Presenting Intelligent Iterative Control: PID Replacement for Setpoint Control

Implementing an intelligent-iterative-control algorithm

Editor's note: This is the second part in a three-part series.

By THOMAS HARTMAN, PE
Principal
The Hartman Co.
Marysville, Wash.

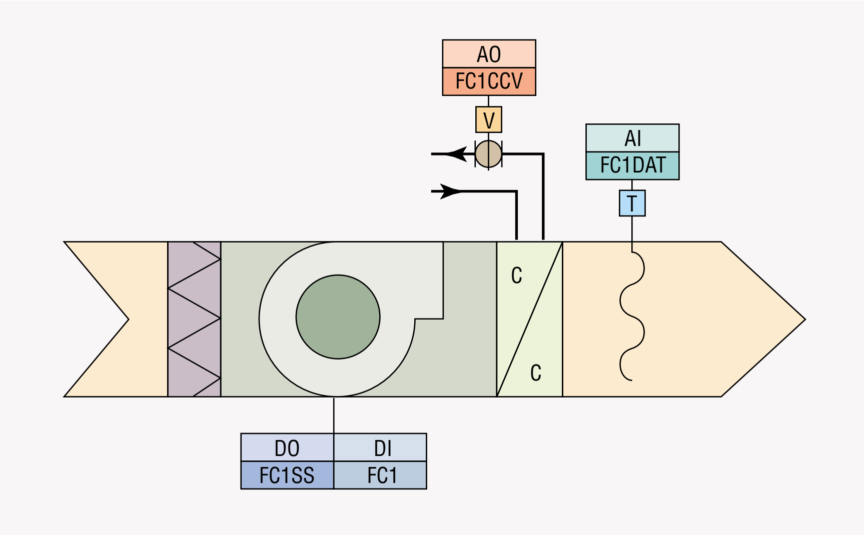

FIGURE 1. A simple cooling-only fan coil.

Though widely used to solve complex problems, iterative logic is well-suited to providing modulation control. Indeed, the iterative process can be easily applied to control applications.

For the fan-coil-valve-control application in Figure 1, assume the unit is operating, but the discharge-air temperature is deviating from the setpoint value. The iterative process of control involves: (1) estimating the change in valve position required to achieve the desired change in discharge-air temperature, (2) making the change, (3) waiting to see how close to the desired temperature the result is, and (4) adjusting the estimation logic (if necessary) and repeating the process. The process becomes "intelligent" when the estimate is enhanced by logic and real-time information derived from other elements of the fan coil and/or related systems. With an iterative-control scheme, control for the cooling valve in Figure 1 (FC1CCV) involves estimating and implementing a capacity-adjustment algorithm intended to bring the loop to its current setpoint. Because this is an iterative process, the algorithm employed to make the estimate does not need to be precise; however, it should be as accurate as possible and contain the primary data points needed to affect the desired change in valve position, along with a factor that correlates the relative importance of each.

A member of HPAC Engineering*'s Editorial Advisory Board, Thomas Hartman, PE, is an internationally recognized expert on the use of advanced building controls and control networks. Contact him at* tomh@hartmanco.com.

Among the substantial benefits of applying intelligent iterative control are that the intervals between valve repositionings can be much longer than those typically employed with PID control, and there is no need for interval times to be fixed. This has the potential to greatly reduce the frequency of required valve repositioning and, thus, extend valve and actuator life.

From an energy-performance perspective, a significant benefit of replacing PID control with intelligent iterative control is system head pressure can be reduced dramatically at all load conditions because valves can be line-sized and full-ported. With intelligent iterative control, there is no requirement for a linear response between valve actuation and the cooling effect it produces. To achieve the linear relationship required for PID control, valves must be undersized with significant pressure drops, while controllability requires only that the relationship between actuation and coil cooling output be a continuous curve and that the slope be positive at all times. As shown in Figure 2, these conditions are not difficult to meet with line-sized valves. A linear relationship as shown by the red line in Figure 2 can be established when the valve is undersized and consumes 25 to 50 percent of the total system head-pressure drop. Though not offering a linear relationship between actuation and coil cooling capacity, the line-sized valve in Figure 2 does meet the requirements for controllability.

The iterative logic used to effectively control a line-sized valve such as the one in Figure 2 is based on information coming from the fan coil and across the

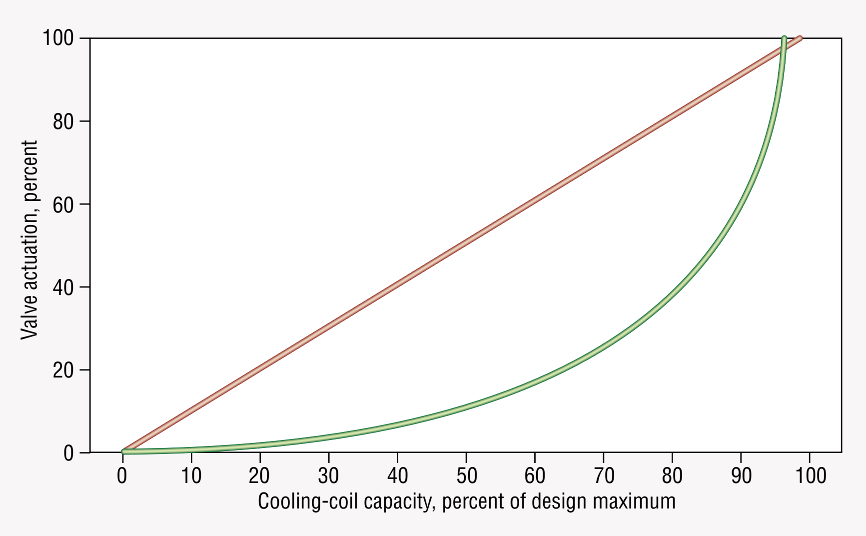

FIGURE 2. Actuation vs. coil capacity for undersized valve used in PID control (red line) and line-sized valve in intelligent-iterative-control application (green line).

network from other system elements.

During the design process, it is useful for designers to learn what factors are involved for each modulating control loop. A reasonably effective approximation of the incremental cooling effect required to meet a change in load is not difficult to approximate. When a line-sized valve is employed, and the chilled-water temperature is fixed for the application shown in Figure 1, the iterative-control algorithm often can be:

$$FC1CCV_{new} = FC1CCV + \left(C2 \div \left(1.0 + C1 \times \left(PumpRPM - 0.3\right)\right)\right) \times \left(FC1DAT - FC1DATSP\right) \times \left(C3 + FC1CCV\right)$$

where:

$FC1CCV_{new}$ = the estimated new valve position required to meet the load

FC1CCV = the current valve position

C2 = a constant that depends on the relative capacity of the cooling coil and the chilled-water lines that connect it. Initially, it can be set at 0.05. From there, it generally is adjusted between 0.02 and 0.10, with the lower values ensuring less potential for hunting and the higher ones ensuring faster response

C1 = a constant between 0.0 and about 20 that is adjusted depending on the proximity of each valve to the pump (C1 should be 0.0 for the load farthest from the pump)

PumpRPM = the speed of the distribution pump (100-percent speed = 1.0)

FC1DAT = the current fan-coil discharge-air temperature in degrees Fahrenheit

FC1DATSP = the current fan-coil discharge-air-temperature setpoint in degrees Fahrenheit

C3 = a constant developed to compensate for the slope of coil capacity vs. the actuation curve in Figure 2 as a function of valve position. The value of C3, which must be greater than zero, generally ranges between 0.1 and 0.5

Note that this algorithm calculates valve position as a fractional value (0.0 to 1.0) and uses pump speed as a fractional value. Some control systems provide these percentages as whole numbers (0 to 100) or other values. Constant and variable values may need to be scaled to be compatible with the control system employed.

As discussed earlier, one of the purposes of this type of control is the minimization of valve repositioning. For the fan-coil application shown in Figure 1, the minimum valve-repositioning interval typically is 30 sec. Following is the final form of a useful control algorithm for the fan coil in Figure 1:

Do every 30 sec:

$$A = \left(C2 \div \left(1.0 + C1 \times \left(PumpRPM - 0.3\right)\right)\right) \times \left(FC1DAT - FC1DATSP\right) \times \left(0.2 + FC1CCV\right)$$

If absolute value $\left(B\right) > 15\% \left(0.15\right)$, then $FC1CCV_{new} = A + FC1CCV$ and $B = 0$;
otherwise, $B = B + A$

where:

A and B = intermediate variables used in this algorithm only

This iterative process compensates for the fact that changes in pump speed may cause changes in pressure across the valve and, thus, change the valve movement necessary to attain the same change in cooling effect at different pump speeds.

This valve-control algorithm is simpler than many of the PID algorithms now in use. With some experience, intelligent-iterative-control algorithms are relatively easy to sketch out during design and can be easily inserted into sequences of operation. Initial values for C1, C2, and C3 usually can be estimated based on coil characteristics and the location of the fan coil in the distribution circuit. If a fan-coil supply fan is variable-speed, and a variable chilled-water-supply temperature is incorporated, additional factors can be implemented easily. Sometimes, it is helpful to incorporate into iterative algorithms self-learning, whereby constants are automatically set and adjusted to reduce startup-time requirements and the frequency of repositioning. Lastly, in critical systems, it sometimes is useful to adjust valve position for system-dynamics features, in addition to marginal cooling-load requirements. Such adjustments may be made independently of valve-position changes attributed to changes in load requirements. All of these controls are surprisingly easy to set up, as long as the digital controller has flexible control-programming capabilities.

Next month, the third and final part in this series will discuss limitations in designing with line-sized chilled-water valves and benefits of intelligent iterative control.