

A Heat Exchanger Simulator for Testing and Demonstrating Automation Algorithms

R. Russell Rhinehart, Fellow ISA & AIChE

Abstract— This offers a simulation model of a process heat exchanger, which can be used for testing and demonstration of automation algorithms such as: control techniques, controller tuning, data reconciliation, steady state detection, model adaption, noise filtering, constraint prediction, fault detection, NN or fuzzy modeling, etc. Based on a full-scale exchanger, the simulator is nonlinear, with variable transport delay, third-order lags, and subject to both unmeasurable and measurable disturbances. Although rich in the expression of process features, the simulator is relatively simple to code. A VBA version of the simulator and user guide are accessible to the public on www.r3eda.com. (Note to reviewers, this might not be posted, but other simulators are.)

I. INTRODUCTION

This article offers a process heat exchanger simulator, which is convenient for use (coding, understanding, etc.), yet is also reasonably grounded in an actual exchanger (See Figure 1) to legitimately guide technique development and to establish credibility.



Figure 1 – Heat Exchanger Unit in the Unit Operation Lab of Oklahoma State University

Researchers commonly use simulators to guide development of novel automation techniques, then use the simulators to demonstrate their technique, and to benchmark theirs to other techniques. Instructors use simulators to aid learning. Often the simulator is conveniently simple, which is important for ease of understanding, ease of coding, and immunity from execution errors. However, most simulations are too idealized to be able to establish credibility of the technique for the practitioner. Further, when using transfer

function models, simulators lose phenomenological support of process understanding. Real processes are nonlinear, non-stationary, have measurement noise, and are subject to continually varying inputs and disturbances.

II. MODEL DESCRIPTION

The heat exchanger is a liquid-liquid, counter-current, single-pass type [1]. The hot fluid is in the shell side and modeled as an oil. The process fluid being heated is modeled as water. The control objective is to control the exit water temperature by sending a signal to the flow control valve on the hot fluid entrance line. The control valve has a linear inherent characteristic, but installed in the line, it is a moderately quick-opening installed characteristic [2]. The valve is air actuated, with a stem position response that is roughly first-order to the controller output, the manipulated variable (MV). Each of the entrance and exit lines have a thermowell to measure temperature, the measurement has a first-order lag, and also a delay due to fluid transport between the sensor and the heat exchanger.

Besides the flow rate of the hot fluid (controlled) the uncontrolled inputs to the heat exchanger are the cold fluid flow rate, the hot and cold fluid temperatures, and ambient heat losses. These four variables continually fluctuate using an autoregressive moving average, ARMA(1,1), model driven by normal (Gaussian) independent deviations with zero mean and a standard deviation, NID(0, σ) [3]. The user can also make intentional changes to these variables.

A key and undesirable characteristic of a heat exchanger is fouling of the heat transfer surface. It might be due to buildup of scale, a component that precipitates or chars when hot, etc. As fouling progresses, the hot fluid flow rate must be increased to sustain the outlet fluid at its setpoint. Fouling typically progresses over time, or it may happen fairly rapidly if the inlet fluid contains a batch of the undesirable material.

More than just control, of interest is the identification of fouling to schedule maintenance and to forecast constrained operation.

There are several stages to the model, which will be developed independently.

A. The Control Valve

The output of the controller, the MV, is u , in the 0-100% range. The valve stem position is x , the fraction of full travel,

R. Russell Rhinehart (Russ) is a chemical engineering professor emeriti at Oklahoma State University. Retired, he is now providing resources related to optimization, modeling, and process control through his web site www.r3eda.com. E-mail Russ@r3eda.com or rrr@okstate.edu.

in the 0 to 1 range. Because of the air actuator, x has a nearly first-order response to u with a time-constant of τ_v .

$$\tau_v \frac{dx}{dt} + x = .01u \quad (1)$$

Numerically,

$$x := \lambda_v 0.01u + (1 - \lambda_v)x \quad (2)$$

$$\lambda_v = 1 - e^{-\Delta t/\tau_v} \quad (3)$$

In Equation (2) the $:=$ represents an assignment statement. On the right-hand side x represents the valve stem position at the prior time step, Δt , which is used to predict the new value at the end of the time interval. Differentiated from an algebraic ARMA(1,1) model, in computer code subscripts are not needed. Use Equation (2) to determine how x responds to u .

The valve equation describes how steady state flow rate responds to stem position.

$$F = C_v f(x) \sqrt{\frac{\Delta P_v}{G\xi}} \quad (4)$$

Here, C_v is the valve capacity coefficient with the same units as F , the volumetric flow rate. ΔP_v is the pressure drop across the valve, G is the fluid specific gravity, and $\xi = 1$ has the same units as ΔP_v . $f(x)$ is the inherent valve characteristic, which reveals how steady state flow rate depends on stem position, $0 \leq f(x) \leq 1$. For a linear valve, $f(x) = x$.

For in a pipeline, the liquid mass needs time to accelerate from a prior velocity value to a new steady state value. However, typical process flow lines are short enough so that this acceleration time-constant is much smaller than the valve actuator τ_v , and the flow can be considered at the steady state associated with x [4].

However, as the flow rate changes, the friction loss pressure drop, ΔP_f , in the line changes. Assuming that the system pressure drop, ΔP_s , does not change

$$\Delta P_s = \Delta P_v + \Delta P_f \quad (5)$$

Using a classic turbulent flow model for losses in piping and fittings [5], with k representing the sum of all pipe and fitting influences,

$$\Delta P_f = k \frac{1}{2} \rho v^2 = k \frac{1}{2} \rho \frac{1}{A_{xc}^2} F^2 \quad (6)$$

Solving Equation (4) for ΔP_v , inserting Equations (4) & (6) into (5), combining constants into coefficients a , b , c , using the linear inherent characteristic, and solving for F , converts the linear inherent characteristic to the installed characteristic.

$$F = ax \sqrt{\frac{1}{b+cx^2}} \quad (7)$$

Use Equation (2) to determine how x responds to u , then Equation (7) to determine how the controlled flow rate responds to x .

B. The Heat Exchanger

The design equation for a single-pass, counter-current heat exchanger is [1]:

$$\dot{Q}_t = UA\Delta T_{lm} = UA \frac{(T_{h,i}-T_{c,o})-(T_{h,o}-T_{c,i})}{\ln\left(\frac{T_{h,i}-T_{c,o}}{T_{h,o}-T_{c,i}}\right)} \quad (8)$$

Here \dot{Q}_t is the rate that heat is transferred, U is the overall heat transfer coefficient, A is the heat transfer area, and T represents fluid temperature. The subscripts h and c represent the hot and cold fluids and the subscripts i and o represent inflow or outflow locations. ΔT_{lm} is called delta-T-log-mean and represents the effective average temperature difference along the tube length for heat transfer. Given values for UA and for $T_{h,i}$ and $T_{c,i}$, there are three unknowns in Equation (8).

These are solved by using equations for the heat gained by the cold fluid and lost by the hot fluid, ideally modeled as:

$$\dot{Q}_c = F_c \rho_c C p_c \Delta T_c = F_c \rho_c C p_c (T_{c,o} - T_{c,i}) \quad (9)$$

$$\dot{Q}_h = F_h \rho_h C p_h (T_{h,i} - T_{h,o}) - \dot{Q}_{amb} \quad (10)$$

Where \dot{Q}_{amb} represents the losses from the heat exchanger to the environment, here credited only to the hot shell-side fluid.

Since $\dot{Q}_t = \dot{Q}_h = \dot{Q}_c$, Equations (8-10) can be solved to determine \dot{Q} , $T_{h,o}$ and $T_{c,o}$, given values for the inlet flow rates, fluid properties, \dot{Q}_{amb} , U , A , and the inlet temperatures.

The equations do not seem to admit an explicit solution, so a root-finding approach is needed. Guess at \dot{Q}_t as \dot{Q}_g , solve Equations (9) & (10) for the associated outlet temperatures, then use Equation (8) to see if the associated \dot{Q}_t is equal to \dot{Q}_g . Successive Substitution and Newton's method can be unstable. Interval halving from high and low extreme possible values for \dot{Q}_g is stable and fast. The low guess is that the cold fluid only rises by a small amount (1 degree). The high guess would be the minimum of either the hot fluid exiting very close to the cold inlet T , or the cold fluid exiting very close to the hot inlet T .

The overall heat transfer coefficient depends on the fluid heat transfer coefficients, the conduction resistance through the tube walls, and fouling resistance [1].

$$\frac{1}{U} = \frac{1}{h_h} + r_{cond} + \frac{1}{h_c} + r_{foul} \quad (11)$$

And experimentally for turbulent flow $h \propto F^{0.8}$ [1], so

$$U = \frac{1}{\frac{1}{h_{h,ref} \left(\frac{F_{h,ref}}{F_h}\right)^{0.8}} + r_{cond} + \frac{1}{h_{c,ref} \left(\frac{F_{c,ref}}{F_c}\right)^{0.8}} + r_{foul}} \quad (12)$$

This procedure determines the steady state exit temperatures of the heat exchanger. Experimentally, the dynamics of the heat exchanger is found to be nearly first-order, and small relative to transport delay of the fluid. And the lags in the temperature sensors and can be modeled as first order [6].

Assuming plug flow (nearly valid for turbulent flow) and adiabatic flow (no heat transfer to or from the piping system), the transport delay from the HX to the sensors is simply $\theta = V/F$, where V is the volume from the middle of the HX to the T sensor, and digitized to the number of simulation time steps,

$$N_{delay} = INT\left(\frac{V}{F\Delta t}\right) \quad (13)$$

The temperature of the fluid at the i^{th} simulation interval would be the temperature exiting the HX N_{delay} samples prior. This

is for both the cold and hot fluids. Further, the inlet temperatures are sampled prior to the fluid entering the heat exchanger, so the fluid entering the HX during any particular sample is that at the sensor N_{delay} samples after.

The clock method is used to store and retrieve delayed values. Perhaps use an array of 100 elements. Instead of viewing it as a linear column array, consider that the array is circular, like the numbers on a clock, with the array element 100 adjacent to the 1 like the clock hour 12 is adjacent to the 1. At each sampling increment the store index, i_{put} , by 1, and if $i_{put} = 101$ then reset it to 1. Store the value in the $x(i_{put})$ array element. Determine the i_{read} index value as $i_{put} - N_{delay}$. If i_{read} is negative then reset it using $i_{read} := i_{read} + 100$.

Finally, all of the sensors have a lag to respond to the process fluid which is modeled as a first-order lag.

$$\tau_{tw} \frac{dT_{sensor}}{dt} + T_{sensor} = T_{fluid} \quad (14)$$

Numerically,

$$T_{sensor} := \lambda_{tw} T_{fluid} + (1 - \lambda_{tw}) T_{sensor} \quad (15)$$

$$\lambda_{tw} = 1 - e^{-\Delta t / \tau_{tw}} \quad (16)$$

All in all, to determine the outlet temperatures, use Equation (12) to determine the value of U , then the N_{delay} past inlet temperatures and the interval halving procedure to determine the HX outlet temperatures. Then use Equation (15) on the delayed outlet temperatures to determine the sensor T.

C. Noise and Stochastic Inputs

A recommended way [3] to generate stochastic inputs is to consider that they are first-order responses to disturbances. In Equation (17), the variable x now represents the first-order perturbation on a process variable caused by a disturbance d , with a first-order time constant of τ .

$$\tau \frac{dx}{dt} + x = d, \quad x(t=0) = x_0 \quad (17)$$

Analytically, the solution can be converted to an incrementally updated new time sample by Euler's explicit method.

$$x_{new} = \lambda d_{new} + (1 - \lambda) x_{prior} \quad (18)$$

$$\lambda = 1 - e^{-\Delta t / \tau} \quad (19)$$

Here, Δt is the simulation time step, or sample-to-sample time interval. If the influence, d , is not a constant, but continually changes, then the new x -value is influenced by the new d -value.

If d_{new} is modeled as randomly changing in a Gaussian manner with a mean of 0, and variance of σ_d , then it can be modeled using the Box-Muller [7] formula $d_{new} = \sigma_d \sqrt{-2 \ln(r_1)} \sin(2\pi r_2)$ where r_1 and r_2 are independent random numbers uniformly distributed on an interval from 0 to 1 (a standard random number generator). Then, the first-order persistence driven by $NID(0, \sigma)$ noise and averaging about a value of x_{base} can be modeled as:

$$x := \lambda \sigma_d \sqrt{-2 \ln(r_1)} \sin(2\pi r_2) + (1 - \lambda) x \quad (20)$$

$$x_{process} = x_{base} + x \quad (21)$$

In creating a simulation with a stochastic influence, the user would choose a time-constant for the persistence that is reasonable for the effects considered, and a σ_x -value that would make the disturbance have a reasonable variability. At each sampling, Equation (20) would provide that stochastic value for the variable x , the perturbation for the process value, $x_{process}$. The stochastic variable could represent barometric pressure, ambient heat load, raw material composition, or any such ever-changing influence or process character. In its first use, the value for x in Equation (20) should be initialized with zero.

How should one choose values for λ and σ_d ? First consider the time-constant, τ , in Equation (17). It represents the time-constant of the persistence of a particular influence. Roughly, $\tau \approx 1/3$ of the lifetime of a persisting event, (because the solution to the first-order differential equation indicates that after 3 time-constants, x has finished 95% of its change toward d). So, if you considered that the shadow of a cloud persists for 6 minutes, then the time-constant value is about 2 minutes. Once you choose a τ -value that matches your experience with Nature and decided for a time interval for the numerical simulation, Δt , calculate λ from Equation (19).

To determine the value for σ_d , propagate variance in Equation (18). The result is Equation (22). Use your choice of σ_x (and λ , which is dependent on your choices for Δt and τ) to determine the value for σ_d .

$$\sigma_d = \sigma_x \sqrt{\frac{2-\lambda}{\lambda}} \quad (22)$$

Choose a value for σ_x , the resulting variability on the x -variable. To do this, choose a range of fluctuation of the disturbance. You should have a feel for what is reasonable to expect for the situation that you are simulating. For instance, if it is barometric pressure the normal local range of low to high might be 29 to 31 inches of mercury, if outside temperature in the summer it might be from 70 to 95 °F, or if catalyst activity coefficient it might be from 0.50 to 0.85. The disturbance value is expected to wander within those extremes. Using the range, R , as

$$R = HIGH - LOW \quad (23)$$

And the standard deviation, σ_x , as approximately one-fifth of the range, then

$$\sigma_d = \frac{R}{5} \sqrt{\frac{2-\lambda}{\lambda}} \quad (24)$$

As a summary: To generate stochastic inputs for dynamic simulators, choose a time constant for persistence of events and a range of the disturbance variable. Use Equation (19) to calculate λ , then Equation (24) to calculate σ_d . Then, at each simulation time interval, use Equations (20) and (21) to determine the stochastic input.

Figure 2 is an illustration of one 400-min realization of a stochastic variable calculated as above. The time-constant is 40 min, the nominal value is 20, and the range is 3 units. In this one illustration, notice that the variable averages about its nominal value of 20 and that the difference between the high and low values is nearly 3. From a period between 150 and

275 the value is below the average, indicating a persistence of 275-150=125 min, about three times the 40 min time-constant. Some persistence values are shorter, some longer.

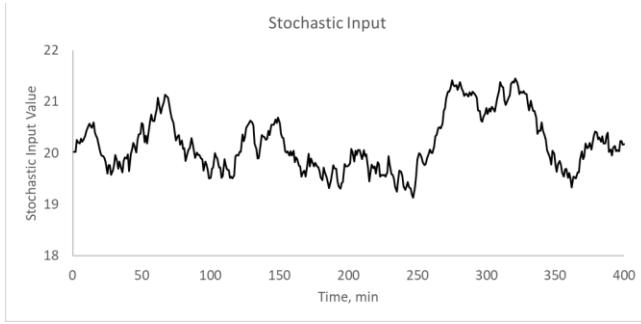


Figure 2 – Illustration of generating a stochastic influence for a process

Measurement values are subject to many random uncontrolled variables (vibration, electromagnetic activity, electronic thermal noise, flow turbulence, etc.). Modeled as a collective influence that is randomly changing in a Gaussian manner with a mean of 0, and variance of σ_{noise} , you might again use the Box-Muller [7] formula

$$x_{noise} = \sigma_{noise} \sqrt{-2\ln(r_1)} \sin(2\pi r_2) \quad (25)$$

Where r_1 and r_2 are independent random numbers uniformly distributed on an interval from 0 to 1. Then, add the noise perturbation to the true process sensor value to generate the measurement value.

$$x_{measurement} = x_{sensor} + x_{noise} \quad (26)$$

D. Coefficient Units and Values

For the valve:

u		controller output, %
τ_v	1	sec
x		Fraction of full stroke, dimensionless
F		m^3/hr
a	500	m^3/hr
b	1	dimensionless
c	5	dimensionless

For the heat exchanger:

\dot{Q}		kJ/hr
U		$kJ/(hr m^2 K)$
A	0.25	m^2
$T_{h,in}$	500	K
$T_{c,in}$	285	K
$h_{h,ref}$	700,000	$kJ/(hr m^2 K)$
$h_{c,ref}$	700,000	$kJ/(hr m^2 K)$

$F_{h,ref}$	100	m^3/hr
$F_{c,ref}$	20	m^3/hr
r_{cond}	0.000,001	$(hr m^2 K)/kJ$
r_{foul}	0	$(hr m^2 K)/kJ$
Cp_h	3	$kJ/(kg K)$
ρ_h	900	kg/m^3
Cp_h	4	$kJ/(kg K)$
ρ_h	1,000	kg/m^3
\dot{Q}_{amb}	50,000	kJ/hr
V	0.01	m^3
τ_{tw}	3	sec

For measurement noise:

σ_T	0.5	K
σ_F	0.1	Fraction of F

For stochastic inputs:

τ_{Fc}	100	sec
R_{Fc}	10	m^3/hr
τ_{Tc}	100	sec
R_{Tc}	50	K
τ_{Th}	100	sec
R_{Th}	100	K
$\tau_{\dot{Q}_{amb}}$	100	sec
$R_{\dot{Q}_{amb}}$	10,000	kJ/hr

III. SIMULATOR RESULTS

The installed valve characteristic is illustrated in Figure 3. Although the inherent characteristic is linear, when installed it has a mildly quick opening functionality.

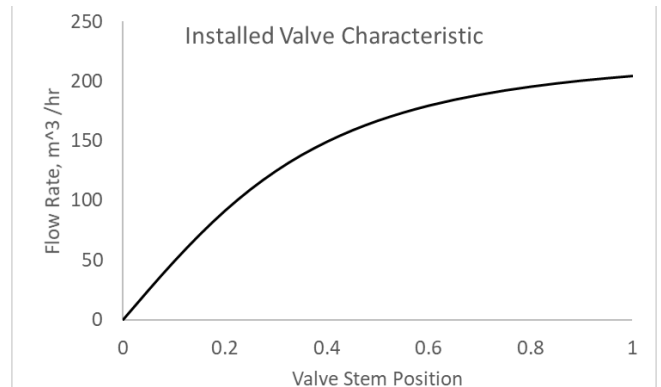


Figure 3 – Installed Valve Characteristic

Figure 4a presents a sequence of steps in the MV, and Figure 4b indicates the measured temperature responses when there is no measurement noise or input ARMA disturbances. The upper trace in Figure 4b is the exit hot temperature and the lower trace is the exiting cold temperature. The low range gain is about 4.5 K/% and the high range gain is about 0.5 K/%, about a 9:1 range. Reasons for the nonlinear gain include the installed valve characteristic, the flow rate dependence on the overall heat transfer coefficient, and the heat exchanger itself.

Figure 4c reveals the detail at the step at 100 s, indicating a several second delay on the slower flowing cold exit Temperature.

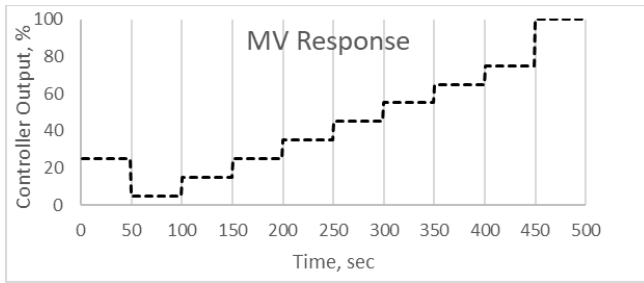


Figure 4a – Steps in MV

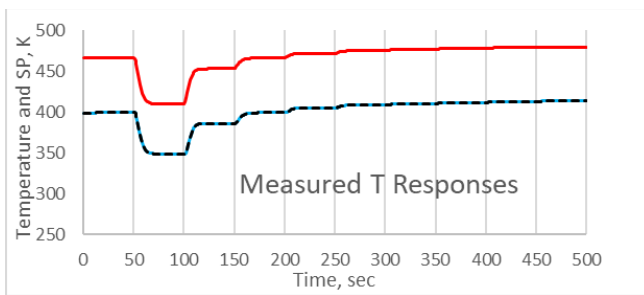


Figure 4b – Temperature Responses

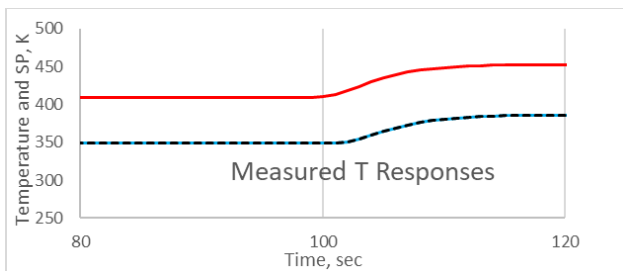


Figure 4c – Revealing Delay and Lags

For the same input sequence, but with 1/3rd of the cold fluid flow rate, Figure 4d shows the same detail as that in Figure 4c.

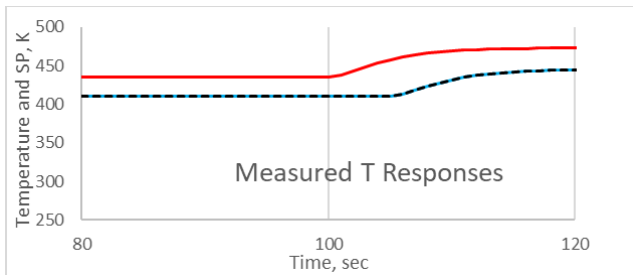


Figure 4d – Repeated Trial with Lower Cold Flow Rate.

Note that the delay in the cold outlet T is about 3 times longer, and that the dynamics of the hot fluid T response is substantially the same. However, note that both outlet temperatures are substantially higher. Also note: Both the gains and delays depend on the two flow rates.

Figure 5a illustrates a sequence of steps in the influences. The upper solid trace is the measured hot inlet fluid temperature and the lower solid trace is the measured cold inlet fluid. Each makes step changes and recoveries, but the lags are due to the sensor dynamics. The dashed lines are the inlet cold (lower) and hot fluid (upper) flow rates.

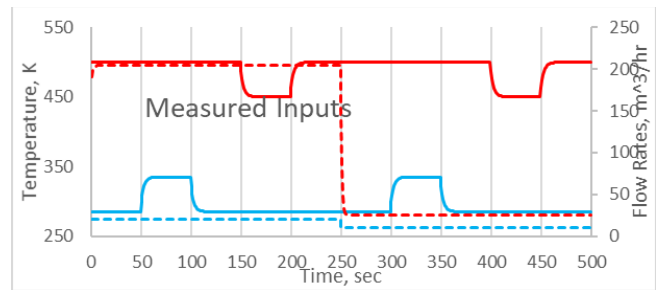


Figure 5a – Step Inputs in Influence Variables.

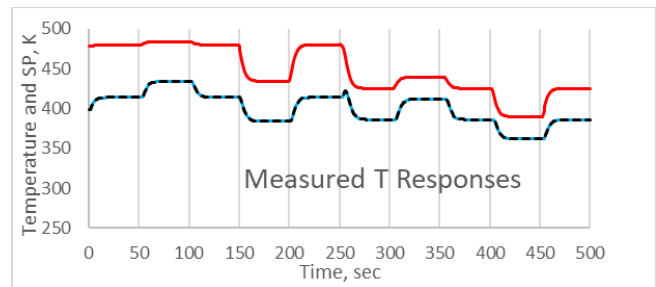


Figure 5b – Response Variables to Figure 5a.

Although each of the input steps has the same magnitude, the change in the response variables are not the same. The responses depend on flow rates and temperature differences. The gains of the process are also dependent on influence conditions.

At a time of 250 sec both flow rates changed. Note that the cold response T rose before it fell. This is due to the dynamics of the hot flow rate valve being slower to respond than a step in the cold flow rate. Not visible on the 500 sec graph, detail shows that the delay in both hot and cold outlet Ts vary with respective flow rates.

Figure 6 presents two nonstationary effects. The signal to the flow control valve is fixed. At a time of 80 sec, the fouling factor begins a ramp increase, then at 350 sec, it drops back to zero. The drop in the lower trace (cold outlet T) and slight increase in the hot temperature, is due to the fouling factor change. But also, throughout the simulation, stochastic influences on the inputs on inlet pressures and temperatures are continually drifting to create the irregular outlet T responses.

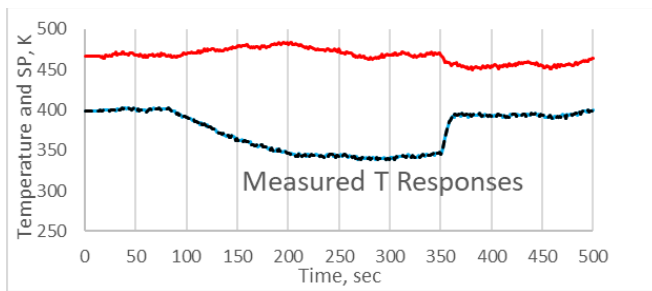


Figure 6 – Measured Response to Fouling and Stochastic Inputs.

Not illustrated, the gains also change with the fouling.

IV. DISCUSSION

In all, this is a nonlinear interactive and non-stationary process. Gains and dynamics change with the operating conditions. However, characteristics of this test case are not contrived to illustrate any difficulties that procedures for control strategy, tuning, fault detection, steady state identification, etc. might encounter. They are naturally expressed in this realistic, yet relatively simple simulator.

Although this is a SISO control application, control is difficult because of the changes in gains and delays. As a test case for model adaption, backing out fouling factor values from data is difficult because 1) the noise on measurements has a nonlinear impact, and 2) because transient conditions continually require time-shifts in matching input to output variable values. Similarly, the same issues confound procedures for fault identification and constrained operation forecasting. As a test case for identification of steady state conditions, there is autocorrelation in the measurements, not just ideal NID noise.

Although rich in the expression of process features, the simulator is relatively simple to code. A VBA version and user guide for the simulator is accessible to the public on www.r3eda.com.

Many investigators are exploring model-based techniques where the models are of the phenomenological type, as is this simulator. A rule of the game is that you cannot use the same model for the supervisory application as you use in the process simulator. Nature is always one step more complicated than human devised models. So, to credibly demonstrate your technique be sure that your model for the application is not identical to the model in the process simulator. Change coefficient values, and functionalities (such as those in the valve, overall heat transfer coefficient model, etc.) Change dynamic relations such as time-constants. And although you can observe variable values in the simulator, the only variables that your application is permitted to observe are the two noise-corrupted measured fluid flow rates and the four noise-corrupted measured temperatures. Reveal that your application model is similar, but reasonably different from the simulation model.

Although it may be useful to run the simulator without stochastic inputs and noise-corrupted measurements, as illustrated in Figures 4 and 5, be sure to test your procedure with those effects.

V. CONCLUSION

Use credible simulators to demonstrate application techniques and to preserve phenomenological process understanding.

REFERENCES

- [1] Any standard heat transfer textbook, such as Incropera, F. P. and D. P. DeWitt, *Introduction to Heat Transfer*, John Wiley & Sons, NY, 1985.
- [2] Any standard textbook about valve characteristics, such as, *Perry's Chemical Engineers' Handbook*, 7th Edition, Perry and Green editors McGraw-Hill, NY, 1997.
- [3] Any standard statistics textbook, such as Rhinehart, R. R. and R. M. Bethea, *Applied Engineering Statistics*, 2nd Edition, CRC Press, Boca Raton, FL, 2022.
- [4] From the author's analysis in teaching fluid dynamic courses.
- [5] Any standard fluid flow textbook, such as the CRANE Technical Paper No 410, *Flow of Fluids through Valves, Fittings, and Pipe*, Crane Valve Co, Joliet, IL, 1969
- [6] Manimegalai-Sridhar, U.; A. Govindarajan, and R. R. Rhinehart, "Demonstration of Leapfrogging for Implementing Nonlinear Horizon Predictive Control on a Heat Exchanger", *ISA Transactions*, Vol 60 (2016) pp 218-227
- [7] Box, G. E. P.; Muller, Mervin E. (1958). "A Note on the Generation of Random Normal Deviates". *The Annals of Mathematical Statistics*. 29 (2): 610–611.