

# **Using MV overshoot as a tuning metric in choosing DMC move suppression values.**

Corresponding Author  
GZ Gous  
gustaf.gous@bluesp.co.za  
Chief Engineer (APC)  
BluESP (Pty) Ltd  
PO Box 1244  
Jukskei Park  
2153  
South Africa  
+27 11 251 5900

Prof P L de Vaal  
philip.devaal@up.ac.za  
Head: Department of Chemical Engineering  
University of Pretoria  
Pretoria 0001  
South Africa  
+27 12 420 2475

# Using MV overshoot as a tuning metric in choosing DMC move suppression values.

## Abstract

Traditionally the tuning of dynamic matrix control (DMC) type multivariable controllers is done by trial and error. The APC engineer chooses arbitrary starting values and tests the performance on a simulated controller. The engineer then either increases the values to suppress movement more, or decreases them to have the manipulated variables move faster. When the controller performs acceptably in simulation, then the tuning is improved during the commissioning of the controller on the plant. This is a time consuming and unscientific exercise and therefore often does not get the required attention. This leads to unacceptable controller behaviour during commissioning and sub-optimal control once commissioning is completed.

This paper presents a new method to obtain move suppression factors for DMC type multivariable controllers by using a Nelder Mead search algorithm to find move suppressions that will provide acceptable control behaviour. Acceptable behaviour is described by characterising the dynamic move plan calculated by the controller for each of the manipulated variables.

## Keywords

Multi-variable control tuning

DMC tuning

MPC tuning

APC tuning

Manipulated variable overshoot

# 1. Introduction

Model based multivariable control using the method now known as DMC or MPC was developed in industry in the early 1970's with the first application in 1973 (Cutler, 1983, Qin & Badgwell, 2003) and Prett and Gillette(1980) presenting the first application of DMC on a FCC unit. Numerous advances have been made such as non-linear controllers, matrix conditioning, real time gain scheduling etc. (Qin & Badgwell, 2003). The one part of the technology that is still largely without a scientific base has been the selection of move suppressions to tune the controller behaviour. The accepted approach in industry is to use trial and error to find tuning values that will provide acceptable controller behaviour in an offline simulation. These values are then refined during the commissioning of the online controller. Using trial and error on a small (4 manipulated variable) controller is already time consuming, and there are controllers in the field with more than 100 manipulated variables. Using this approach is bound to lead to oversights that will lead to difficulty during the commissioning of the controller. During the commissioning, the problems that do occur are often addressed by over suppressing the manipulated variable movement, leading to a badly tuned controller.

The challenge in choosing or guessing move suppressions lies in the multivariable nature of the controller. Changing the move suppression on one manipulated variable will not only change the behaviour of that manipulated variable, it will also change the behaviour of every other manipulated variable with models to the same controlled variables. In the same way, changing the equal concern error of a controlled variable will also affect the behaviour of every other manipulated variable with shared models. This makes the iterative approach as used throughout industry even more time consuming and unscientific.

Another challenge is that the "right" behaviour in a DMC controller is very ill-defined and is often based on the practitioner's personal experience and preferences. More or less aggressive controller behaviour is the outcome of the engineer's aversion to risk. Should the engineer make the controller too aggressive, circumstances such as model error may cause the controller to produce unwanted cycles on the plant. On the other hand, too little aggression in tuning may result in a controller that is sluggish and does not control the process properly.

A further level of complexity is added by the model-based nature of a DMC controller. If the model is an exact match of the plant behaviour, very aggressive tuning may be used. If factors like non-linearity, process response changes or process noise lead to substantial plant/model mismatch, aggressive tuning will once again lead to undesired controller behaviour and performance.

In order to address these complexities a metric was defined to describe desirable controller behaviour. This metric is based on the dynamic move plan calculated for the manipulated variables over the control horizon of the controller.

After a brief discussion of DMC tuning and current work in this field, the claim that tuning will vary between engineers will be examined. This will be done by showing widely varying tuning obtained by asking several engineers to tune the same two controllers on two simulated plants. The tuning metric will then be defined and its

results in obtaining tuning values will be compared with the tuning from the engineers. It will be shown how more or less aggressive tuning can be obtained by using the metric. The effect of tuning aggression on controller performance will be demonstrated by running simulations under perfect controller conditions and simulations with challenges such as model error. In conclusion an approach will be outlined to use this method to obtain good starting tuning values before DMC commissioning commences.

## 2. Tuning DMC controllers

### 2.1.1. Traditional DMC tuning

Traditional DMC tuning is done by first choosing the equal concern errors or controlled variable weights for all controlled variables. The equal concern errors are values that indicate a comparable value of controlled variable error in AspenTech's DMCplus<sup>TM</sup>. This is a good way of comparing controlled variables with dissimilar units of measure, like temperatures and pressures. The values are often chosen by asking the question "What magnitude of controlled variable violation would cause operator distress?" and using this value as the equal concern error. In other words, if a violation of 2 kPa on a pressure would get the operators attention, and a violation of 5 °C on a temperature would have a similar effect, these are good values for equal concern errors. The values are chosen in an arbitrary fashion, but represent the comparative value of keeping the controlled variables within limits. After step testing control engineers often have a very good feel for these values, as they have probably caused operator discomfort during the entire step testing process.

Next the move suppression values must be chosen for all manipulated variables. Traditionally this is a non-scientific, laborious affair of trial and error (Iglesias, Sanjuán and Smith, 2006) where initial move suppression values are chosen and the controller response to upsets and setpoint changes simulated. The move suppressions are then adapted until the engineer is satisfied with the rate of change on the manipulated variables when controlled variable error exists in the simulation. This approach is still taught in advanced control courses in industry. Next the controller is commissioned on the plant and the tuning parameters refined by observing the controller response (Qin & Badgwell, 2003).

There are two major shortcomings with this approach. Firstly the definition of a "reasonable" response has never been defined and is left to the judgment of the control engineer. This will lead to the result that two engineers will tune the same controller and come up with very different tuning parameters that are based on experience and personality.

Secondly, the methodology is made difficult due to the multivariable nature of the tuning. If the move suppression of one manipulated variable is increased, this will cause the controller to allow less movement in that manipulated variable, which will inevitably lead to more controlled variable error as well as more movement on all manipulated variables that have models to the same controlled variables.

These points were proven by asking eight control engineers to tune the same two controllers and by comparing the chosen move suppression values. Details regarding the controllers are given in Appendix A. They all followed the traditional trial and error approach. No guidance was given regarding acceptable tuning, they had to use their own discretion and experience. The engineers were instructed to only change move suppression values, no changes to other tuning parameters like equal concern errors were allowed. The experience of the engineers varied from 2 to more than 10 years in APC as shown in table 2.1.

Table 2.1 Experience of participating engineers

	Years APC Experience	Number of controllers implemented	Largest number of manipulated variables
Engineer 1	10+	25	25+
Engineer 2	2-5	25+	25+
Engineer 3	2-5	10-25	10-25
Engineer 4	5-10	10-25	10-25
Engineer 5	2-5	5-10	10-25
Engineer 6	2-5	5-10	10-25
Engineer 7	5-10	10-25	10-25
Engineer 8	2-5	20	10-25

The different tuning results are shown in table 2.2 and 2.3.

Table 2.2 Move Suppression values chosen for a distillation plant

	Engineer 1	Engineer 2	Engineer 3	Engineer 4	Engineer 5	Engineer 6	Engineer 7	Engineer 8	Average	Standard deviation
SupMov1	10.0	10.0	7.0	100	3	0.02	1	1	16.5	36.1
SupMov2	1.0	5.0	8.0	5	10	0.25	0.1	1	3.8	3.9
SupMov3	5.0	5.0	5.0	5	20	0.25	0.1	5	5.6	6.7

As can be seen, the engineers' chosen tuning values differed widely. There was no discernible difference between the values chosen by the engineers with more than 5 years experience and those with less experience.

Table 2.3 Move suppression values chosen for a reactor

	Engineer 1	Engineer 2	Engineer 3	Engineer 4	Engineer 5	Engineer 6	Engineer 7	Engineer 8	Average	Standard deviation
SupMov1	6.0	2.5	5.0	3	5	6	4	8.5	5.0	1.4
SupMov2	12.0	5.0	6.0	3	12	12	1	14	8.1	4.7
SupMov3	15.0	5.0	4.0	3	12	12	1	15	8.0	5.1
SupMov4	5.0	2.0	3.0	3	6	6	1	11	4.8	2.2
SupMov5	1.0	2.0	5.0	9	7	7	1	20	7.1	4.3

Here the engineers' chosen tuning values also differed widely. The engineers with more than 5 years experience chose tuning values that were more conservative.

This is an indication of the current situation in industry, with initial controller tuning varying according to the experience and personality of the control practitioner.

### 2.1.2. Recent developments in DMC tuning

There have been attempts to calculate the required move suppression factors mathematically. Shridar and Cooper (1998) noted that move suppression factors serve a dual purpose in DMC. Increasing the move suppressions will decrease manipulated variable movement, but will also decrease the matrix conditioning number. They used an approach that is based on the premise that these two effects are interrelated. They deduced that move suppression factors that present a well-conditioned controller matrix will provide a well behaved controller in terms of tuning and developed a tuning strategy that will calculate move suppressions to provide a predefined matrix conditioning number.

Other authors (Iglesias, Sanjuán and Smith, 2006) report that this method leads to unacceptably aggressive tuning. They developed a method of simulating the control behaviour and minimising a cost function. The cost function is the integral of the controlled variable error added to the integral of the manipulated variable movement multiplied by a weighting factor or

$$PP = \int_0^{\infty} |e(t)| dt + \Gamma \int_0^{\infty} |m(t)| dt \dots\dots\dots 2.1$$

where:

- e(t) is the controlled variable error
- m(t) is the manipulated variable moves
- $\Gamma$  is the weighting factor placed on the manipulated variable movement .

Increasing the weighting factor will decrease manipulated variable movement. Analysis of variance was then used to find the significant variables to calculate a tuning equation. The tuning equation developed provides significantly larger values of move suppression, with the shortcoming that it will only provide tuning values for SISO systems with first order models, severely limiting its use in industry.

Kai Han et al (2006) proposed a min-max algorithm that will select tuning parameters that will cause controlled variables to move sharply to steady state values with slightly oscillatory behaviour.

Ghazzawi, A et al. (2010) introduced an online tuning strategy that will re-tune the controller in real time, based on the predicted closed loop controlled variable response. Acceptable dynamic limits are set on setpoint changes and disturbance rejection.

While it is true that this approach will keep retuning the controller to handle controlled variable deviations consistently even when manipulated variables are constrained, this also introduces the risk that the control action will become excessively aggressive when all primary manipulated variables are constrained and the controller must use secondary relationships to bring a controlled variable to setpoint. As it is often difficult to obtain good models for these weaker relationships, this approach may then lead to unacceptable controller behaviour due to model error.

This risk is mitigated by placing upper and lower bounds on the move suppressions. These limits on the allowed move suppression values must once again be found using the traditional tuning approaches.

### 3. Discussion

#### 3.1. *Lack of definition of acceptable control performance*

During a tuning exercise, comments like “That manipulated variable is moving too fast” are often heard. These comments are based on the experience and personal taste of the individual engineer, with no common language that can be used to compare the level of aggressiveness of the chosen tuning variable values. It is also not possible to compare the tuning between different controllers or even different manipulated variables in one controller with each other.

Many control engineers focus on magnitude of manipulated variable movement when tuning a controller and are indeed taught this way. The problem with this approach is that the magnitude of the manipulated variable movement depends on the size of the controlled variable error. As is intuitively clear, if the controlled variable is far from the desired steady state value, the manipulated variable will have to move over a larger range to make this possible. Choosing the magnitude of manipulated variable movement as indication of tuning aggression can therefore be misleading. In figure 3.1 it is shown that doubling the size of the controlled variable error will double the size of the manipulated variable movement in a linear fashion. This happens because the DMC algorithm assumes linearity.

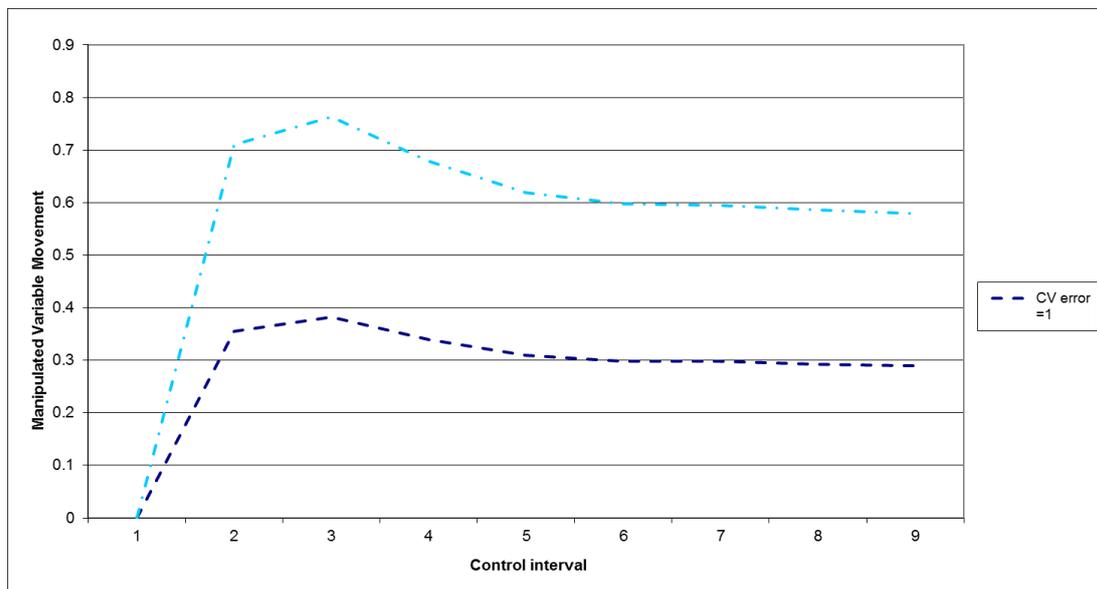


Figure 3.1 Effect of increasing controlled variable error

## 4. Proposed Solution

### 4.1. Description of method developed

#### 4.1.1. Characterisation of acceptable control behaviour – manipulated variable overshoot

What is clear in figure 3.1 though is that even if the controller has to double the magnitude of the manipulated variable movement in order to address the controlled variable error, the shape of the move plan does not change. This fact can be exploited to define aggressiveness of DMC tuning.

If a manipulated variable is tuned very slowly, it will rise (or fall) steadily over the control horizon, almost in a linear fashion. The controlled variable will typically respond by moving slowly to the desired steady state value. More aggressive tuning will cause the manipulated variable to rise quite sharply to the steady state value, causing the controlled variable to rise quicker to its steady state target. If the move suppression is decreased more, the manipulated variable will tend to go beyond its steady state value, and then return to it at the end of the control horizon. This will typically cause the controlled variable to go to its steady state target quickly, in some cases even crossing it before settling to it at steady state.

The tendency of a manipulated variable to go beyond its steady state value and then change direction to settle at the steady state value will be called manipulated variable overshoot. The magnitude of the overshoot will be calculated as shown in figure 4.1.

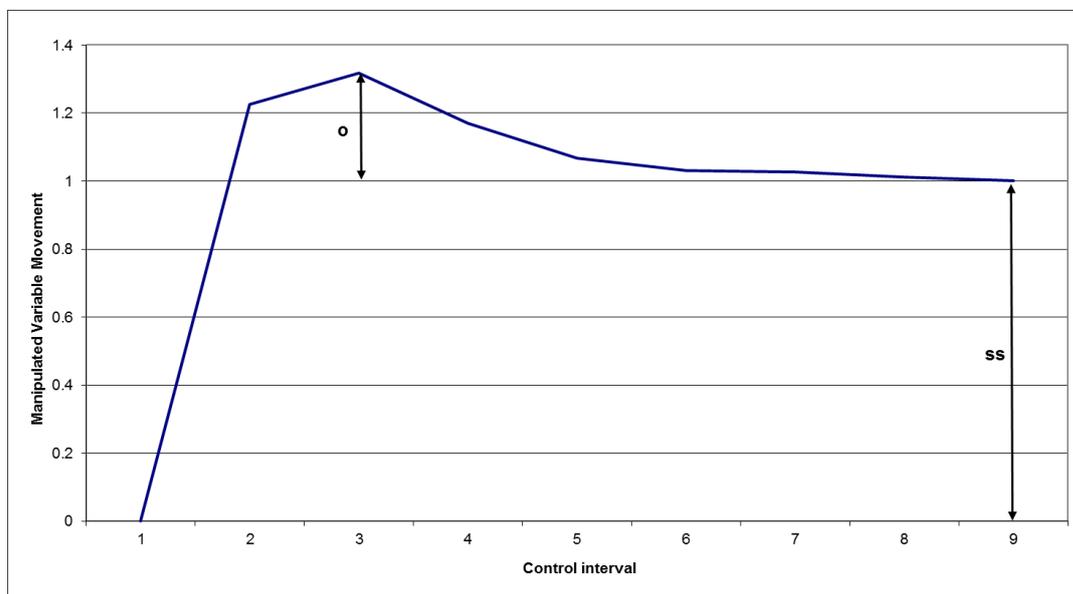


Figure 4.1 Calculating manipulated variable overshoot

If overshoot is defined as the magnitude of movement of the manipulated variable beyond the steady state value, shown by “o” in figure 4.1, with the steady state value shown as “ss”, then:

$$\text{Percentage MV Overshoot} = (o * 100) / ss \dots\dots\dots 4.1$$

As shown before the percentage manipulated variable overshoot is not dependent on the controlled variable error, but it is a good indication of the level of aggression of tuning.

If a typical multi-variable controller is considered, it can be seen how the level of manipulated variable overshoot changes if the move suppression values are changed on one manipulated variable, when all other variables are kept constant.

Table 4.1 Manipulated variable overshoot as function of move suppression

Move suppression	Manipulated variable overshoot
1	30.7%
5	7.0%
9	1.3%
13	0.0%

Table 4.1 clearly shows how manipulated variable overshoot decreases in a non-linear fashion when the move suppression values are increased. This response is not only dependent on the move suppression of the manipulated variable, it also depends on the specific model and the move suppressions of the other manipulated variables.

The use of manipulated variable overshoot is equivalent to the use of manipulated variable overshoot in PID or SISO tuning (King, 2011). It is used in Lambda tuning first developed by Dahlin (1968). Lambda tuning is a form of internal model control for which Chien (1988) and later Chien and Fruehauf (1990) developed tuning methods using direct synthesis. In PID tuning manipulated variable overshoot is also seen as a measurement of acceptable tuning aggression.

It is therefore suggested that in multivariable control manipulated variable overshoot can also be used as an indication of aggression of tuning and the metric that can be used to quantify this behaviour is simply the manipulated variable overshoot.

#### **4.1.2. Characterisation of acceptable controller behaviour – first order manipulated variable movement**

Table 4.1 shows that at large values of move suppression, manipulated variable overshoot goes to zero, and increasing move suppression more will not affect this. If a controller is desired with tuning that is less aggressive than tuning that will cause manipulated variable overshoot, another indication of tuning is if the chosen parameters cause the manipulated variables to gently go to their steady state values, moving along a first order path.

A controller that is tuned to follow a first order manipulated variable move plan will be quite robust when there is significant model uncertainty, but will still react fast enough to provide reasonable control performance.

The metric that was developed to characterise a manipulated variable that follows a first order path is to calculate the sum of the squared error between a first order move plan and the actual manipulated variable movement.

$$\text{First order metric error} = \sum^{\text{controlmoves}} (\text{first order MV path} - \text{actual MV path})^2 \dots\dots\dots 4.2$$

In the controller example of table 4.1, a move suppression of 17 will cause the manipulated variable to follow a first order path to its steady state value.

#### **4.2. Description of method developed**

Once a decision regarding which metric to use was made, many search algorithms can be used to find the multiple tuning constants that will provide the desired behaviour in the control action. For this project the Nelder-Mead or downhill simplex method (Nelder and Mead, 1965) was chosen. The implementation was done in Microsoft Excel, using a Visual Basic macro to implement the search algorithm and calculate the DMC responses. Once the metric error has been measured, Nelder-Mead is used to minimise the metric error by adjusting the move suppression, and optionally the move suppression multipliers or the steady state costs of the manipulated variables.

## 5. Analysis of method performance

### 5.1. Comparison of different definitions of optimal tuning

#### 5.1.1. Controller performance with no model error

To compare the performance of the different tuning constants obtained by using the definitions of acceptable control, two simulated plants and controllers were used. The different metrics for acceptable DMC tuning as described in section 4.1 were applied and the macro found move suppression values that will cause the behaviour as defined by the metrics.

The different move suppression values were loaded in an online controller and controller setpoints were changed. Data was collected on all controller variables to compare the performance. To demonstrate the change in manipulated variable move plans and controlled variable responses, one manipulated variable and one controlled variable of the reactor controller are shown in figures 5.1 and 5.2 and a manipulated variable and controlled variable of the distillation process are shown in figures 5.3 and 5.4. Only one manipulated and controlled variable from each controller is shown for succinctness as the behaviour is typical of all manipulated and controlled variables.

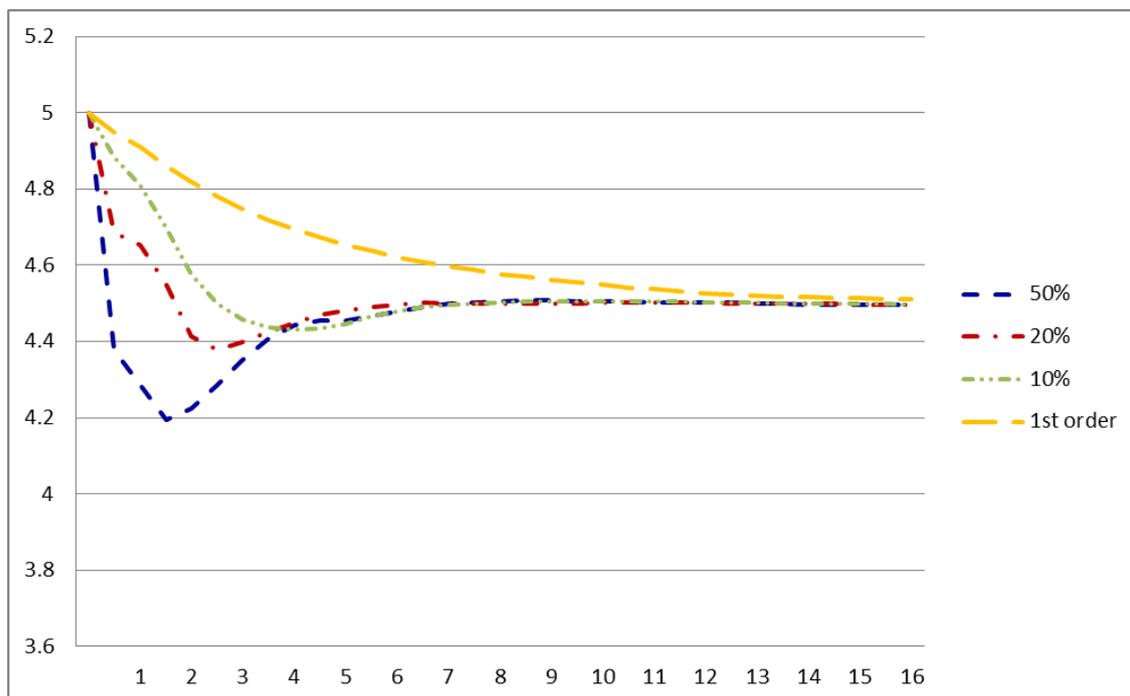


Figure 5.1 Reactor MV1 movement without model error

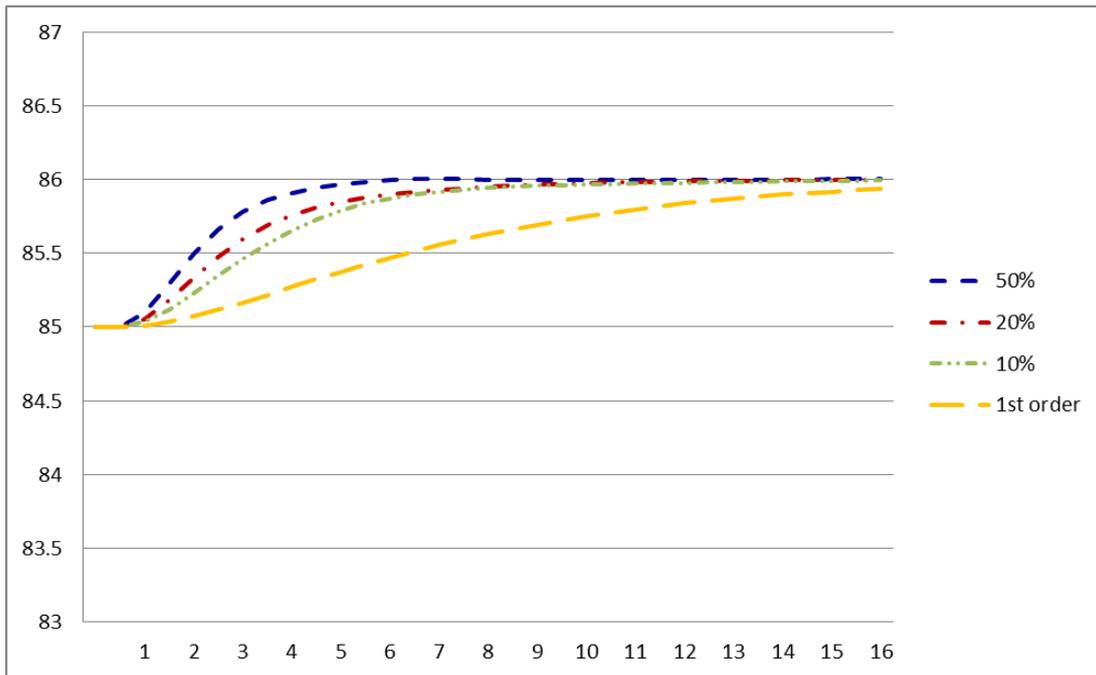


Figure 5.2 Reactor CV1 movement without model error

Figure 5.1 shows how the manipulated variables of the reactor controller move to their steady state value faster when the more aggressive tuning metrics like 50% overshoot are used. The controlled variables also move to setpoint faster as seen in figure 5.2. Using a first order move plan for the manipulated variables as metric and using a first order response path for the controlled variables provide much slower tuning as shown.

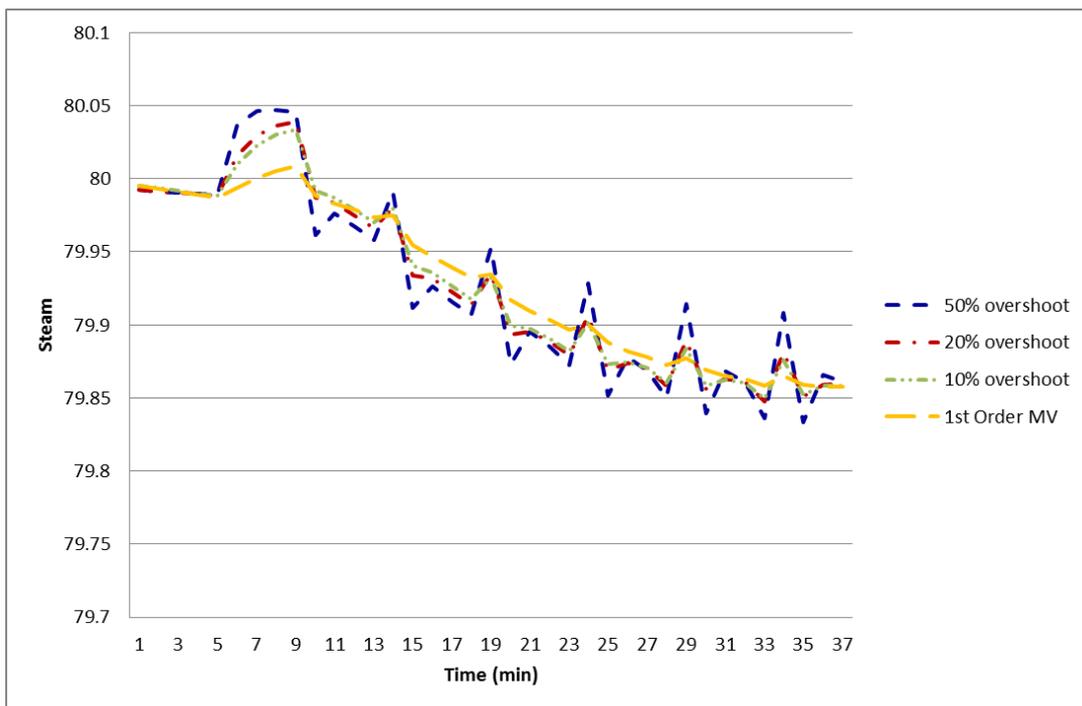


Figure 5.3 Distillation plant steam movement with no model error

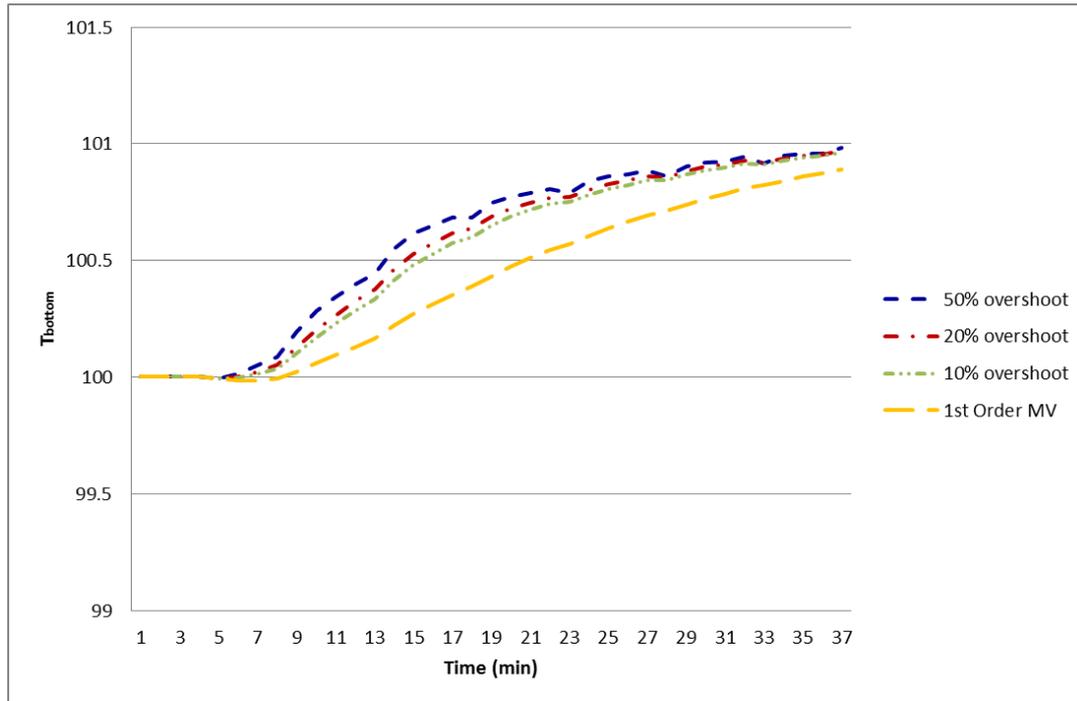


Figure 5.4 Distillation plant  $T_{bot}$  response with no model error

In figure 5.3 and 5.4 the effect of an ill-conditioned controller matrix can be seen, especially with the more aggressive tuning parameters. Because all controlled variables are controlled to a setpoint, slight numerical differences between the controller prediction and the simulation response cause the cycle.

### 5.1.2. Controller performance with model error

Model error was created by changing the model gains on the simulations. If the plant or simulator response is smaller than the prediction created by the controller models, this will lead to a sluggish controller. While this is not optimal, it is much less destructive than when the plant or simulator response is much larger than the prediction. This will lead to the controller cycling, especially if aggressive move suppressions were used. All gains on the reactor simulator were increased by 100% to simulate and test the controller response with different tuning constants under worst case conditions. These responses are demonstrated by showing the movement of the first manipulated variable and controlled variable of the reactor in figures 5.5 and 5.6.

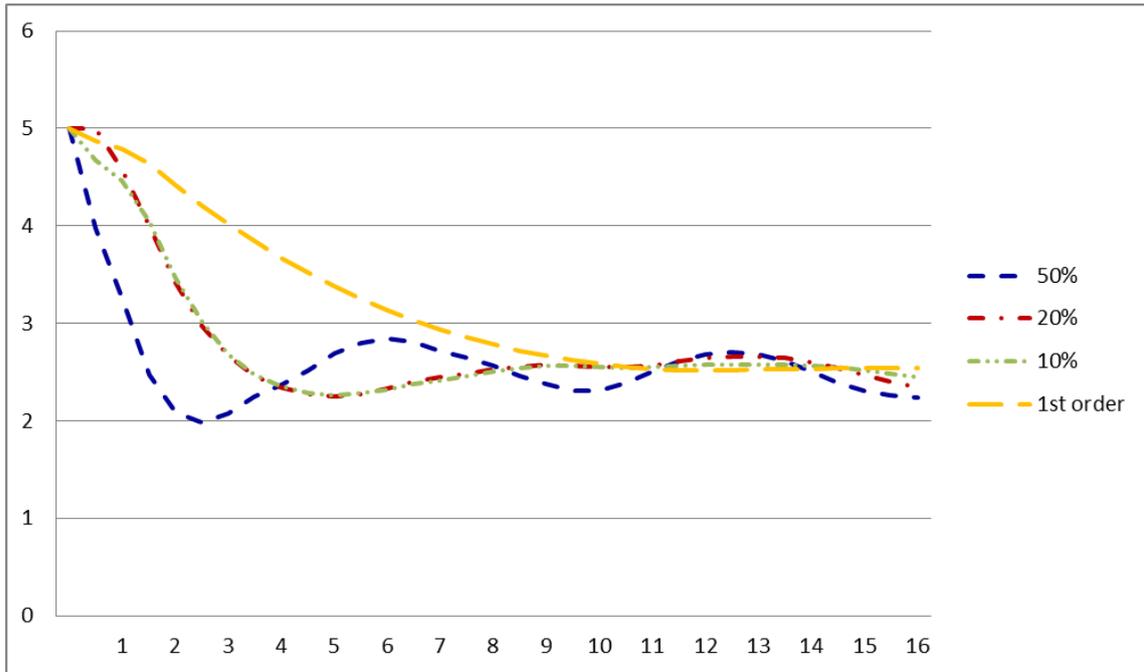


Figure 5.5 Reactor MV1 movement with model error

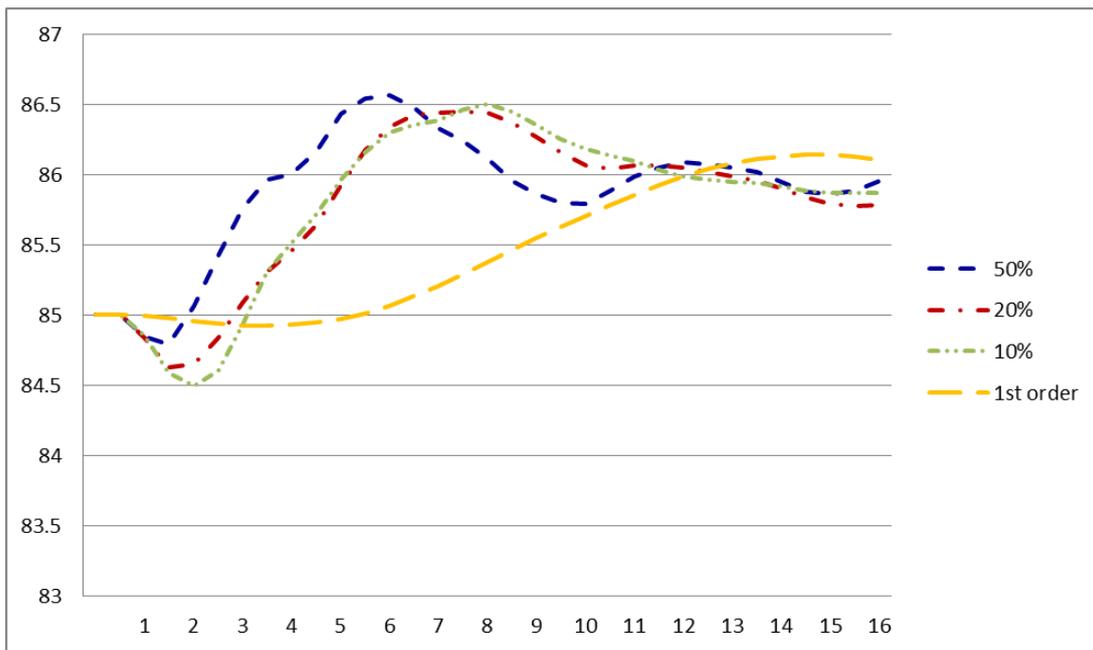


Figure 5.6 Reactor CV1 movement with model error

The data clearly shows that the more aggressive tuning has much more of a tendency to start a cycle on the process, with the less aggressive tuning not cycling at all.

Even with no model error, the distillation controller already had performance issues caused by ill-conditioning as shown in section 5.1.1. All gains for the bottoms temperature on the distillation simulator were increased by 100%. This further hampered controller performance as shown in figures 5.7 and 5.8.

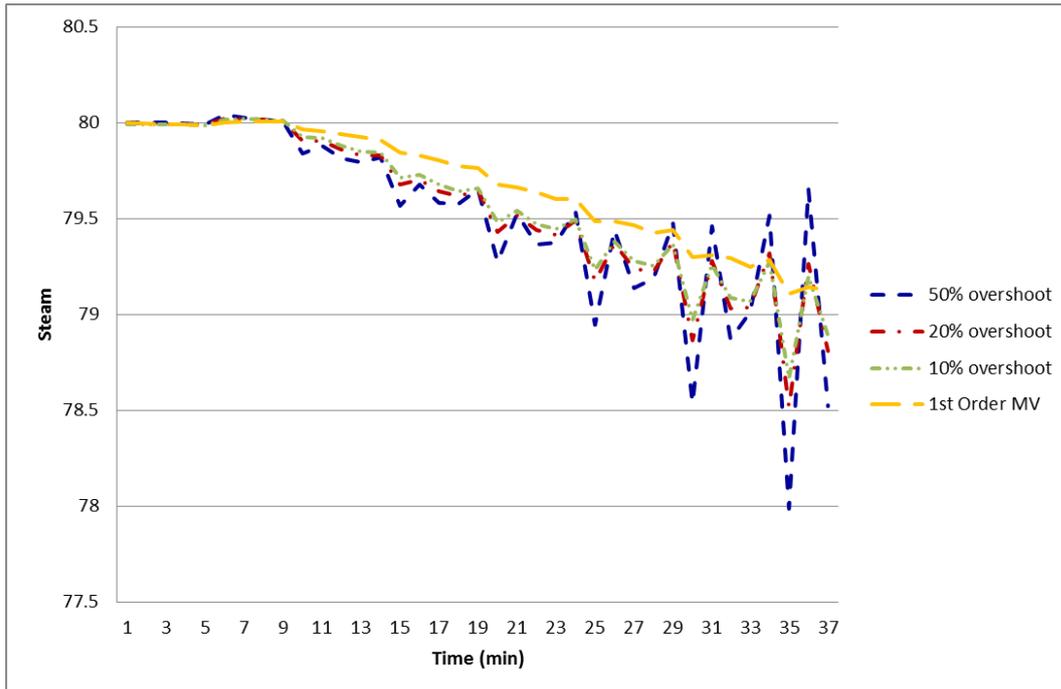


Figure 5.7 Distillation plant steam movement with model error

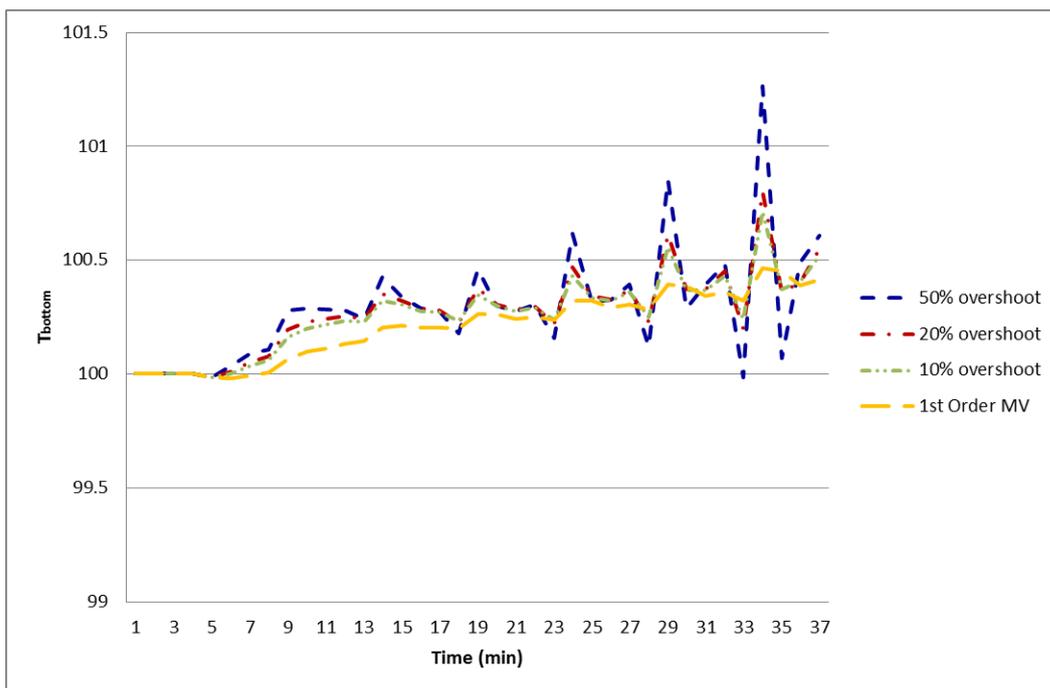


Figure 5.8 Distillation plant  $T_{bot}$  response with model error

Once again the slower tuning constants showed less of a tendency to cause unstable behaviour.

## 5.2. Comparison with traditional tuning

The initial tuning values that the engineers decided on using trial and error were compared with the results obtained from applying the different metrics. The different results are shown in table 5.2 and 5.3.

Table 5.2 Results of different tuning methods for Distillation plant

	50% MV Overshoot	20% MV Overshoot	10% MV Overshoot	1st Order MV Response	Engineer 1	Engineer 2	Engineer 3	Engineer 4	Engineer 5	Engineer 6	Engineer 7	Engineer 8
SupMov1	1.2	2.5	3.4	17.6	10.0	10.0	7.0	100	3	0.02	1	1
SupMov2	9.2	14.4	16.7	442	1.0	5.0	8.0	5	10	0.25	0.1	1
SupMov3	7.9	14.2	18.4	58.2	5.0	5.0	5.0	5	20	0.25	0.1	5

If the calculated move suppressions are compared to the tuning values chosen by the engineers, it is seen that the tuning method found more conservative values than the engineers. The values chosen by most of the engineers would cause no overshoot in the first manipulated variable and very large overshoot in the second and third manipulated variables.

This is in line with the experience of the authors, where most engineers are uncomfortable with move suppression values that are not between 1 and 20. It is also common in most projects for engineers to choose aggressive tuning values at first, and then detune the controller during commissioning.

Table 5.3 Results of different tuning methods for Reactor

	50% MV Overshoot	20% MV Overshoot	10% MV Overshoot	1st Order MV Response	Engineer 1	Engineer 2	Engineer 3	Engineer 4	Engineer 5	Engineer 6	Engineer 7	Engineer 8
SupMov1	1.8	3.0	3.8	57	6.0	2.5	5.0	3	5	6	4	8.5
SupMov2	9.6	11.4	12.8	100	12.0	5.0	6.0	3	12	12	1	14
SupMov3	11	13.5	15.6	76	15.0	5.0	4.0	3	12	12	1	15
SupMov4	6.6	8.8	9.7	404	5.0	2.0	3.0	3	6	6	1	11
SupMov5	1.0	1.8	1.9	9.8	1.0	2.0	5.0	9	7	7	1	20

As expected, the tuning method results show less aggressive behaviour as the amount of manipulated variable overshoot is decreased. Tuning for a first order movement in the manipulated variables yields much larger move suppressions.

On the reactor controller the move suppressions chosen by the more experienced engineers were much more in line with those chosen by the method while the engineers with less than 5 years experience once again chose much more aggressive tuning values.

## 6. Conclusion

Historically the tuning of multi-variable controllers or DMC has been a matter of personal taste of the engineer and trial and error methods were used to tune controllers that influence multi-million dollar processes or can have an impact on environmental and safety issues. The problem is compounded by the lack of agreement of what acceptable controller behaviour is, with the level of aggressiveness of controller tuning depending on the judgement of the engineer. This also means that comparing different tuning constants is vague and unscientific.

Using the shape of the predicted manipulated variable move plan or the shape of the controlled variable response as an indication of tuning aggressiveness addresses this problem. Using the amount of manipulated variable overshoot as an indication of tuning aggressiveness provides satisfying results. The amount of overshoot can vary from zero if a first order path is chosen, to any positive value chosen by the engineer. Using a solver algorithm to find the desired amount of overshoot in a manipulated variable can be used to find good starting values for tuning parameters.

Using the amount of manipulated variable overshoot is a valid way to characterise aggressiveness of tuning. By using it as a metric it is possible to compare different tuning values.

Even though the different metrics that have been introduced have the potential to enable APC practitioners to compare the aggressiveness of tuning parameters, this will by no means close the book on the subject. Other major considerations when choosing move suppressions are:

- Conditioning number of the controller matrix
- Quality of process models
- Non-linear process responses
- Severity of disturbances
- Consequences of controlled variable limit violation.
- The number of controller execution cycles from the time when a disturbance manifests and when the effect of the disturbance causes unacceptable behaviour in the controlled variables.

Taking these factors into consideration, it is recommended that the APC engineer obtains move suppression values that will provide 20% manipulated variable overshoot and tuning that will provide a first order move plan for the manipulated variables. Interpolation between these values can be done to use 20% manipulated variable overshoot for aggressive tuning, and moving towards the first order manipulated variable moves if less aggressive tuning is required if issues like ill-conditioning or model uncertainty exist. As per best practises, these tuning values will have to be refined by observing the actual controller response during the project commissioning phase.

The value in using manipulated variable overshoot is not that it will provide ideal tuning values for a live controller. It will firstly provide a way to characterise aggression of tuning, and a way to compare different sets of tuning values. Secondly it will provide a good set of initial tuning values for online controllers that will save

the engineer time and process upsets during commissioning. Thirdly it is much faster in getting acceptable initial tuning values than the traditional trial and error approach, yielding results in minutes rather than hours.

## Appendix A – Plants and models used in tuning exercises.

### Reactor simulation

The plant consists of 4 continuous stirred tank reactors in series, or one reactor with 4 chambers, with a preheated feed undergoing an exothermic reaction. Each of the 4 chambers has a cooling water coil with an associated flow controller. The feed flows into the first chamber, and from there overflows into the next. Product is let out of the last chamber under level control.

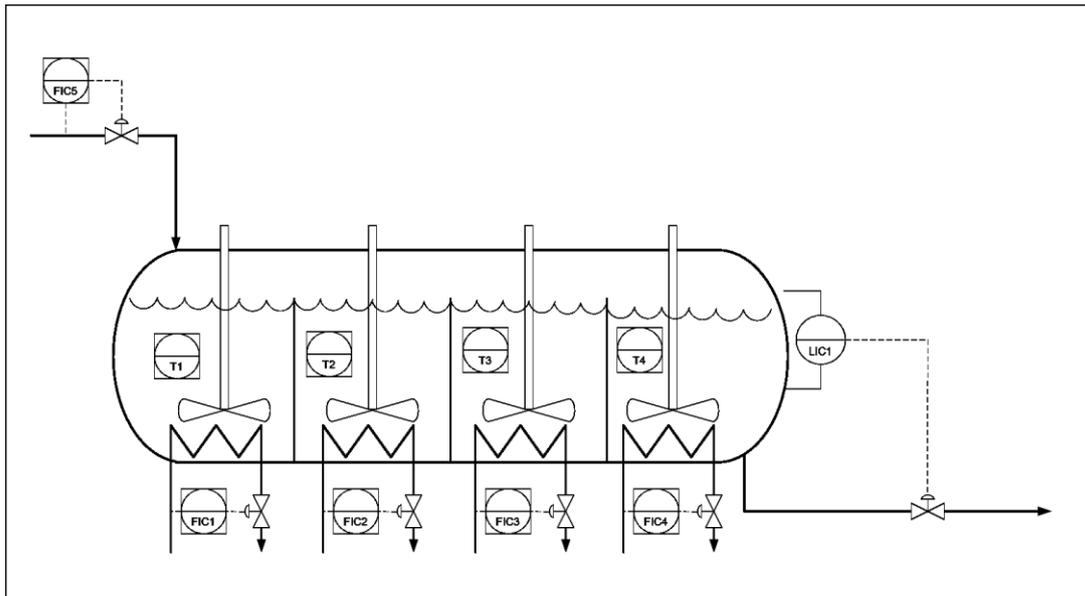


Figure C1 Reactor process flow diagram

An increase in feed into the reactor increases the exotherm.

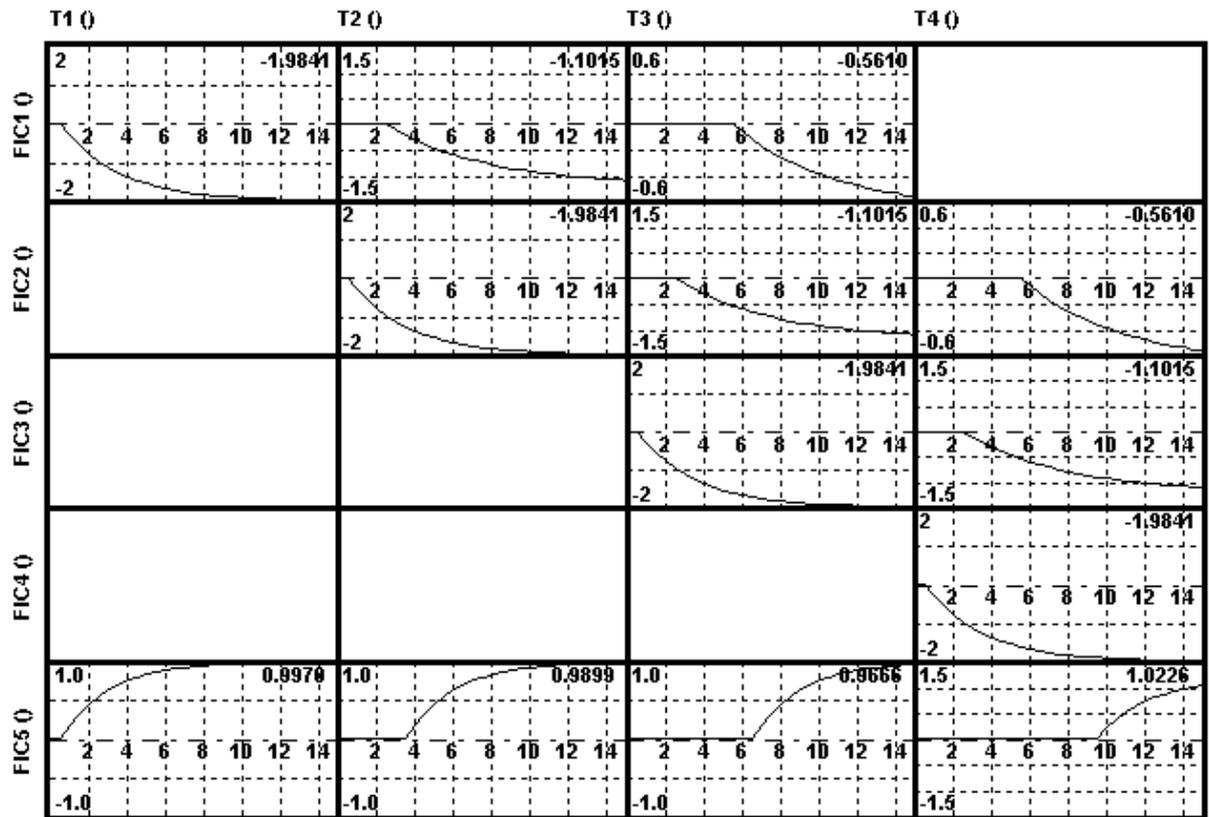


Figure C2. Reactor models

### Distillation Simulation

This is a normal distillation column with 3 MVs and 3 CVs. MVs are feed, steam and reflux cooling water. CVs are top and bottom temperature, and column dP. The model is shown below.

This model was specifically chosen because of the ill conditioning that exists between the top and bottom temperature and the feed and steam manipulated variables.

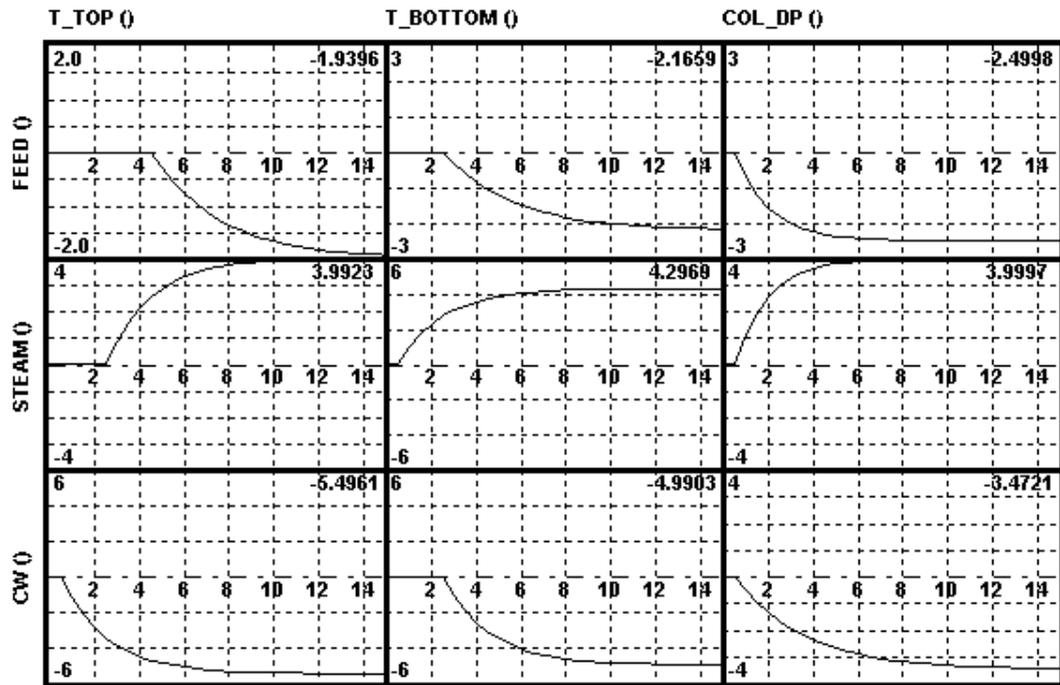


Figure C3 Distillation plant models

## References

- Chien, I.L. (1988) IMC-PID controller design – an extension. *Proceedings of the IFAC Adaptive Control of Chemical Processes Conference, Denmark*, pp.147-152
- Chien, I.L. and Fruehauf, P.S. (1990) Consider IMC tuning to improve controller performance. *Chemical Engineering Progress*, 86, pp.33-41
- Cutler, C.R., Haydel, J.J., Morshedi, A.M. (1983) An Industrial Perspective on Advanced Control *Annual Meeting - American Institute of Chemical Engineers*
- Dahlin, E.B. (1968) Designing and tuning digital controllers, *Instruments and Control Systems*, 2(6), pp.77-83
- Ghazzawi, A., Ali, E., Nouh, A. and Zafirou, E., (2010) On-line tuning strategy for model-predictive controllers, *Journal of Process Control 11*, pp.265-284
- Iglesias, E.J., Sanjuán, M.E. and Smith, C.A (2006) *Ingenaria & Desarrollo*, Vol. 19, pp. 88-100
- Kai Han, Jun Zhao, Jixin Qian (2006) A Novel Robust Tuning Strategy for Model Predictive Control *Proceedings of the 6th World Congress on Intelligent Control and Automation, June 21 - 23, 2006, Dalian, China*
- King, M. (2011) *Process Control A Practical Approach*, Wiley, pp. 60-65
- Nelder, J.A. and R. Mead, “A Simplex Method for Function Minimization”, *The Computer Journal* (1965) 7 (4): pp. 308-313.
- Qin, S.J. and Badgwell, T.A. (2003) A survey of industrial model predictive control technology *Control Engineering Practise 11*, pp. 733-764
- Prett, D. M., & Gillette, R. D. (1980). Optimization and constrained multivariable control of a catalytic cracking unit. *Proceedings of the joint automatic control conference.*
- Shridar, R. and Cooper, D.J. (1998) A novel tuning strategy for multivariable model predictive control, *ISA Transactions*, Vol. 36, No. 4. pp. 273-280