\frac{k_i\sin\frac{\mu\pi}{2}}{\omega^{\mu}} \equiv -B$$

$$k_i = \frac{B\omega^{\mu}}{\sin\frac{\mu\pi}{2}} \tag{14}$$

Integral gain is computed first before combining (13) with (14) to get proportional gain. Integral time can be obtained as:

$$\tau_i = \left|\frac{A}{k_i} - \frac{\cos\frac{\mu\pi}{2}}{\omega^{\mu}}\right|.$$

These formulae are used to calculate the initial gains of the controller parameter for each loop. Furthermore, given the FOPDT model in (6), process's relative dead-time (T) is defined as: $T = L/(L + \tau_1)$. A guide to selection of fractional order based on T is available (Monje, et al., 2010). This is given in **Table 1.**

FOPI controller settings are determined individually for each *jth*-diagonal transfer function using (15), (16) and (17).

$$k_{i,j-imc} = \frac{B\omega^{\mu}}{\sin\frac{\mu\pi}{2}} \tag{15}$$

$$\tau_{i,j-IMC} = \left|\frac{A}{k_{i,j-IMC}} - \frac{\cos\frac{\mu\pi}{2}}{\omega^{\mu}}\right| \tag{16}$$

$$k_{c,j-imc} = \frac{A}{1 + \omega^{\mu}\tau_{i,j-IMC}\cos\frac{\mu\pi}{2}} \tag{17}$$

4. PROPOSED TUNING OF IMC FOPI CONTROLLER

The derived FOPI controller gains given in (15) and (17) can be fine-tuned using BLT to meet a defined frequency domain specification. These parameters are tuned to meet set-point tracking objective as well as disturbance rejection using BLT approach. A summary of the procedure is given next.

- Consider each diagonal PI controller; determine the IMC gains for each diagonal loop using (15) and (17). Here, the IMC tuning parameter (α) is unused as it is set to one.
- Initial value of the BLT detuning factor F is initially chosen as 0.7 if relative gain array $\lambda_{ij} < 1$. If the relative gain array is greater than one, F is initially assumed to be 1.5.
- The preliminary gains of the controller are calculated as follows:

$$k_{c,j} = \frac{k_{c,j-IMC}}{F} \quad (18)$$

$$\tau_{I,j} = F \times \tau_{I,j-IMC} \quad (19)$$

- The diagonal controller matrix is calculated as $C(s)$ where $C(s) = \begin{bmatrix} C_{11} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & C_{ij} \end{bmatrix}$; $i = j = 3$ for a three input three output system.
- Determine a corresponding multivariable Nyquist diagram of the scalar function:

$$W(j\omega) = -1 + \left| I + C(j\omega)G(j\omega) \right| \quad (20)$$

- Determine the multivariable closed loop log modulus L as shown below in (21).

$$L(j\omega) = 20 \times Log_{10} \left| \frac{W(j\omega)}{1 + W(j\omega)} \right| \quad (21)$$

- The peak of L over the entire frequency range is the biggest log modulus termed $L_{\max}$.
- Finally, the factor F is varied (with 0.01 incrementally) until $L_{\max}$ is equal to $2n$ (4 dB for two-input two-output system and 6 dB for three input three output system). Here, n is the number of independent loops in the system.

Final gains are obtained using F when $L_{\max}$ is equal to $2n$. FOPI controller is realised using (18) and (19).

4.1 Stability and Performance Analysis

Robust stability analysis is required in order to know the degree of stability of the control system in the presence of plant-model mismatch and other uncertainties. Many dynamic perturbations that may occur in different parts of a system can be lumped into a single perturbation block Δ.

In this paper, inverse maximum singular value (ISV) method is considered to analyse robust stability because of suitability for MIMO system analysis. Given a process multiplicative input uncertainty $G(s)\left[I + \Delta_I(s)\right]$, if (22) holds, then the system is stable.

$$\left\| \Delta_I(j\omega) \right\| < \frac{1}{\bar{\sigma}\left\{ \left[I + C(j\omega)G(j\omega) \right]^{-1} C(j\omega)G(j\omega) \right\}} \quad (22)$$

where $\bar{\sigma}$ is the maximum singular value of the closed loop system. For the process multiplicative output uncertainty $\left[I + \Delta_O(s)\right]G(s)$, the closed loop system is said to be stable if (23) holds.

$$\left\| \Delta_O(j\omega) \right\| < \frac{1}{\bar{\sigma}\left\{ \left[I + G(j\omega)C(j\omega) \right]^{-1} G(j\omega)C(j\omega) \right\}} \quad (23)$$

$\Delta_I(s)$ and $\Delta_O(s)$ are assumed to be stable. Matlab program can be developed to plot the right hand side terms of (22) and (23) in order to reveal regions of stability for each control system. The greater the area under the curve, the greater the stability of the system. Therefore, a more robust controller will yield larger area under the curve. This index is used throughout this paper to compare controllers in terms of robust stability.

5. DISTILLATION COLUMN CONTROL EXAMPLE

A 19-plate, 12-inch diameter distillation column was experimentally set up and studied by Ogunnaike and Ray (Ogunnaike & Ray, 1983). The column (identified as *ORA*) had side-stream draw off as well as variable feed input with measurements taken for plate temperatures, overhead composition, reflux, feed flow rate and product lines. Details of the model is found in the paper. The transfer function matrix $G(s)$ for the process is given below:

$$\begin{pmatrix} \dfrac{0.66e^{-2.6s}}{6.7s+1} & \dfrac{-0.61e^{-3.5s}}{8.64s+1} & \dfrac{-0.0049e^{-s}}{9.06s+1} \\ \dfrac{1.11e^{-6.5s}}{3.25s+1} & \dfrac{-2.36e^{-3s}}{5s+1} & \dfrac{-0.012e^{-1.2s}}{7.09s+1} \\ \dfrac{-34.68e^{-9.2s}}{8.15s+1} & \dfrac{46.2e^{-9.4s}}{10.9s+1} & \dfrac{0.87(11.61s+1)e^{-s}}{(3.89s+1)(18.8s+1)} \end{pmatrix}$$

Let the output variables be represented as shown below:

y_1 = overhead composition;

y_2 = side-stream composition and y_3 = bottoms composition (19th tray temperature in Celsius). The input variables are:

u_1 = reflux flow rate (m^3/s); u_2 = side-stream product flow rate (m^3/s); u_3 = reboiler steam pressure (kPa).

The disturbances are:

d_1 = feed flow rate changes (m^3/s); d_2 = feed temperature changes (Celsius).

The relative gain array matrix is calculated first:

$$\lambda = \begin{bmatrix} -0.1904 & 1.1625 & 0.0278 \\ 1.9928 & -0.1854 & -0.8074 \\ -0.8024 & 0.0229 & 1.7796 \end{bmatrix}.$$

Thereafter, the proposed algorithm is used to obtain the controller parameters. The three diagonal transfer functions are considered independently. That is:

$$\frac{y_1}{u_1} = \frac{0.66e^{-2.6s}}{6.7s+1}; \frac{y_2}{u_2} = \frac{-2.36e^{-3s}}{5s+1}; \frac{y_3}{u_3} = \frac{0.87(11.61s+1)e^{-s}}{(3.89s+1)(18.8s+1)}.$$

The initial IMC gains are calculated as explained in the algorithm. The transfer function of the third loop is first approximated as a FOPDT model using Taylor series before calculating IMC settings. If the second order transfer function is used directly, a derivative component will be required. In this paper, only proportional and integral gains are required using the FOPI control structure. These gains are tuned accordingly as F is varied until L_{max} equals 6 dB. Resultant parameter gains are tabulated in **Table 3**. The controller is implemented in a multiloop (feedback) control configuration.

5.1 Performance and Disturbance Rejection

The proposed controller is simulated under drastic perturbations. A 20% step disturbance signal ($d1$) is introduced at t = 500 *minutes* while a 30% step disturbance signal is simultaneously introduced at t = 600 *minutes* as changes in feed temperature ($d2$). The simulation is ran for 800 *minutes* and results are shown in Fig.1 – Fig. 6. It is desirable to see how this proposed controller compares with optimal PI controller developed by the original authors who modelled the *ORA* column. Therefore, the MIMO FOPI controller is compared with an Optimum PI controller (OPI) proposed by Ogunnaike and Ray under exact conditions and disturbances. Inverse maximum singular value analysis is used to quantify robustness of the FOPI control system and results are plotted in Fig. 3 (blue line). ISV result for the OPI controller is shown by the red line in Fig.3. The area below each curve represents stability region as the lines depict stability bounds. It can be observed that the blue line covers a greater area and that shows a greater stability region provided for by the proposed FOPI controller.

Set-point tracking or steady state error reduction is judged using integral absolute error. This is tabulated in **Table 2**. In terms of performance, the proposed method compares favorably with the optimum PI method as reflected in the tabulated IAE index. However, the proposed method does not require any extensive optimisation routine. This reduces computational burden when compared with optimal methods like the *ORA* optimum PI controller. In addition, it yields a more robust control system as shown by the ISV analysis of sensitivity in Fig. 3.

6. CONCLUSIONS

The main contribution of this paper is the development of a simple design and tuning method for fractional-order PI controller for MIMO process control system. The proposed FOPI controller is first realised using internal model control method. IMC setting for each diagonal controller is further tuned using BLT approach to obtain better settings for proportional and integral gains. Analysis of system's robustness using inverse maximum singular value of sensitivity shows greater region of stability compared to the conventional PI controller.

ACKNOWLEDGEMENT

I am grateful to the management of Niger Delta Development Commission and the University of Strathclyde for supporting my research.

REFERENCES

Besta, C. and Chidambaram, M. (2016). Tuning of multivariable PI controllers by BLT method for TITO systems. *Chemical engineering communications,* 203 (4), pp. 527 – 538.

Huang, H.,Jeng, J.,Chiang, C., and Pan, W. (2003). A direct method for multi-loop PI/PID controller design systems. *Journal of process control,* 13 (8), pp. 769 – 786.

Jevtovića, B. T., and Mataušek, M. R. (2010). PID controller design of TITO system based on ideal decoupler. *Journal of process control,* 20 (7), pp. 869-876.

Loh, A., Hang, C., Quek, C., and Vasnani, V. (1993). Autotuning of multiloop proportional-integral controllers using relay feedback. *Ind. eng. chem. res.* 32 (6), pp. 1102-1107.

Luyben, W. (1986). Simple method for tuning SISO controllers in multivariable systems. *Ind. eng. chem. process des. dev.,* 26 (3), pp. 654-660.

Monje, A. M. et al. (2010). *Fractional-order systems and controls: Fundamentals and applications.* pp. 93-99. Springer-Verlag, London.

Ogunnaike, B. and Ray, W. (1983). Advanced multivariable control of a pilot plant distillation column. *AlChe journal,* 29 (4), pp. 632-640.

Padula, F., Vilanova, R., and Visioli, A. (2014). H-infinity optimization based fractional-order PID controllers design. *International journal of robust and nonlinear control*, 24 (17), pp. 3009–3026.

Skogestad, S. and Grimholt, C. (2012). The SIMC method for smooth PID controller. In: Vilanova, R. and Visioli, A., *PID control in third millenium: Lessons learned and new approaches.* pp. 147-174. Springer-Verlag, London.

Table 1. Selection of fractional order

Relative Dead Time	Recommended Order
$T<0.1$	0.7
$0.1 \le T < 0.4$	0.9
$0.4 \le T < 0.6$	1.0
$T \ge 0.6$	1.1

Table 2. Performance Comparison

Step Change	IAE y1	IAE y2	IAE y3
FOPI	38.4	31.0	33.9
Optimum PI (OPI)	12.42	53.48	12.06
Settling T.(m)-FOPI	10	20	100
Settling T.(m)-OPI	10	90	100

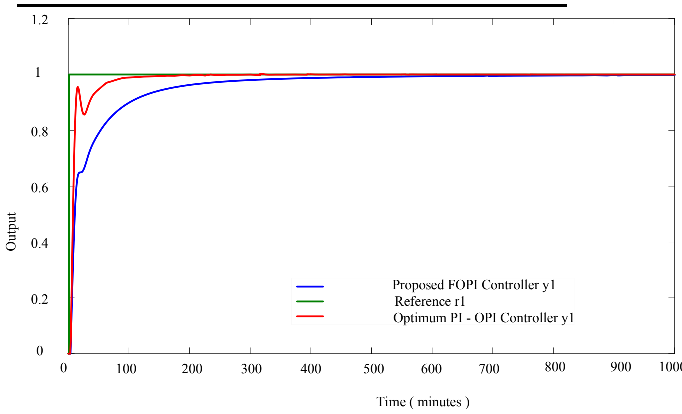

Fig.1 Top composition set-point tracking comparison with $r1$ =1.

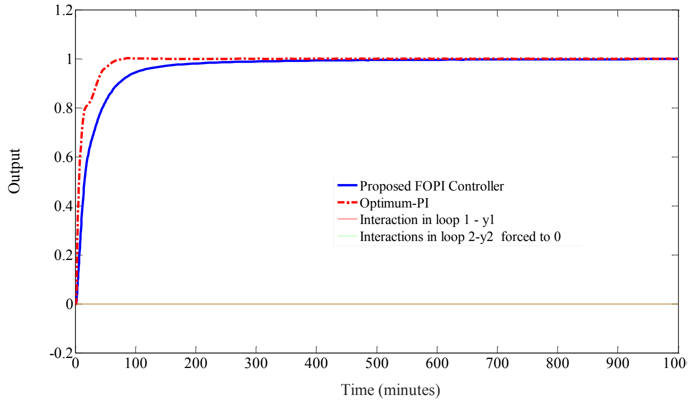

Fig.2 Bottoms composition set-point tracking comparison with $r3$ = 1.

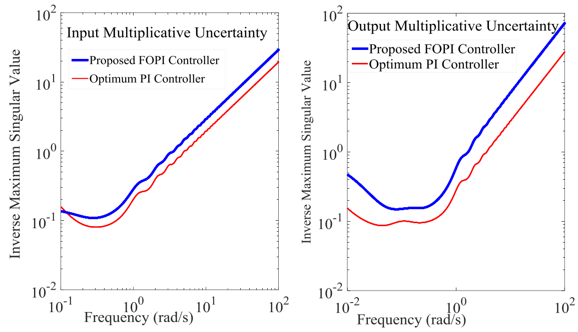

Fig.3 Stability regions for input and output uncertainties.

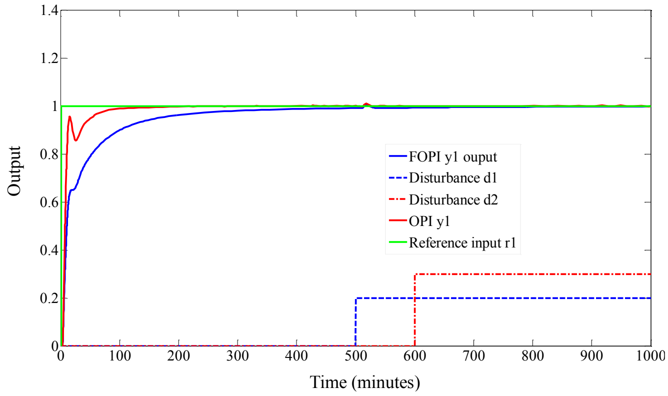

Fig.4 Disturbance rejection: Top composition.

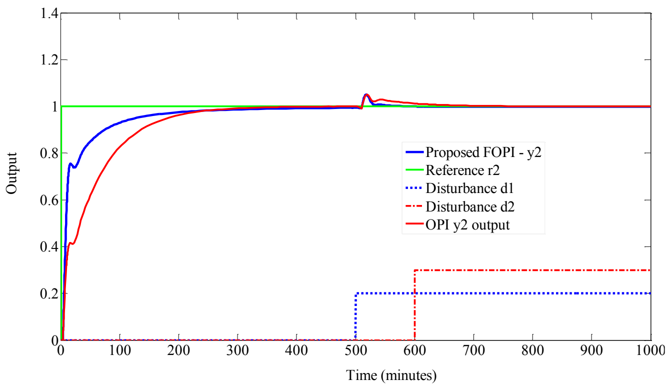

Fig.5 Sidestream composition loop: Disturbance rejection.

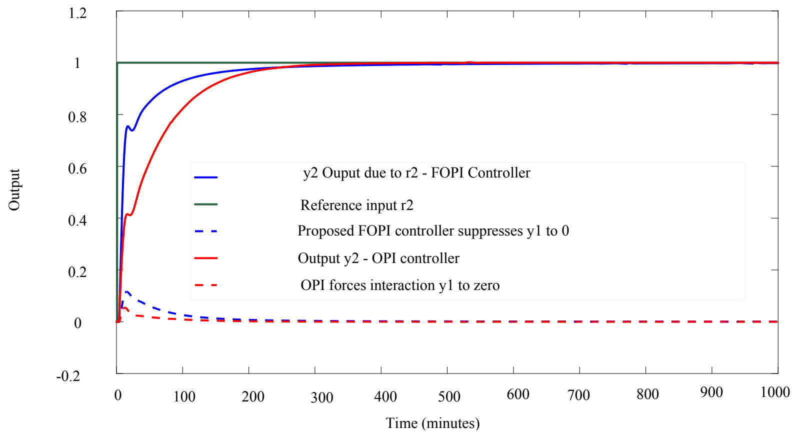

Fig.6 Sidestream composition set-point tracking comparison with $r2$ =1.

Table 3. Controller parameters

Settings	BLT-IMC	Optimum PI (OPI)
k_c	$\begin{bmatrix} 0.7881 & 0 & 0 \\ 0 & -0.1991 & 0 \\ 0 & 0 & 0.2306 \end{bmatrix}$	$\begin{bmatrix} 1.2 & 0 & 0 \\ 0 & -0.15 & 0 \\ 0 & 0 & 0.6 \end{bmatrix}$
k_i	$\begin{bmatrix} 0.1452 & 0 & 0 \\ 0 & -0.0491 & 0 \\ 0 & 0 & 0.0151 \end{bmatrix}$	$\begin{bmatrix} 0.24 & 0 & 0 \\ 0 & -0.015 & 0 \\ 0 & 0 & 0.15 \end{bmatrix}$
μ	0.9	1