6AS3 Final Project

Applying Machine Learning To Industrial Control

Reg Marr, Tom Versteeg

12.10.2017

SEP6AS3 - Advanced Systems Component Control and Integration

Introduction

This project utilized a neural network to develop a model for the process of separating ethanol and water components through distillation.

The still will attempt to more efficiently control the reflux ratio by controlling a mesh valve within the reflux pipe. This will provide a secondary output for the neural network to control and use to maximize output composition control and throughput efficiency.

Distillation is one of the most common industrial processes utilized for a variety of industrial processes. Often the process is done in a batch process where the distillation unit (often referred to as a “still”) is sized such that the batch can be distilled in a single run. This is often done in part due to an ongoing non-linear process being easier to run without disturbances. However this often requires equipment oversized and requiring more cost than would otherwise be needed if the still could be run while accounting for disturbances caused by adding cool pre-product to the still.

Neural networks have been found to produce highly optimized models (particularly for nonlinear chemical processes such as distillation) and as such have the potential to control the process in such a way that the system can be controlled in an efficient manner.

GOALS & OBJECTIVES

The overarching objective of this project was to determine the effectiveness of using models developed by neural networks versus the classical application of PID controllers. The effectiveness was to be measured in relation to overall energy consumption, throughput, and time required to complete the distillation. A distillation (for the purposes of this project) was to be considered complete when the production of ethanol has met within 10% of the estimated potential product volume. The neural network was to be considered viable when it has been trained and can effectively model the interactions of the input layers effect on the process output with a finer tuned response than the PID controller.

Fundamentally the goal of this project is to develop a deeper understanding of modern control systems and what role machine learning may play into them.

The tasks of this project consist of: Seamless integration of Dynamic Data Exchange (DDE) for communication to Matlab using the OPC server from the remote PLC, training the neural network in real time using Matlab, increasing throughput efficiency as compared to PID control. The trained neural network will then produce a model for model predictive control (MPC). Additionally this project will be using the Cogent Data Hub to create a web based HMI that follows the standards/ guidelines of Situational Awareness, and provide access from a remote terminal for real time access. The data analytics will be completed using a MySQL server for historical trending.

MATERIALS

Resources	#	Description & Use
Reflux Control Valve	1	The Reflux Valve was used with a Arduino, to control the position of the valve and to feed back the position to the MySQL database
20 litre keg	1	Standard 20L keg with bushing for instrumentation welded on
Short Packed Distillation Column	1	Schedule 40 Food Grade Stainless Steel pipe, with bushings welded on for instrumentation
Vapour Condenser	1	¼ Copper Tubing Wrapped in a helical manner as a heat exchanger to condense the ethanol vapours

Solid State Relay	1	Used to control the 120VAC going to the Immersion Heating Element
2000W Immersion Heating Element	1	Used to heat the wart up to boiling temperature in the keg
0-30 psi Pressure Sensor	1	Used to measure the pressure of the basin
PT100 Resistance Temperature Detector	2	Used to measure the Vapour and Basin Temperature
Endress Hauser RSG30	1	Datalogger used to collect analog inputs from the distillation column
CompactLogix 5370	1	Programmable Logic Controller, that relayed the outputs from the distillation column sensors to the KepServer over OPC UA
KepServerEX	1	OPC UA Server with datalogging to MySQL and Modbus abilities
MySQL Database	1	Central Database hosted on the Control PC
Custom Neural Network Application	1	Custom Neural Network application used to calculate optimal settings for the Still to obtain a desired Output at any point in the Distillation Process

Distillation Column Piping & Instrumentation Diagram & Pictures of Apparatus

HYPOTHESIS

While current PID based control systems can competently account for some level of disturbance new studies have shown that back propagation neural networks, particularly multilayered perceptron (MLPs) are better equipped to develop optimized models which can often account for disturbances and control complex outputs more effectively. The overarching goal is that throughput efficiency will increase and a greater level of composition control will be attained in using a MLP neural network as compared to a simple PID controller.

Neural Network Implementation outline

The NN implementation plan and design can be broken down into the following components and subcomponents.

Neural Network Design:

Topology

The initial topology proposed is one with two hidden layers (with six and three neurons respectively). This is likely to be altered during the first few training runs however there is reasoning behind this initial setup.

It is expected that this Neural Network (hereafter referred to as NN) will require some deep learning capacity due to the relative complexity of the process being controlled and such will require multiple hidden layers. It is recommended that the number of neurons in each hidden layer starts with an initial size representative of the number of relevant factors (in this case this is more than the number of inputs) and scales down towards the number of outputs.

Use of Bias Neurons

Each compatible layer will also include a bias to start, this is done somewhat arbitrarily as there is no known reasoning as to why this would be necessary, however it seems common practice to include bias.

Input configuration

The inputs chosen include the column temperature, basin pressure, Alcohol production rate of the last measurement, change to the alcohol production rate since the last measurement, and current run time. It has also been proposed to include the basin temperature and column vapour pressure. If this is included the inputs will be altered such that temperature and pressure are represented as a ratio between column and basin (thus the number of inputs would not increase).

Output configuration

The proposed output configuration is set to include both the desired duty cycle for the heating element and the position of the reflux ratio control valve. Although this configuration is somewhat outside of the norm it is not unprecedented, and given the NN operates with a certain level of abstraction it is hoped that this configuration will operate appropriately and result will be an operation type that falls somewhere between a softmax and regressive NN.

Neural Network Training Mode Operation:

Training Data Source

The training data will come from a MySQL database that receives data from the OPC UA server over ethernet and can be interacted with the neural net by use of the MySQL C API. It is expected that we will only need one or two batches logged to have sufficient data as each batch runs for about six to eight hours and data is logged at approximately one second intervals (yielding a range of between 21,600 to 28,800 samples and 108,000 to 144,000 data points per batch). Note that the number of samples can be reduced by only feeding the NN points at greater than the default interval (this is likely to be the case anyways as the training intervals should match the run time intervals).

Neural Network training process

Once data has been collected the NN will undergo training using backpropagation (using stochastic gradient descent) with the above proposed topology and a variety of activation functions (Sigmoid, ReLu, or Leaky ReLu), learning rates, momentum gains, and error smoothing factors. The initial training will be done on a randomized sampling of 50% of the available data. This will determine any topological changes that need be made and suggestions towards the most effective activation function, as well as some indication of the appropriate settings for learning rates, momentum and error smoothing factors. Once this step is done another 25% of the available data will be used to validate the settings and determine with greater confidence what changes, if any need be made. At this time the topology and activation function is set and the bias and weight setup is saved. Finally the remaining 25% of the training data will be used to test the operation with the determined settings and structure. Note that testing is done with the NN model and no backpropagation.

Note this will be done using the custom Qt Creator (aka C++ with gui program) application.

Neural Network Runtime Mode Operation:

Runtime Data Source

The runtime data will be retrieved from KepServerEX by using a linux OPC UA client driver to integrate the custom program within the industrial control setup.

Runtime Operation

During runtime the C++ program must be set to runtime mode and will disable backpropagation, at this point is is simply operating as a complex model which returns two control values based on the receiving five inputs as arguments. The only input considered a control input will be the desired change in alcohol production rate. This will be a variable which is determined by use of a function will undergo IF THEN and ELSE statements based on the users custom or default input. This functions operation will resemble minimalistic fuzzy logic.

The NN output control variables will be sent to the PLC along with a tag indicating the NN is operating in run mode and the PLC will use these tags as control variables if the NN tag indicates run mode and the user has also indicated that the process should operate using the NN control variables. If the user does not wish to use the NN control variables then the PLC will simply control the process using the column temperature to respond using PID.

STILL DESIGN

Description

The current still design is a work in progress and is based on relevant literature and resources including consultation from experienced distillers.

The still contains a heated basin, a single recycling column, and a reflux column mounted to the recycling column. This design was chosen as it will best demonstrate the impact of the two control methods and will be relatively easy to operate.

Visual Representation

Data Management Architecture

The data management architecture is found under data management for neural network still in the appendix.

Neural Network Design

Visual Representation

The Visual representation of the neural network and how it integrates within the control system can be found in the appendix under neural network topology.

Comments and Considerations

Note that a diagram has been attached to this document illustrating the proposed integration design. Many of the features proposed in this document are to be considered starting points and may be altered during training and runtime testing dependent on error rate and overall operation quality.

Conclusion

Over the course of this projects implementation we have managed to develop a complete control system that is very analogous to a real-world industrial distillation system. This system is able to be controlled directly over KepserverEX and Cogent Datahub. Additionally we have developed a neural network application which is able to train based off of a mySQL connection or a txt file, it is also able to control the system via a direct Modbus TCP connection (Modbus RTU is expected to be added soon).

A more in depth review of the systems operation can be found in the appendix with a video link titled system operation explanation.

While we were not able to fully test the efficacy of neural network control vs PID control this project serves as an excellent starting point from which further development can be done.

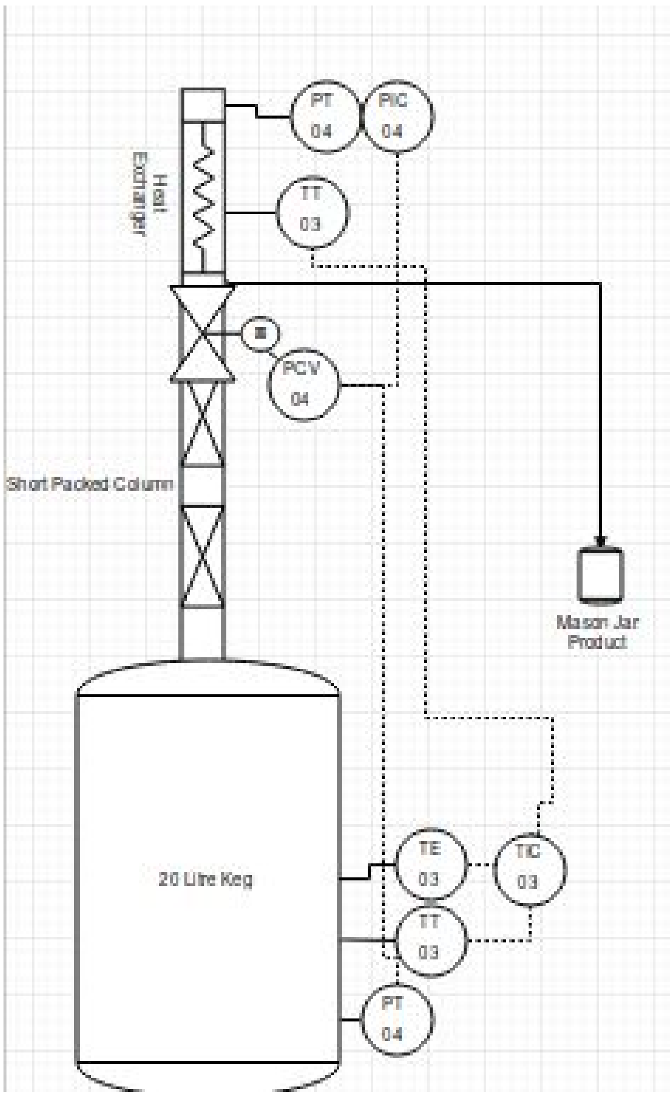
Furthermore this project was able to show that a complex industrial control system could be developed from the ground up using relatively inexpensive components and inexperienced engineering students. It is recommended that this project report be used as a guideline for future endeavours related to the application of machine learning and other methods of note to real-world industrial systems.

REFERENCES

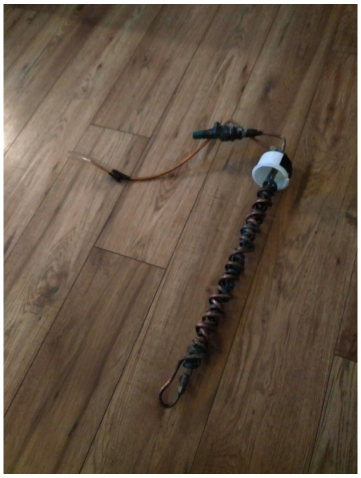
All References can be found here:
https://drive.google.com/file/d/0B1pwjjC84BX5SVBodzZpSVFYQzg/view?usp=sharing

Appendix

Piping and Instrumentation Diagram

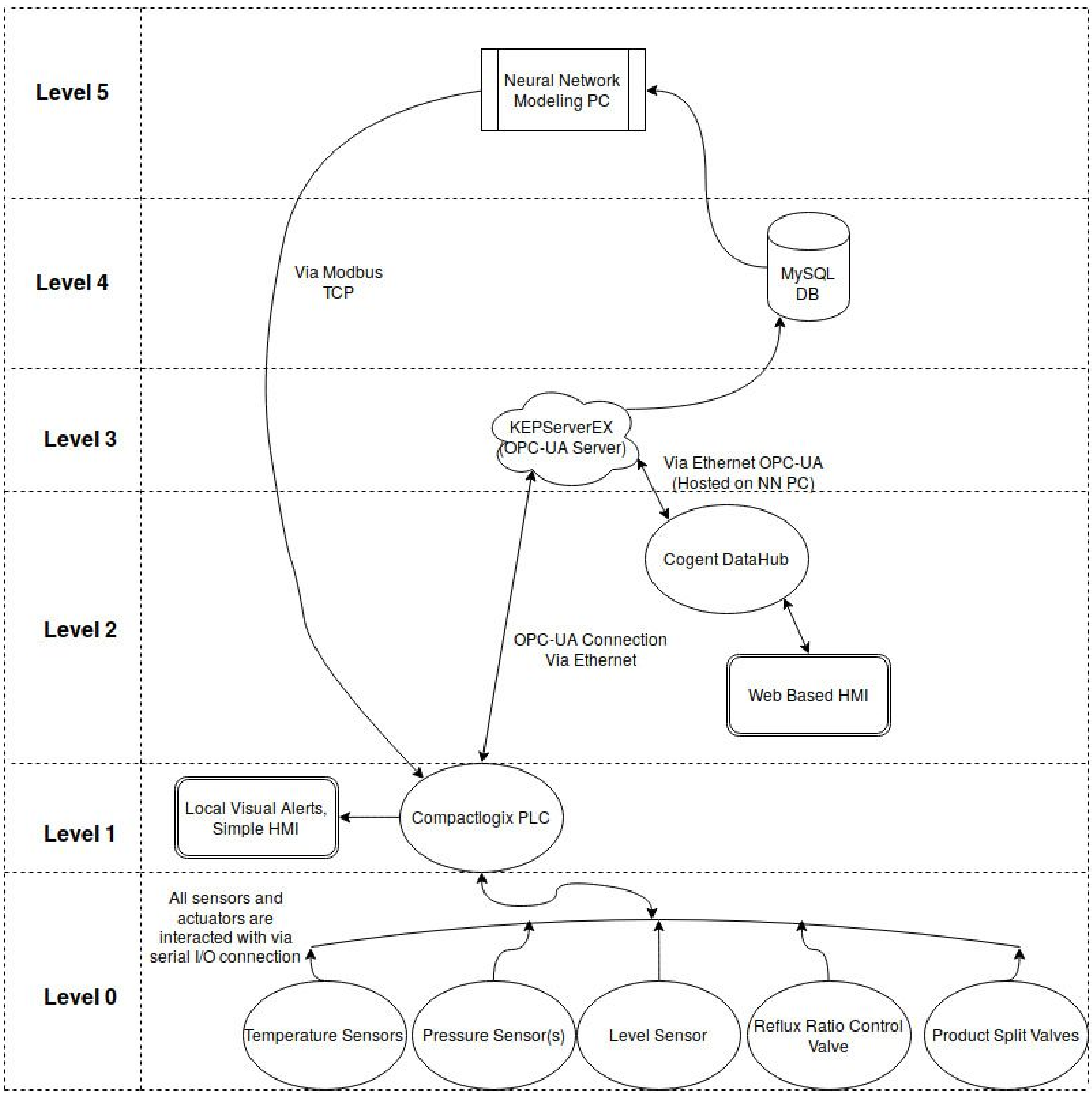


Pictures of Apparatus



Data Management for Neural Network Still

Data Management Architecture



System Operation Explanation

Neural Network Topology