Characterization and Optimization for Liquid Cooling Technology

This project integrates automatic gain tuning into CoolIT's Coolant Distribution Unit (CDU), alleviating the demand for manual on-site tuning. System characterization in MATLAB yielded a linear system model for the valve and pump. With a training and validation data set, the valve achieved a 62.6% and 56.6% fit while the pump achieved an 81.3% and 91.1% fit respectively. Initial Ziegler-Nichols gain tuning in Python resulted in reduced valve gains by 47% and improved stability while reducing overshoot by 5.5%. Pump gains decreased by 55% and also improved stability. Relay Auto-Tuning algorithm was developed in Python to automatically determine gains, ensuring a stable system response with sufficient response time. Further exploration into Model Predictive Control (MPC) and using a non-linear model for the valve could help further system optimization capabilities.

SCHULICH School of Engineering

CoolIT systems™

Authors and Acknowledgements

Students

A.D. Mohammad, Jason Chiu, Safwan Reza, Tejbir Minhas, Michael Hanson, Justin Hui

Mentors

Dr. Reza Najjari (P.Eng, Ph.D.)

Chevy O'Dell (E.I.T.)

Faculty Advisors

Dr. Mahdis Bisheban

Dr. Jeff Pieper

Our Problem

Currently, any issues with the controller systems require manual gain tuning by a technician. This creates a large burden on CoolIT and their technicians. The risk of error and resource demand increase as this process continues to happen. This capstone aims to alleviate this issue.

Our Goal

The goal can be broken into the following:

- **Characterizing the System through Advanced Modeling**
- **Optimizing Controller Gains**
- **Developing Auto-Gain Tuning Algorithm**

Our Solution

Through on-site testing of the CDU, the system dynamics were characterized in MATLAB giving insight into how it interacts with operational changes. Using Python, gains for the controller were determined and the Auto-Gain Tuning algorithm was developed.

Results

A linear system model was created in MATLAB Simulink using a transfer function. Using the training and validation data sets:

- **The valve achieved a 62.6% and 56.6% fit respectively**
- **The pump achieved a 81.3% and 91.1% fit respectively**

With the Ziegler-Nichols closed-loop tuning method, the initial set of gains were determined and implemented using Python. Performance was benchmarked and the improvements were measured through better stability and response.

- **The valve gains were decreased by 47%, improving stability and reducing oscillations. Overshoot was reduced by 5.5%.**
- **The pump gains were decreased by 55%, improving stability.**

A Relay Auto-Tuning algorithm was developed in Python to automatically identify and determine the required gains for the specific system. The determined gains provide a stable system response with sufficient response time and reduced overshoot.

Discussion

- **Model Predictive Control (MPC) system modelling was initially explored as an advanced control technique. Further investigation into MPC could help optimize the system's performance.**
- **The non-linear model for the valve performed better, but the design for gain tuning and MPC modeling with a non-linear model would become significantly more complicated. Non-linear modelling could be explored in the future to help find further optimization opportunities.**

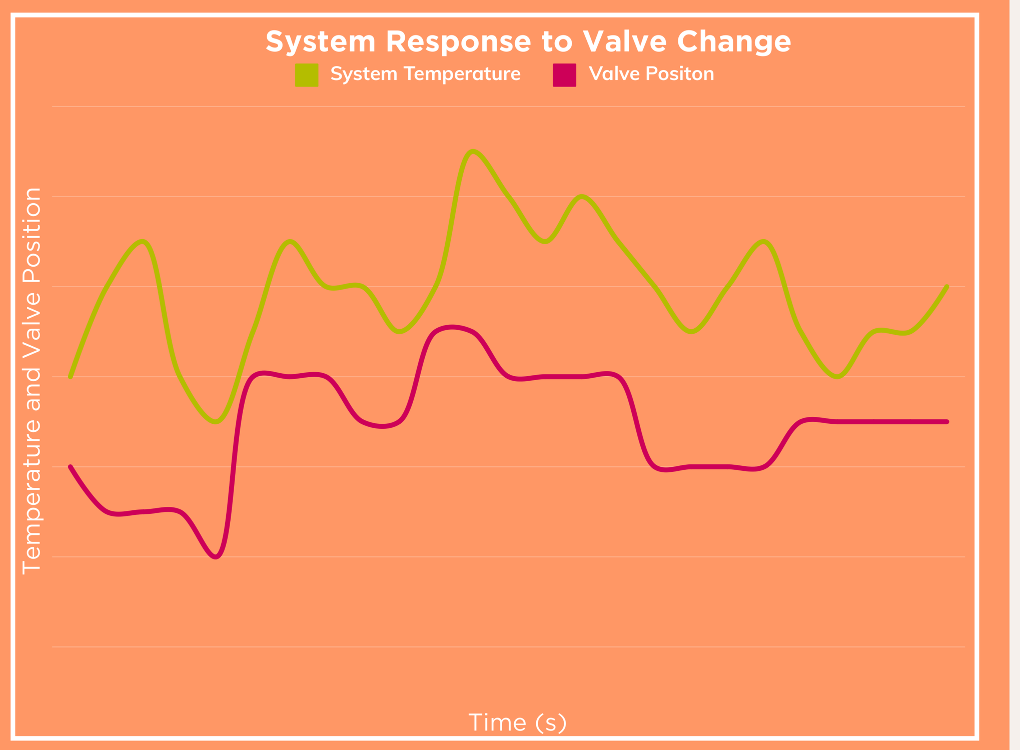
Methods and Analysis

Performed on-site testing of the CDU to collect necessary datasets, which were subsequently analyzed utilizing advanced control system theory to gain a comprehensive understanding of our system.

Test Plan

Repeatability Test	Sustained Random Input Test	Sustained Random Input Test
Verifed system performance and ensured data collected is reproducible	Increased hold time of random input to obtain reasonable input and response training data	Sine waves applied as inputs to mimic real world system performance and used for validation

Test plan execution gave us the data to understand and model our physical system and its dynamics.



System Characterization

Characterizing the system was a multi-step process that involved both iterative design and a fine tuning of modeling methodology.

The system was defined by Two Linear Models

Transfer Function Model

$$\frac{as + b}{cs^2 + ds + e}$$

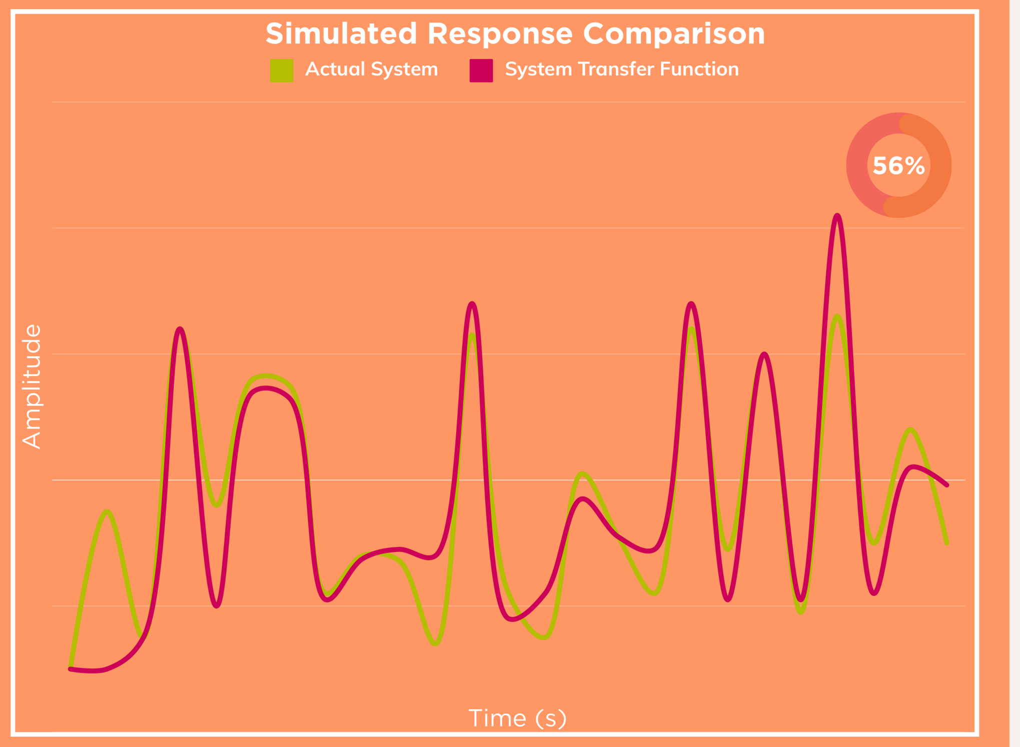
The number of poles and zeros was modified

State Space Model

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

The size of the coefficient matrix was modified

- First, the initial actual system data needed to be evaluated on the influence of noise by looking at the residual correlation between the actual and estimated data
- Second, model parameters such as state space matrix size or transfer function poles and zeroes needed to be determined
- Finally, a model generated using a chosen methodology such as least square estimation would be produced with a percentage fit comparison for model evaluation



Auto-Gain Tuning

Initially, Heuristic Tuning was applied to attain an initial understanding of the systems transient response characteristics. Ziegler-Nichols table was applied via ultimate gains obtained through sustained oscillations.

- Square wave signals are sent as inputs, causing the output signal to oscillate. Once a steady state response was achieved, the ultimate gain and period can be identified.
- Using the Zeiger-Nichols table, the PID gains of the system are determined.

The entire gain tuning process is automated.