



# 22: Monitoring

*In the last article we looked at the assessment of a potential inferential. Here we cover techniques for monitoring its performance and automatically updating it to maintain its accuracy*

**W**HILE we may have installed an effective inferential, this will not usually replace the existing quality measurement – whether it be laboratory sampling or an on-stream analyser. So, we can use these measurements to check the accuracy of the inferential and potentially correct it automatically.

First, let us consider the use of a laboratory result for the property  $Q$ . The issue is that the result is reported some considerable time after the sample is taken. To validate the inferential, we need to know its value at the time of sampling. While sample time is recorded in many LIMS (laboratory information management systems), it is often the scheduled rather than the actual time. Processes are rarely perfectly at steady state and so comparison between laboratory and inferential becomes unreliable. This, of course, is an issue when developing an inferential. However, the difference is that in using a large number of records, such time-stamping errors are averaged close to zero. While they will cause a reduction in  $R^2$  they will have less effect on accuracy. The problem arises if today's result is very different from the installed inferential. The solution, of course is reliable time-stamping. This is commonplace in highly regulated processes, such as pharmaceuticals. Sampling points include a limit switch which is activated when a sample is taken. The time is logged automatically and a scannable sample label printed – which also helps avoid sample mix-ups.

Traditionally, a bias term in the inferential is updated to force it to agree with the latest laboratory result. However, we recognise that the laboratory is also prone to error and so we take a cautious approach. We introduce the parameter  $K$ , typically set to around 0.35, so that the correction ramps in over several samples:

$$E = Q_{inferential} - Q_{laboratory}$$

$$bias_{n+1} = bias_n - KE_n \quad \text{where } 0 \leq K \leq 1$$

Fortunately, much of the industry now recognises this approach is flawed. Any random error in  $Q_{laboratory}$  will be added to the *bias* and so appear in the corrected inferential. The variance of the error will therefore be increased by the factor

## QUICK READ

- **Inferential Accuracy and Bias Correction:** Inference should be validated using quality measurements, but discrepancies arise due to time-stamping inaccuracies and laboratory errors. A cumulative sum of errors (CUSUM) approach helps distinguish bias from random errors, improving inferential accuracy
- **Dynamic Compensation for Analyser Updates:** On-stream analysers and inferences have different response times. Dynamic compensation techniques, such as deadtime and lead-lag algorithms, align their measurements to improve inferential performance and update reliability
- **Handling Complex Process Dynamics:** Inferences must be designed with dynamic behavior in mind. In cases where inputs respond at different rates to disturbances, reducing the number of variables can simplify dynamics and improve control system performance

$(1 + K^2)$ , so reducing  $\phi$  – recalling this is defined as:

$$\phi = 1 - \frac{(\sigma_{error}^2)_n}{(\sigma_{property}^2)_{n-1}}$$

Clearly unsuitable for random error, this technique is important in dealing with bias errors. Such might occur if there is a change to the process – for example, in feed composition or catalyst activity. So how do we separate bias error from random error?

Table 1 records 20 consecutive laboratory results alongside the value of the inferential at sample time. The final column is the cumulative sum of errors (CUSUM). This is plotted as Figure 1. If the error were random then the CUSUM trend would be a noisy horizontal line. The slope of the trend is the bias error. In this example, this was determined using the last six records and applied as a correction to the inferential's bias term. While waiting for six results might seem an excessive delay in applying an update, the method it replaces (with  $K$  set to 0.35) would have implemented only 92% of the correction.

**Table 1: Laboratory versus inferential**

SAMPLE	INFERENTIAL	LABORATORY	ERROR	CUSUM
1	5.08	4.81	0.27	0.27
2	4.97	4.79	0.18	0.45
3	4.93	5.25	-0.32	0.13
4	5.05	5.02	0.03	0.16
5	5.20	4.86	0.34	0.50
6	5.55	4.96	0.59	1.09
7	5.22	5.08	0.14	1.23
8	5.52	5.17	0.35	1.58
9	5.56	4.98	0.58	2.16
10	5.56	4.90	0.67	2.82
11	5.64	4.86	0.78	3.61
12	4.80	4.98	-0.18	3.43
13	5.16	4.94	0.23	3.65
14	4.95	5.17	-0.22	3.43
15	4.93	5.01	-0.09	3.35
16	4.95	5.17	-0.22	3.13
17	5.17	5.09	0.08	3.21
18	5.17	5.16	0.01	3.22
19	5.16	4.75	0.41	3.63
20	4.84	4.81	0.03	3.66

And it's likely that we could make the correction after fewer samples.

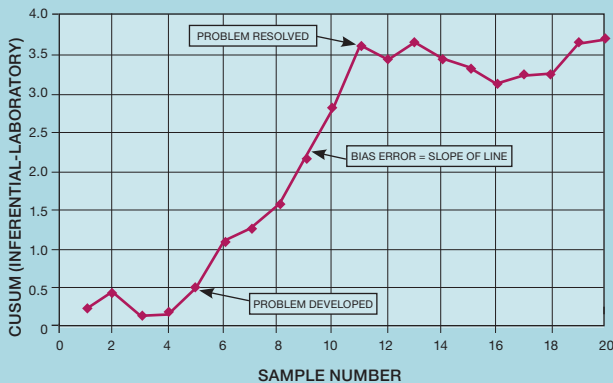
Without going into the proof, for what might seem an obvious result, if we use the slope of the CUSUM for the last three samples:

$$bias_{n+1} = bias_n - K(0.5E_n + 0.5E_{n-1})$$

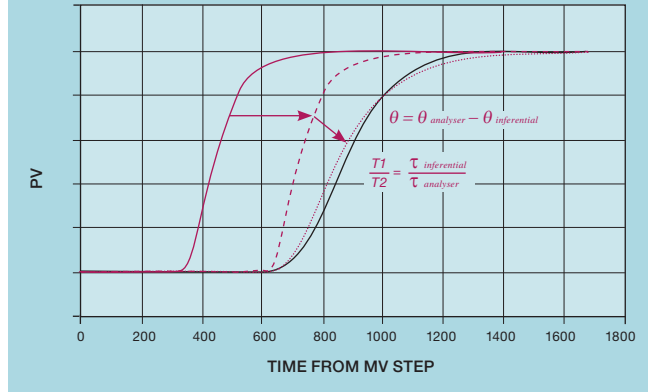
Less obviously, using the last four samples:

$$bias_{n+1} = bias_n - K(0.3E_n + 0.4E_{n-1} + 0.3E_{n-2})$$

**Figure 1: Use of CUSUM**



**Figure 2: Dynamic compensation**

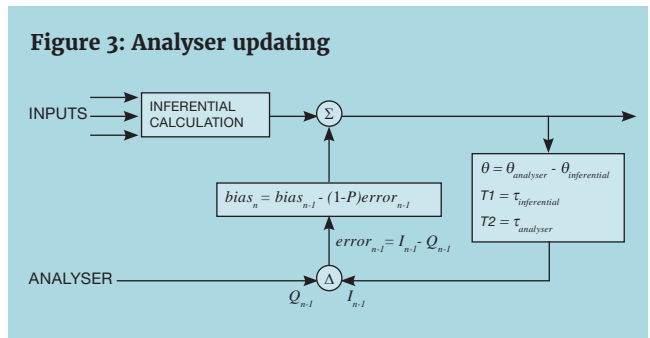


Inclusion of the term  $K$  may be unnecessary – certainly its value can now approach 1. In fact, its value is better optimised as part of the regression analysis. Note that, in determining future corrections, the bias correction also has to be applied retrospectively to those predictions which will be used to determine the next correction.

## ON-STREAM ANALYSER

If installed, we can also use an on-stream analyser to update the inferential. While the inferential and the analyser might agree at steady state, they will not do so during a disturbance. This is because the dynamic response of the inferential will be faster than that of the analyser (otherwise the inferential has little purpose!). We could wait until steady state is reached before updating but a better approach is to apply dynamic compensation. We covered the technique as part of our article on feedforward control (see TCE 999). We step-test to obtain the dynamics of both analyser and inferential. As shown in Figure 2, using a deadtime and lead-lag algorithm delays the inferential measurement so that it has the same dynamics as the analyser. Figure 3 shows the configuration. It includes a filter parameter ( $P$ ). For continuous analysers this can be set close to 1. Discontinuous analysers produce a staircase trend, so a

**Figure 3: Analyser updating**



value of 0.7 or less is advisable to prevent this adversely affecting the inferential. Or a better approach is to use the analyser read-now contact to trigger an update.

While we must take account of analyser dynamics when monitoring and updating an inferential, they also influence the precision to which we can develop an inferential. One approach is to be sure that the data are collected at steady state. This may limit the number of records available and might miss those occasions where the process is away from target – so reducing data scatter. There is, however, a simple way of including dynamics in the regression analysis. Considering, first, a single-input inferential, we regress the equation:

$$y_n = a_0 + a_1 y_{n-1} + a_2 x_{n-(\theta/ts)} + a_3 x_{n-(\theta/ts)-1}$$

This equation is effectively the same as the FOPDT (first order plus deadtime) model that we developed in TCE 981. To apply it, imagine we have a spreadsheet with the dependent variable  $y$  in the first column and the independent variable  $x$  in the second. We first insert a column between the two and copy into it the values of  $y$  – displaced downwards one row. The column will

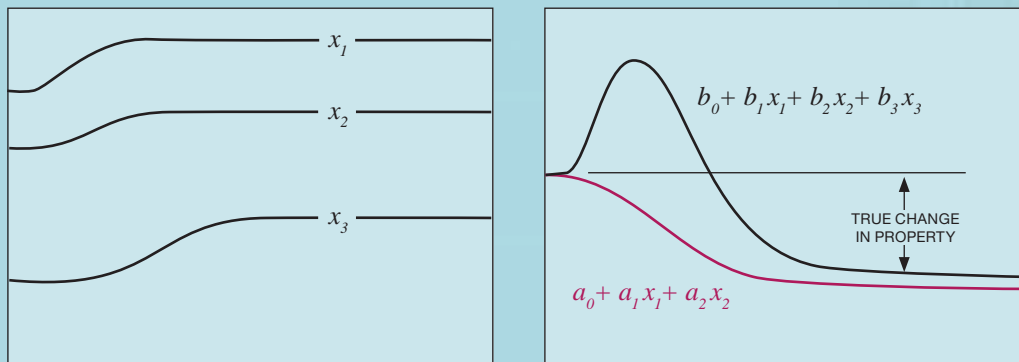
now contain  $y_{n-1}$ . We'll show later that this helps us identify the lag ( $\tau$ ). To obtain the deadtime ( $\theta$ ), we similarly copy  $x$  into the next column, displaced by one row, to give  $x_{n-1}$ . We repeat this for several columns to include  $x_{n-2}, x_{n-3}$  etc – adding enough columns to cover the likely deadtime. We then delete any incomplete rows from the beginning and end of the spreadsheet.

We use regression to identify the best three-input inferential. If there is a clear dynamic model, the best inputs will be those in the equation above. In addition to  $y_{n-1}$ , it should include values of  $x$  that are one data collection interval ( $ts$ ) apart. These allow for  $\theta$  not being an exact multiple of  $ts$  (if  $a_3$  is close to 0, then it is). The values of  $\theta$  and  $\tau$  are not required for the inferential itself but will be of use when we later install analyser updating. They are calculated from:

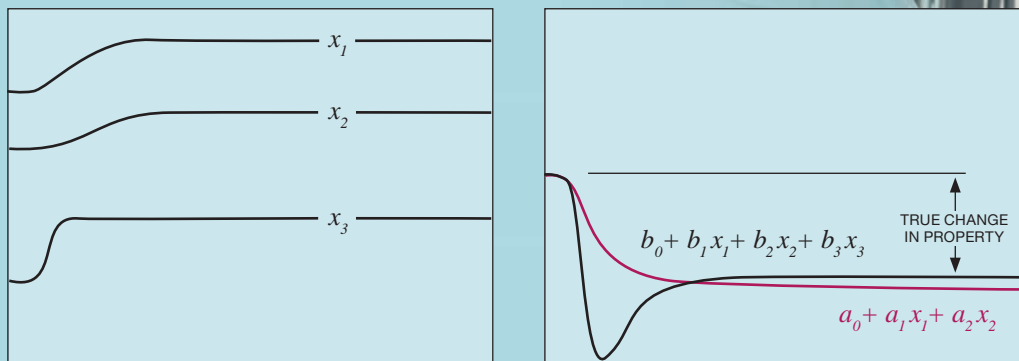
$$\theta = \left[ \left( \frac{\theta}{ts} \right)_{best\ fit} + \frac{a_3}{a_2 + a_3} \right] ts \quad \tau = \frac{-ts}{\ln(a_1)}$$

While we include dynamic compensation when developing the inferential, we do not in the installed inferential. We want it to give the earliest possible indication of any change, not have the

**Figure 4: Inferential input dynamics (case 1)**



**Figure 5: Inferential input dynamics (case 2)**



same dynamics as the analyser. The implemented inferential would therefore be:

$$y = \frac{a_0}{1-a_1} + \frac{a_2+a_3}{1-a_1}x$$

This technique can equally be applied when developing an inferential that includes more than one independent variable – but only if the dynamics are similar between the analyser and each input. If they are not, then consideration should be given to reducing the number of inputs.

### DIFFICULT DYNAMICS

Figure 4 shows a near-perfect inferential, in that it accurately predicts the property at steady state. But its dynamic behaviour would give considerable controller tuning problems. The inverse response is the result of  $x_3$  changing sometime after the other inputs. For example, a disturbance at the top of a distillation column would more quickly affect tray temperatures nearer the top than one nearer the bottom. In theory it would be possible to lag the other inputs to match those of  $x_3$  but is unlikely to be practical. Figure 5 shows the same inferential responding to a different disturbance. This might be at

the bottom of our column and so affect  $x_3$  first. The inferential now shows very different dynamics. Since we don't know the source of the disturbance, or if there are several occurring simultaneously, dynamic compensation is not practical. In this case, we would sacrifice some accuracy by omitting  $x_3$  from the inferential. As the figures show, this now shows much simpler dynamic behaviour. ■

### NEXT ISSUE

In the next issue we'll present a number of examples of inferential development, aimed at illustrating some of the key issues.

*Myke King CEng FICHEME is director of Whitehouse Consulting, an independent advisor covering all aspects of process control. The topics featured in this series are covered in greater detail in his book Process Control – A Practical Approach, published by Wiley in 2016*

*Disclaimer: This article is provided for guidance alone. Expert engineering advice should be sought before application.*

Registration now open



# Advances

in the Digitalisation of the Process Industries

16–17 October 2025, Manchester, UK

Advances is the must-attend event for industry professionals applying and working with digital technologies and techniques. Join the experts and connect with peers to explore best practice, emergent thinking, and future implications of a dynamic and rapidly changing technology environment.

Hear from an international line-up of high-profile plenary speakers on the following programme themes:

- AI & machine learning
- Digital twins & IIoT
- Advanced automation
- Data integration & risk management

Find out more and register: [www.icheme.org/advances](http://www.icheme.org/advances)

ICHEME