A Closed Loop Deadtime/ Parameter Estimator for Self-Tuning Process Control

John D. Lane
Senior Application Engineer
Bailey Controls Company
Wickliffe, Ohio 44092

Presented at
American Controls Conference
Pittsburgh, PA
June, 1989

Bailey Controls
Babcock & Wilcox, a McDermott company
A Closed Loop Deadtime/Parameter Estimator

for

Self-Tuning Process Control

John D. Lane
Senior Application Engineer
Bailey Controls Company

Abstract

Processes with large deadtime to time constant ratios have traditionally proved difficult to control effectively. In practice, PID controllers (tuned to assure robustness at the expense of performance) are generally employed for the control of these processes. Conceptually better performance is achievable through deadtime compensation using a Smith Predictor or Model Based Controller. Unfortunately, these techniques are rarely used in practice due, in part, to unfamiliarity of technicians and operators with the tuning of these controllers. Recent work in self tuning control has had little impact on the control of large deadtime processes. This may be attributed to reliance on most self tuning controllers on the PID algorithm and the absence of techniques for the identification of process deadtime.

This paper describes a model based control algorithm that directly incorporates deadtime compensation and a novel closed loop process identification technique capable of identifying process deadtime directly. This combination provides an effective system for the control of large deadtime processes. The resulting controller is easy to implement and commission. Its auto tuning feature provides a simple mechanism for retuning the controller on a periodic basis. Extensive simulation results and preliminary test results are presented in addition to a brief description of the controller and auto tuner.

Introduction

The application of self tuning controllers to processes with large, unknown or time varying deadtimes, has been the goal of much research. A majority of research in this area has attempted to extend recursive least squares parameter estimation to include deadtime identification. Two particular methods, one employing a bank of estimators with different assumed values for the deadtime [1] and a second employing a single estimator to identify a set of parameters extended over the time horizon in the neighborhood of assumed deadtime [2], have received much attention. Unfortunately these techniques, along with other signal processing applications (see [3] for a survey), have proven impractical for general purpose application in an industrial setting.

Itakura [4] and Nishikawa, et al. [5] describe a technique for estimating process parameters using deadtime for systems operating in closed loop. This technique, henceforth referred to as a deadtime/parameter estimator (DPE), calculates time integrals of system variables reacting to a step or impulse input. Proper selection of these
integrals (centre areas) results in a set of non-linear equations which can be solved numerically. The DPE identifies the complete dynamic characteristic of processes, and provides an effective base for the implementation of an auto-tuning controller.

This paper describes an auto-tuning implementation using the DPE concept to tune a model-based controller.

First, the details of the model-based controller are reviewed.

Second, the derivation of the control area parameter estimator of the model-based controller is reviewed. A rigorous description can be found in [6].

Last, simulation results are presented to support the selection of the necessary test algorithm and illustrate the strengths and limitations of this technique.

**Model Based Controller Description**

The initial implementation of the DPE auto-tuner is based on a model-based controller. The model-based controller was selected over a more conventional PID controller for two reasons. The primary reason was the controller's ability to compensate for the effects of process time delay. This allowed it to be applied to a more general class of processes than the class of processes suitable for PID control. Another advantage of the model-based controller is the use of a separate parameter correction scheme to update parameters. In contrast, PID parameter selection requires closed-loop performance to be controlled in the selection of tuning parameters.

Figure 1 presents a block diagram of the model-based controller and its definition of transfer matrices.
This particular formulation assumes that the controlled process can be approximated by a first order lag with deadtime. The controller is tuned in terms of the observed process gain, time constant and deadtime. A fourth tuning parameter, the filter time constant specifies the desired time response to a change in setpoint or to a disturbance.

**Deadtime/Parameter Estimation Technique**

The use of the previously described model based controller requires knowledge of the dynamic characteristics of the controlled process. To extend the applicability of this system to processes with unknown or time varying dynamics, a feature based parameter estimation technique is employed. This technique has the advantage of estimating all process parameters, including deadtime, in closed loop. The technique selected for this work parallels that described by Bakura [4] and Nishikawa [5]. The basic idea is that certain quantitative measures of the system response to a step setpoint change can be directly related to the process parameters. Using various control areas involving the process output, one can develop and thus solve a linear equation in terms of the process parameters.

The following steps describe the procedure used for estimating the deadtime and process parameters of a first order linear system using the controller (1).

1. The maximum deviation of the process variable from setpoint is determined over a fixed time horizon along with the average value for the controller output. This information is used to determine when the process has reached steady state and to provide an initial value of the controller output.

2. The setpoint to the controller is increased (decreased) by a prescribed amount to start the estimation procedure.

3. The integrals in (2) and (3) are calculated numerically as the process responds to the setpoint change. Note that the values of both of these integrals converge as the process output approaches the new setpoint. Therefore, these integrals can be evaluated over the finite time horizon from time 0 to tss. Figure 2 illustrates the calculation of these integrals.

4. The process is considered to have achieved steady state conditions when the error between the process variable and the new setpoint is less than twice the maximum deviation measured in step 1 or a fixed amount of time.

5. The final value of the controller output is calculated by averaging over a fixed time horizon. This minimizes the effects of process noise on the test.

After observing the process response to a step setpoint change, 4 and 5 are used to estimate the process parameters. Clearly $r_{est}$ and $T_{est}$ cannot be computed directly in these equations. Fortunately, they have been obtained by solving the exponent in the form of the successive substitution technique and starting from an initial $r_{est}$ of 0.

\[
1_{1} = \int_{0}^{\infty} \left[ m_{ss} \right]^{t} dt
\]

\[
1_{2} = \int_{0}^{\infty} \left[ m_{ss} \right]^{t} e^{at} dt
\]
\[ k_{est} \]
\[ m_{ss} \]
\[ m_{init} \]
\[ r_{est} \]
\[ a_{est} \]
\[ b_{est} \]
\[ c_{est} \]
\[ \tau_{est} \]
\[ z + ln \]
\[ a \]
\[ c \]
\[ \tau_{est} \]

where \( I_1, I_2 \) Control Area Integrals

\[ m_{ss} \] Final Value, Control Output
\[ m_{init} \] Initial value, Control Output
\[ m(t) \] Control Output at time \( t \)
\[ SP \] Change in Setpoint

\[ k_{est}, r_{est}, \tau_{est} \] Estimated Parameters

\[ k_m, \tau_m, \tau_f \] Controller Parameters

\[ a \] weighting parameter \( (\tau_m + \tau_f) \)

and

\[ z = k_m (\tau_f + \tau_m) \]
\[ k_{est} I_1 \]

\[ a(\tau_m + \alpha) \]
\[ k_{est}^2(\tau_m + 1) \]
\[ k_m^2(\tau_m + 1) \]

\[ b \]
\[ \alpha \]

\[ c \]
\[ k_{est}(\tau_m + 1) \]
\[ k_m \]
Simulation Results

Simulation testing was conducted using a Bailey Controls Multi function Controller (MFC03). The DPE algorithm was implemented as a C program, the existing model based controller was used and two generic processes were simulated using standard function code logic. The generic processes used for the simulation testing, consisted of a third order lead lag network with deadtime and a second order polynomial with deadtime specified in terms of natural frequency and damping ratio. Both linear processes were designed to operate around a bias of 500 and with a gain of 10. Zero mean, gaussian white noise with a specified standard deviation was added to the output of each process. The simulation was designed to run the process at a sampling rate 3 times that of the controller to approximate the behavior of a digital controller operating on an analog process. A supervisory program controlled each simulation run to assure repeatability and consistency between all tests.

Preliminary simulations used a first order lag with deadtime process to determine the effects of the ratio between process time constant and deadtime and selection of the model based controllers time constant on estimator performance. Table 1 summarizes the results of initial testing by showing the worst case fraction generated on a first order processes with deadtime to time constant ratio of 1/6 to 4. The controller parameters were varied to match those of the actual process with filter time constants ranging between 100 and 900 seconds.

<table>
<thead>
<tr>
<th>$\tau_p$</th>
<th>$T_p$</th>
<th>$\tau_{est}$</th>
<th>$T_{est}$</th>
<th>$\tau_f$ (worst case)</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>10</td>
<td>0.18</td>
<td>1.41</td>
<td>0.45</td>
</tr>
<tr>
<td>30</td>
<td>30</td>
<td>0.32</td>
<td>0.2798</td>
<td>0.31</td>
</tr>
<tr>
<td>60</td>
<td>60</td>
<td>0.32</td>
<td>0.3837</td>
<td>0.30</td>
</tr>
<tr>
<td>120</td>
<td>120</td>
<td>0.36</td>
<td>0.1379</td>
<td>0.30</td>
</tr>
<tr>
<td>30</td>
<td>5</td>
<td>0.3150</td>
<td>0.361</td>
<td>0.30</td>
</tr>
</tbody>
</table>

Table 1 shows that smaller values of the filter time constant resulted in improved estimator performance down to the point at which controller saturation occurs in response to the required setpoint change. For small filter time constants, the value of $\tau_f$ remains large for a greater percentage of the total process response increasing the amount of information captured by the integral $I_t$ and thus improving estimator accuracy. The error between the estimated and actual parameters increases as a function of the deadtime value used in the controller resulting from inaccuracies in the controllers implementation of the deadtime. For estimators parameters were implemented in the controller, this error was found to have no impact on the overall performance.

Additive output noise with a standard deviation of less than 0.25 had minimal impact on estimator performance. The impact of noise, however, cannot be fully understood by analyzing the results of simulation testing alone. It is assumed that the type of noise present and the magnitude of the noise with respect to the size of setpoint change used for estimation and the process gain will affect estimator performance in ways that cannot be understood through field testing.

A second set of simulation tests were used to determine the effect of initial parameter estimates on estimator performance. A first order plus deadtime process with $\tau_p$ 300 seconds and $T_p$ 610 seconds was used. Controller parameters were varied as follows:

<table>
<thead>
<tr>
<th>$\tau_m$</th>
<th>100</th>
<th>600 seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_m$</td>
<td>100</td>
<td>600 seconds</td>
</tr>
<tr>
<td>$\tau_f$</td>
<td>150</td>
<td>600 seconds</td>
</tr>
</tbody>
</table>
The DPE identified both the process time constant and deadtime to within + 1.5 seconds or 2%. For processes with either the model time constant or the model deadtime of the model based controller set significantly faster than the correct value and with a small filter time constant, no estimates were obtained. Observation of the closed loop response for these cases revealed marginal stability or instability of the closed loop system, preventing convergence of the integral I1.

The above simulation studies validated the performance of the DPE for the described first order and second order processes. Excellent results were obtained independent of initial controller settings in the presence of moderate levels of process noise. The estimator experienced difficulty when the controller switched during the course of the closed loop test or when the initial controller settings resulted in an unstable closed loop system.

Further simulation testing was conducted to explore the ability of the DPE to approximate high order dynamic systems with a first order process model. Two processes were used for this testing:

\[
G_1(s) = \frac{10s + 1}{30s} \quad G_2(s) = \frac{2}{s^2 + 2s + 1}
\]

The DPE was tested on \( G_1(s) \) using controller settings:

\[
\begin{align*}
\tau_m &= 300 \quad 9.0 \text{ seconds} \\
T_m &= 300 \quad 9.0 \text{ seconds} \\
\tau_f &= 300 \quad 17.0 \text{ seconds}
\end{align*}
\]

For all controller settings, evoking a stable closed loop response, the estimated process corresponded to a first order process with:

\[
G_{\text{test}}(s) = \frac{e^{75s}}{s + 1}
\]

Figure 3 shows the open loop response of the estimated process, \( G_{\text{test}}(s) \), superimposed over the test response of the process it approximates, \( G_1(s) \). Figure 4 shows the closed loop response of the process \( G_2(s) \), under the control of the model based controller using the estimated model, \( G_{\text{test}}(s) \), with a time constant of 60.0 seconds.

This series of tests clearly show that the DPE is capable of approximating high order process dynamics with a first order model even in the presence of numerically-determined process parameters. Further more, these tests show that using the estimated parameters to tune the model based controller provides excellent coarse control of high order processes.

Simulation testing with the process \( G_2(s) \) revealed an inability of the DPE to approximate processes using a first order model. Specifically, the estimator provided suitable approximations for processes with damping ratio above 0.8. Similarly, the first order model-based controller provided stable closed loop response on processes with a damping ratio below this value only with the use of a large time constant.
of tuning resulted in sluggish closed loop performance. Clearly neither the DPE nor the model based mode controller are well suited for use with highly underdamped processes.

Conclusion

Bailey Controls is developing a powerful autotuning controller, using a deadtime/parameter estimation algorithm to identify process characteristics and a mode-based controller to provide deadtime compensated control. Simulation results show that this autotuner is applicable to a wide range of industrial processes. Currently, this autotuner is undergoing further evaluation at several test facilities. In a steady-state configuration, the DPE portion of the autotuner is able to respond to naturally occurring step setpoint changes. Also, an external triggering mechanism can be used to force the DPE to identify the process.

The development of the autotuning controller is nearing completion. After validation of the algorithm, implementation, and documentation through field testing, the DPE will be added to Bailey's library of standard function codes. Nevertheless, several areas of interest have been identified as targets for continuing development. First, simulation results suggest that the DPE is not applicable to processes that exhibit large, slow changes in the closed-loop responses. Fortunately, one can often stabilize closed-loop responses by increasing the filter time constant of the model-based controller. Therefore, a first goal of continuing research will be to identify a mechanism for automatically increasing the filter time constant of the model-based controller to stabilize slow closed-loop system response and allow identification of process characteristics. Second, PID control is currently used in a majority of applications. Initial development of the DPE identified and validated equations for process identification for use with PID controllers. Therefore, a second goal of continuing research will be to identify a version of the DPE for use with a PID controller.
References


