Combat Process Disturbances and Interactions: New Tools and a Practical Guide to Implementing Feedforward and Decoupling Control

Henri A. (Hank) Brittain, Ph.D., Jean-François Dubé
Top Control Inc.

ABSTRACT

PID loop interactions and process disturbances are leading sources of process variability and instability in all industrial settings. The advanced methods to deal with these (decoupling and feedforward control) have existed for decades, yet are seldom used or used incorrectly. This is because these techniques rely on 'textbook' methods to obtain the needed process models. This paper will show you how to employ new tools to easily obtain the effects of interaction and disturbances on your process (the models) and how to use these to reduce or eliminate loop interactions and the effects of disturbances. The tools can be used to perform tests while the process is running normally. A highly interactive process common to industry with a continuous disturbance will be presented to illustrate the solution with before and after results shown.

INTRODUCTION

There are a multitude of papers and textbooks on the theory of how to mitigate the effects of disturbances in PID control and the interactions between PID loops. This theory will be discussed briefly and then the differences with this paper will be addressed. Results will follow. Classic feedforward theory requires one to obtain a model of both the process and the effects of the disturbance on the process. These models are in the form representing a first-order plus dead time (FOPDT) process. Though higher order models can be used, they are usually not necessary. For the process, the model consists of the process gain (G_p) the process dead time (t_{dp}), and the process time constant (τ_p). The process gain represents how much change the controller output (CO) imparts on the process variable (PV). The units of this gain, %PV/%CO, are self explanatory and are the result of dividing the steady state value of the change in PV by the change in CO that is driving the PV response (e.g. ΔPV/ΔCO). The deadtime is the time between when a change in the CO is made and when the PV begins to respond to this change. The time constant is when the PV reaches 63.2% of its final change due to the change in CO. One method of obtaining this model is to place the controller in manual, 'step' the CO, and allow the PV to reach steady state. There are a number of methods and software tools to obtain a model from this type of test. The result of this textbook approach is shown on the left in Figure 1 for the 'Process Test'. For feedforward control, the same discourse is true for the disturbance model, except it is the disturbance that causes the PV movement versus the PID controller output. If one were able to 'step' the disturbance with the PID controller in manual to prevent any correction on the PV, this model could also be obtained and used in feedforward control. This is also shown on the right in Figure 1 as the 'Disturbance Test'.

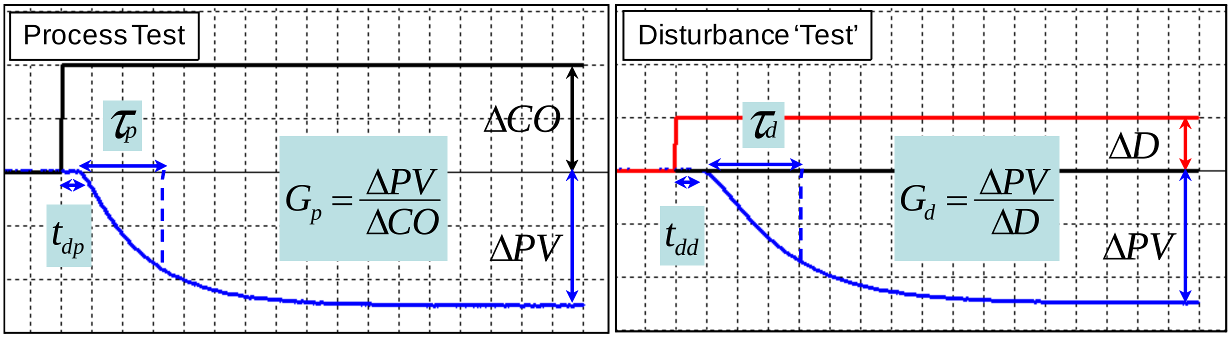


Figure 1: Typical textbook presentation of example FOPTD model for the process (left) and a disturbance (right) needed to implement feedforward control.

Once one has obtained these models, a 'compensator' can be determined as shown in Table 1, which becomes the feedforward controller. The steady state portion of the feedforward controller is the steady state feedforward gain ΔCO/ΔD and represents how much the controller must move its output to cancel the effects of the disturbance. This, multiplied by the change in disturbance, and added to the PID controller output (with the correct sign), moves the CO to the desired position when the disturbance occurs, *before* the PV is disturbed. This is the essence of feedforward control. The dynamic element of the feedforward controller are contained in the lead and lag portions as well as the delay time of the compensator. These dynamics determine when and how much of the steady state gain to apply.

	Process Model	Disturbance Model	Compensator
Gain	G_p	G_d	$-\left[{G_d}/{G_p} \right] = \dfrac{\Delta CO}{\Delta D}$
Time Constant	τ_p	τ_d	$lead\,\tau_p \quad lag\,\tau_d$
Dead Time	t_{dp}	t_{dd}	$t_{dd} - t_{dp}$

Table 1: Calculation of the feedforward compensator from 1st-order models for the process and the disturbance. Deadtime compensation can only be used if the delay time of the disturbance on the process t_{dd}, exceeds that of the dead time of the controller output on the process variable, t_{dp}.

Unfortunately we cannot usually 'step' the disturbance as shown in the right hand side of Figure 1. It is also usually difficult to step the controller output and wait for the process to settle, as this may be disruptive to the process. Yet the above discussion is how textbooks present the solution to feedforward control. If one is fortunate enough to have a disturbance that lasts long enough for the process to reach a new steady state, one could simply observe where the PID CO moves to in order to cancel the disturbance and this would at least provide the steady state feedforward gain. This is rarely the case, so most are left without a means to obtain the disturbance model to implement even steady state feedforward control.

We follow a similar 'textbook' discussion for decoupling, which is really a special case of feedforward control, where the 'disturbance' originates with the CO from another PID loop. In this case, we do have the ability to 'step' this 'disturbance'. For a two by two system, we would place both controllers in manual and step them one at a time, allowing ample time for each loop to come to steady state after each CO step. For a 3x3 system, we would have to place all three loops in manual and do the same. As we did for feedforward we would obtain a compensator, which in this case is called a decoupler, as shown in Table 2.

	Process 1 Model	Process 2 Effect on Process 1	Decoupler
Gain	$G_p{}_{11}$	G_{p12}	$-\left[{G_{p12}}/{G_{p11}} \right] = \dfrac{\Delta CO_1}{\Delta CO_2}$
Time Constant	τ_{11}	τ_{12}	$lead\,\tau_{11} \quad lag\,\tau_{12}$
Dead Time	t_{d11}	t_{d12}	$t_{d12} - t_{d11}$

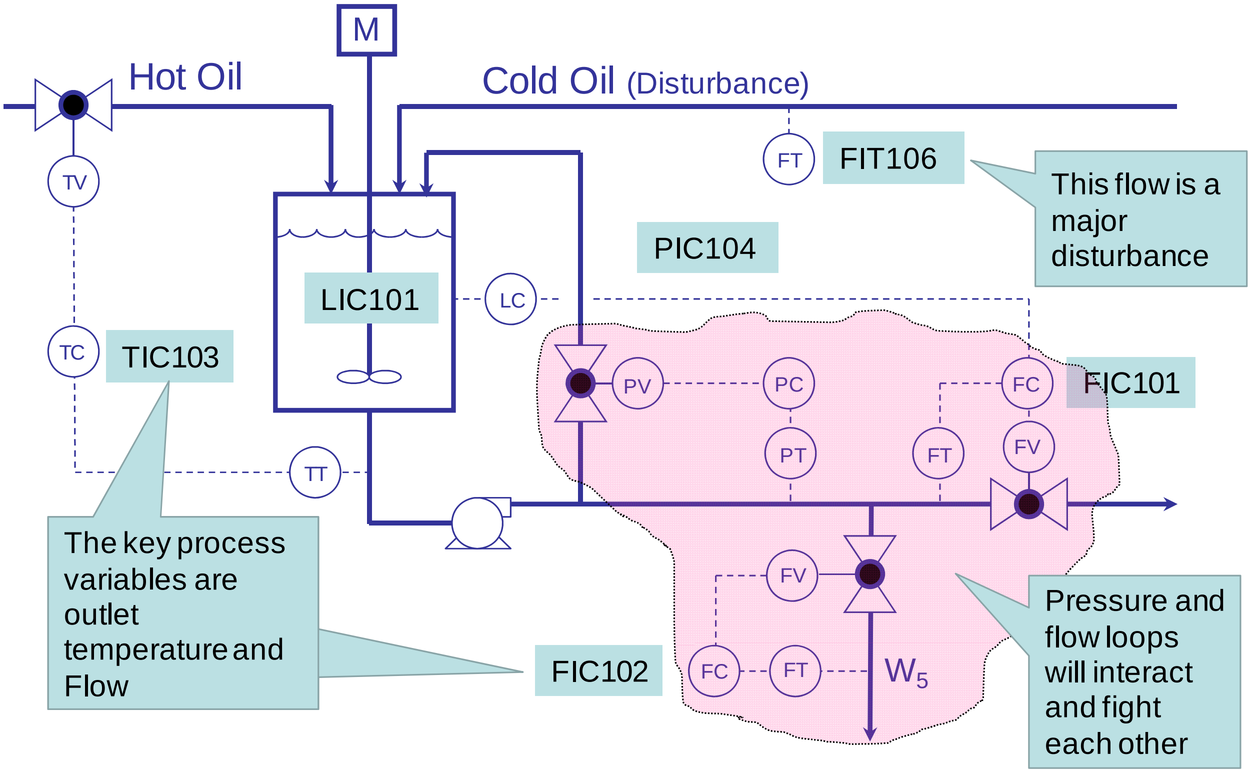
Table 2: Calculation of the decoupler shown for one way decoupling, canceling the effects of process 2 on process 1. This is the same as that for feedforward control, except the 'disturbance' in this case is process 2 (from CO_2).

The steady state gain of the decoupler in Table 2 tells us how much the CO from one controller needs to be moved to compensate for the effects from another controller's CO. As in feedforward, there can be dynamic compensation as well, using a lead and lag. In practice we would not use any dead time compensation, as this adds dead time to the effected loop and decreases the loop stability. Though we have the ability to 'step' each controller output, it may still be too disruptive to the process to do so and wait for the process to settle out.

This paper will show results from a new tool recently available that allows for feedforward and decoupling solutions to be reached much more easily than before. The tool permits one to measure the disturbance for a feedforward model while testing the key loop in automatic mode. It also allows simultaneous testing of multiple loops in automatic mode and can calculate the array of models. These models show the degree of interaction and can be used to develop either a simple tuning solution that may reduce the interaction, or if required, a more complete decoupling solution. An example process will be described with a constant disturbance and highly interactive loops followed by the progression of work leading to both feedforward and decoupling solutions using the new tool.

EXAMPLE PROCESS AND CONTROL OBJECTIVES

The process displayed in Figure 2 below consists of a tank receiving a cold fluid stream and a hot fluid stream. The goal is to control the temperature of the outlet stream by manipulating the inlet flow of hot fluid. The level is also controlled by manipulating the setpoint to a flow controller in typical cascade fashion. Another outlet flow distributes the controlled temperature fluid to a user. Finally, the pressure in the exit piping is controlled with a valve in the recirculation stream. The key process variables here are flow for the user maintained by FIC102 and outlet temperature maintained by TIC103. The inlet cold fluid flow is a major, ongoing disturbance.



This flow is a major disturbance

The key process variables are outlet temperature and Flow

Pressure and flow loops will interact and fight each other

Figure 2: Example process.

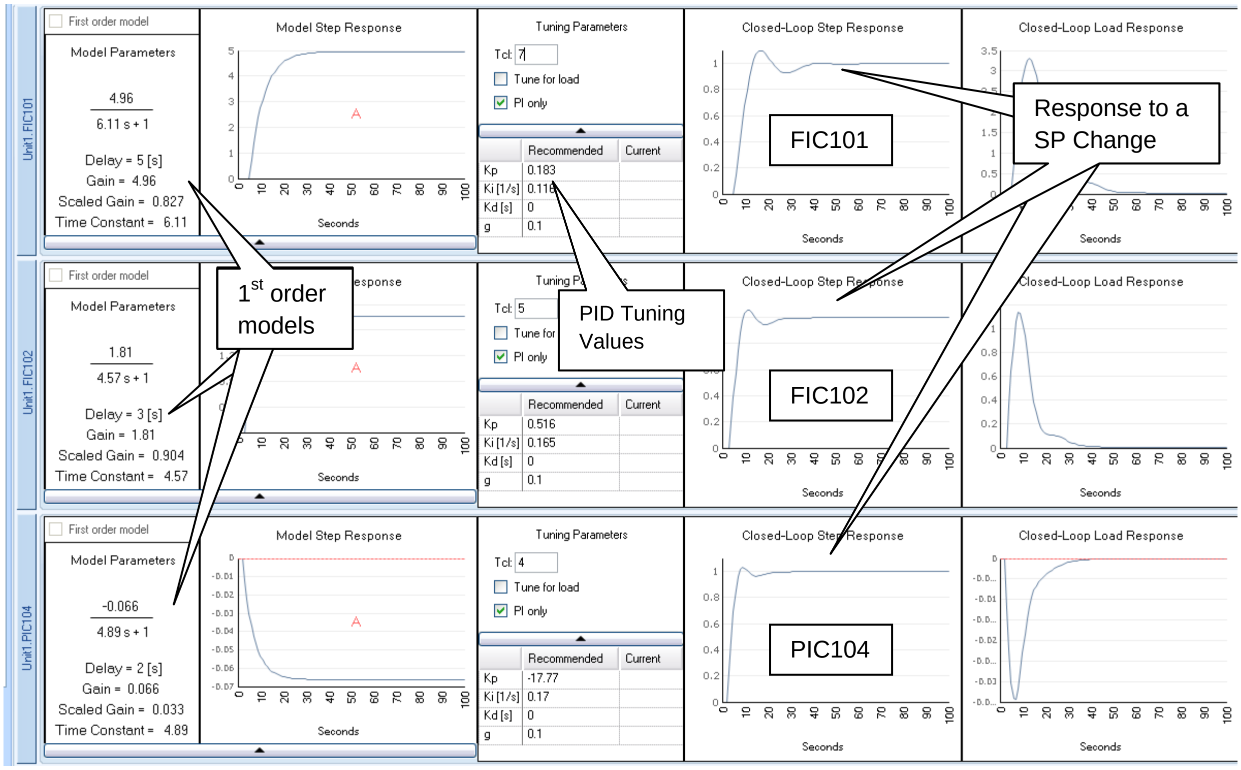
One challenge in this process will be how to handle the highly interactive flow and pressure loops (FIC101, FIC102, and PIC104) in the piping branch leaving the tank. This type of interaction in piping junctions is a common problem in many industrial settings. The cold fluid flow, measured by FIT106, is a continuous disturbance to the system and poses a challenge in controlling the temperature leaving the tank.

The procedure to follow to optimize the performance of these loops is based on principles learned after many years of tuning experience. Fast loops are always tuned first. The pressure and flow loops will be tested for interaction and we will first attempt to de-tune these loops to break the interaction. If this is not acceptable, MIMO models can be used to engineer a decoupler. Slow loops are tuned next. Tuning the cascade loop will not be addressed in this paper. Finally we will tune the temperature loop, and employ the disturbance and process models to engineer a feedforward solution. Results will be shown for each of these steps.

DECOUPLING CONTROL

PID control loops interact when changes in one loop's output disturbs another loop. This can be illustrated in the example process as follows: When LIC101 responds to a disturbance, the setpoint to FIC101 will be altered, causing a change in that flow. If, for example, that change results in an increase in flow for FIC101, this causes a drop in the pressure controlled by PIC104, and also causes a decrease in flow for FIC102. The controllers PIC104 and FIC102 will both respond. PIC104 will close its valve in an attempt to restore pressure. FIC102 will open its valve to try and restore flow. As the pressure rises due to PIC104, flow through both FIC101 and FIC102 will increase. This exacerbates the move that FIC102 made, and that flow will increase even more. This is positive interaction, and as we will see, seemingly stable tuning can approach instability for this loop under these circumstances.

Testing and Tuning the Interacting Loops as Individual Loops (SISO): The most common approach to tuning PID loops is by trial and error but this is not recommended and in the case of interacting loops, is rarely successful, very disruptive to the process, and time consuming. Preferable to trial and error is to put the loop(s) in manual, perform what is commonly known as a 'bump' or a 'step' test, and obtain a 'model'. This will usually work for single loops, but poses more challenges for interacting loops and does not consider the interaction if tuned as single loops. The most common model obtained from 'step' or any other testing is of the first order plus dead time variety. For the pressure and flow loops in the example process, one would need to put all three loops in manual simultaneously and test each loop in turn. An example of the result of this approach, is shown in Figure 3, below.



Unit1.FIC101

□ First order model

Model Parameters

$$\frac{4.96}{6.11\,s+1}$$

Delay = 5 [s]
Gain = 4.96
Scaled Gain = 0.827
Time Constant = 6.11

Tuning Parameters

Tcl: 7
□ Tune for load
☑ PI only

	Recommended	Current
Kp	0.183	
Ki [1/s]	0.118	
Kd [s]	0	
g	0.1	

Unit1.FIC102

□ First order model

Model Parameters

$$\frac{1.81}{4.57\,s+1}$$

Delay = 3 [s]
Gain = 1.81
Scaled Gain = 0.904
Time Constant = 4.57

Tuning Parameters

Tcl: 5
□ Tune for load
☑ PI only

	Recommended	Current
Kp	0.516	
Ki [1/s]	0.165	
Kd [s]	0	
g	0.1	

Unit1.PIC104

□ First order model

Model Parameters

$$\frac{-0.066}{4.89\,s+1}$$

Delay = 2 [s]
Gain = 0.066
Scaled Gain = 0.033
Time Constant = 4.89

Tuning Parameters

Tcl: 4
□ Tune for load
☑ PI only

	Recommended	Current
Kp	-17.77	
Ki [1/s]	0.17	
Kd [s]	0	
g	0.1	

Figure 3: Open-loop testing of the flow and pressure loops, with the resulting models and closed loop response tuned as if the loops did not interact. In this case, the loops are treated as single input, single output (SISO) and tuned to each have a slight overshoot upon a setpoint change.

Actual System Response with Single Loop Tuning (MIMO process loops with SISO Tuning): In reality, the system is interactive. Should one apply the tuning values obtained from SISO testing as shown in Figure 2 to the process, the results are quite different than the predicted SISO closed loop responses. These results are shown in Figure 4.

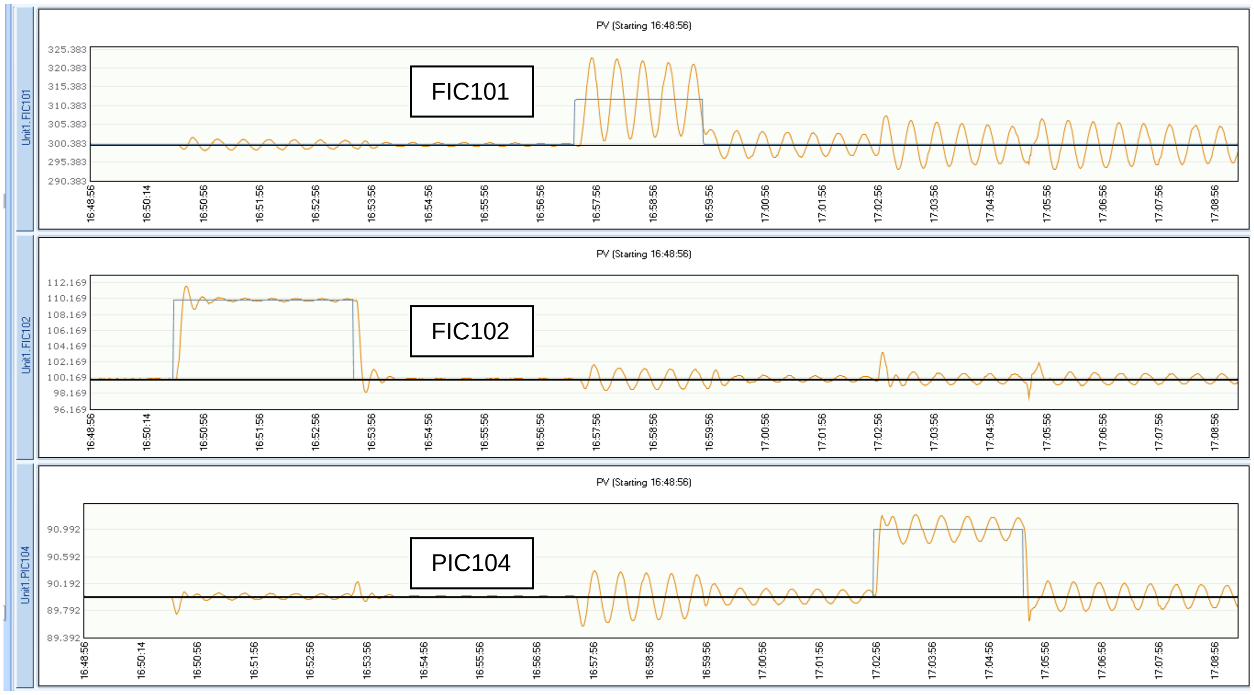
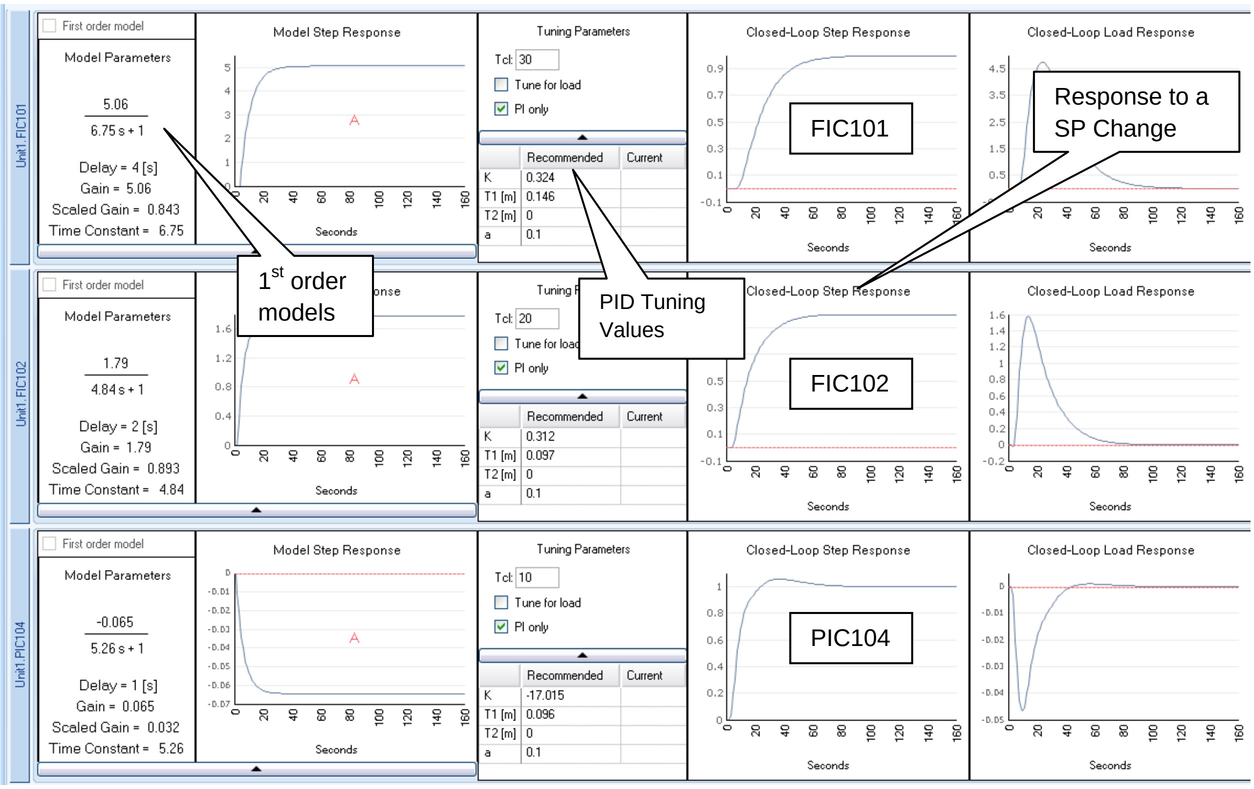


Figure 4: Actual response of the system to setpoint changes using tuning obtained from SISO testing. The tuning is nearly unstable due to the interactions between the loops.

Detuning the MIMO System to Reduce Interaction:

Detuning is a method used to make one or more loops sluggish in response. This can often reduce or break the interaction between loops. Some hierarchy in loop performance needs to be determined for this method to work. The hierarchy implemented in this study is to tune the pressure loop fastest, to maintain header pressure. This will also allow the pressure to break the interaction with the flow loops if they are tuned slower. Since FIC101 is very interactive with pressure, we tune it 3x slower. This is done using IMC tuning rules and choosing a closed loop time constant (Tcl) 3 times greater than the pressure loop PID104. We tune the other flow, FIC102, with a Tcl in between that of the pressure and flow loop, in this case, 20 seconds. In theory, we should tune all interacting loops 3x or more slower, but that is often too slow. Figure 5 shows the anticipated results of detuning the loops in this way, using MIMO models. Figure 6 shows the results on the actual process.



Unit1.FIC101

□ First order model

Model Parameters

$$\frac{5.06}{6.75\,s+1}$$

Delay = 4 [s]
Gain = 5.06
Scaled Gain = 0.843
Time Constant = 6.75

Tuning Parameters

Tcl: 30
□ Tune for load
☑ PI only

	Recommended	Current
K	0.324	
T1 [m]	0.146	
T2 [m]	0	
a	0.1	

Unit1.FIC102

□ First order model

Model Parameters

$$\frac{1.79}{4.84\,s+1}$$

Delay = 2 [s]
Gain = 1.79
Scaled Gain = 0.893
Time Constant = 4.84

Tuning Parameters

Tcl: 20
□ Tune for load
☑ PI only

	Recommended	Current
K	0.312	
T1 [m]	0.097	
T2 [m]	0	
a	0.1	

Unit1.PIC104

□ First order model

Model Parameters

$$\frac{-0.065}{5.26\,s+1}$$

Delay = 1 [s]
Gain = 0.065
Scaled Gain = 0.032
Time Constant = 5.26

Tuning Parameters

Tcl: 10
□ Tune for load
☑ PI only

	Recommended	Current
K	-17.015	
T1 [m]	0.096	
T2 [m]	0	
a	0.1	

FIC101

Response to a SP Change

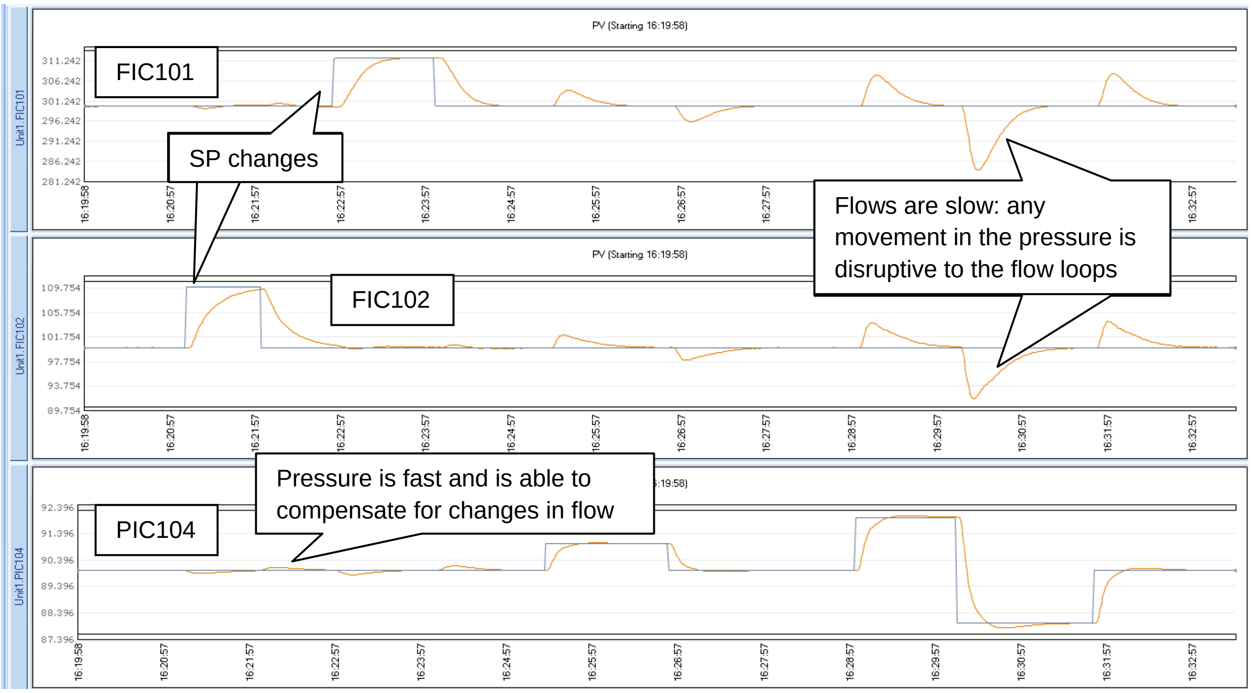
1st order models

PID Tuning Values

FIC102

PIC104

Figure 5: Open-loop testing of the flow and pressure loops, with the resulting models and closed loop response tuned with the understanding that they will interact. In this case, the loops are treated as single input, single output (SISO) but are tuned progressively slower in an attempt to reduce the interaction. Using IMC tuning rules, we use a Tcl of 10 seconds for the pressure, and 20 and 30 seconds for FIC102 and FIC101, respectively.



FIC101

SP changes

Flows are slow: any movement in the pressure is disruptive to the flow loops

FIC102

Pressure is fast and is able to compensate for changes in flow

PIC104

Figure 6: Actual response of the interacting system to setpoint changes in PIC104 using tuning obtained from SISO testing, but tuned to reduce interaction. The tuning works well for the pressure loop, but the flows are sluggish.

When the loops are tuned progressively slower, we observe that we can successfully reduce the interaction between the two flows by tuning the pressure loop 2 to 3 times faster. Essentially the pressure loop quickly recovers from any pressure changes so as to mitigate the effects of one flow on the other.

The consequence of detuning the flow FIC101 in this manner is that the level loop that is primary in the LIC101-FIC101 cascade strategy must be tuned 3 or more times slower than the flow. This may adversely affect the control of the level if tight level control is needed. If tight level control is not needed, this 'detuning' strategy may provide suitable results.

Testing and Tuning the Loops as an Interacting Loop System (MIMO): In order to assess the interaction between the loops, and obtain the models, and determine the effects of tuning without resorting to trial and error methods, we would need to obtain the 3x3 array of models showing how each loop's output affects its own process variable and that of the other loops. One way to do this is to place all three loops in manual, and 'step' the outputs one at a time, while recording the three process variables. Newer tools allow a simpler way to do this. These can test the loops simultaneously either in manual (open-loop) or automatic (closed-loop) mode as shown in Figure 7 below. The results of this closed-loop testing are shown in Figure 8. During this testing, the process was run normally and small pulse-like changes were made to the setpoint for each loop simultaneously.

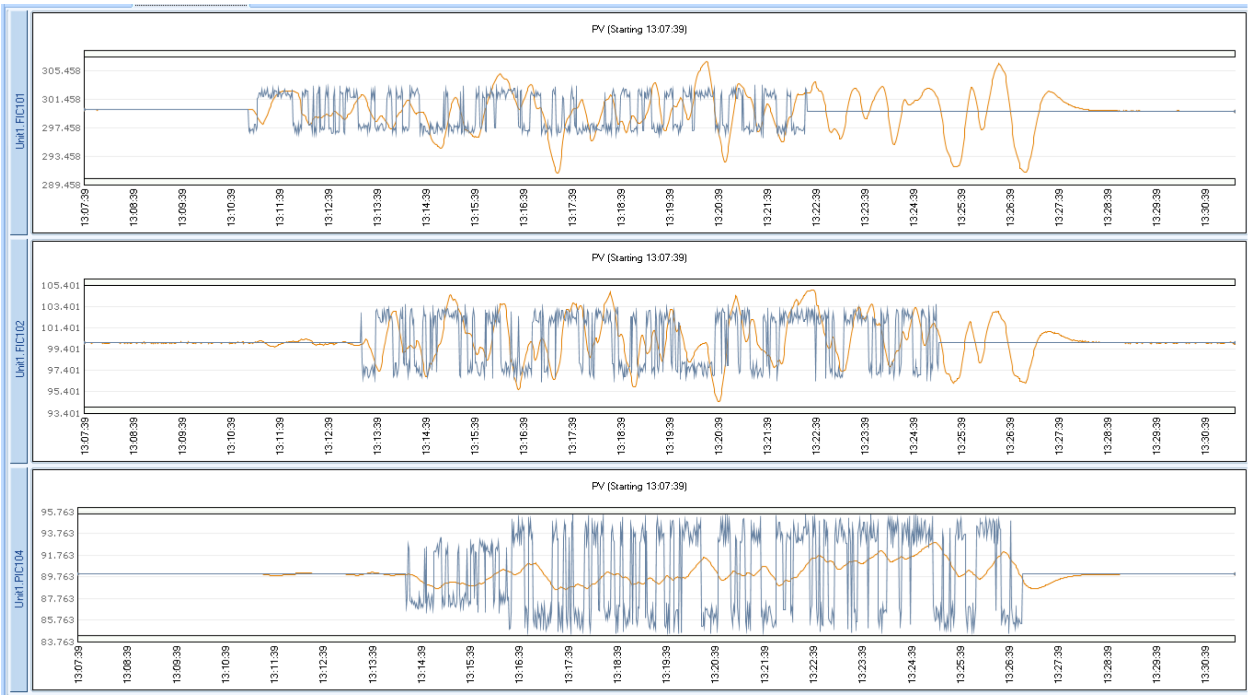
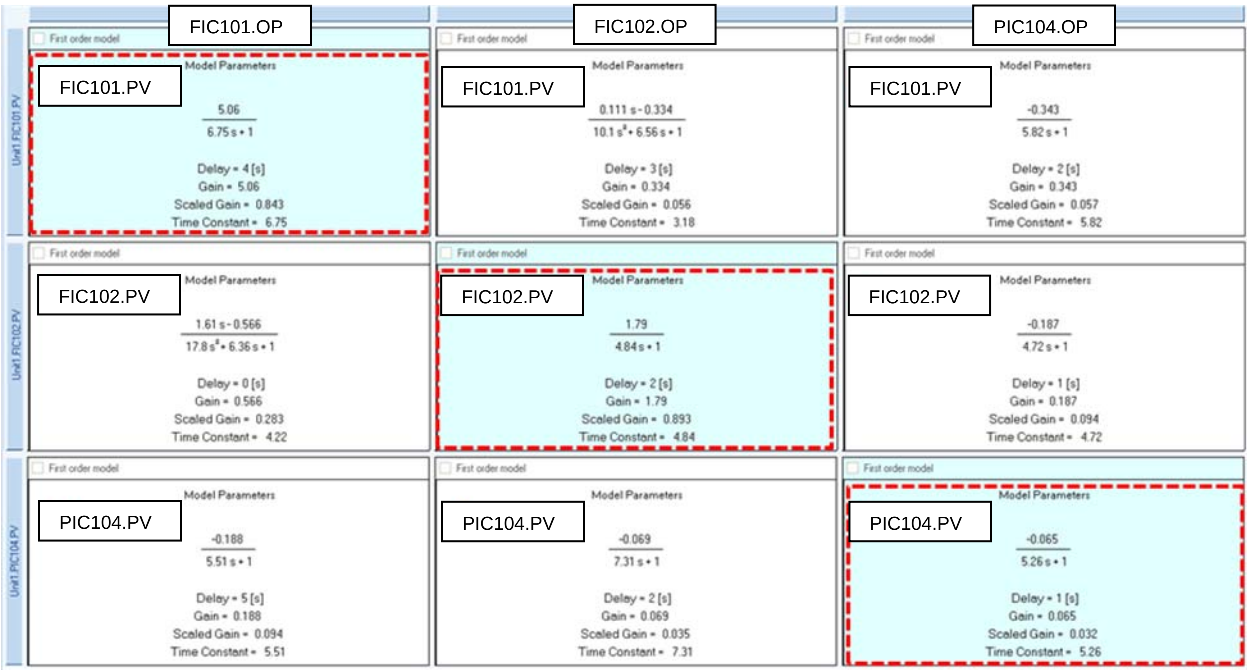
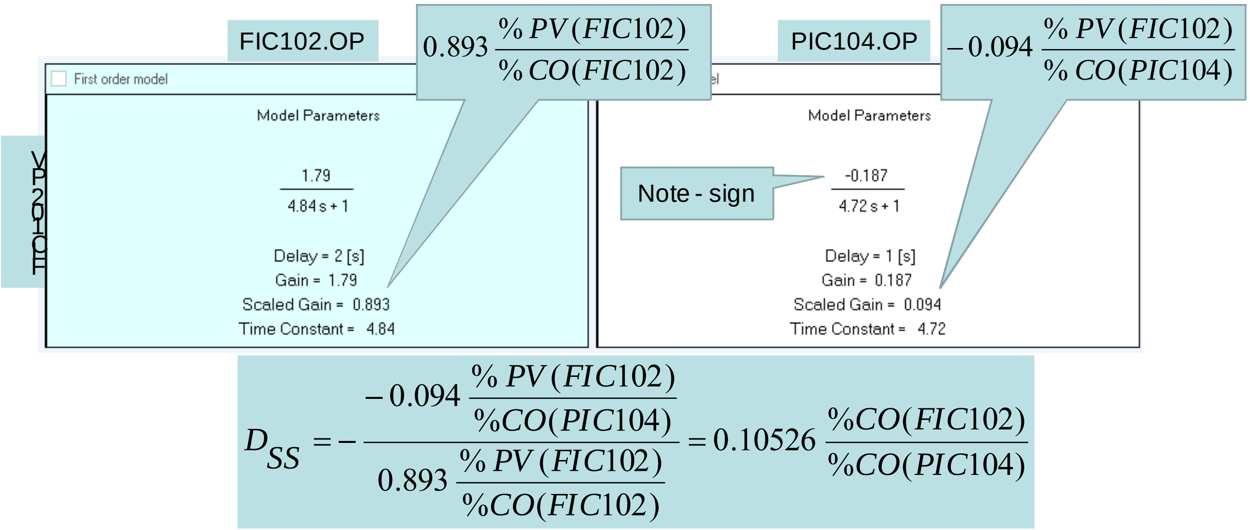


Figure 7: Closed loop simultaneous testing of the 3 x 3 system. The stepoints of each loop are moved with a user-prescribed amplitude with a 'dither' added to this signal. FIC100 was placed in automatic mode from cascade mode for this test. The setpoints were moved +/- 2 lbm/min for FIC100 (top), +/- 3 lbm/min for FIC102 (middle), and +/- 8 PSIG for PIC104 (bottom).



	FIC101.OP	FIC102.OP	PIC104.OP
Unit1.FIC101.PV FIC101.PV	□ First order model Model Parameters $\frac{5.06}{6.75s+1}$ Delay = 4 [s] Gain = 5.06 Scaled Gain = 0.843 Time Constant = 6.75	□ First order model Model Parameters $\frac{0.111s-0.334}{10.1s^2+6.56s+1}$ Delay = 3 [s] Gain = 0.334 Scaled Gain = 0.056 Time Constant = 3.18	□ First order model Model Parameters $\frac{-0.343}{5.82s+1}$ Delay = 2 [s] Gain = 0.343 Scaled Gain = 0.057 Time Constant = 5.82
Unit1.FIC102.PV FIC102.PV	□ First order model Model Parameters $\frac{1.61s-0.566}{17.8s^2+6.36s+1}$ Delay = 0 [s] Gain = 0.566 Scaled Gain = 0.283 Time Constant = 4.22	□ First order model Model Parameters $\frac{1.79}{4.84s+1}$ Delay = 2 [s] Gain = 1.79 Scaled Gain = 0.893 Time Constant = 4.84	□ First order model Model Parameters $\frac{-0.187}{4.72s+1}$ Delay = 1 [s] Gain = 0.187 Scaled Gain = 0.094 Time Constant = 4.72
Unit1.PIC104.PV PIC104.PV	□ First order model Model Parameters $\frac{-0.188}{5.51s+1}$ Delay = 5 [s] Gain = 0.188 Scaled Gain = 0.094 Time Constant = 5.51	□ First order model Model Parameters $\frac{-0.069}{7.31s+1}$ Delay = 2 [s] Gain = 0.069 Scaled Gain = 0.035 Time Constant = 7.31	□ First order model Model Parameters $\frac{-0.065}{5.26s+1}$ Delay = 1 [s] Gain = 0.065 Scaled Gain = 0.032 Time Constant = 5.26

Figure 8: Closed-loop testing of the flow and pressure loops yields a 3x3 matrix of models. The models on the diagonal, highlighted with a dotted box, are SISO models and show how each loop's output effects its process variable. The off-diagonal models show how one loops output effects another loops process variable.



FIC102.OP

$0.893 \frac{\% PV(FIC102)}{\% CO(FIC102)}$

□ First order model

Model Parameters

$\frac{1.79}{4.84s+1}$

Delay = 2 [s]
Gain = 1.79
Scaled Gain = 0.893
Time Constant = 4.84

PIC104.OP

$-0.094 \frac{\% PV(FIC102)}{\% CO(PIC104)}$

Model Parameters

Note - sign

$\frac{-0.187}{4.72s+1}$

Delay = 1 [s]
Gain = 0.187
Scaled Gain = 0.094
Time Constant = 4.72

$$D_{SS} = -\frac{-0.094 \frac{\% PV(FIC102)}{\%CO(PIC104)}}{0.893 \frac{\% PV(FIC102)}{\%CO(FIC102)}} = 0.10526 \frac{\%CO(FIC102)}{\%CO(PIC104)}$$

Figure 9: To decouple the effects of PIC104.OP on FIC102.PV, as shown previously in Table 2 we divide the steady state gains to obtain the steady state decoupler. Note that the dynamic portions of the models, the time constant, and the dead time, are very close, as we would expect in a closely coupled hydrodynamic system. Because the dynamics are close, we will implement only the steady state portion of the decoupler.

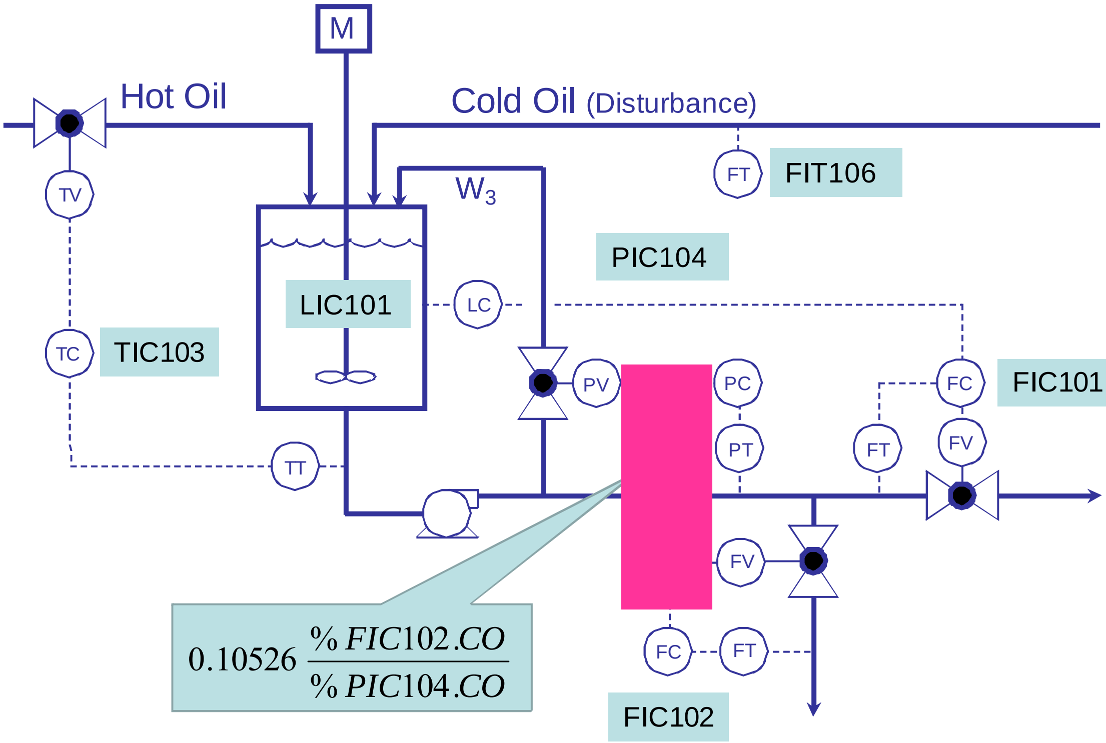


Figure 10: The decoupling strategy required cancel the effects of the output of the pressure controller PIC104 on the PV for FIC102.

Figure 9 shows the calculations to obtain the decoupler. Figure 10 shows the one-way decoupling strategy to cancel the effects of PIC104.CO on the PV for FIC102. The effect here is that for every 1% increase in opening for the pressure valve, we also open the flow valve for FIC102 by 0.105%. This will offset the decrease in pressure caused by opening the pressure valve, and cancel the reduction in flow that would otherwise result. A two-way strategy could also be implemented if needed, using the same approach. We could also use the models in Figure 8 to decouple FIC101 from PIC104 and FIC101 from FIC102.

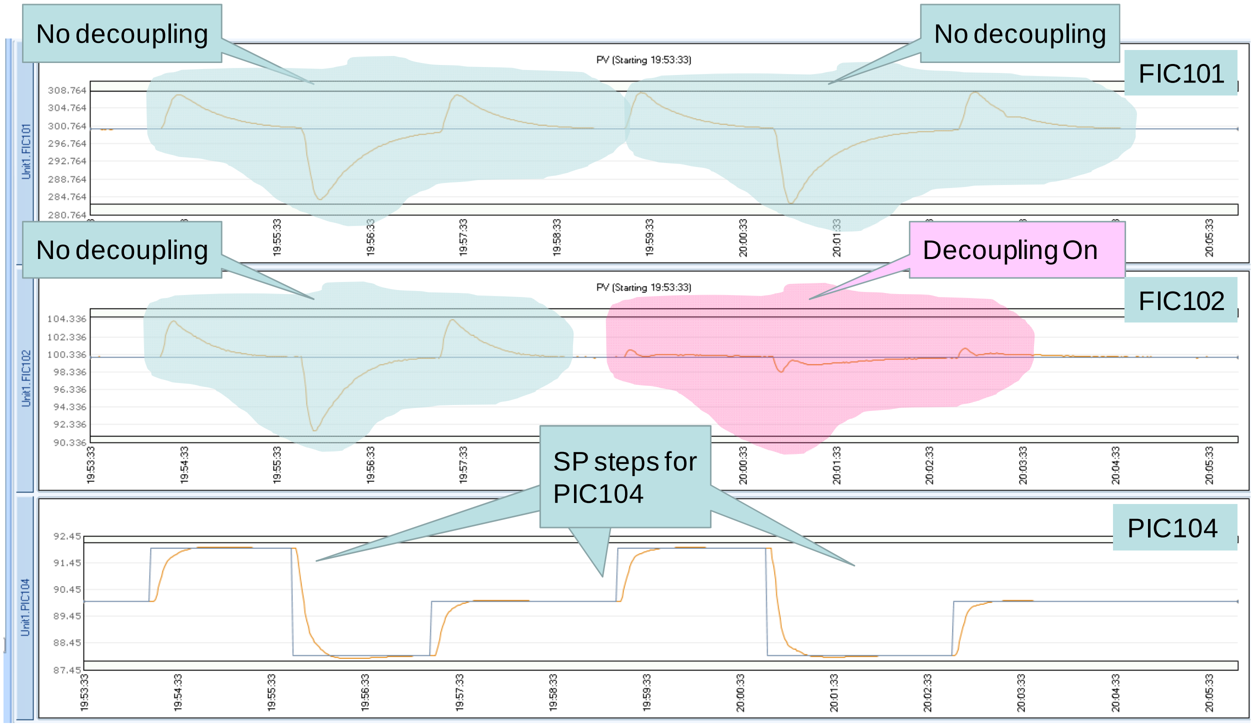
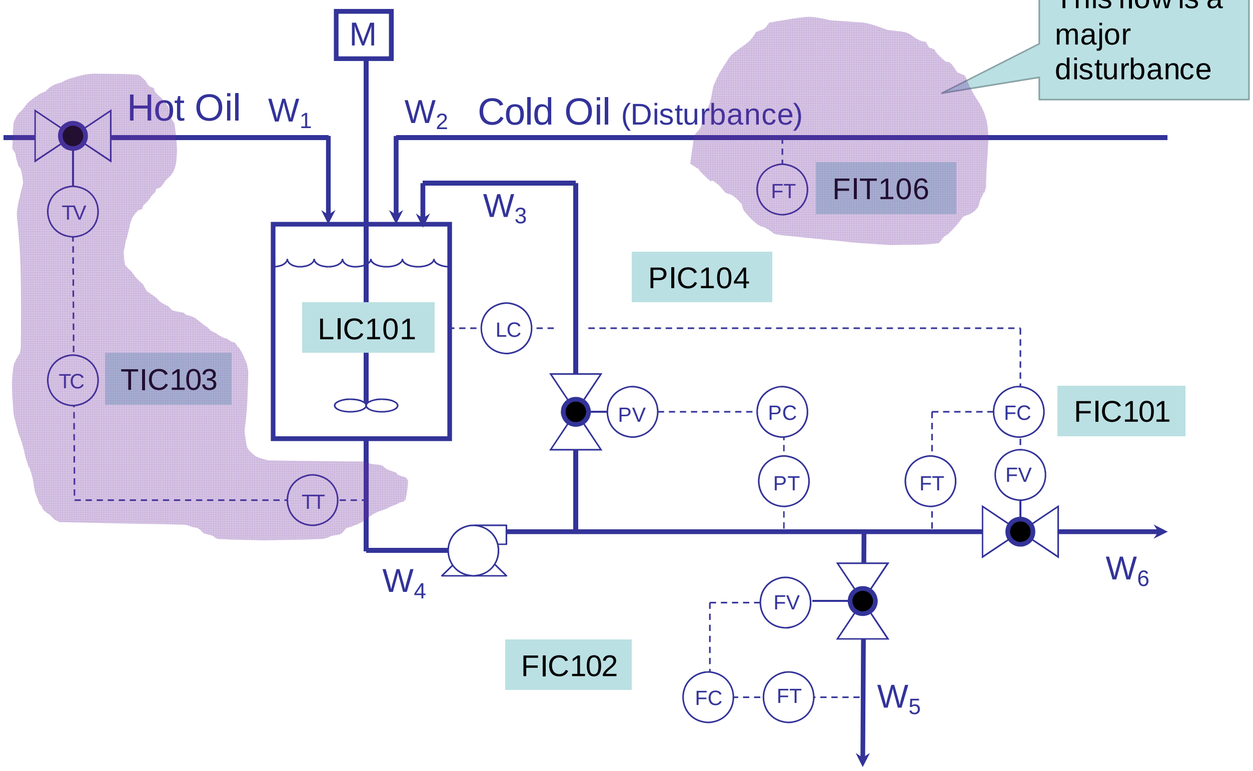


Figure 11: Results of the decoupler implemented to break the interaction between the pressure and the flow FIC102. The left hand side shows results with no decoupling. Note the effect on the right hand side. No decoupling has been implemented for FIC101. For FIC102, the control is not perfect due to dynamic differences or model error.

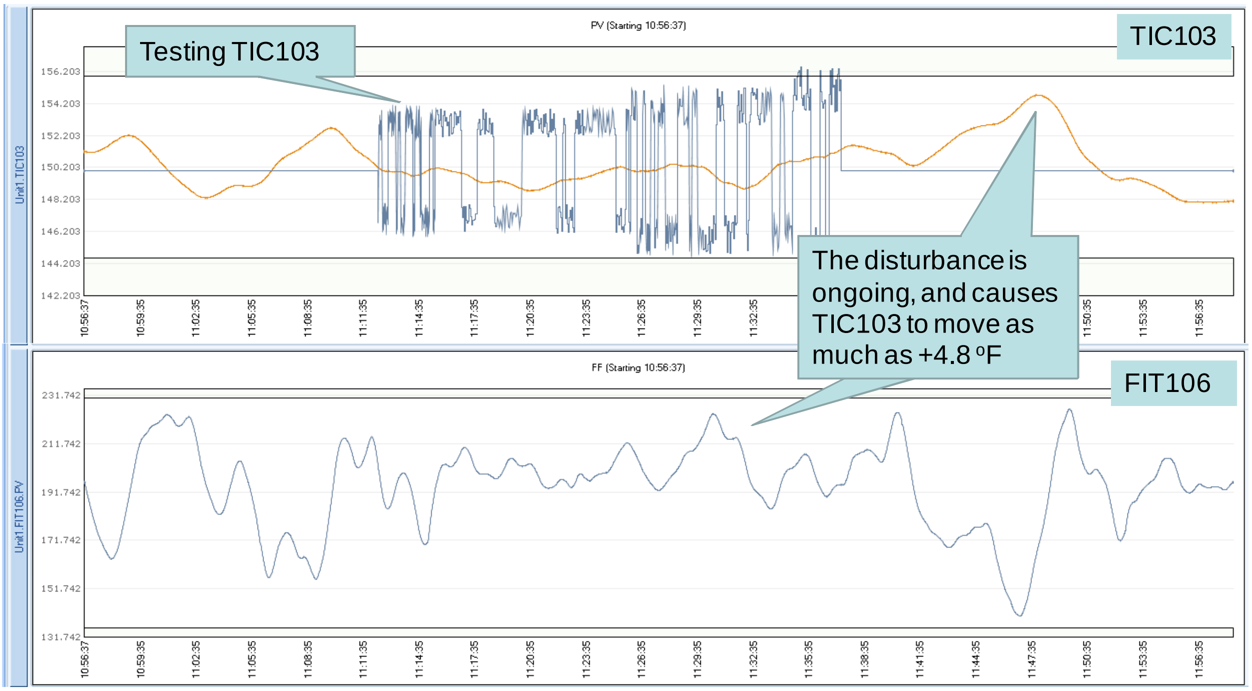
FEEDFORWARD CONTROL

Feedforward control is like decoupling control in that a model is used to correct for a disturbance external to the loop. In decoupling, the disturbance is the output from another loop. When implementing feedforward control, the disturbance used is a measured process variable. For the example process, the disturbance is the flow measured by FIT106. This flow is continuously changing and also undergoes major changes at times. In this paper, we will investigate how to implement feedforward control to cancel the effects of FIT106 on TIC103. The same process could be used to cancel the effect of this disturbance on other loops.



This flow is a major disturbance

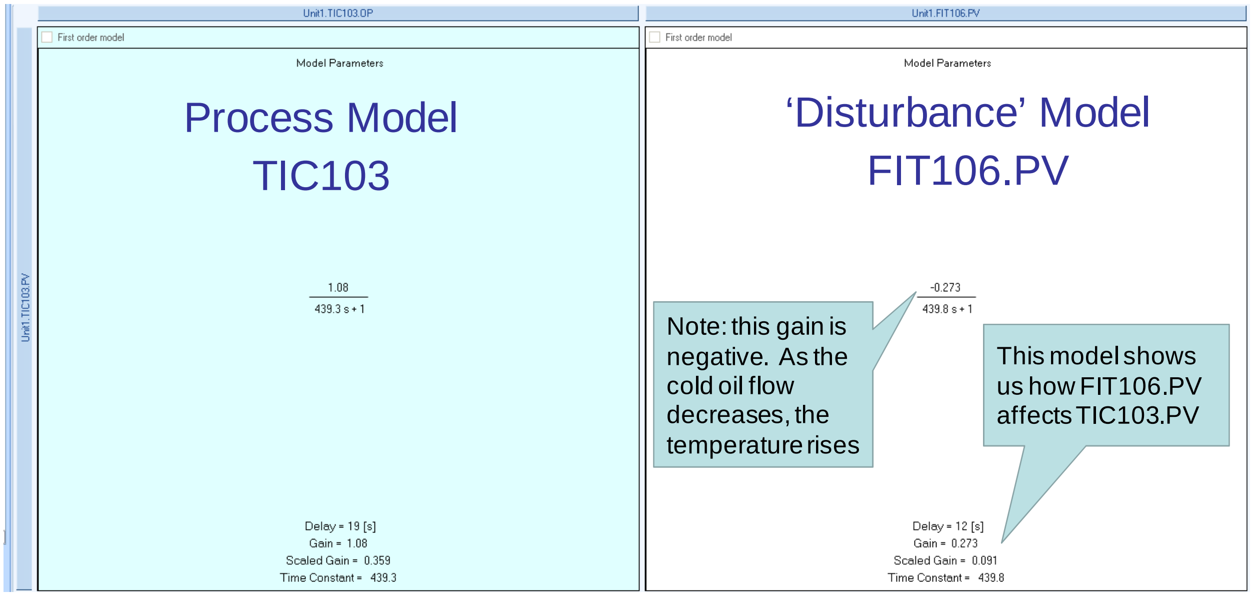
Figure 12: The flow measured by FIT106 is a disturbance to the rest of the process.



Testing TIC103

The disturbance is ongoing, and causes TIC103 to move as much as +4.8 °F

Figure 13: A test of the loop TIC103 is shown (top), while also collecting data for the flow FIT106 (bottom). There is an ongoing disturbance and some major disturbances in FIT106 (bottom). While the process is operating normally, we implement a test with the loop in automatic by moving the setpoint with a specified amplitude (+/- 4 °F) and duration (rapidly moving line in top plot). The slowly moving line is the temperature response to the setpoint changes and the disturbance FIT106.



Unit1.TIC103.OP

□ First order model

Model Parameters

Process Model

TIC103

$$\frac{1.08}{439.3\,s+1}$$

Delay = 19 [s]
Gain = 1.08
Scaled Gain = 0.359
Time Constant = 439.3

Unit1.TIC103.PV

Unit1.FIT106.PV

□ First order model

Model Parameters

'Disturbance' Model

FIT106.PV

$$\frac{-0.273}{439.8\,s+1}$$

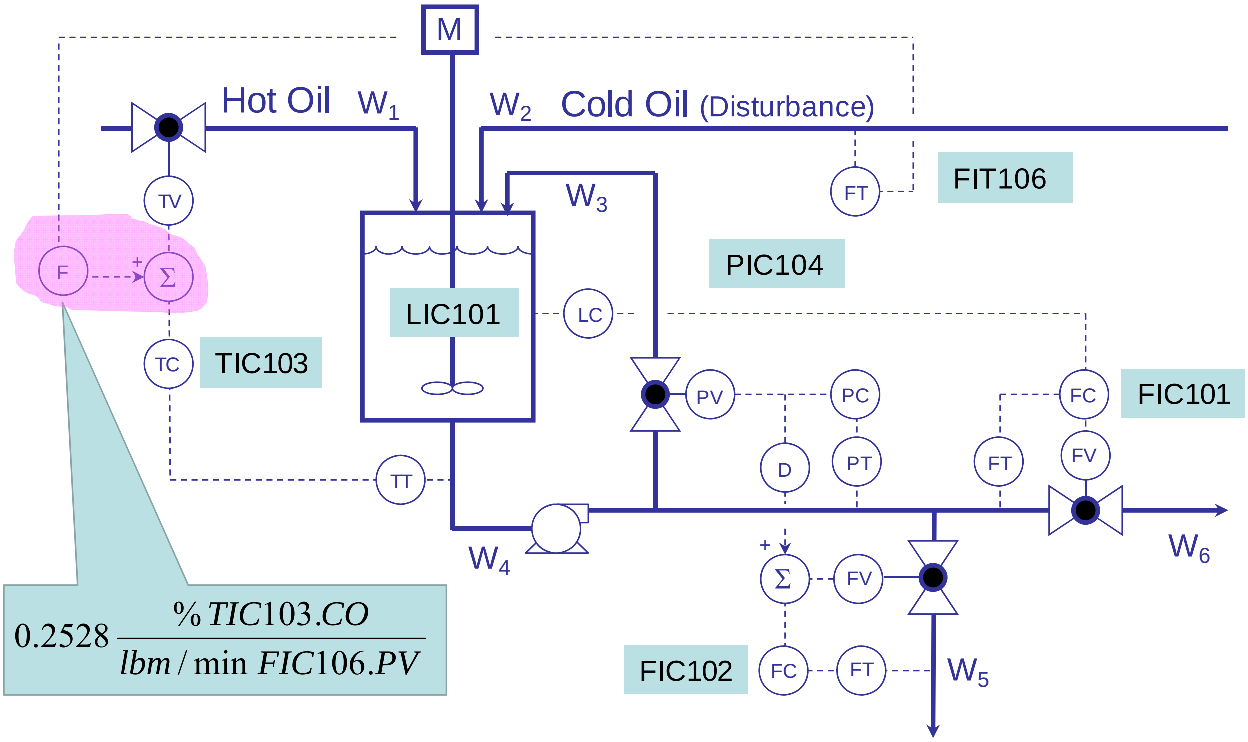
Note: this gain is negative. As the cold oil flow decreases, the temperature rises

This model shows us how FIT106.PV affects TIC103.PV

Delay = 12 [s]
Gain = 0.273
Scaled Gain = 0.091
Time Constant = 439.8

$$FF_{SS} = -\frac{-0.273\ {}^{o}F/lbm/min\ FIT106}{1.08\ {}^{o}F/\%CO\ TIC103} = 0.2528\,\frac{\%CO(TIC103)}{lbm/\min(FIT106)}$$

Figure 14: The models obtained from the test in Figure 13. Using the negative of the ratio of gains (according to Table 1), the steady state portion of the feedforward controller is obtained. Since the time constants are similar, compensation by lead/lag is not required. There is a difference in the time delays, but the dead time of the disturbance is less than the dead time of the process, so we will be unable to implement dead time compensation.



$$0.2528\,\frac{\%\,TIC103.CO}{lbm/\min\ FIC106.PV}$$

Figure 15: Implementation of feedforward control to cancel the effects of FIT106 on TIC103. What the steady state gain means is that for every 1 lb/min increase in cold oil flow, 0.2528% is added to the controller output for TIC103.

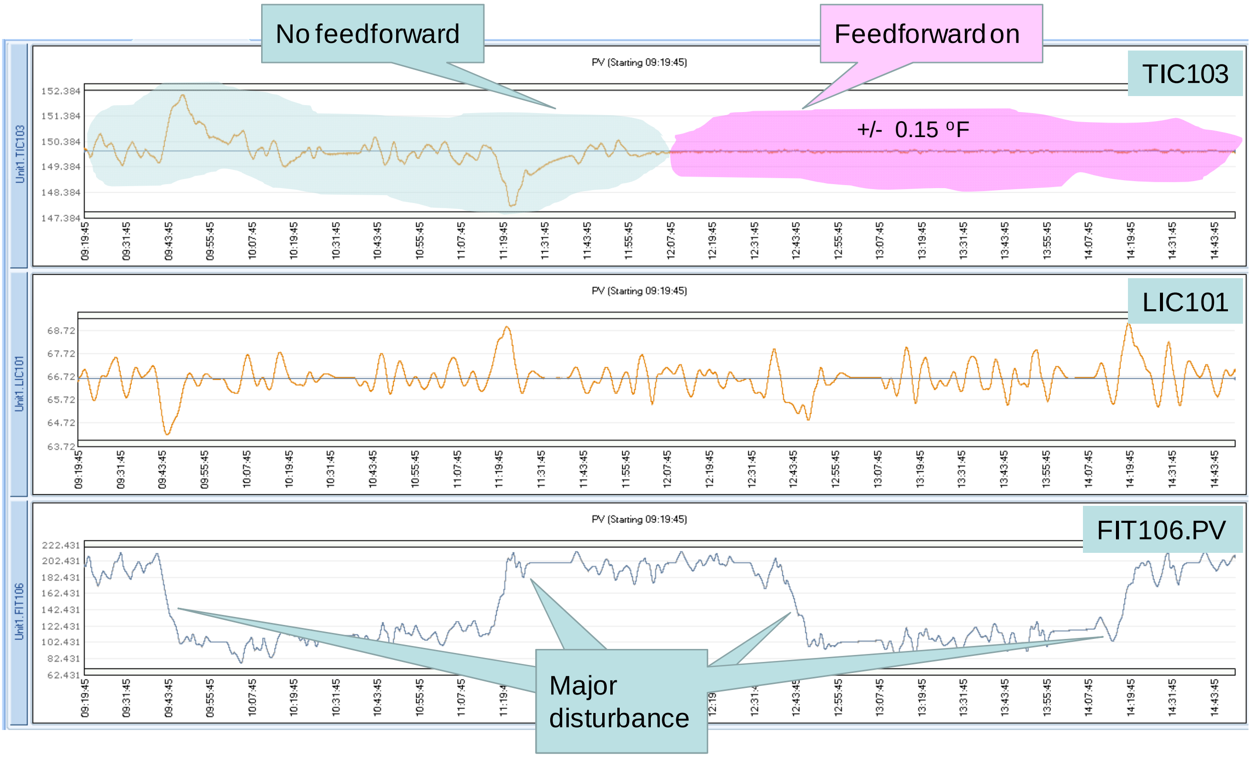


Figure 16: Results with the feedforward solution, with control improved to a range of 0.3 ^{o}F from a range of 5 ^{o}F.

CONCLUSIONS

Loop interaction and disturbances are clear challenges in industrial control. We have presented detuning as a possible approach to reducing loop interactions and have also presented a practical decoupling solution using new software tools. The detuning approach represents a first step; if this meets the needs of the process this is an excellent solution, but has drawbacks. If more than one loop in an interacting system needs to be tightly controlled, then decoupling can be used, provided one can obtain the models easily enough. This paper shows how well new tools can complete this testing and calculate the models while the process is operated normally.

The same tools can be used to model disturbances for use in feedforward control implementation while the process runs at normal conditions, provided there is enough movement in the disturbance.

The results presented show how simple it is to implement classical feedforward and decoupling control solutions to improve process performance, leading to less variability in the process. Anyone can now effectively combat disturbances and interactions in their process.

ABBREVIATIONS

CO – Controller Output
FOPDT – First Order Plus Dead Time
MIMO – Multiple Input Multiple Output
PV – Process Variable
SISO – Single Input Single Output

REFERENCES

1. Automated Continuous Process Control, Carlos A. Smith, Wiley, 2d ed., 2002
2. Process Dynamics & Control, D.E. Seborg, T.F. Edgar, D.A. Mellichamp, Wiley, 2d ed., 2004