1

Introduction

In common with many introductions to the subject, process control is described here in terms of layers. At the lowest level is the process itself. Understanding the process is fundamental to good control design. While the control engineer does not need the level of knowledge of a process designer, an appreciation of how the process works, its key operating objectives and basic economics is vital. In one crucial area his or her knowledge must exceed that of the process engineer, who needs primarily an understanding of the *steady-state* behaviour. The control engineer must also understand the *process dynamics*, i.e. how process parameters move between steady states.

Next up is the field instrumentation layer, comprising measurement transmitters, control valves and other actuators. This layer is the domain of instrument engineers and technicians. However the control engineer needs an appreciation of some of the hardware involved in control. He or she needs to be able to recognise a measurement problem or a control valve working incorrectly and must be aware of the accuracy and the dynamic behaviour of instrumentation.

Above the field instrumentation is the DCS and process computer. These will be supported by a system engineer. It is normally the control engineer's responsibility to configure the control applications, and their supporting graphics, in the DCS. So he or she needs to be well-trained in this area. In some sites only the system engineer is permitted to make changes to the system. However this does not mean that the control engineer does not need a detailed understanding of how it is done. Close cooperation between control engineer and system engineer is essential.

The lowest layer of process control applications is described as *regulatory control*. This includes all the basic controllers for flow, temperature, pressure and level. But it also includes control of product quality. Regulatory is not synonymous with basic. Regulatory controls are those which maintain the process at a desired condition, or SP, but that does not mean they are simple. They can involve complex instrumentation such as on-stream analysers. They can employ 'advanced' techniques such as signal conditioning, feedforward, dynamic compensation, overrides, inferential properties etc. Such techniques are often described as *advanced regulatory control (ARC)*. Generally they are implemented

Process Control: A Practical Approach Myke King

within the DCS block structure, with perhaps some custom code, and are therefore sometimes called 'traditional' advanced control. This is the domain of the control engineer.

There will be somewhere a division of what falls into the responsibilities between the control engineer and others working on the instrumentation and system. The simplistic approach is to assign all hardware to these staff and all configuration work to the control engineer. But areas such as algorithm selection and controller tuning need a more flexible approach. Many basic controllers, providing the tuning is reasonable, do not justify particular attention. Work on those that do requires the skill more associated with a control engineer. Sites that assign all tuning to the instrument department risk overlooking important opportunities to improve process performance.

Moving up the hierarchy, the next level is *constraint control*. This comprises control strategies that drive the process towards operating limits, where closer approach to these limits is known to be profitable. Indeed, on continuous processes, this level typically captures the large majority of the available process control benefits. The main technology applied here is the *multivariable controller* (*MVC*). Because of its relative ease of use and its potential impact on profitability it has become the focus of what is generally known as *advanced process control* (*APC*). In fact, as a result, basic control and ARC have become somewhat neglected. Many sites (and many APC vendors) no longer have personnel that appreciate the value of these technologies or have the know-how to implement them.

The topmost layer, in terms of closed loop applications, is *optimisation*. This is based on key economic information such as feed price and availability, product prices and demand, energy costs etc. Optimisation means different things to different people. The planning group would claim they optimise the process, as would a process support engineer determining the best operating conditions. MVC includes some limited optimisation capabilities. It supports objective coefficients which can be set up to be consistent with process economics. Changing the coefficients can cause the controller to adopt a different strategy in terms of which constraints it approaches. However those MVC based on linear process models cannot identify an unconstrained optimum. This requires a higher fidelity process representation, possibly a rigorous simulation. This we describe as *closed-loop real-time optimisation* (*CLRTO*) or more usually just *RTO*.

Implementation should begin at the base of the hierarchy and work up. Any problems with process equipment or instrumentation will affect the ability of the control applications to work properly. MVC performance will be restricted and RTO usually needs to work in conjunction with the MVC. While all this may be obvious, it is not necessarily reflected in the approach that some sites have towards process control. There are sites investing heavily in MVC but which give low priority to maintaining basic instrumentation. And **most** give only cursory attention to regulatory control before embarking on implementation of MVC.