

# Control Modeling of Fluid Temperature in Heat Exchangers

I.E. Ohue; M.O. Petinrin; O.E. Zebulun; and R.A. Kazeem\*

Department of Mechanical Engineering, University of Ibadan, Ibadan, Nigeria.

E-mail: [kazeemadobayo85@yahoo.com](mailto:kazeemadobayo85@yahoo.com)

## ABSTRACT

Heat exchanger systems are widely used in chemical plants because they can sustain a wide range of temperatures and pressures. The main purpose of a heat exchanger system is to transfer heat from a hot fluid to a cooler fluid, so the temperature control of outlet fluid is of prime importance. To control the temperature of an outlet fluid of the heat exchanger system, a conventional PID controller can be used, but for this work, a written MATLAB code serves as the temperature controller.

The mathematical model of a counter-flow heat exchanger was derived and modeled in the COMSOL Multiphysics (version 5.0). The simulated model was saved as a .m file, so that it could be opened in a MATLAB (R2012b version) environment. This MATLAB model was edited and integrated in the SIMULINK workspace, after which the heat exchanger model was coupled and then simulated. The designed controller regulates the temperature of the outgoing fluid to a desired set point in the shortest possible time irrespective of load and process disturbances. The developed MATLAB code (which served as the controller) demonstrates an effective regulation of the outlet fluid temperature, and the resulting temperature plot shows that the user of such a model has optimum control over the specific temperature desired.

(Keywords: feed-forward controller, MATLAB, heat exchanger, SIMULINK)

## INTRODUCTION

The heat exchanger is a type of equipment with numerous applications targeted to the efficient transfer of heat from a hot fluid flow to a cold fluid flow without moving parts. It transfers heat between two fluids without mixing them up. The heat exchanger depends on many factors like temperature difference, heat transfer area, flow

rate of fluids, and flow patterns. Although heat flows from a hot fluid to cold fluid by thermal conduction through the separating wall (except in direct contact types), heat exchangers are basically heat convection equipment, since they are the convective transfer what governs its performance (Martinez, 2015).

In a counter-flow or countercurrent exchanger, the two fluids flow parallel to each other but in opposite directions within the core. In processes which exist in various industrial plants, it is imperative for process fluids to be at specific temperatures at different points in time, hence setting a target for a specific controlled parameter which in turn has an important effect on the performance of the system and impact on the environment.

Due to the large surface area needed for the heat transfer, and the distance the fluid travels through the long ducts, achieving efficient control becomes a major problem and since the complexity of the eventual heat exchanger structure attracts heavy cost. This work is therefore hinged on addressing optimal process operation through efficient temperature control thus minimizing cost (proper and efficient regulation leads to effective use of energy and much cost savings) and this is achieved by modeling an optimal control of the output stream temperature heat exchanger fluid in space and time and developing and simulating a mathematical model of the heat exchanger system, the actuator, valve, sensor, etc. Heat exchangers find wide-spread applications in different industries such as the food, spacecraft, petrochemical, power generation, and petroleum industries, and so on (Padhee, 2014).

As mentioned above, the distributed parameter model on a control process will be modeled in this work. In this model, the state space is infinite-dimensional, systems described by partial differential equations (PDEs) and delay

differential equations (DDEs) are examples of this.

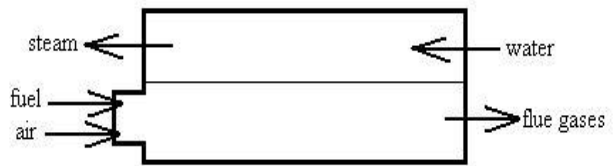
This study dwells basically on the use of the counter-flow heat exchanger. Many processes have been described using lumped-parameter models but which are inherently distributed parameter models, that is, defining the process variables in space and time. Thus, the distributed parameter model on a control process will be modeled in this work. The thermal control of the temperature needs to be maximized and absolutely regulated as the temperature of a system depicts the internal energy level of that system. Advanced study and further development of temperature controllers which can be further advanced by engineers specialized in control systems for shell and tube heat exchangers in industrial applications can be a great improvement on this work.

### THE HEAT EXCHANGER SYSTEM

Nowadays, heat exchangers are used in a wide variety of applications and they have to be designed in order to increase their effectiveness without neglecting the costs associated with their life cycle. Maximizing the heat transfer area and the heat transfer coefficient minimizes the volume required by the heat exchanger. Heat exchangers designed in this way are called compact heat exchangers. The compactness of the heat exchanger minimizes the capital cost of the component and also minimizes the volume of space consumed by the component. The complex shapes of some of the designs, however, are difficult to manufacture, adding to the cost of those cores.

Counter flow Heat exchangers are key devices used in a wide variety of industrial applications. Control of a heat exchanger is a complex process due to its non-linear behavior and complexity caused by many phenomena such as leakage, friction, temperature dependent flow properties, contact resistance, unknown fluid properties, etc. Therefore, IT2 FLC+ANFIS controllers can be a better alternative to the PID control, although many industrial applications use PID control to maintain constant process variables. Distributed parameter systems (well-known in process industry e.g. heat exchangers, tubular and packed bed reactors, solar collector systems) are usually modeled with partial differential equations (PDEs) and have been studied since the middle of the last century.

Figure 1 shows the geometry of a heat exchanger.



**Figure 1:** Heat Exchanger Geometry in 2-D (Martinez, 2015).

### COMSOL MULTIPHYSICS, MATLAB and SIMULINK

Three programs were used for this work are COMSOL Multiphysics, MATLAB and SIMULINK (which is integrated in MATLAB as well). COMSOL Multiphysics is a general-purpose software platform, based on advanced numerical methods, for modeling and simulating physics-based problems. The software comes with several modules in its library for specific applications. Each module contains modeling tools and equations for the application described. Modeling tools from multiple modules can be coupled together to accurately depict complicated systems and processes.

With COMSOL Multiphysics, one is able to account for coupled or Multiphysics phenomena. Several approaches were taken during the development of this model. The final model is a combination of text-based MATLAB code that uses multiple Simulink models as functions. This approach was an optimization of the conveniences and limitations of Simulink combined with the transparency of text-based code.

The resulting simulation is useful for designing counter flow compact heat exchangers, including selecting the heat transfer surfaces, the materials, and the geometry of the overall heat exchanger. After picking the geometry of the heat exchanger, the simulation can be used to determine the steady state and transient behavior of the heat exchanger. The simulation requires the user to define the inlet pressure, temperature and mass flow rate for each side of the heat exchanger as a function of time. The outlet temperature and the pressure drop are

determined, as functions of time. COMSOL Multiphysics uses the finite element method to analyze and solve complex problems.

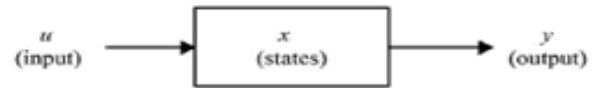
Different measurable factors were used in setting the conditions of its operation. These parameters are shown in Table 1.

## PROBLEM DESCRIPTIONS

Usually, for the lumped parameter models, the Proportional Integral Derivative (PID) controllers are used where gain derivations and PID tuning are necessary, but for this work which is based on the distributed parameter model these results in possible drawbacks. The usual way to deal with these systems and the associate drawbacks are either to simplify them to lumped parameter systems described by ordinary differential equations (ODE).

By this way, it is possible to obtain either a first or a second order transfer function to easily tune PID controllers (Camacho et al. 1997), but in turn, this causes that relevant system dynamics are not taken into account when tuning the PID controller; OR to find a high order transfer function which is subsequently used in a more complex (high order) control system, e.g. model predictive control (Camacho, et al., 1997.; Johansen et al. 2000 and Alvarez, et al., 2012). In this work, we avoid going through PID controllers and other conventional controllers, a whole new approach is adopted. A

MATLAB code is written to serve as the controller. This code is later integrated into Simulink in an S-Function block.



**Figure 2:** Representation of the S-Function General Characteristics (Schjindel, 2014).

## SIMULATIONS AND TESTING

The geometry for this experiment was designed in COMSOL Multiphysics (version 5.0). The counter-flow heat exchanger comprises of two channels each of length: one meter (1m), entrance height: 0.1m, and an assumed breadth/depth of 1m. Each channel (domain) contained a process fluid each, the first domain which was selected to contain Air (selected from COMSOL's 'built-in' material properties). The parameters required for this experiment, ranging from inlet velocity of both fluids to the mass flow rates. The second channel/domain was set to contain Water, which was also selected from COMSOL's built-in properties. The boundary conditions specify the flow and thermal variables on the boundaries of the physical domain. The velocity and temperature values are specified at the inlet of the water and air channels.

**Table 1:** Parameters Used for the Counter-Flow Heat Exchanger.

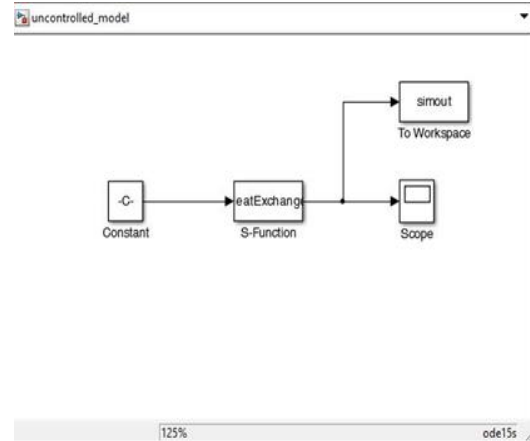
Fluid	Notation	Air	Water
Inlet Temperature	$T_{in}$	320.15 K	303.15K
Outlet Temperature	$T_{out}$	-	-
Mass flow rate	-	0.034 kg/s	1.38 kg/s
Density	$\rho$	1.092kg/m <sup>3</sup>	996kg/m <sup>3</sup>
Surface Area	$A$	0.1 m <sup>2</sup>	0.1 m <sup>2</sup>
Inlet Velocity	$V$	0.31136 m/s	0.013855 m/s
Dynamic viscosity	$\mu$	1.9341 e-5kg/m.s	0.798 e-3 kg/m.s

The direction of the inlet velocity is normal to the surface. The inlet velocity was calculated from channel mass flow rate and channel inlet cross-sectional area. The mesh structure specifies the resolution at which COMSOL Multiphysics® analyses the model. Before analyzing the heat exchanger model in COMSOL Multiphysics, it was necessary to create a mesh structure. Therefore, a grid independence study was carried out to ensure the results accuracy.

For final simulations, the mesh was created for the heat exchanger using triangular mesh with around 5,910 cells and about 1,144 quadrilateral cells to ensure the simulated results are grid independent. Edge elements and Vertex elements were about 485 and 6, respectively. The segregated solver was used to solve the governing equations for the conservation of mass, momentum and energy equations. The semi-implicit method for pressure-linked equations algorithm was used for the calculation of the pressure and the velocity, which were needed for the solution of the energy equation. The numerical domain consisted of 500,000 elements and a first-order upwind differencing scheme was used at all interior grid points. The solution was converged and the overall heat balance throughout the domain was reached (Sarairoh, 2012).

The COMSOL model was saved as a '.m' file, which automatically translated the entire work done in COMSOL to a MATLAB code. The MATLAB software was then launched and the program was initiated. This MATLAB code was then edited in order to ensure it was adaptable in Simulink later on. This '.m' file is then saved to be used as an integral part of the Simulink model in an 'S-function'.

SIMULINK block sets are arranged to represent the equation and its parameters are inputted accordingly after which a simulation can be run to give results. From the library, the appropriate block sets for setting up this model such as the input (the constant), output (the scope), S-Function and simout blocks are dragged into the SIMULINK Model. This model is set-up in order to show the output/resulting temperature profile varying with time without control. This model is saved as the 'uncontrolled model'. Figure 3 shows an uncontrolled model created in SIMULINK. The solver configuration is set. The blocks are linked up by arrows conveying signals.

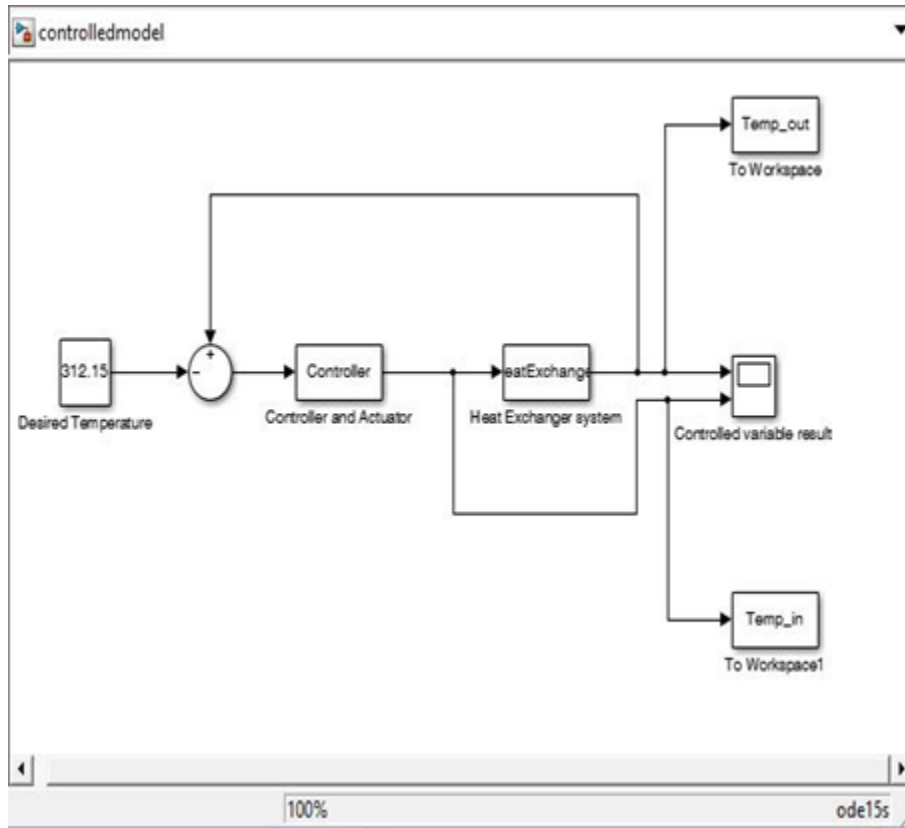


**Figure 3:** An Uncontrolled Model Created in SIMULINK.

Simulation of this model is then carried out by clicking on the play icon after the simulation time e.g. 50s has been inputted. This gives a plot of the outlet temperature profiles of the counter-flow heat exchanger side without control. A MATLAB code is written to serve as the controller. This code is later integrated into Simulink in an S-Function block.

The appropriate block sets for setting up the counter-flow heat exchanger model such as the input (the constant), output (the scope), S-Function, sum and simout blocks are dragged into the SIMULINK Model. In this model, there are two S-function blocks. One of which houses the MATLAB code of the heat exchanger model from COMSOL, while the other is the MATLAB controller code. The sum block serves as a comparator, evaluating the returning signal from heat exchanger outlet with the desired temperature at which the controller will regulate the temperature.

The controller block, as stated earlier, is a code written in MATLAB with the aim of governing the periodic results of the counter flow heat exchanger. Now, the desired temperature or temperature range can only be a wild guess, judging from the fact that the essence of this work is to first observe the nature of the temperature output (which is the Water outlet temperature) with respect to time. It is from the result of the uncontrolled model that a particular temperature range is selected to serve as a desired set point temperature which the controller ensures that the new resulting temperature does not exceed.



**Figure 4:** A Controlled Model in SIMULINK.

The controlled parameter can only be one that, when varied, affects the water outlet temperature. The Air inlet temperature was made to be the controlled parameter after running series of results in order to determine a suitable parameter that efficiently regulates the resulting temperature (this is briefly explained in later sections). This is where the controller comes in. The 'set point temperature' block is connected to the comparator and controller.

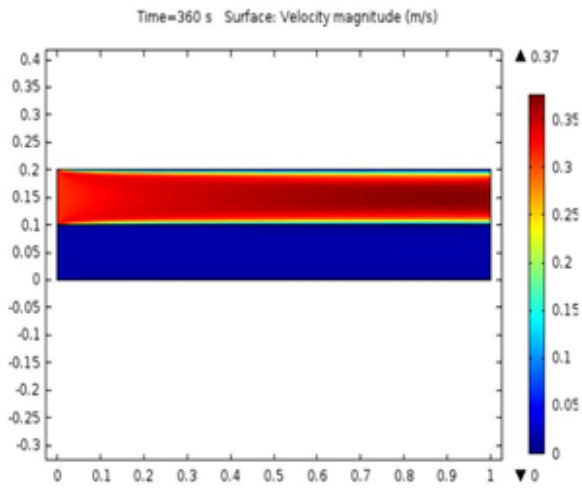
A signal is sent from the controller to the Heat exchanger system, that same signal is branched out and sent directly to the scope (the variable result). A feedback/return loop is sent from the output signal from the heat exchanger system back to the comparator. The reason for this loop is that, after the first signal is sent, the resulting temperature data is sent back to the sum block and compared to the set point temperature, the sensor detects the resulting difference and determines whether the controller increases or decreases the controlled parameter. For instance, if the returning temperature (the most recent

temperature output from the heat exchanger system) is higher, the sensor detects the difference and triggers a decrease in the controlled parameter by multiplying it by a 'reducing-factor/value'. This process continues until the stipulated end time is reached. When the returning signal which conveys the outlet temperature from the heat exchanger is less than the desired temperature, the air inlet temperature ( $T_{in,a}$ ) is reduced by multiplying it by a factor (0.995), this is done until the returning temperature is greater than the desired temperature, in this case, the air inlet temperature ( $T_{in,a}$ ) is increased by multiplying it by a factor (1.005).

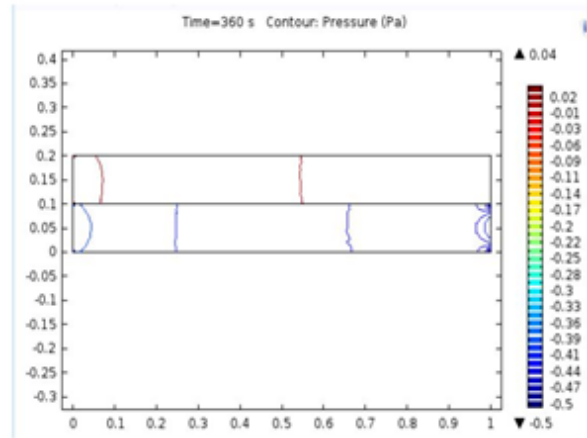
## RESULTS AND DISCUSSION

Figures 5 and 6 show the velocity distribution in both domains from zero seconds up till 360 seconds. It can be seen that the hot air coming into the counter-flow heat exchanger moves at a

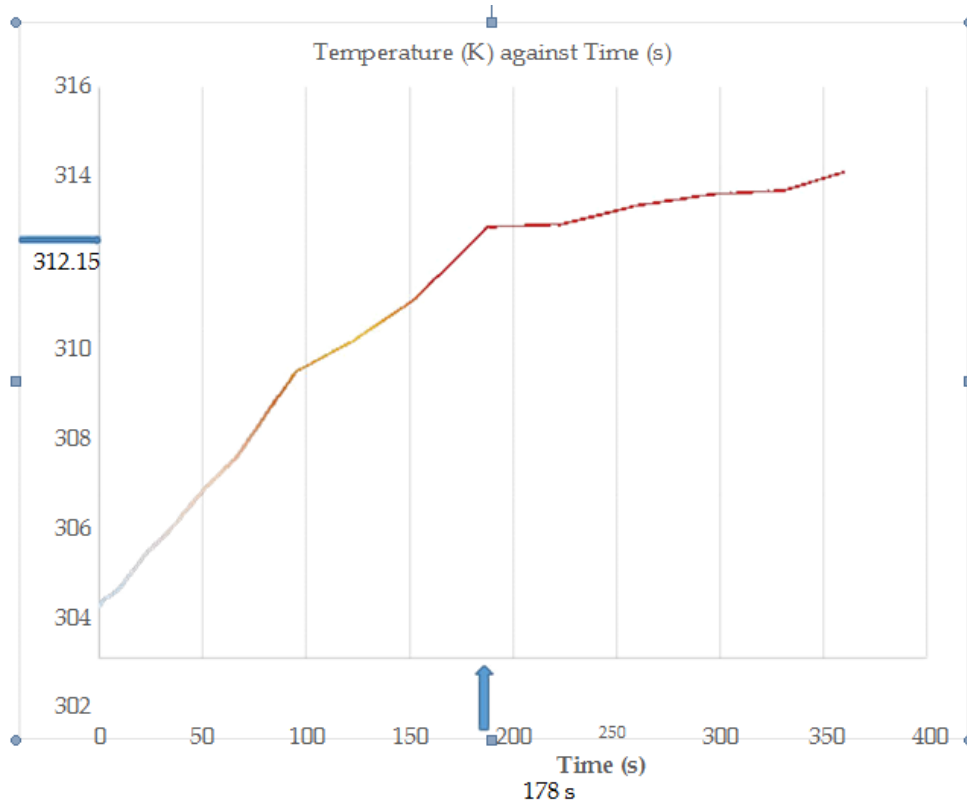
slightly higher velocity than the water in domain 2 over time.



**Figure 5:** Velocity Magnitude Distribution across the Domains at 360 seconds.

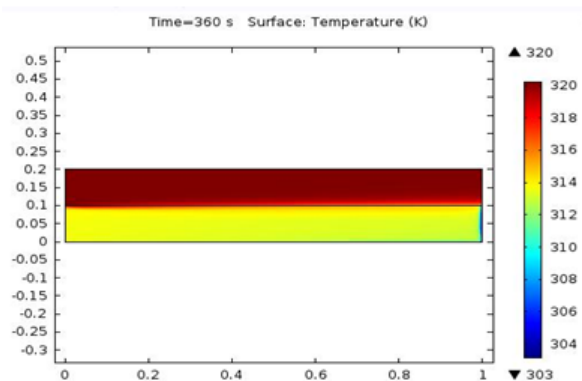


**Figure 6:** Pressure Distribution across the Domains at 360 seconds.

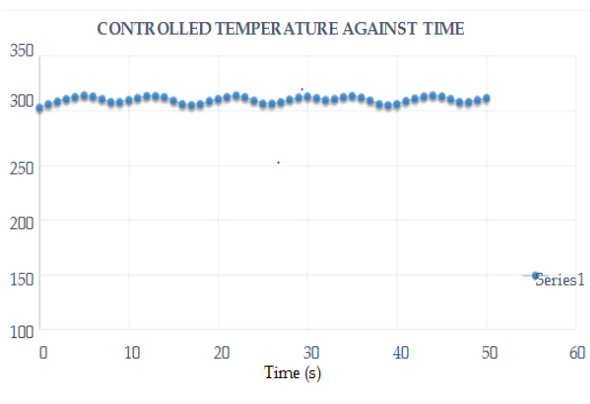


**Figure 7:** Plot of Water Outlet Temperature varying with Time.





**Figure 8:** Plot of Water Outlet Temperature varying with Time with the Desired Temperature for Control.



**Figure 9:** Plot Showing Controlled Temperature against Time.

## CONCLUSIONS

In conclusion, the controller showed efficient Temperature control of the counter flow heat exchangers in general. Optimal control of the temperature of the heat exchanger outlet fluid temperature was achieved and the steps employed can be used in the design of controllers for other counter flow heat exchangers. The results of the simulation were successfully validated against results available in the literature.

The primary advantage of this simulation is the ability to directly include the code in the model of a larger system. Another advantage of this code is that it is written in a current and convenient programming language with sufficient

documentation to make it portable and usable by other engineers. However, only computers with good processing speed can run these simulations conveniently. The higher the processing speed of the computer, the longer the period of simulation can run and achieve results in reasonable time.

Successful development and validation of the influence of the controller on the water outlet temperature was the primary result of this project.

## REFERENCES

1. Srivastava, N., D.K. Tanti, and Md.A. Ahmad. 2014. "MATLAB Simulation of Temperature Control of Heat Exchanger using Different Controllers. Automation". *Control and Intelligent Systems*. 2(1):1-5. doi: 10.11648/j.acis.20140201.11
2. Reddy S., et al. 2012. "Optimization of Shell and Tube Heat Exchangers for Sea Water cooling by COMSOL Multiphysics". *International Journal of Emerging Technology and Advanced Engineering*.
3. Van Schijndel, A.W.M. 2014. "A Review of the Application of SimuLink S-Functions to Multi Domain Modeling and Building Simulation". *Journal of Building Performance Simulation*. 7(3): 165-178.

## SUGGESTED CITATION

Ohue, I.E., M.O. Petinrin, O.E. Zebulun; and R.A. Kazeem. 2018. "Control Modeling of Fluid Temperature in Heat Exchangers". *Pacific Journal of Science and Technology*. 19(1):5-11.