FOPDT Modeling

R. Russell Rhinehart

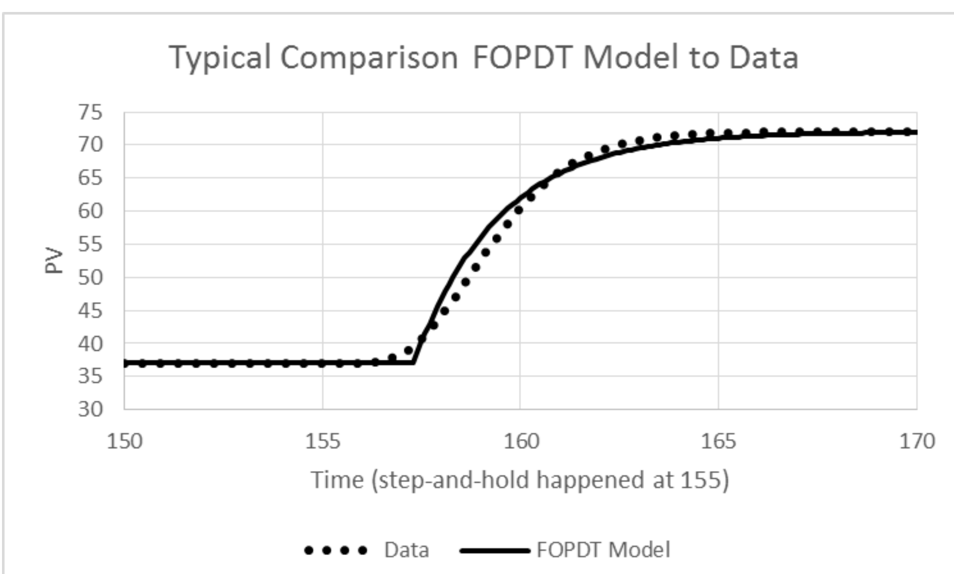
Develop Your Potential Series in CONTROL Magazine, November 2016, Vol. XXIX, No. 11, pages 46-48.

A First-Order Plus Deadtime (FOPDT) model is a simple approximation of the dynamic response (the transient or time-response) of a process variable to an influence. It is alternately termed first-order lag plus deadtime (FOLPDT), or “deadtime” may be replaced with “delay” changing the acronym to FOLPD. The FOPDT model is often a reasonable approximation to process behavior; and has demonstrated utility for controller tuning rules, for structuring decouplers and feedforward control algorithms, in communicating essential process attributes, and as a computationally simple surrogate model in simulations for training and optimization.

There is no claim that the FOPDT model is a true representation. The process is likely higher order, and nonlinear. However, a FOPDT model is a practicable representation, balancing multiple aspects of utility.

In FOPDT modeling, typically, we consider that the influence has remained constant in the recent past, and that the process variable (PV) had achieved a steady value. Then, we consider that the influence makes a step-and-hold, and holds that new value until the PV reaches its new steady state. The deadtime represents the time duration after the influence changes for which the PV does not change. It is like a transport delay in a pipeline with plug flow, or a laboratory analysis time.

This figure compares the model to data. Here, for clarity, the data is ideally noiseless, s-shaped, and indicated by the dots. The step-and-hold in the influence happened at a time of 155; but note, the high-order process does not begin to reveal a response until a time of about 156 where it starts to rise slowly, then achieving its fastest rate of change at a time of about 159. For a best fit to the data, the FOPDT model (solid curve) has a longer delay, and does not begin to make a change until a time of about 157. Because it is a single lag, its fastest rate of change is when it starts to respond. The model has a delay, longer than the process, then must change rapidly to catch up to the process. The model rises above the process in the 158-162 time period, then relaxes to the final steady state value a bit slower. This model best balances the “+” and “-“ deviations from the data.



Mathematically, the model could be stated as an ordinary differential equation.

$$\tau_m \frac{d\tilde{y}'(t')}{dt} + \tilde{y}'(t') = K_m u'(t' - \theta_m), \quad \tilde{y}'(t' = 0) = 0$$

The model gain, K_m, is the multiplier for the influence change that determines the new steady state value for the PV. The FOPDT model pretends that once the delay, θ_m, duration has past the PV follows a first-order exponential trajectory to the final steady state value. The FOPDT time-constant, τ_m, is an indicator of how fast the PV moves toward the new value. Contrasting some conventions, I have used the subscript "m" for "model" not the subscript "p" for "process" to acknowledge that model is not the process. I have explicitly placed a squiggle hat over the model response variable, $\tilde{y}'(t')$, to indicate that it is the model, not the process. And, I have used the prime mark to indicate that the model influence, response, and time are each a deviation from the initial steady conditions as well as the time for the step-and-hold influence. In the figure above the change happens at a time of 155. Although $t = 155$, at that instant $t' = 0$. Similarly, the initial process value is $y = 37$, but the deviation value is $y' = 0$.

Although the concept for the model is a response to a step-and-hold influence from an initial steady state, and although this makes for convenient analytical solutions, it is a generic model, and not so restricted when solved with numerical methods. And, although the model can be equivalently stated in Laplace, or z-transform notation, I won't!

The classic textbook method to generate FOPDT models is the reaction curve technique, a pre-computer era technique. It is simple to understand and to implement, and it can be derived from the analytical solution of the ODE; so it serves the current content of undergraduate engineering education appropriately; however, I believe reaction curve techniques do not express best practices in the computer era.

However, often, a crude approximation FOPDT model is all that is needed. In such cases, a reaction curve technique can be a simple and fast method to get a good-enough model.

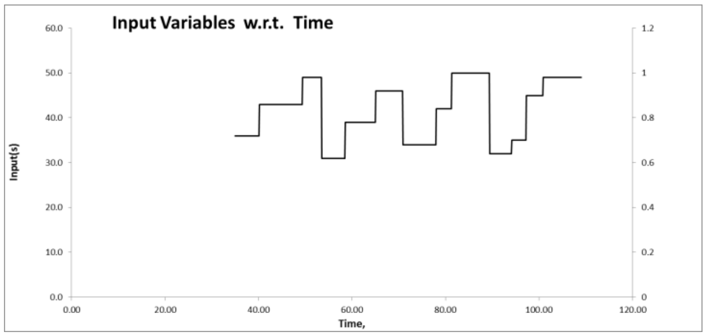
The reaction curve technique asks you to make a step-and-hold change in the process input, from an initial steady state, and hold the input until the response variable levels to an ending steady state. Unfortunately, noise and drifting alternate influences confound the response. And, a single step pushes the process away from a desired set point. Further, a push to one side of a nominal value will misrepresent nonlinear aspects. So, for effective reaction curve tests, we often use an up-down-down-up pattern in the influence step-and-hold values. This generates four reaction curves, and their average can temper the influence of noise and disturbances. Further, the pattern explores both sides of the original MV value, making compensating upsets and, ideally, returning the process to the original value.

The steps must be large enough to make a noticeable change in the response. If the change is small relative to normal noise and drifts, the FOPDT model coefficients will have a large uncertainty.

Once the response curves are completed the model coefficients are calculated from a few points on the response curve. There are multiple twists on the method.

This approach, however, requires operator attention for an extended time to wait for 4 steady state periods, it may create process deviations that impact downstream quality, it requires the human to interpret the signal to provide data for the mathematical analysis, only uses a small part of the data generated, and can be substantially confounded by uncontrolled disturbances.

In the computer era, by contrast, nonlinear least squares regression is simple to implement; and a skyline input function has advantages in operational duration, magnitude of upsets, and number of excitations over classical methods. A skyline pattern in the controller output could look like this example:

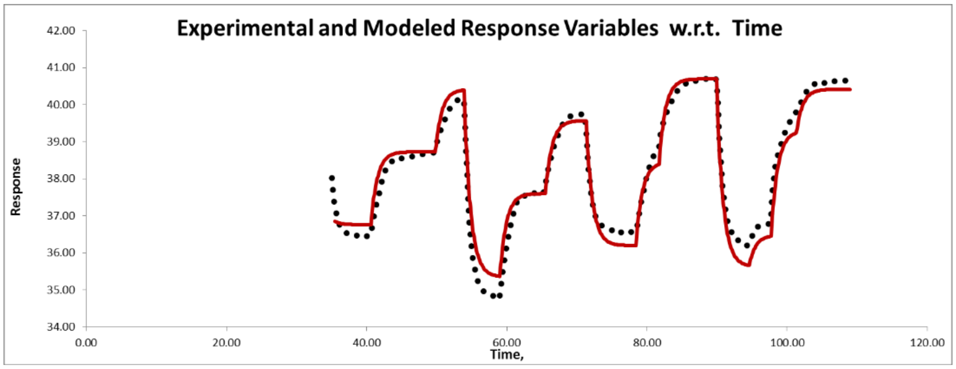


The nonlinear regression method seeks to fit the model to all data points, not just the selected several points in a classic reaction curve fit.   So, it better rejects noise and disturbances.

The skyline and regression method does not require operator attention or judgment, which lessens the possibility for operator error or bias. And the skyline input has many ups and downs, accordingly this tempers the influence that environmental drifts have on confounding the CV response to the MV. The skyline and regression method does not extend the period of off-nominal production, each "+" or "-" period is shorter, creating less objection by a quality manager. The skyline and regression method does not require an initial SS, the entire test period takes less time.

For a nonlinear process, gains, delays and time-constants change with MV and CV.   The FOPDT model is linear, and may not provide a great match to the process over a wide operating range. Just because the optimizer converges on a best model, does not mean that that model actually fits the data. So, after the regression, look to see if the fit is satisfactory for your model use purposes.

The following graph reveals the data and best FOPDT model from the input sequence above, for a pilot-scale process flow rate response to controller output. The model is the solid line, the data are the dots. Note that early-time data is usually below the model, but late-time data is usually above. Perhaps some drifting influence was affecting the data. Also note the kinks in the model at times a bit after 80 and 100. These are expected responses to small changes in the MV at those times, but these are not expressed in the data. Perhaps valve sticktion prevented the valve from moving even though the MV made changes. Finally, this data is not expected to be a linear response. Although the linear FOPDT model does not perfectly match the data, it is a very good representation of the process dynamics.



Visit www.r3eda.com, and select the "Regression" Menu to access a more extensive tutorial and my Excel/VBA file for generating FOPDT models.