This section discusses some of the advanced software packages that are embedded in modern distributed control systems (DCS). They serve control, estimation, monitoring, simulation, and tuning functions. For the advanced control strategies that are not discussed in this section, refer to the following sections in this volume:

- Section 2.20: “Optimizing Control”
- Section 2.33: “State Space Control”
- Section 2.34: “Statistical Process Control”

For in-depth discussions of auto-tuning, fuzzy logic, and model predictive control (MPC), which are also discussed in this section, the reader is referred to the following sections:

- Section 2.13: “Model-Based Control”
- Section 2.14: “Model-Based Predictive Control”
- Section 2.16: “Modeling and Simulation of Processes”
- Section 2.17: “Model Predictive Control and Optimization”
- Section 2.18: “Neural Networks for Process Modeling”
- Section 2.19: “Nonlinear and Adaptive Control”
- Section 2.22: “Self-Tuning Controllers”
- Section 2.31: “Fuzzy Logic Control”
- Section 2.38: “Tuning by Computer for Optimizing”

INTRODUCTION

The computational requirements of software products for advanced control have traditionally limited their implementation. For that reason, their associated software resided either in a host computer or in dedicated control units that were...
external to the basic DCS package and were “layered” onto the plant’s control system. This situation is changing, as some of the more advanced software packages can be embedded in the basic DCS system.

Recent advances in the processing power available within some DCS control systems have allowed them to include some advanced control capability. These embedded software packages include:

- Performance monitoring
- Loop tuning
- Fuzzy logic control
- Model predictive control
- Neural network-based variable estimation
- Control simulation

Most of the successful advanced control applications, such as MPC, that are in use today are used on high-throughput processes, where the savings justify the high cost of developing, installing, and maintaining these layered control applications. This new generation of advanced control is often designed so that the average process engineer can use these features with the same level of confidence as he or she used the more traditional controls. Fuzzy logic control, model predictive control, and property estimation may be implemented as a function block that can be assigned to execute in the standard redundant controllers.

This operating environment has improved security and performance in comparison to the traditional approach of leaving the advanced controls outside of the basic control system, which is referred to as “layering.” For this embedded implementation, the engineering tools for the configuration and troubleshooting of traditional control may also be used with these advanced software blocks.

Also, the standard dynamos of these blocks support operator displays that include advanced controls without the need to map parameters. Thus, the engineer and the operator can have consistent interfaces to both the advanced and the traditional control applications.

Embedding advanced control applications has many other advantages as well. Probably the most important is that the advanced control application software runs inside the embedded controllers where the applications can fully participate in controller redundancy, alarming, configuration downloads, and online upgrades. This also means that all parameter references are managed by the runtime and the configuration system, and such terms as “Mode” and “Status” have meanings consistent with the terminology of other software packages in the system. To the operator and other operations personnel, the advanced control applications are largely invisible — they are just other function blocks and parameters.

**PERFORMANCE MONITORING**

The maximum production rate and efficiency of a plant are determined by the process design and the capacity of the equipment used. However, many plants never reach the production capacity that is inherent in the plant design and equipment. In addition, there usually is little time to study the process operation to determine whether it could be improved because attention is concentrated on maintenance and on abnormal conditions that can become major sources of process disturbances.

To improve on this situation, some distributed control systems are now designed to support automatic and continuous monitoring and detection of abnormal operation of control and field devices. When control is based on the Foundation fieldbus architecture, the status associated with each function block output gives a direct indication of the quality of the measurement of control signal. Also, the actual and normal mode attributes of these blocks may be used to determine whether a function block is operating in its designed mode. Thus, by continuously monitoring the status and mode supported by fieldbus function blocks, it is possible to automatically determine the following abnormal conditions in control and input-output (I/O) function blocks:

- A condition exists that could be reducing the accuracy of the measurement.
- Control effectiveness is limited by a downstream condition.
- The block is not running in its designed mode of operation.

Based on the inherent accuracy of the sensor and its maximum error, it is possible to compute a variability index that compares the actual performance to the best achievable. These calculations can be built into the control and I/O function blocks to support control analysis. By this approach, the control performance of even the fastest processes can be evaluated.

Within the control system, the status and mode values are automatically processed to determine the percent of time when an abnormal condition existed during the current and previous hour, shift, or day. If this percentage exceeds a preset limit, then the condition is flagged as abnormal and the module containing the function block will show up in the monitoring interface. An example of an abnormal condition monitoring interface is shown in Figure 4.7a.

**CONTROLLER TUNING**

Loop tuning can be done either when needed (on demand) or continuously by adaptive tuning. On-demand tuning is initiated by the operating personnel. Continuous adaptive tuning is performed automatically after set-point changes or significant disturbances, or when noise is experienced. On-demand tuning is often used during startup to obtain approximate tuning constants during the initial commissioning of a control system. Adaptive tuning corrects the tuning in response to changes in process dynamics.
On-Demand Tuning

One of the most successful techniques for providing on-demand tuning is based on using the relay oscillation tuning method to identify the process ultimate gain and ultimate period. (For a detailed discussion of tuning by computer, refer to Section 2.38 and for an explanation of the determination of ultimate process gain and ultimate time period, see Figures 2.38c and 2.38i in that section.)

This “relay oscillation” tuning method can be used on both self-regulating and integrating processes but is not suited for pure dead time processes, which require sample and hold control. Once the ultimate gain and ultimate time period of a loop are known, the tuning rules (see Sections 2.35 to 2.38) may be automatically applied to obtain the initial controller settings.

The relay oscillation technique of tuning is performed in an open loop under two-state control. Each time the controlled variable (process output) crosses the set point or initial value of the controlled variable, the relay is switched. Relay action causes the loop to oscillate at its ultimate period, $T_u$. The ultimate gain, $K_u$, is determined as the ratio between the amplitude of the two-state controller output and the amplitude of the oscillations in the controlled variable.

Upon completion of the relay oscillation test, the autotuner calculates the ultimate gain, ultimate period, dead time, time constant, static gain, and integrating gain of the process. The user can input the type of process (only self-regulating or integrating), the type of tuning rules to use, and perhaps an additional tuning factor such as the closed-loop time constant. The tuning rules are then applied and the calculated settings are provided for the controller, as illustrated in Figure 4.7b.

Adaptive Tuning

Adaptive tuning can be used to tune both feedback and feedforward controllers. There are two ways to design an adaptive PID controller: direct and indirect or identifier based. An identifier-based approach is advantageous when model switching and parameter interpolation are used. In this case a set of models is available together with a switching strategy among them, and parameter interpolation is based on the integrated squared error assigned to every value of the model parameter.

The adaptation cycle continues through a declared number of scans or until there is enough excitation on the inputs. As soon as a model has been updated, controller redesign takes place based on the updated model parameters. Adaptation can proceed for the whole model or separately for the feedback or feedforward portions of the model. The external excitations can be injected into the feedback loop automatically. The applied excitations can be a small change of the set point or controller output in either the Manual or in the Automatic modes.

Since the process model obtained is first-order plus dead time, any tuning rules can be used, including applied, typically lambda or IMC tuning. A general adaptive PID controller structure with model parameter interpolation is shown in Figure 4.7c.

Fuzzy Logic Control

Many industrial processes operated by humans cannot be automated using conventional control techniques since the...
performance of these controllers is often inferior to that of the operators. One of the reasons is that linear controllers, which are commonly used in conventional control, are not appropriate for the control of nonlinear processes.

Another reason is that humans aggregate various kinds of information and combine control strategies, which cannot be integrated into a single analytic control law. The underlying principle of knowledge-based (expert) control is to capture and implement experience and knowledge available from experts (e.g., process operators).

One would usually consider fuzzy logic control only after all attempts at controlling the process by optimized PID

**FIG. 4.7b**
Operator’s display after “on-demand” tuning been performed. The lower line in the graph shows the oscillating step changes that are made in the manipulated variable (usually the control valve opening), and the upper line shows the response of the controlled variable.

**FIG. 4.7c**
The configuration of the blocks of software in an adaptive PID controller, which includes both feedback and feedforward parts.
controllers have failed. Because fuzzy logic control requires exhaustive amounts of process data, the other essential requirement is that the process be somehow manually controllable. For a detailed discussion of fuzzy logic control, refer to Section 2.31.

The concept of fuzzy logic originated in 1965 when Lofti Zadeh of the University of California at Berkeley proposed what he called “fuzzy set theory.” Theoretically, fuzzy logic-based controllers can provide significant performance improvements over conventional PID-based controllers. The nonlinearity introduced in the fuzzy PID controller can reduce overshoot significantly without compromising disturbance rejection capability, but fuzzy logic has not been widely adopted in the process industries.

One reason is that the tuning is difficult since no standard guidelines exist for the establishment of membership function scaling. To address this issue, many manufacturers provide preengineered versions of fuzzy control that require little user input to commission. In addition, manual tuning may be substituted by relay-based automatic tuning. Such controllers can be used as nonlinear single-loop replacements for conventional PID controllers.

Fuzzy logic controllers combine the concept of membership functions with rule inference (see Section 2.31 for definitions and detailed explanations). The rules are used to determine a controller output based on the input membership functions. Tuning is accomplished through adjustment of the scaling associated with the membership functions. Figure 4.7d illustrates the software block components of a fuzzy logic controller that has already been defined.

The fuzzy logic controller in many cases provides increased performance and improved response with little or no overshoot relative to PID control. According to some sources, the improvement can range from 20% on the basis of integral of absolute error (IAE) to almost 50% for integral of absolute error multiplied by time.

MODEL PREDICTIVE CONTROL

In internal model controllers (IMC), MPC, and pole-cancelation, lambda, and Dahlin controllers, the model used by the controller is based on the controlled process (see Sections 2.13, 2.14, 2.17, and 2.19 for details). (Editor’s note: According to some users, the MPC model relates to the inverse of the process model. Therefore, when the process dynamics are unchanged (set point changes), these controllers are effective; if the model is exactly known, they are better than PID. Therefore, people use them, when the process cannot tolerate set-point overshoots. Their main limitation is their sluggish response to changes in process dynamics (process load changes). In some applications it is the response to load upsets that is critical.)

Since the introduction of MPC in the 1980s until 2004, over 5000 plant sites have utilized this technology. Modern DCS software packages include embedded MPC control for small applications (models of eight inputs by eight outputs in size) as well as larger applications, which traditionally were handled externally to the DCS (layered MPC). Fast execution rates and ease of use in commissioning allow the new embedded MPC control to be used where previously only traditional control techniques based on PID control had been applied. Improvements in the user interface and in the tools for commissioning and operating the MPC allow some experienced process or control engineers to apply this technology without the assistance of outside experts.

**FIG. 4.7d**
Application of preengineered fuzzy control.
In some simple MPC applications, the controller configuration might consist of only a few parameters. Examples of such parameters include:

- Controlled variable — Set-point limits
- Manipulated variable — Maximum rate of change, controller output limits
- Constraint parameter — Constraint high and low limits

The MPC controller is based on a dynamic model of the process. Traditionally, the dynamic model of the process is determined by manual manipulation of the control valve openings (process inputs). When the model is small enough for embedded MPC control, automated testing of the process is provided as part of the MPC engineering tools.

Once data on the dynamics of the process is collected, the process model and controller the MPC, can be generated automatically. Both the step response of the MPC model and of the actual process can be displayed. An example of the overview step response model and a detail of an individual output response for a particular input are shown in Figure 4.7e.

MPC identification tools compare the measured process variables with those calculated based on the model, if the manipulated variables to both the model and the actual process are the same. This comparison may be displayed in different formats to allow the user to determine how good the model is, i.e., how closely the model dynamics match those of the actual process.

A simulated environment is used to evaluate the control response to both set-point and load changes before the MPC control is commissioned. Using the model of the process for simulation, it is possible to see the actual process response in advance of real time. Similarly, on very fast processes, the ability to simulate the control and process response slower than real time can also be valuable. The ability to adjust the speed of execution allows the control response to be quickly verified for a variety of operating conditions. Figure 4.7f illustrates the testing of the MPC model by viewing the simulated controlled variable response of the MPC model to a change in the manipulated variable.

Once the MPC control has been tested offline using simulation, then the control can be commissioned.

(Editor’s note: If the process model accurately reflects the process, the control response to set-point changes is likely to be good. The response to load disturbances is usually poorer if the load disturbances result in substantial changes in process dynamics; in such cases, adaptive control should be considered as a possibly better option.)

NEURAL NETWORK APPLICATIONS

Artificial neural networks (ANNs) can learn complex functional relations by generalizing from a limited amount of training data. Hence they can thus serve as black-box models
of nonlinear, multivariable static and dynamic systems and can be trained by the input–output data of these systems. Applications that are suitable for ANN solutions and the detailed theory of ANN are discussed in Section 2.18. ANN is similar to fuzzy logic in that the mathematical model that relates the inputs and the outputs of the model does not need to be known. To use ANN, it is sufficient to know the behavior or response of the model. The main difference between fuzzy logic for property estimation and ANN is that the gains and functions cannot be modified in ANN; they can only be “trained” with data.

The use of estimated process variables can improve control and monitoring when critical measurements are only available through laboratory analysis. The intermittent laboratory analysis results can be used as inputs to provide online inferred or estimated variable values when actual analytical readings are not available. Neural networks are often used for process variable estimation even if the inferring relationships are nonlinear. ANN provides a nonlinear model of a process, which uses those input variables that indirectly influence the composition that is intermittently detected by laboratory analysis.

After a neural network has been commissioned, the offline laboratory analysis is still required to correct for unmeasured disturbances and changes in the process. Without this correction, the prediction provided by the model of the neural network will drift away from the true measurement value as time passes. For example, the laboratory analysis may be used as an input to the ANN model to obtain automatic adaptation of its prediction in response to changes in process.

Modern tools embedded in DCS control systems have reduced the complexity of designing and training neural networks. ANN can be integrated as a function block into a distributed control system and that block can be configured similarly to other function blocks. The tools provided to train the neural network take into account the dynamic nature of the processes for which a neural network is developed.

Once a potential application for a neural network is identified, the first step in the ANN development is to configure the function block. For example, as a first step in configuring the ANN block, the user may browse the past history of the process to identify the variables that tend to influence composition and use them as function block parameters. As actual laboratory analysis takes place, the results are provided to the ANN model for updating the model of that analytical variable that is being estimated by ANN (Figure 4.7g).

Once the neural network block is configured and downloaded, all measurements used in the ANN block are assigned to a historian for collection. After a sufficient amount of process data have been collected, the operator can graphically view the historic data and make corrections by adding/removing inputs. Once the input data have been screened, the ANN model can be generated. During this generation process, the dynamics of the input delays and output sensitivity to every input are calculated. To help the user determine the accuracy of the neural

FIG. 4.7f
The MPC model of the process is tested by introducing a change in the manipulated variable and viewing the resulting response of the controlled variable of the simulated process.

FIG. 4.7g
The MPC model of the process is tested by introducing a change in the manipulated variable and viewing the resulting response of the controlled variable of the simulated process.
network, the value calculated by the neural network may be compared to the lab analysis, as illustrated in Figure 4.7h.

**PROCESS AND CONTROL SIMULATION**

Process simulation packages are used to train operators prior to plant startup; they are also used to examine the dynamic response and performance of a process during operation, and finally, they are used to prepare the operators for complex and critical shutdown procedures.

The cost of developing the simulation software rises exponentially with the required accuracy of the process model. Yet, it is desirable that the model reflect not only the process dynamics but also the operation of the actually implemented control system and that it incorporate the simulation of process upsets to train the operators in a real-time setting. It is also important that the keyboard, strokes, sequences, mouse, and touch screen used during simulation be identical to the actual ones.

Some commercial systems allow all features of the process control system to be executed on a single PC platform or distributed between multiple computers. This high-fidelity process control simulation capability may be used to check out control system logic and operator interface as well as to train operators on the continuous and discrete control, diagnostic, and alarming under a variety of conditions.

The software and the associated configuration of a control system may be designed to be used in multiple operating environments without change. When this approach is taken,

---

**FIG. 4.7g**

Neural networks can be used to estimate analytical laboratory measurements on the basis of the past history of such readings by sending the associated laboratory readings to the ANN block. Later composition measurements from the laboratory are used to update the ANN model.

**FIG. 4.7h**

Comparison between the actual laboratory readings and the ANN-predicted values of composition.
it is possible to distribute control system functionality in a variety of ways without needing to reengineer the software or reconfigure the associated applications.

All features of the control system can be combined on a single platform or distributed between multiple PCs to support configuration and dynamic checkout of the controls and interface associated with a process control system, as illustrated in Figure 4.7i.

When a control system utilizes the Fieldbus Foundation function block architecture, a SIMULATE parameter is available in each I/O function block. Through this parameter, a value and status may be provided to I/O function blocks to simulate the field measurement or actuator in the control system.

This capability may be used to check out control logic and operator displays. Also, where the control systems support
the object linking and embedding (OLE) interface for processing control (OPC), the process simulation package may write the simulated value and status. This capability, combined with the ability to redistribute control functionality without reconfiguration, provides a strong foundation for the integration of high-fidelity simulation with control system simulation in a single or multi-PC environment.

Depending on the modeling rigors and computing resources, the process simulation can potentially run faster or slower than real time. To allow the control simulation to match the process simulation execution, the control system executes function blocks faster or slower. In addition, a trainer or an application may coordinate the execution of control and process simulation. An example interface for such coordination is shown in Figure 4.7).

CONCLUSION

Modern distributed control systems can provide embedded advanced control solutions. The DCS system supports the advanced control capability similarly to the support provided for the traditional control tools. Among the major advances in DCS system designs are the easy-to-use engineering and commissioning tools.

Reference


Bibliography


