Which Tuning Method Should You Use?

by Tom Kinney and John Gerry, P.E. © 2003 ExperTune Inc.

There are many, many different tuning rules available. Quarter Amplitude Damping, Cohen-Koon, IMC, Trial-and-error, Lambda, Ziegler-Nichols, Shinskey, minimum IAE, to name just a few. Which one should you use?

Any tuning method will have proponents who push the merits of the method saying it guarantees this and that in terms of response and at the same time is robust and not very sensitive to the process changing. The proof is by actually looking at the response to **both** set point changes and load upsets **and** to compare the robustness of the tunings. There is always a tradeoff between speed of response and robustness or sensitivity to the process changing. Detailed explanation of robustness plots and how to use them is the subject of another presentation "Robustness Plots - The Other Side of the PID Tuning Story" available from the articles page at ExperTune.com.

Match the Tuning Objectives to What the Business Needs

However, first you need to match the strategy to the business objectives. Do you want good rejection of loads or good set point response? Do you want fastest response at the expense of overshoot and low robustness? Answer these questions for what the control loop should do, and pick the method to get the response and robustness you want. It is also important to look at the controller output response and see if what it does is OK for your process.

In many chemical plants and refinery processes there is a quality constraint. The graph to the right illustrates this showing profit vs. set point on this quality constraint. The green area in the graph is profitable, the red area, a loss. The vertical line in the middle of the graph represents a constraint in the process variable. Exceed this process variable and you start losing money - you have throw-away product. To make the most profit possible, you want your set point as close as possible to the constraint. However, to not exceed the constraint, you must keep a margin of roughly the standard deviation away from it. If control is poor then you must keep farther away. If stability is poor you will have to run even farther away from the constraint. If the loop is controlled better you can get nearer the quality constraint. In this type of loop, (with a quality constraint) tuning for minimizing the effects of upsets gets you the closest to the constraint and will make the most profit for your plant.

As an example, let's compare the response of 4 tuning methods:

1. Tuning for minimum integrated absolute error to load upsets. This method looks at the error between the set point and process variable after an upset. Its objective is to drive this error to the smallest amount. The set point response may overshoot, however.
2. Tuning for quarter amplitude damping. This is roughly approximate to the response Ziegler and Nichols ended up with.
3. Tuning for 10% overshoot on load upsets. This is a slower version of quarter amplitude damping.
4. Tuning optimized for set point response.

Above is a family of tuning responses. The process has a dead time of 5 seconds and lag time of 10 seconds. Derivative action is included in the tunings. The safety factor was set to 1 so you can see the true intended shapes of the responses. Detailed explanation of the safety factor is the subject of another presentation "Safety Factor: The Most Important Tuning Parameter" available from the articles page at ExperTune.com. For the set point tuning, the speed of response was set to get fastest possible response with the least amount of overshoot. It is important to view the tradeoff of response with robustness. As the tuning gets faster, the robustness goes down. From these plots you can also compare and contrast response to set points, load upsets and how the controller output responds. For example, it may be important that the controller output does not overshoot in a loop that responds differently up than down.

Here is a family of tuning responses on the same process but this time tuned with a safety factor of 2.5. The set point tuning is also backed off to be slower by a factor of about 2. Note that everything becomes slower, less oscillatory and more robust. The controller output is now a smooth response.

This is a family of tuning responses on a different process - this one having a dead time of 2.5 seconds and 2 large (large compared to the dead time) lags of 20 seconds each. Load tuning, tuned with a safety factor of 2.5 and again set point tuning backed off from optimal by a factor of 2. Note that the family changes quite a bit from the last process. One method of tuning may not be what you expect when you compare it on a different process. The best way to be sure, is to simulate the responses like we are doing here so you end up with a control loop that meets the business objectives of your plant.

Recommendations

In general, what we recommend using for a tuning method is either tuning for good rejection of load upsets or tuning for set point response. If you use the tuning for load rejection, adjust the speed and robustness with the safety factor. For the set point tuning, adjust the response time. In all cases, you should look at the simulated response to both load upsets and set point changes and contrast these to the robustness plots in the loop. Also note the shape and extent of the controller output. Tradeoff all these items to get the closest match to the business objectives for this loop. These tradeoffs and simulations are all made very easy by clicking the "Analysis" button (above the automatically found PID parameters) in ExperTune's PID Tuner software.

Simulating the time response and robustness of the process lets you make informed decisions about the best tuning method to use. Always keep in mind the business objectives of the plant. Why is the control loop there? For example, in temperature control you probably want no overshoot and good load response because temperature often reflects an index of product quality. On the other hand, with a pressure controller it is probably OK to have some overshoot to achieve the fastest possible response. All this is made easy with ExperTune's PID Tuner software.

No editing has been made except images. Original writing is available at http://www.expertune.com