Study of fuzzy logic in safety and control systems in Nuclear Power Plant

By

Harsh Deol

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Master of Applied Science

in

The Faculty of Energy Systems and Nuclear Science

Nuclear Engineering

University of Ontario Institute of Technology

March 2016

Abstract

Safety and control systems in Nuclear Power Plants are essentially run without intelligent systems. As nuclear industry starts to become more advanced and vendors are starting to manufacture more advance products, old structures and systems in place will eventually become obsolete needing alternatives to resolve discrepancies. Control System is considered as one protective layer with plant safety systems. Research demonstrates that intelligent control systems improve safety performance of power plants. This research studies fuzzy logic control (i.e. intelligent control system), its methodology, design and applicability to safety related systems in Nuclear Power Plant. Study further investigates fuzzy logic control approach and how it impacts improving safety related system performance. Critical system parameters and trends were studied, research plan developed and results recorded using Matlab & Simulink applications. Two case studies selected demonstrate proposed fuzzy logic with effective results. First study demonstrates how implementing fuzzy logic improves operability of self-contained pneumatic assemblies that impact improving breathing air system performance. Second study demonstrates how fuzzy logic implementation can improve instrument air compressor performance with early detection of solenoid valve failures. Both studies reveal that using fuzzy logic results in better control, precision and performance in maintaining safety related systems and encourages its use to many other applications.

***Key words**:* PID controller, fuzzy logic controller, control system design in nuclear power plants, breathing air system, reliability, safety instrumented system, risk reduction factor, instrument air, compressor, service water

Acknowledgements

I would like to thank Dr Hossam A.Gabbar and Energy Safety and Control Lab (ESCL) members for their guidance and support during my study period. I am extremely grateful for feedback, coaching and mentoring to help complete this research.

Many regards to Nuclear Engineering faculty for giving me the opportunity to pursue Masters of Nuclear Engineering (MASc) and providing balanced environment and resources to help me become better nuclear professional.

Thanks a lot to all my classmates, family members and specially my wife, Navdeep Deol, for being there for me and guiding me help achieve success.

Thank you to all for being there for me.

Table of Contents

List of Figures

List of Tables

List of Acronyms

SIS	Safety Instrumented System
PNGS	Pickering Nuclear Generating Station
OPG	Ontario Power Generation
CP	Air Compressor
RB	Reactor Building
RC	Air Receiver
IA CP	Instrument Air Compressor
HX	Heat Exchanger
SV	Solenoid valve
FT	Flow transmitter
PM	Preventative Maintenance
PdM	Predictive Maintenance
CM	Corrective Maintenance
PRA	Probability Risk Assessment Model
SCR	Station Condition Record
CAP	Corrective Action Plan
SPMP	System Performance Monitoring Plan

FMEA	Failure Mode and Effect Analysis
ECR	Engineering change request
MOD	Modification
WR	Work Request
PFD	Probability of failure on demand
SIF	Safety Instrumented Function
SIL	Safety Integrity Level
PFD	Probability of failure on demand
RRF	Risk reduction factor
VVVF	Variable-voltage and variable-frequency hydraulic system

List of Symbols

$T_{water,in}$	$Compressor\ water\ temperature\ in$ (℃)
$T_{water,out}$	$Compressor\ water\ temperature\ out$ (℃)
$T_{oil,in}$	$Compressor\ oil\ temperature\ in$ (℃)
$T_{oil,out}$	$Compressor\ oil\ temperature\ out$ (℃)
C_{water}	$Heat\ capacity\ rate\ of\ water\ (^{KW}/_{\circ C})$
C_{oil}	$Heat\ capacity\ rate\ of\ oil\ (^{KW}/_{\circ C})$
$c_{p,water}$	$Specific\ \ heat\ of\ water\ (^{KJ}/_{Kg}\cdot{}^{\circ}C)$
$c_{p,oil}$	$Specific\ heat\ of\ oil\ \ (^{KJ}/_{Kg}\cdot{}^{\circ}C)$
U	$Overall\ Heat\ Transfer\ Coefficient\ (\frac{W}{m^2}{}^{\circ}C)$
$\dot{m}_w$	$Water\ mass\ flow\ rate\ (^{kg}/_{sec})$
$\dot{m}_O$	$Oil\ mass\ flow\ rate\ (^{kg}/_{sec})$

R_f Fouling factor $(m^2 \cdot {}^\circ C / W)$

Nu Nusselt Number

ρ_w Density of water (Kg/m^3)

ρ_o Density of oil (Kg/m^3)

Pr Prandtl number

k Thermal Conductivity $(W/m \cdot {}^\circ C)$

μ & v_w Kinematic Viscosity (m^2/s)

h_i Water Convection heat transfer coefficient $(W/m^2 \cdot {}^\circ C)$

h_o Oil Convection heat transfer coefficient $(W/m^2 \cdot {}^\circ C)$

D_h Hydraulic diameter (m)

A_c Area of circular tube (m^2)

A_s	$Total\ surface\ area\ of\ inner\ tube\ (m^2)$
$\dot{Q}_{max}$	$Maxıum\ \dot{h}eat\ transfer\ rate\ (KW)$
$\in$	$Effectiveness\ factor\ for\ heat\ exchanger$
$\dot{Q}$	$Actual\ \dot{h}eat\ transfer\ rate\ (KW)$
C	$ratio\ of\ min/max\ heat\ capacity\ rate$
m	$mass\ of\ the\ valve$
$\ddot{y}$	$acceleration\ of\ valve\ movement$
$\dot{y}$	$velocity\ of\ valve\ stem$
y	$friction\ displacement\ of\ valve$
F	$force$
P	$pressure\ imposed\ on\ the\ valve$
S	$area\ of\ valve\ diaphragm$

$K1\ to\ K8$	$constants$
y	$displacement\ of\ valve\ travel$
ΔP	$pressure\ diffrence\ across\ the\ valve$
q	$air\ flowrate\ across\ the\ valve$
∂	$density\ of\ air$
Q	$Total\ volume\ of\ breathing\ air\ in\ the\ system$
Q_0	$Initial\ volume\ of\ breathing\ air\ in\ the\ system$
$q(t)$	$Air\ flow\ going\ into\ the\ system$

Chapter 1 - Introduction

1.1 Background

Safety of workers, environment, equipment and public is of paramount importance to Nuclear Sector. A Nuclear Power Plant produces energy that is used for various purposes in safe and reliable manner. Critical focus of any Nuclear Plant is reliable operation of instrumentation and control systems. Functionality of control systems and its instrumentation serves as a nervous system to the plant [9]. Various detection methods are used to proactively act on problems before they become reactive. Nuclear power plants meet more than 50 per cent Ontario electricity demand [31].Therefore, it is imperative to select systems that would run them reliably and safely at all times.

Value for money for safer operation is also an important factor in operating Nuclear Plants considering its controlled environment, employees and expensive equipment. Serious events such as reactor trip, radiation release, fire explosion or turbine trip may result due to equipment failure or undetected errors in plant operation, which could lead to repair or replacement expenditure. In order to ensure safe operation, control systems with right sensors & detectors to monitor critical parameters are to be purchased with expenditure allocated for safer operation.

Industrial processes are not always defined and modeling them could be challenging task. Experimental modeling at times could also not be feasible since system outputs are not always measurable. When feasible, models are complicated using control algorithms that could reduce control bandwidth to result in unacceptable time lags and often can slow the process [10].

Fuzzy logic is an “intelligent control” system, which generates computer automated control decisions to alleviate problems with simultaneous monitoring, control of speed and variables. Fuzzy rules represent controlling processes to ensure output is of desired quality [10].

Identification of parameters for conventional control is analogous for this approach. Fuzzy logic uses rule based process consisting of IF-THEN rules that relate to defined ranges and represent outputs based on range of inputs that are matched with rules per fuzzy control interfaces.

Focus of this research is to study fuzzy logic in safety and control systems in Nuclear Power Plant. Two safety related systems (breathing air and instrument air system) are selected to analyse scenarios using fuzzy logic to demonstrate results, which signify if use of fuzzy control is advisable in Nuclear Power Plants or not.

1.2 Problem Definition

Safety and control systems in Nuclear Power Plants are essentially run without intelligent systems. As the industry becomes more advanced with computer technology (for the right reasons), it be critical to resolve old equipment & part obsolesce issues that impact Nuclear safety in long run. Research demonstrates installing intelligent control systems will improve safety performance, reduce operational risks and associated costs of power plants.

This research will study fuzzy logic control and two relevant case studies to investigate how self contained pneumatic assemblies that have difficulty maintaining optimized design control can be resolved using fuzzy logic and how problem of using silted lake water to cool equipment that causes compressor degradation can be resolved using fuzzy logic. Analysis of both cause studies is used to further support fuzzy logic implementation to other Nuclear applications.

Study of fuzzy logic to improve nuclear safety is critical in today's time for any power plant. It demonstrates why it is important to use existing and past research for betterment of Nuclear future and demonstrate effective research is put into good use for right reasons. Research conducted in this thesis involves various steps ranging from building intelligent control methods, safety designs, safety life cycle activities, risk analysis, risk reduction, safety system requirements, transfer functions, steady-state error, Kp, Ki, and Kd to ultimately improve plant performance.

1.3 Solution Approach

In order to approach solution to the problem defined in Section 1.2, functionality of fuzzy logic was studied and validated against nuclear safety related applications. Known problems were considered and reviewed for why they have not been solved without intelligent control. What were the implications, consequences and costs for not using fuzzy logic to fix the problem were studied. This included reviewing limitations of safety related systems with conventional PID controllers and other field deficiencies, developing plan to resolve existing deficiencies with fuzzy logic, analyzing system critical parameters that lead to system failures, studying tread of failures and implementing fuzzy logic to improve system performance.

Essentially, this research is looking at an industry problem of not utilizing potential of fuzzy logic (i.e. intelligent control system) in Nuclear sector, developing realistic plan for execution (conducting this research) and providing results (for two safety related case studies) to signify use of intelligent control for other Nuclear applications.

1.4 Objectives

This study focuses on investigation of intelligent control system (i.e. Fuzzy logic) and its application to improve performance of safety related systems. Objectives of the study are as follows:

- Design a self-tuning control system and apply on breathing air system (which is a safety-related system) in a Nuclear Power Plant to enhance performance of the system.

- Design fuzzy logic system to detect solenoid valve failures proactively and improve the performance of instrument air compressors (i.e. safety-related system).

- Determine feasibility of fuzzy logic implementation to other Nuclear applications.

1.5 Research Scope

The scope of research conducted involves studying intelligent control methods, safety designs, safety life cycle activities, risk analysis, risk reduction, safety system requirements, reviewing related journal papers, system performance monitoring plans (breathing air and instrument air systems), health reports and running simulations in Matlab (Simulink) applications to justify use of fuzzy logic to Nuclear applications.

Intent of this research is to prove fuzzy logic is beneficial to Nuclear Power Plant applications and must be considered for its current and future use. This thesis uses two safety related systems as case study to provide justice for fuzzy logic usage and why it must be used in other applications.

1.6 Thesis Structure

Intent of research is to provide justice to fuzzy logic usage and applicability for Nuclear Plants. Chapter 1 starts with introduction, fuzzy logic background, problem definition, solution approach, objectives, research scope and structure. It builds background on why fuzzy logic research is needed to justify its applicability and provides reason for research conducted for betterment of intelligent systems. Chapter 2 covers literature review involving safety related systems, comparison of self-tuning fuzzy logic controllers and conventional PID controllers, challenges, solutions and limitations of implementing resolutions to Nuclear Plant, background on intelligent control systems and current trends of controls systems in Nuclear industry. Chapter is covering fuzzy logic methods, techniques and how it applies to the industry. Chapter 3 provides framework and methodology in form of flowchart to aid research. Chapter 4 covers first case study to design fuzzy logic based self-tuning control for breathing air system, Chapter 5 covers second case study to design fuzzy logic system to detect solenoid valve failures proactively and improve instrument air performance. Chapter 6 covers results and discussions and Chapter 7 states conclusions and future work.

Chapter 2 - Literature Review

2.1 Fuzzy Logic Control

Fuzzy Logic explains the thinking behind human brain with a fact that human reasoning is approximate, non-binary and non-quantitative. In most cases, there are shades of grey but no exact answers. Temperature is the simplest example for this as quite often, people don't say temperature is "25.36 degrees", but as its "pretty cold" or "really hot".

The approach to Fuzzy logic control (FLC) mainly consists of 5 steps as follows:

1. Defining input and output variables
2. Define subset's intervals
3. Select the membership functions
4. Setting IF-THEN rules
5. Adjust rules and perform calculations

The problem of controlling the distance between two cars is another example to explain fuzzy control steps.

Initially defining the input and outputs: The D, distance between the cars, and v, the velocity of following car are the two inputs. The B, amount of braking to apply to the following car (force) is the only one output. Figure 1 pictorially depicts input and outputs.

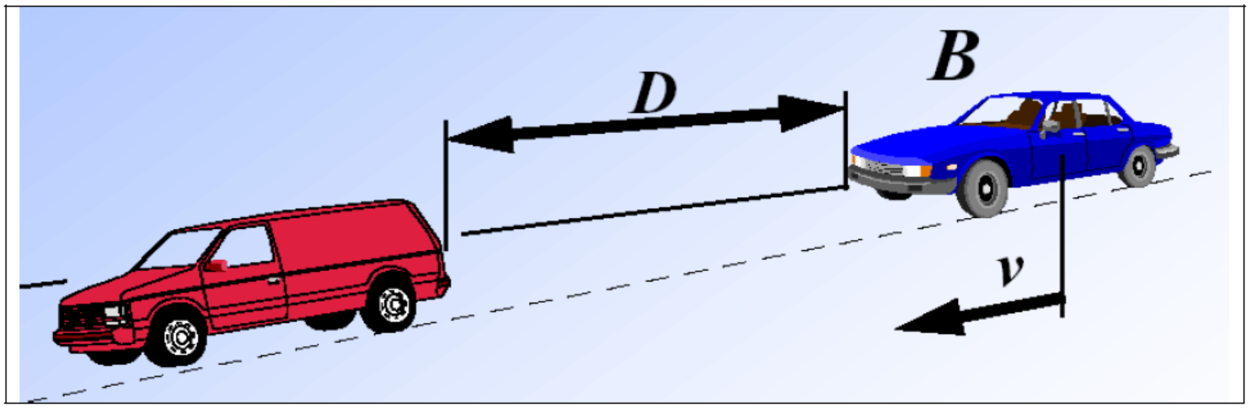

Figure 1: Pictorial description of Fuzzy logic control of the distance between two cars [45]

Second step is to define subset intervals: To make it simple, each variable has chosen three subset intervals. These are small, medium and big for braking force and low, medium and high for distance and velocity. Figure 2 explain these subset intervals.

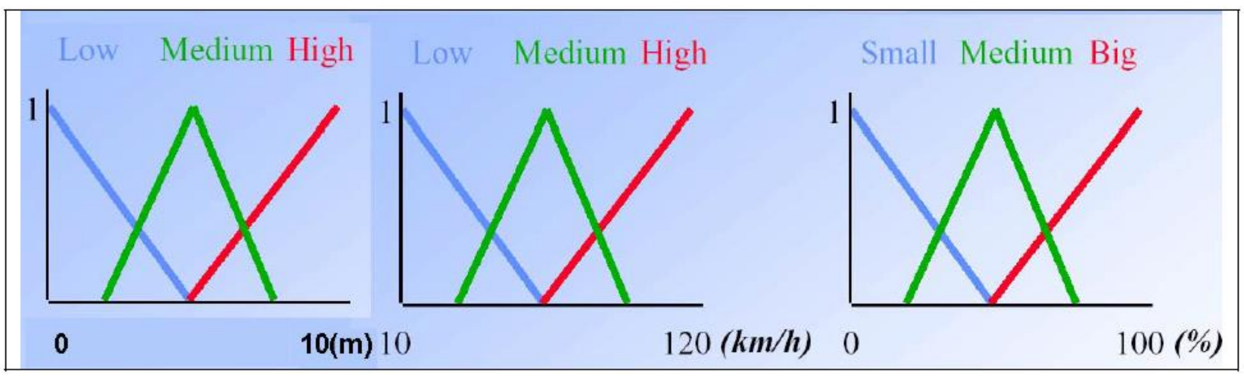

Figure 2: Pictorial description of Fuzzy subset intervals for Distance, Velocity and braking force. [45]

Thirdly, to select membership functions: The membership functions shape be a linear transition between the different subsets, as shown in this example. Figure 2 is illustration of as distance goes from 0 to 5 metres; the membership function for low distance goes down linearly from 1 to 0.

Fourth step is to set the IF-THEN rules: Output is determined by combinations of input. As an example, “IF the distance, D, between the cars is low AND the velocity of the following car is high, THEN the braking to apply is big.”[45]. In same way, it defines other rules with non-quantitative human reasoning.

Adjusting the rules and performing calculations is fifth step: As by rule, to optimally control the vehicles distance non- exact adjustments may be necessary. For example, if the speed of the car is 100km/hr and distance between the vehicles is 2.5 meters. In Figure 2, referring to distance subset, 2.5 meter distance renders into 0.5 low distances plus 0.25 medium distance. Similarly, 100 km /hr speed renders into 0.75 high speed. To determine the output, different methods can be used. We are using two inputs here for two possible subset memberships. A 0.75 high speed and 0.25 medium distance would give 0.25 medium braking. The quantum of braking to be applied for vehicles is based on and computed from the centre of gravity of the area under the breaking curve due to these two portions. There are different techniques for determining the output as for purpose to explain fuzzy logic theory in this example, centre of gravity is used.

2.2 Safety-Related Systems in Nuclear Power Plants and Their Classifications.

Following studies were reviewed to understand positive impact of intelligent systems to Nuclear Power Plants, which further helps to justify usage of fuzzy logic (i.e. intelligent control system) in Nuclear Plants.

Intelligent Control For A Nuclear Power Plant Using Artificial Neural Networks [36]

This paper presented an approach based on neural networks for the control system design of a pressurized water reactor (PWR), which is able to control the nuclear reactor in a robust manner under parameter variations originated from the uncertain parameter α_f and α_c**,** and on the bases of this feasibility study; it is suggested that artificial neural networks could be successfully implemented on the control system of a PWR-type nuclear power plant.

A Case Study In Developing Complex Safety Critical Systems [37]

This paper is reviewing a case study on 'the stepwise development of a distributed control program for a safety critical technical production process by highlighting elicitation of adequate modeling ideas, the development of precise and alterative descriptions of system functions and safety requirements and carry out a careful analysis of specifications and design solutions. The study results demonstrate it is effective to get feedback through trial use and error by using light versions of software specification, design and programming tools supporting the techniques.

An Intelligent Decision Support System For Spare Parts Joint Replenishment [38]

This paper is about integrating the artificial neural network and gene algorithms-based spare parts criticality class identifying system to confirm the target service level, and the web-based joint replenishment IDSS to obtain reasonable inventory control parameters that can be helpful for reducing of total inventory holding costs by modifying the unreasonable purchase applications while maintaining the predefined target service level. Study results demonstrate that the use of artificial neural network (ANN) model can be a persuasive analytical tool in deciding whether the criticality of a spare part should be classified as a category H, M, or L although these classification models do have their limitations, which can be eliminated by increasing classification accuracy of ANN-based spare parts criticality class identifying system (ANNCCIS)to improve the decision support ability of spare parts joint replenishment IDSS (SPJRIDSS).

Advanced Control Of A Steam Generator [39]

This paper is presenting a structure for addressing the problem of the violation of safety limits on the water level which is common at low operating power where the plant exhibits strong non-minimum phase characteristics based on a method of advanced control based on fuzzy model predictive control. As a result from validations of this system a new concept of modular advanced control system designed for a seamless and gradual integration into the existing distributed control system is proposed.

As proposed in this paper, the advanced control system can be integrated without production interruption into the primary distributed control system (the identification, modeling, control and validation stages are done on-line using a real image of the I/0 process data, without affecting the existing control system). Because of high level of interconnectivity between system components, it is necessary to provide the highest independency between communication and control modules of the designed system to achieve unified API of extended generality and extendibility in order to unify access and information retrieval from various wireless and wired technology wherein communication interfaces are developed .A Client/Server architecture for advanced controller that run on the Windows environment with real-time characteristics is proposed too.

Hardware Reliability Prediction Of Computer Based Safety Systems Of Indian Nuclear Plants [40]

For forthcoming Computer Based Systems, new standardized Versa Module European (VME) bus based family of microcomputer boards are developed by Reactor Control Division, BARC. These boards and systems are configured using boards that need to be qualified to stringent requirements of nuclear industry. Paper briefly outlines microcomputer boards' description and qualification tests carried out on the boards. Board failure rate estimation is done by summing component failure rates. The board failure rate is then modified by various factors corresponding to process, environment, reliability growth and infant mortality characteristics. MIL-STD-217 Plus methodology is adopted for failure rate calculation of the components and boards. A fully integrated framework of reliability analysis tools is used that supports Reliability prediction, Reliability block diagram, Fault Tree and Event Tree evaluation, Failure Mode and Effect Analysis as per MIL-STD-1629A and Weibull Analysis. Paper includes details of failure rate analysis of microcomputer boards and results. The paper also presents system reliability analysis carried out for CBS system built using boards. On the basis of sub-system failure rates, various system level reliability metrics like on-demand failure probability, spurious failure probability and system availability can be determined. This is to ensure that the system meet its target reliability values during the design phase.

A Small Climbing Robot For The Intelligent Inspection Of Nuclear Power Plants [41]

This paper is about wall climbing robotic system for intelligent monitoring in nuclear power plants, the robot was designed as a bipedal robot with five degrees of freedom. It was actuated by an embedded controller, which was developed based on an ARM microprocessor and μc/os-ii operating systems. The controller also received and processed tele-manipulation commands from operator. To supply complete information about environment and key devices, robot was equipped with optimal cameras, radiometers, barometer and thermometer. These sensors and their electrical parts constituted the inspection subsystem. Experiment results demonstrate that robot has good capabilities but needs improvement on its movement, wall adsorption ability, enhancing autonomous guidance & control ability, enlarging tele-operation distance, designing anti-radiation electronics and much more.

Control System Of A Small Intelligent Inspection Robot For Nuclear Power Plant Use [42]

This paper is about development of control system for intelligent monitoring robot used in nuclear power plants. The control system is actually a two-level controller, consisting of the host computer and lower computer. The host computer is traditional PC, supplying human-computer interface and also used for mission planning, control parameters setting, monitoring results processing and displaying and so on. The lower computer is an ARM embedded controller. It is directly connected to all actuators and sensors. The real-time operation system uc/os-ii is also migrated to ARM processor which can effectively manage the hardware resource and multiple tasks in real time. Typical experiments verified robot's effectiveness and reliability. Experiment results do demonstrate improvements that are needed in robot's movement, control ability, wall adsorption ability and much more.

Regulatory Review Of Computer Based Systems: Indian Perspectives [43]

AERB safety guide AERBISG/D-25 was prepared to prescribe criteria and requirements to assess qualitative reliability of computer based systems software based nuclear instrumentation. This paper elaborates on the regulatory approach adopted by AERB for regulatory review and control of design modifications in operating phase of Nuclear Power Plants (NPPs). This paper also covers a case study of AERB audit on verification & validation activities for software based safety and Safety related systems used in an Indian plant. Review experience shows that documents provide adequate guidance to qualify software based nuclear instrumentation and control systems. However, use of commercially off the Shelf (COTS) as Pre Developed Software in safety applications at NPPs is still a concern. Further, quantifying the reliability of software used in CBS will go a long way in regulatory decision making.

Intelligent Platform Management Controller For Nuclear Fusion Fast Plant System Controllers [44]

An Intelligent Platform Management Controller (IPMC) is being developed by IPFN/IST. This controller in addition with Shelf Manager module is responsible for management of hardware failure, redundancy procedures and hot swapping of the modules in Advanced Telecommunications Computing Architecture (ATCA) crate. Verification of compatibility between modules that share ATCA resources, the power management of each module, temperature monitoring and fan control are as well as tasks that IPMC has responsibility to manage and programming of ATCA & Advanced Mezzanine Cards (AMC) module firmware, application specific program selection and firmware version control. In this paper, hardware architecture of IPMC implementation at IPFN ATCA modules is also described. The xTCA AMC PCIe Carrier developed by IPFN/IST can be used in applications that require large, fast and distributed control systems such as nuclear fusion experiments, taking advantage of the standard ATCA/xTCA hardware platform management and the enhanced feature of remote reprogramming of FPGA firmware by the IPMC.

2.3 Self-Tuning Fuzzy Logic Controller Compared Against Conventional PID Controllers.

Following references demonstrate self-tuning fuzzy-logic controller is a better choice compared against conventional PID controllers.

“Fuzzy immune PID control in (variable-voltage and variable-frequency) VVVF Hydraulic system” [4] paper proved that conventional PID controller had difficulty maintaining precise pressure in the system whereas, biologically immune and principal adjusted amalgamated fuzzy controller is more effective to maintain system desired pressure to VVVF hydraulic system.

“Fuzzy PID control of intelligent pump” [5] also showed field pressure control problems to aerial hydraulic system solved via designing an intelligent pump. Non linear mathematical model for the pump was developed since load to aerial hydraulic system was complex. A fuzzy PID controlled algorithm was developed to raise output of the load. Simulation was performed and compared against PID controller. Results demonstrated Fuzzy PID controller having better accuracy and rapidity than conventional PID controller in maintaining pressure to the hydraulic system.

“Application of self-tuning fuzzy PID controller on industrial hydraulic actuator using system identification approach.” [6] also demonstrated that self-tuning Fuzzy PID controller is better to optimize electro-hydraulic actuator performance. System Identification technique was used for investigating and estimating mathematical model of the system. Discrete transfer functions were developed, Matlab was used for simulation and fuzzy logic used to tune parameters of PID controller. Results indicated improved performance of hydraulic system with Fuzzy PID compared to conventional PID controller.

“The pump house constant pressure fuzzy self-tuning PID control system simulation”[7] also showed that keeping constant pressure to the water supply system using conventional PID controller produced large delay times and often wasn't reflective of the working condition parameters. A self-tuning fuzzy PID led controller showed better real-time tuning of PID parameters to maintain pressure to the water supply system. Modeling to the system developed in Matlab/Simulink proved that by using Fuzzy led PID controller, short output response is attained and strong robustness was achieved in steady state, PID parameters with no overshoot. It was concluded again that Fuzzy led PID controllers were better solution for complicated pump delay system issues.

"Predictive fuzzy PID control: theory, design and simulation" [8] also reiterated same results. Controller was developed to improve time-delay systems using fuzzy led PID logic. Predictive control concepts and fuzzy PID control were used to develop a structure of a controller based on, on-line model identification, fuzzification, defuzzification, rule base and optimal cost index. Many simulations were performed and advantages to the controller were confirmed. Results indicated predictive fuzzy PID control methods providing better control to complex linear/nonlinear and uncertain systems.

2.4 Challenges, Solutions and Limitations with Impact to Safety and Control System to a Nuclear Power Plant in Ontario.

Research includes reviewing problem at a Nuclear Generation Plant in Ontario wherein, silt (encompassing algae/debris/zebra mussels) in service water is causing constant degradation to instrument air compressors (installed beneath sea level and service water system (with flow diversion impairments) that needs resolution.

a) Problem challenges

1) Silt in service water varies at Nuclear Plant.

Silt contained in Lake Ontario cannot be controlled by Plant Personnel. It could purely be seasonal and carry more quantities of silt in summer compared to winter. Data collected from air compressor walkdowns does demonstrate higher amounts of silt plugging of compressor internals during summer from lake water.

2) Service water intake at Lake water level.

Data collected from various stations does demonstrate more quantities of silt at lake water level (such as in PNGS) compared to underneath (like DNGS). Likewise, DNGS has less station backlog for silt and saves resources to resolve other critical areas [17][19][24].

3) Instrument Air Compressors at PNGS installed in the basement [17].

Location does matter when dealing with silted water. Per design, air compressors at Pick 058 units were installed at the lowest elevation in plant (elev 225) that further enhanced problems pertaining to silt as gravitational pull of silted water is more likely to constantly plug-up SVs and compressor internals contributing to its degradation all the time (diverting Maintenance resources for urgent cleaning). This also leads to higher work request backlogs for the station [17].

4) Breathing air pressure not maintained

The control problem investigated also involves Pressure CVs not operating reliably to maintain design pressure of 620 kPa. Typically, in Nuclear Generation plants, preventative maintenance practices exist to maintain functionality of Pressure CVs. Due to accumulation of dirt (ex rust), they could get stuck closed to further reduce pressure maintaining capability in the system. Consequences include, CVs not regulating system pressure properly and incurring extra costs to the company (e.g., during plant outages, when increased maintenance activities are carried out in the RB (reactor building), breathing air demand goes high but the CVs do not regulate to allow more air to pass through and maintain system pressure at 620 kPa. Therefore, breathing air pressure reduces beneath 550 kPa initiating alarm to the control room and all maintenance activities get stopped resulting in outage delays.)

b) Potential Solutions

1) Dredging at PNGS Forebay

Dredging can be performed at PNGS intake to rid of collected silt from years before to reduce station impact [11].

2) Seasonal Cleaning

Data can be gathered per system surveillance and time based preventative maintenance (PMs) can be implemented for divers to clean station intake channels.

3) Sediment Suction System

There is a sediment suction system installed at PNGS but not operational due to equipment problems. This system can be fully returned to service to reduce silt coming into station. It was operational in the 90s and station did observe less silting amounts at the intake but ever since system has equipment issues, internal station systems are accommodating the impact of silts.

4) Time based/conditioned based flushing of service water system

Instead of cleaning silt particulates at the intake, service water header used to feed air compressor loads can be cleaned/ flushed regularly to avoid plugging of HXs (oil cooler, inter-cooler and after-cooler) inside compressors.

5) Closed loop system installed to feed clean water to air compressors [34]

Supply of service water can be changed to station dematerialized water in a closed loop (to feed instrument air compressors). This way, there are no silting problems and compressor reliability will improve with clean supply of cooling water.

6) Filtration System (i.e. cyclone separators) installed upstream of SVs

Cyclone separators can be installed upstream of SVs to rid of silt feeding compressor internals. Clean supply of service water can be fed to cooling compressors and would lead to less CP trips and internal plugging (Oil cooler, intercooler, aftercooler).

7) Replace SVs with Motorized Ball Valves

To remove SV 'getting stuck' problem, equipment can be replaced with Motorized Ball Valves that would open/ close based on compressor configuration. This way, ball valves would never get stuck in any position and supply cooling service water (containing silt) to compressors. Small hole inside SV to assist with equipment operation be no longer needed (such as ball valves) to resolve the issue.

8) Installing PT (Pressure transmitters) downstream of SVs

PTs can be installed downstream of SVs to measure service water line pressure before water enters compressors. Decrease in line measure would mean SV likely plugged. Annunciations can be installed to inform operators of proactively cleaning SVs (before equipment gets plugged up).

c) Solution Limitations

NOTE: Pickering Station Life is extended till 2020 (proposal in review for 2024). Any solution to a recurring problem is to be cost justified.

1) Dredging

Benefits from dredging activities are short-lived (approx 5 years). Silt taken out of forebay will get accumulated with limited time (approx next 5 yrs) as a recurring problem. Hence, dredging is not be permanent solution and only resolves silting problems in the interim till permanent solution is implemented [11].

2) Seasonal Cleaning/ Sediment Suction System

Calling in divers (for time based PMs) could be expensive as it requires lot of security clearances, approvals and personnel alignment with Operations. It also requires permission to dispose collected silt for further monitoring. Resolution can only be an interim solution as silt can increase anytime of the year. Also, divers cannot be called in short-time frames to remove silt expeditiously [29][30][33].

3) Changing of SVs to motorized ball valves

This is possible but could lead to upstream pipe plugging (before SVs) due to silted water when ball valves are in closed position (i.e. when CP not running). Also, changing of SVs to ball valves is a big modification to existing system and time consuming activity (considering station life till 2020 with proposal in review for 2024). A permanent plant modification could take years for implementation due to various approval levels required.

4) Service Water flush (Instrument air compressors)

This is risky job. Four compressors are installed per unit (5,6,7,8) and common service water header is used to feed all four compressors. Design configuration is such that its hard to isolate one compressor from other for flushing the system. To perform this activity, all four compressors must be taken offline for flushing to be effective. Instrument air supply to any unit cannot be isolated as its available all times. Hence, two units will be inter-tied to perform this activity, which puts both units at risk (as shortage of air on one unit could lead to shutting down both unit reactors) [3][13][17].

5) Closed loop system

This idea is expensive. It requires addition of new system with separate monitoring practices (i.e. system walkdowns, PMs for chemistry sampling, resources to analyze sample results etc). It will be a design change to existing air compressors. Even though the idea can work but considering PNGS life till 2020 (proposal in review for 2024), it is not cost effective.

6) Installing Pressure Transmitters

A Pressure transmitter installed to existing system is also minor modification to existing system design (time consuming activity that could take few years for field implementation). Other than that, annunciations for proactively cleaning SVs are a good measure to maintain compressor reliability till 2020 (proposal in review for 2024).

2.5 Intelligent Control System Methods and Techniques

This section describes concept of intelligent control (IC) at prominent level [45].

2.5.1 No System Modeling

IC works under concept of being controlled system without being precisely modeled. The proper stimulus is contributed by designer to the IC and evaluation is done on the basis of result. The IC is controlled by its own developed model system.

2.5.2 Intelligent Control Examples

Humans can do complicated things being unaware about the mechanism behind them. The following subsections are presenting control problems that are resolved by IC.

Examples of Intelligent Control include fuzzy logic, artificial neural network, genetic programming, support vector machines, reinforcement learning etc.

2.5.3 Artificial Neural Networks

The structure and function of the human nervous system is mimicked by Artificial neural networks (ANN). There are various kinds of ANN methods consisting of hopfield, art, artmap, backpropagation, linear vector quantization designs and few more. Backpropagation is the most common used method wherein, interconnected neurons demonstrate human nervous system. The way the knowledge is stored per human biology is determined by the interconnections between neurons. As shown in Figure 3, electrical pulses travel along the axon which sends the signals between neurons. Axons attached to second neurons by synapse close to a dendrite. A neurotransmitter (small amount of chemical) is released and travels to dendrite when a pulse occurs at the synapse, which triggers a change in potential at dendrite. The electrical pulse triggers along the axon if the strength of all such interconnections is higher than some threshold and the process goes on.

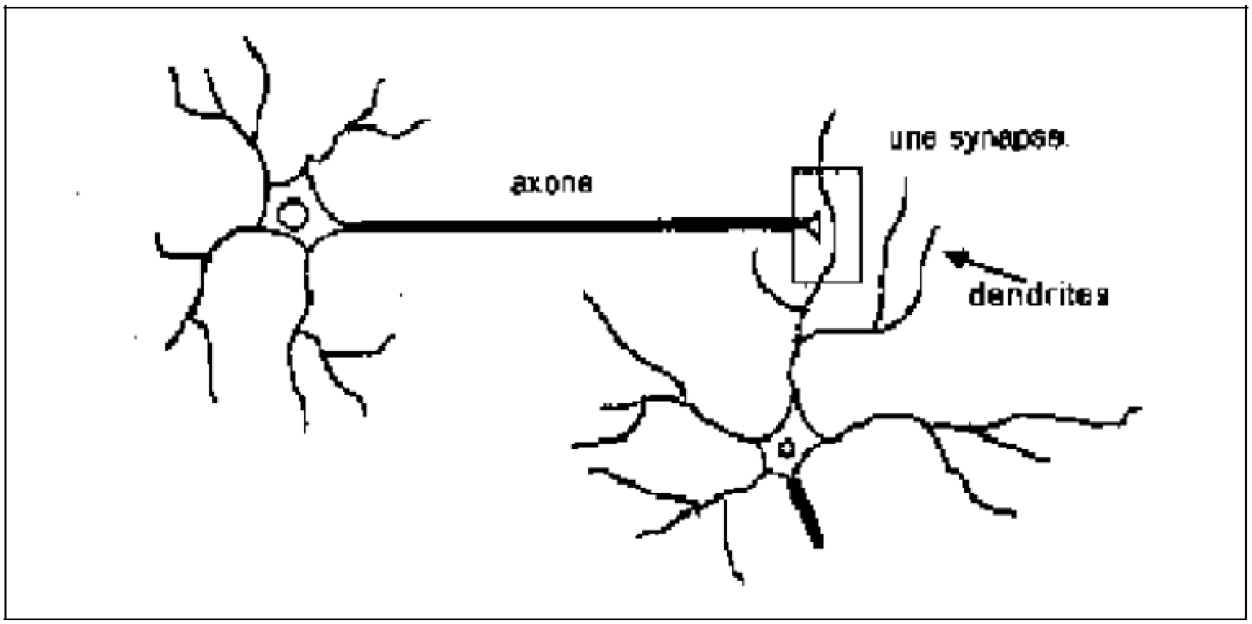

Figure 3: Illustrations of interconnections of biological neurons.[45]

Functionality of ANN (as intelligent system) works on similar principal as human biology. As you can see in figure 4, inputs are received by the artificial neurons from other neurons through a weighing function, which is generally a suppression and an amplification of the signals. On the addition of all the signals connected to the neurons, signal only travels from one neuron to other if the sum is higher than some threshold. The sigmoid function of the input determines the output of neuron not the threshold function which produces a non- linear input to output connection in a neuron. Point to be taken is the input weighs of neuron is storage for knowledge. The ability to store different information in neurons comes by adjusting weights.

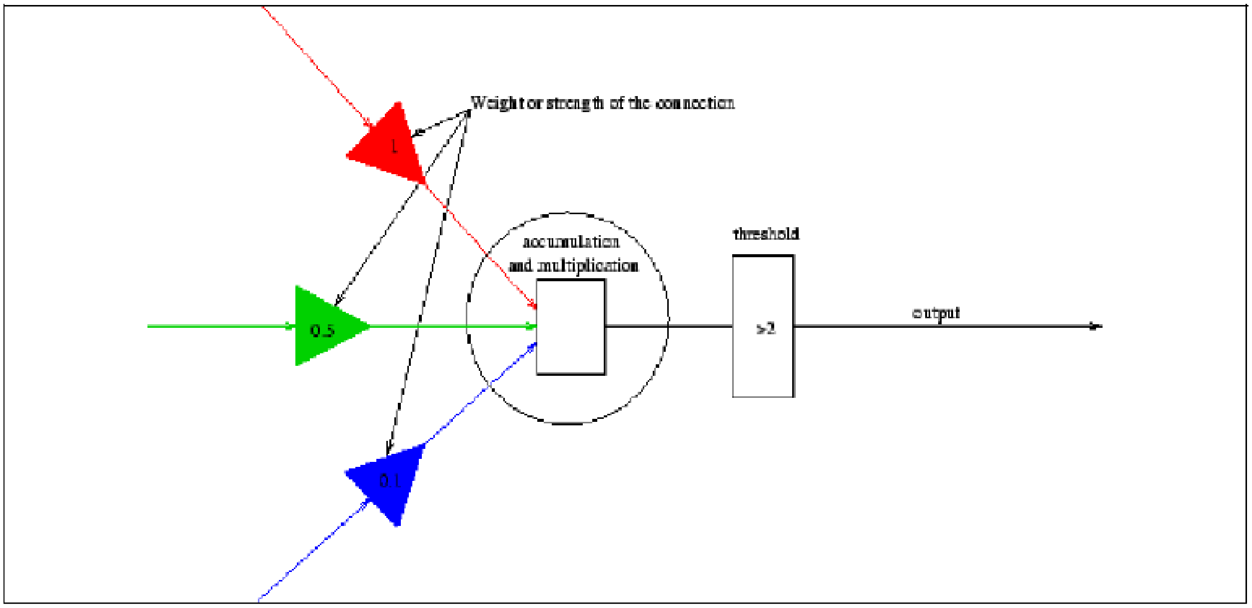

Figure 4: Pictorial description of single artificial neuron and its parts. [45]

Neurons interconnected in many layers have ability to store much more information as compare to one neuron (as referred in Figure 5).

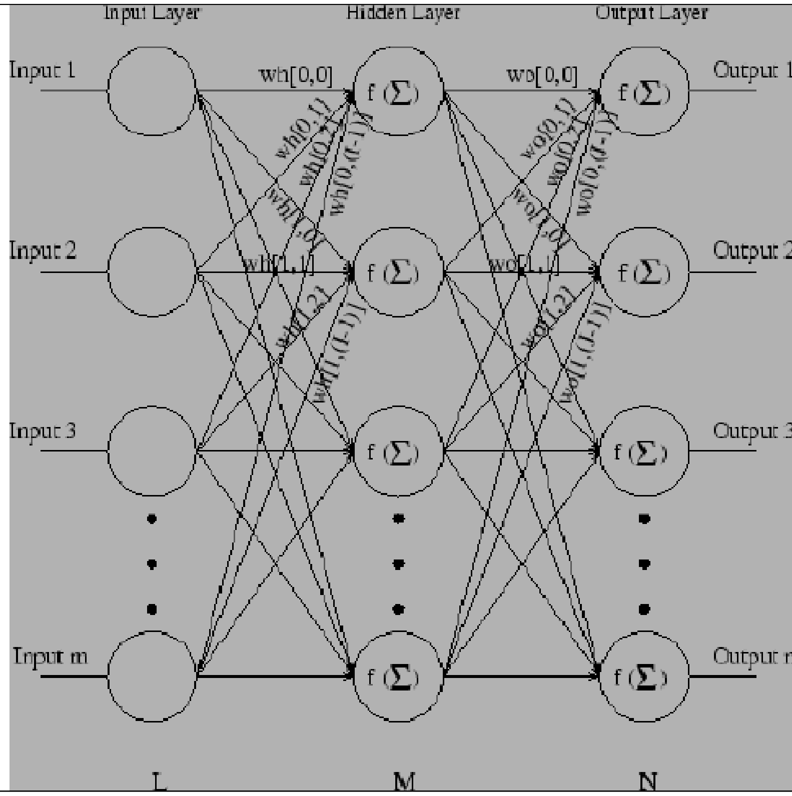

Figure 5: Illustrates A Back propagation – ANN [45]

Figure 6 outlines backpropagation algorithm.

Initialize all weights to small random numbers

For each training example do

- For each hidden unit h: $o_h = \sigma\left(\sum_i w_{hi} x_i\right)$
- For each output unit k: $o_k = \sigma\left(\sum_k w_{kh} x_h\right)$
- For each output unit k: $\delta_k = o_k(1 - o_k)(t_k - o_k)$
- For each hidden unit h: $\delta_h = o_h(1 - o_h)\sum_k w_{hk}\delta_k$
- Update each network weight w_{ij}:

  $w_{ij} \leftarrow w_{ij} + \Delta w_{ij}$ With $\Delta w_{ij} = \eta \delta_j x_{ij}$

Figure 6: Backpropagation Algorithm [45]

2.5.4 Genetic Programming (GP)

GP output is a separate program except Genetic programming evolves from genetic algorithms. The main concept behind GP is to resolve control problem by creating a new program on the bases of programs that work best.

The implementation of GP involves four steps: Generating a random group of terminals and functions is an initial step. A computer program is a part of each random group. Functions operators as -,+,*,/etc… Problems consist of inputs and outputs that are terminals. As an example, earlier looking at the car, terminals would be the amount of braking applied following car, the velocity of car and the distance between the cars.

Executing each program with assigned number (known as fitness value) on basis of performance to solve the problem be the second step.

The next step is creation of new population through crossover, mutation and the fit program. A terminal of program and randomly changing functions is a part of mutation as shown in Figure 7. Exchange of functions and terminal of one program with another program is a crossover.

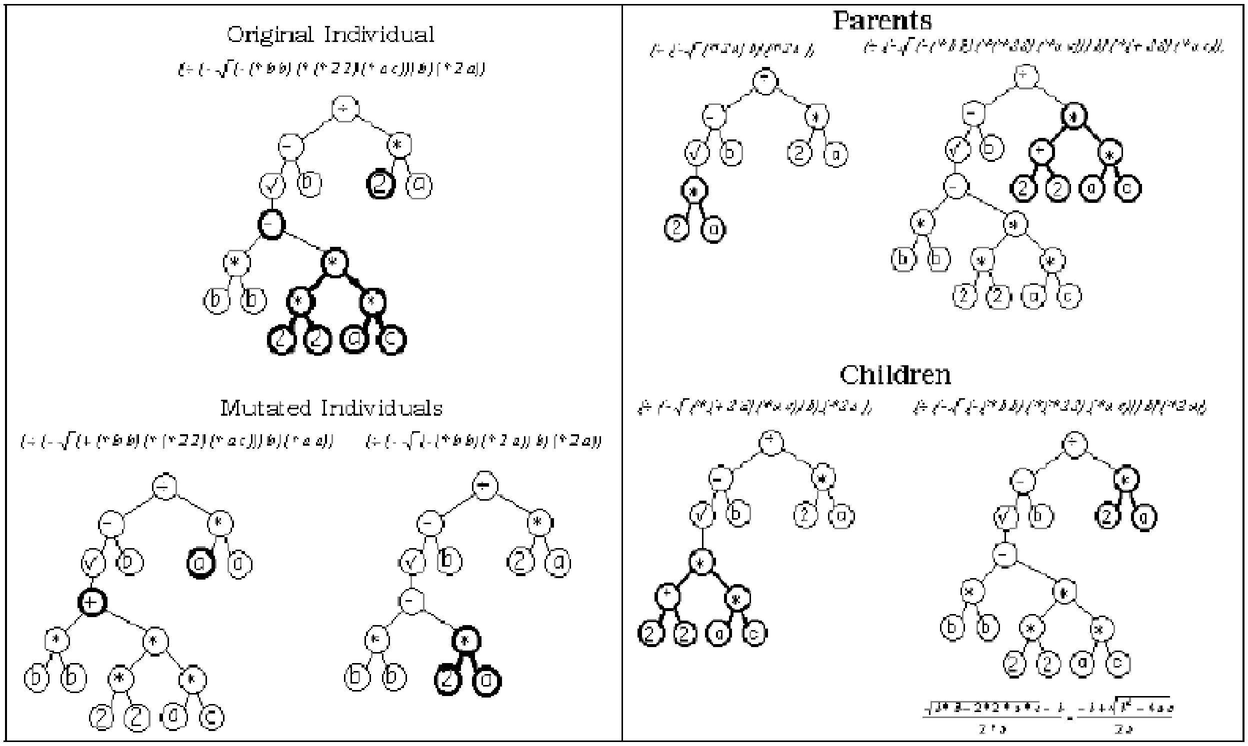

Figure 7: Pictorial description of mutation (left) and crossover operations in genetic programming.[45]

To reach a desired value the above mentioned three steps are repeated and the one that works best is the final result of genetic programming.

2.5.5 Support Vector Machines

The pattern recognition areas (computer vision) are the most common application of support vector machines and can also be utilized in control problems but that's not the most common purpose. Paper written by Suykens et al. be the most well known paper about the use of SVMs in non- linear systems for controls that consist of ball and beam problems (i.e. a ball rolling on a see-saw) and inverted pendulum problems.

2.5.6 Reinforcement Learning

Like other forms of intelligent controls, reinforcement learning is also important concept. A robot or plant act as an agent with already set of action choices as a part of reinforcement learning, prematurely considered to be as equally rewarding. An agent gets rewarded or punished (by a reward function) depends on the choices made by an agent as per behavioral policy. This way agent will learn how to make choices when the similar situation is confronted again.

2.5.7 Conclusion

Control system theory includes intelligent control system as well as classical control systems. In most of situations classical control systems are used when intelligent controls over kill it. When classical systems are unsuccessful, model of the system is impossible or difficult to obtain or areas are highly non- linear that’s when intelligent systems demonstrate their excellence.

2.6 Current Trends of Intelligent Control Systems in Nuclear Power Plants.

"Safety Regulations and Fuzzy-Logic Control to Nuclear Reactors" [12] presents R&D project using fuzzy control logic applied on Belgian Nuclear Reactors at research center. Project aimed at investigating value of fuzzy control implemented on reactors. Online tests that were successfully completed demonstrate fuzzy control is able to control reactor in stable state for various power levels and handle disturbance of rods per power changes. Project demonstrates its is feasible to apply fuzzy logic in nuclear reactors.

"Particle Swarm Optimization (PSO) Based Turbine Control" [16] uses genetic algorithm (GA) and particle swarm optimization (PSO) to optimize gains of proportional internal derivative (PID) algorithm and control steam turbine. Results were investigated and effectiveness of algorithm was evaluated. Numerical data also supports using PID controller coupled with PSO algorithm for better results. "The PSO-based PID controller was able to improve the optimization objective function by minimizing its value 0.51% lower than that of GA with spending 6.23% less time than GA. The PSO-based PID controller is highly recommended over GA-based PID controller." [16].

"Fuzzy-Logic-Based Safety Verification Framework for Nuclear Power Plants" [20]. Referring to nuclear power plants, paper presents practical implementation of safety verification framework per fuzzy logic. Safety and control limits in various plant processes with hazard scenarios are identified. In order to achieve Safety verification, risk is estimated quantitatively with safety limits in real time. Safety rules are defined using fuzzy logic to map hazard conditions with needed safety protection for viewing risk estimates. Proposed real time safety verification framework is analyzed with automated system developed to demonstrate safety limits for various hazard scenarios.

“Capacitive sensing technique for silt suspended sediment concentration monitoring” [21] studies suspended sediment concentration (SSSC) in water reservoirs using capacitance sensor techniques. Due to the fact that dielectric constants of water, air and sediments are different, characteristics of dielectric constants are studied for detecting concentration and soil moisture of water-air two phase flow. Capacitance sensor was used to monitor suspended silt concentration in the paper since it will increase in water-sediment mixture. This leads to dielectric constant of water increasing and also capacitance which is detective by sensing system increasing. Paper “demonstrated that the variations in the concentration of silt sediment correlates positively with the variations in observed capacitance in a linear fashion, and correlates negatively with voltage outputs but also in linear fashion” [21]. Paper demonstrates a good consideration of the technique that could be implemented at various Nuclear Power Plants that use sediment suction system at water intake points.

“Method for Improved Pressurizer System Knowledge Enabling Enhanced Pressure Control” [25] studies hybrid knowledge base with use of Kalman filter, model corrector and Recursive Least Squares Identification (RLS ID). Model updates dynamically per system changes based on measured data by RLS ID empirical identification system. Kalman filter estimates state variables, which are accurate considering uncertainties to improve system knowledge. Model corrector improves model accuracy using Kalman filter estimates. Paper introduces empirical and analytical pressurizer models to provide data sets (simulated) and describes techniques used by Kalman filter, RLS ID and model corrector. Results demonstrate better system knowledge achieved using the methods.

Chapter 3 - Framework and Methodology

3.1 Framework Flowchart

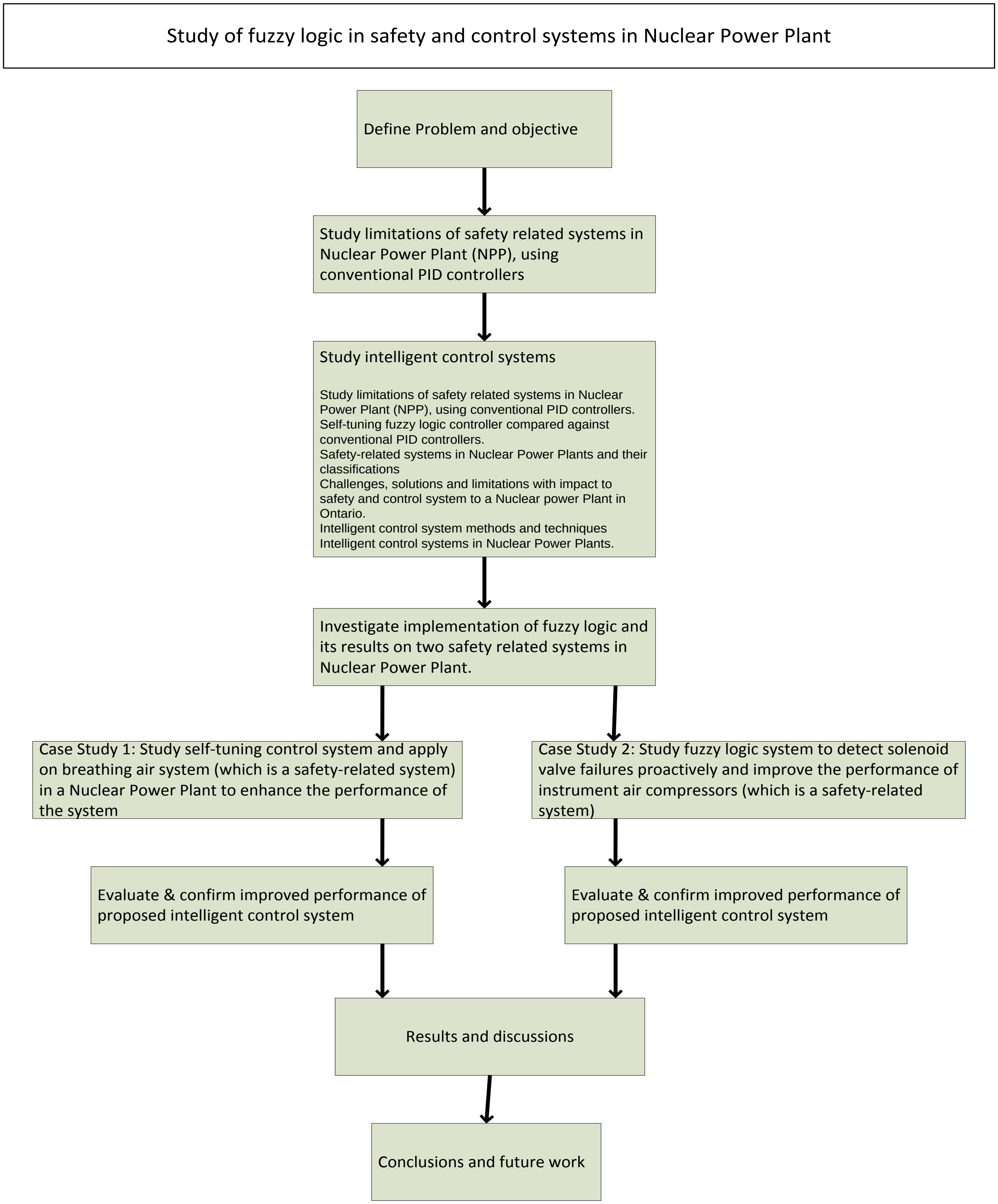

Figure 8: Flowchart depicting research plan

3.2 Framework Details

This research studies fuzzy logic in safety and control systems in a Nuclear Power Plant. Intent of thesis is to add value in improving performance of existing Nuclear Power Plants by using fuzzy logic and be good considerate for future plant implementation. Industry has to be fully aware of fuzzy logic benefits and how much productivity this logic can bring for plant reliability.

Problem and objectives were developed, limitation of safety related systems in Nuclear Power Plant were studied along with intelligent control systems. Study included review of self-tuning fuzzy logic controller compared against conventional PID controllers, safety related systems and classification in Nuclear Power Plant, intelligent control system methods, techniques, challenges, solutions and limitations.

Furthermore, two safety related systems (Breathing Air and Instrument Air) were selected for investigation and analysis with fuzzy logic. Case studies were investigated, one to study self-tuning control system applied on breathing air system to enhance performance, second to study fuzzy logic to detect solenoid valve failures proactively and improve instrument air compressor performance. Both case studies were evaluated to confirm improved performance of proposed intelligent control systems. Results were compared with and without fuzzy logic to prove logic integrity and usage. Results were also discussed at the end with conclusions and future work.

Chapter 4 - Design of Fuzzy Logic Based Self-Tuning Control System for Breathing Air System

Note: Breathing air system is a safety-related system in a Nuclear Power Plant. Design of self tuning control system is studies to enhance the performance of the system.

Nuclear power plants meet more than 50 per cent Ontario electricity demand [31]. It is imperative to run them reliably and safely at all times. This is partly done by executing plant outages wherein, a unit is taken offline and personnel enter reactor building to execute maintenance. Airborne radiation levels in the reactor building can be high, therefore, personnel require breathing air supply to perform work inside these buildings. Hence, pressure to the breathing air system at Nuclear Plant becomes crucial to execute work during outage and could lead to outage delays if not maintained within design limits [1].

Typical Breathing Air system at nuclear generating plant in Ontario is supplied by “three, two stage, water cooled and oil free rotary screw compressors, ZR3B type manufactured by Atlas Copco”[1]. Each compressor discharges air at 650scfm at 860kpa into air receiver that further discharges air to 4inch diameter common header [1].

The compressor internals (i.e. oil coolers, intercoolers, aftercoolers, etc.) are cooled by service water supplied to each compressor to maintain its key parameters under acceptable limits.

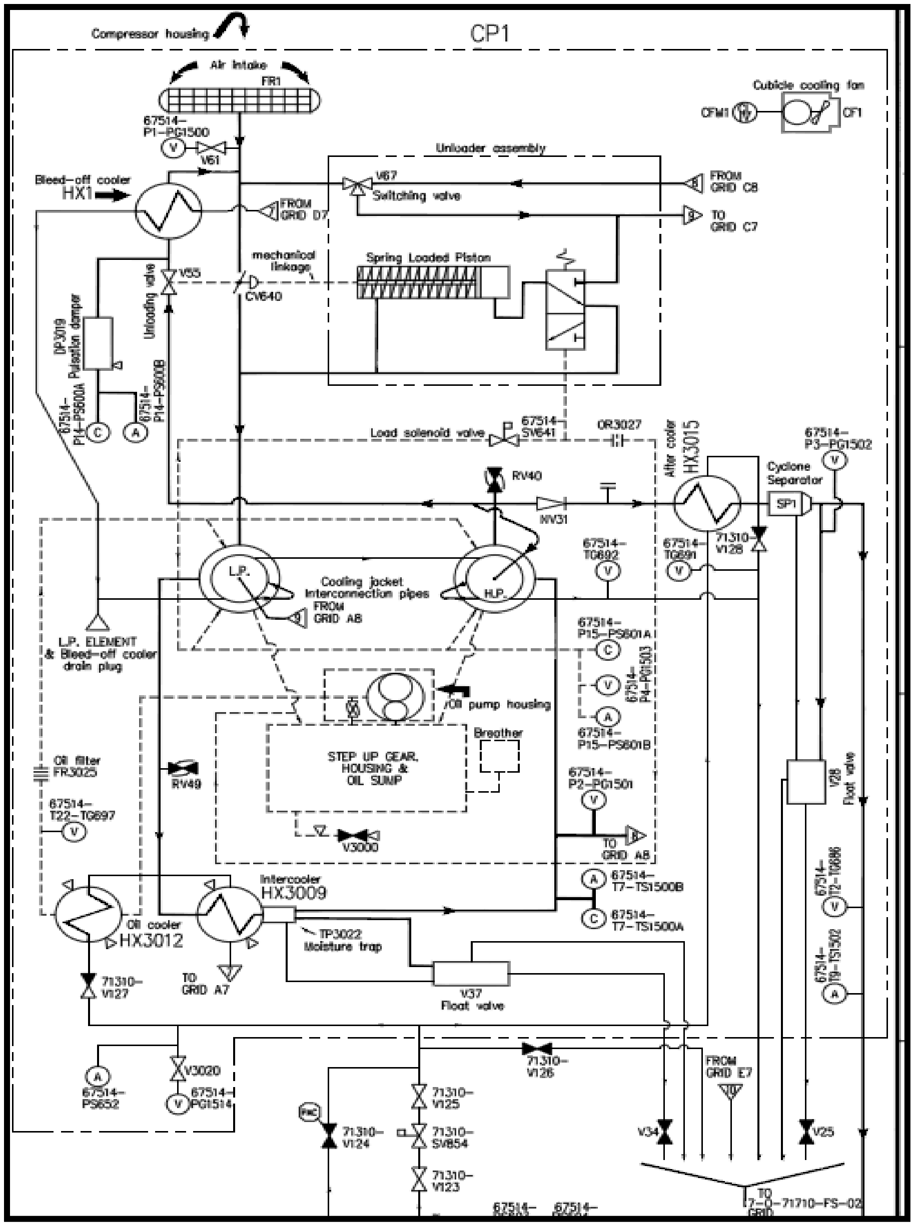

Figure 9: Schematics of Breathing Air Compressor [2]

Each compressor also has a water separator at the aftercooler drain trap to extract moisture from breathing air before it's supplied to station. In addition, it also contains air filter to remove dust and foreign materials from air that is fed downstream.

After compressors and receivers, breathing air is fed to common header that contains two pressure control valves (Pressure CVs) with operating alignment as one valve in and one standby to reduce operating pressure from 860kpa to 620kpa to be compliant with system design pressure.

Drain traps exist to air receivers, piping and stations to remove excess moisture from breathing air to keep its air quality within compliance of Z180.1-00 standards (CSA Compressed Breathing Air and Systems std).

The control problem investigated involves Pressure CVs not operating reliably to maintain design pressure of 620kpa. Typically, in Nuclear Generation plants, preventative maintenance practices exist to maintain functionality of Pressure CVs. Due to accumulation of dirt (ex rust), they could get stuck closed to further reduce pressure maintaining capability in the system. Consequences include, CVs not regulating system pressure properly and incurring extra costs to the company (Ex. during plant outages, when increased maintenance activities are carried out in the RB (reactor building), breathing air demand goes high but the CVs don't regulate to allow more air to pass through and maintain system pressure at 620kpa. Therefore, breathing air pressure reduces beneath 550kpa initiating alarm to the control room and all maintenance activities get stopped resulting in outage delays.)

4.1 Proposed Design

The new proposed design at a nuclear generating station in Ontario involves replacing CVs with Fuzzy+PID controllers as shown in Figure 10. This new model would ensure to sense pressure downstream in reactor building to indicate signals to fuzzy logic to drive Kp, Ki and Kd parameters and becomes self-tuning fuzzy logic driven PID controller.

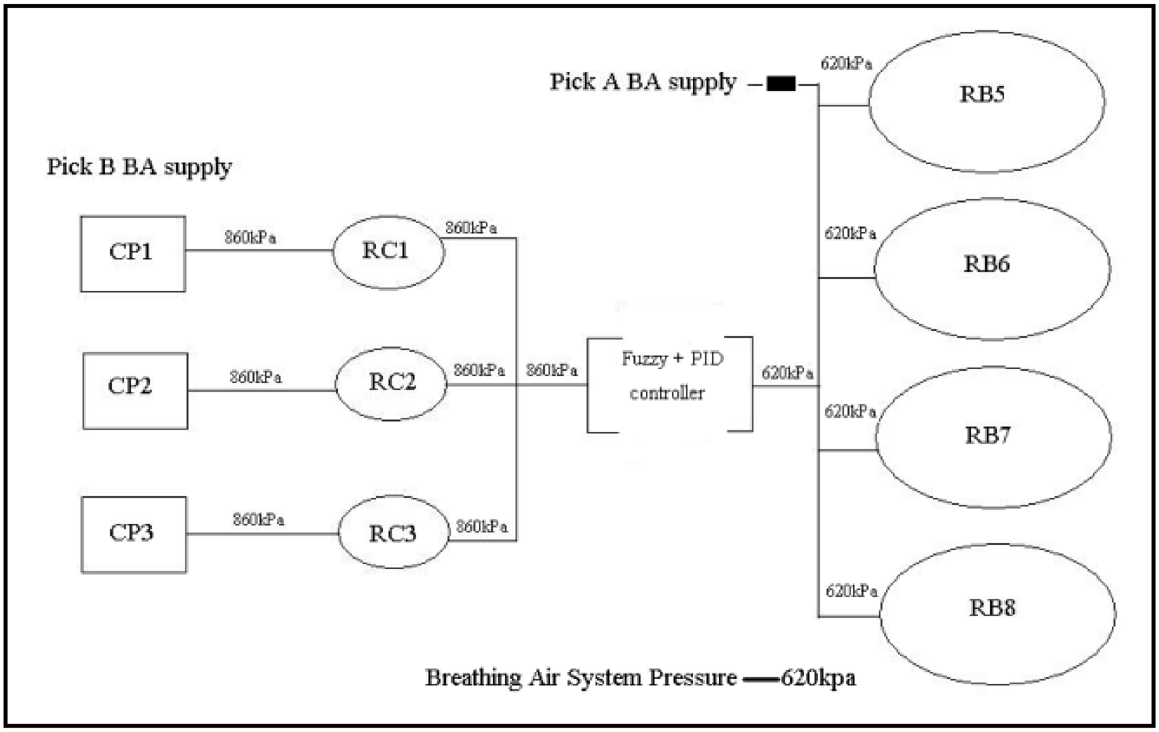

Figure 10: Proposed Design of Breathing Air System

NOTE: CP → Air Compressor, RC → Receiver, RB → Reactor Building

Following process was used to conclude proposed control design in Simulink [14].

1. Control goals established → Maintain breathing air system pressure at 620kpa and never below 550kpa.
2. Variables to be controlled identified→ System pressure, steady-state error, Fuzzy/PID parameters Kp, Ki and Kd.
3. Specifications written→ modeling to PID done repeatedly to understand system behavior to develop fuzzy rules and establish Kp, Ki and Kd numerical ranges.

4. System configuration established→ Block diagram developed with Fuzzy matrix and rules to drive PID controllers.
5. Process model developed in Simulink (Figure 12).
6. Control problem analyzed, controllers developed and key parameters adjusted for simulation.
7. Simulation performance analyzed and parameters adjusted to produce optimum results.
8. Simulation performance adjured to specifications and process repeated to reach control goal (of maintaining system pressure at 620kpa and never below 550kpa). Control Design finalized in the end (refer to Figure 12).

Figure 11 demonstrates sample system configuration of breathing air controlling valves at a nuclear generating station in Ontario [2].

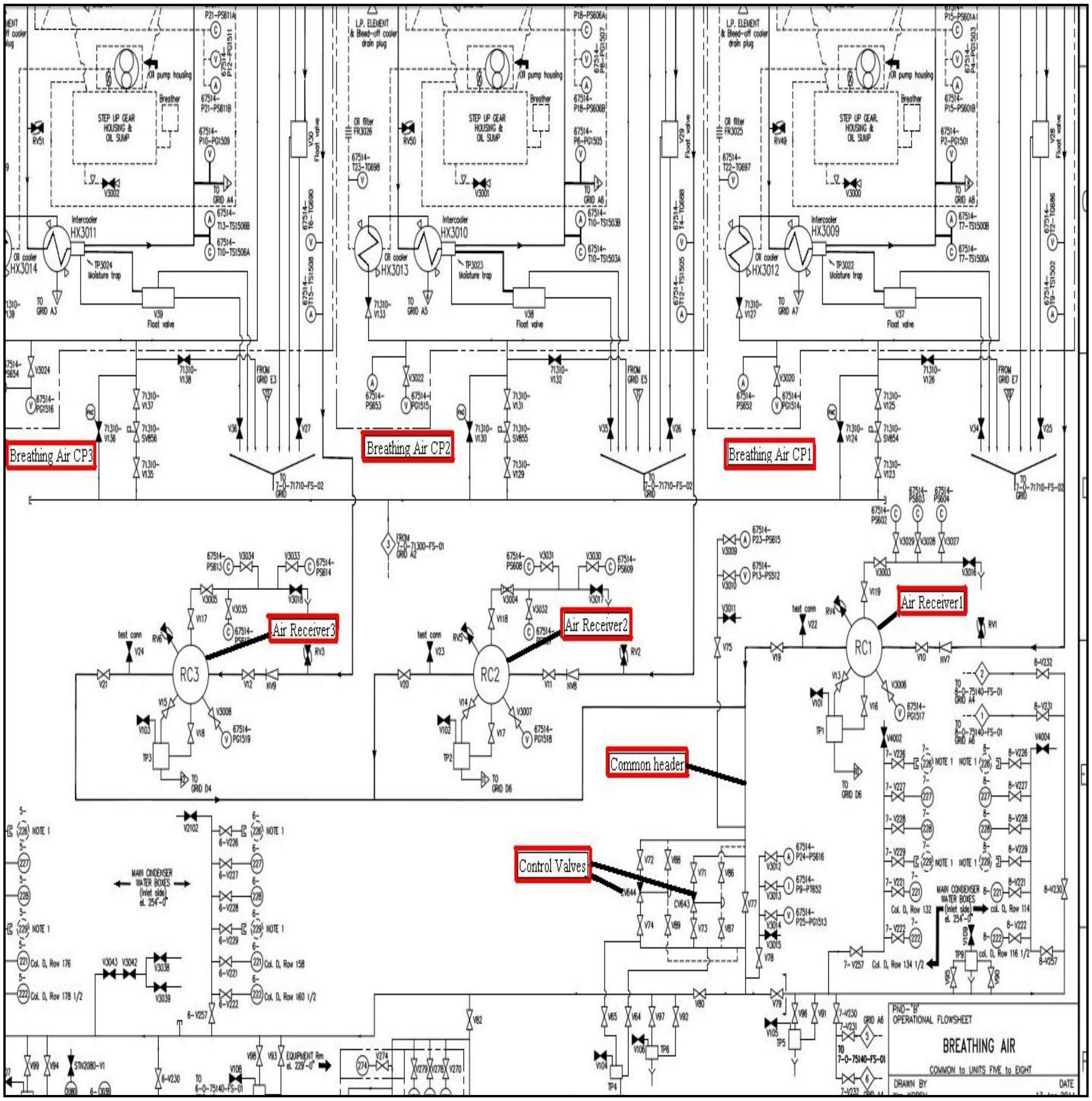

Figure 11: Breathing Air Flowsheet [2]

4.2 Data Sets

Modeling and building transfer functions of the system in Simulink consisted of four parts.

Part 1: Valve input signal to open or close the valve. Generated from an electrical controller, the electrical input signal range is kept between 4-20ma and relative output pressure range is 3-15psi (g).

Therefore Eq 1 calculates the slope.

$$G(Ax) = \frac{p(s)}{u(s)} = \frac{15-3}{20-4} = \frac{12}{16} \quad \text{Equation 1}$$

$$G(Ax) = 0.75 = K0$$

$$\boxed{G\ (Ax) = \ K0} \quad \text{Equation 2}$$

Part 2: Valve travel due to signal input. This includes the movement of valve stem including friction. Following equation was used to drive the transfer functions.

$$m * \ddot{y} = \ -F = (P * S) - (K1 * \dot{y}) - \ (K2 * y) \quad \text{Equation 3}$$

$Where\ (P * S)\ is\ the\ force\ created\ by\ valve\ diaphragm$

$(K1 * \dot{y}) is\ the\ friction\ of\ valve\ movement, propotional\ to\ velocity$

$(K2 * y) is\ the\ spring\ force, propotional\ to\ valve\ travel\ (movable\ part)$

$m = mass\ of\ the\ valve$

$\ddot{y} = acceleration\ of\ valve\ movement$

$\dot{y} = velocity\ of\ valve\ stem$

$y = friction\ displacement\ of\ valve\ \ (force\ created\ by\ spring)$

$F = force$

$P = pressure\ imposed\ on\ the\ valve$

$S = area\ of\ valve\ diaphragm$

$K1\ \&\ K2 = constants$

Simplifying the equation gives:

$$(m * \ddot{y}) + (K1 * \dot{y}) + (K2 * y) = (P * S) \qquad \text{Equation 4}$$

$$\left((\ddot{y}) + \left(\frac{K1}{m} * \dot{y}\right) + \left(\frac{K2}{m} * y\right)\right) = \left(\frac{S}{m} * P\right) \qquad \text{Equation 5}$$

$$\ddot{y} + (K3 * \dot{y}) + (K4 * y) = (K5 * P) \qquad \text{Equation 6}$$

$$Where\ K3 = \frac{K1}{m},\quad K4 = \frac{K2}{m},\quad K5 = \frac{S}{m}$$

$$K3, K4\ and\ K5\ are\ constants$$

Laplace Transform to Eq(6) gives:

$$\frac{y(s)}{p\ (s)} = G(CV) = \frac{K5}{S^2 + (K3 * S) + K4} \qquad \text{Equation 7}$$

$$G(CV) = \frac{K5}{S^2 + (K3 * S) + K4} \qquad \text{Equation 8}$$

<u>Part 3</u>: Air flow through the valve and into breathing air system.

$$q = y * \frac{\sqrt{\Delta P}}{\partial} \qquad \text{Equation 9}$$

$y = displacement\ of\ valve\ travel\ (assuming\ linear\ valve\ characteristics)$

$\Delta P = pressure\ diffrence\ across\ the\ valve\ (assuming\ contact\ pressure)$

$q = air\ flowrate\ across\ the\ valve$

$\partial = density\ of\ air$

Simplifying the equation gives:

$q = K6 * y$

Where $K6 = \frac{\sqrt{\Delta P}}{\partial}$

$$\boxed{G(flow) = \frac{q(s)}{y(s)} = K6}$$

Equation 10

Adding all three transfer functions will produce final function to be used in Simulink for the valve.

Equation 11

$$G(total) = \frac{q(s)}{u(s)} = G(Ax) * G(CV) * G(flow)$$

$$G\ (total) = \frac{p(s)}{u(s)} * \frac{y(s)}{p(s)} * \frac{q(s)}{y(s)}$$

$$G(total) = \frac{q(s)}{u(s)} = K0 * \frac{K5}{S^2+(K3*S)+K4} * K6$$

$$G(total) = \frac{K9}{S^2 + (K3 * S) + K4}$$

Where K9 = K0 * K5 * K6

$$G(total) = \frac{K9}{(S+K7)*(S+K8)} = \frac{K9}{K8-K7}\left(\frac{1}{S+K7} - \frac{1}{S+K8}\right)$$

For simplicity purposes, assume K8>>K7, this constitutes the faster mode of the valve movement (involving valve shaking etc), which is neglected since focus of research is kept on slow movement of the valve (involving K7) that would include opening, closing and valve regulations.

$$G(total) = \frac{K}{S + K8} = \frac{K/_{K8}}{S/_{K8} + 1}$$

$$= \frac{K/_{K8}}{(T * S) + 1}$$

$$T = 1/_{K8}$$

$$G(total) = \frac{K9}{K8 - K7} * \frac{1}{S + K7}$$

$$G\ (total) = \frac{K9/_{K7}}{K8 - K7} * \frac{1}{\frac{1}{K7} * S + 1}$$

$$G\ (total) = K * \frac{1}{T * S + 1}$$

$$Where\ K = \frac{K9/_{K7}}{K8 - K7}, \quad T = \frac{1}{K7}$$

From various experiments on valve stroke tests pertaining to control valves and for simplicity purposes, the time constant (T) has been derived to be 0.0013hrs (5sec) and K assumed to be 1.

Hence, $T = 0.0013$, K =1

Therefore, complete transfer function for the valve is derived to be,

Valve transfer function = $\frac{1}{0.0013S+1}$ **Equation 12**

Part 4: Breathing Air System.

Breathing Air System at PNGS was thought of as volume of air into a system. Therefore,

$$Q = Q_o + \int_0^t q(t)dt \quad \text{Equation 13}$$

$Q = Total\ volume\ of\ breathing\ air$

$in\ the\ system$

$Q_0 = Initial\ volume\ of\ breathing\ air$

$in\ the\ system$

$q(t) = \ Air\ flow\ going\ into\ the\ system$

Derivative of Eq (13) gives us.

$$\dot{Q} = q(t) \quad \text{Equation 14}$$

Laplace transform to Eq (14) gives

$$SQ = q \quad \text{Equation 15}$$

$$\frac{Q(s)}{q(s)} = G_{BA} = \frac{1}{S} \quad \text{Equation 16}$$

4.3 Control System Implementation

Please refer to newly proposed breathing air model (Figure 12) developed using PID + Fuzzy logic controllers in Simulink. Kindly note steps to build fuzzy rules are explained in Section 4.3.1.

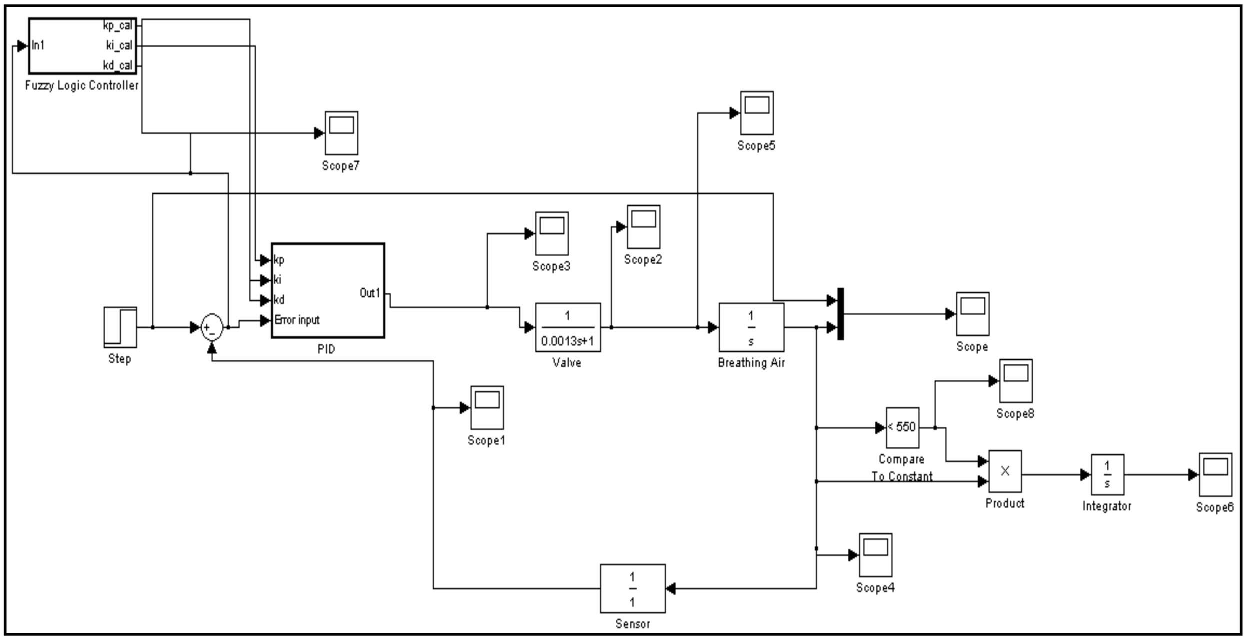

Figure 12: Model of breathing air using Fuzzy+PID controller in Simulink

In order to prove effectiveness of the new model, it was compared against conventional PID controllers (Figure 13).

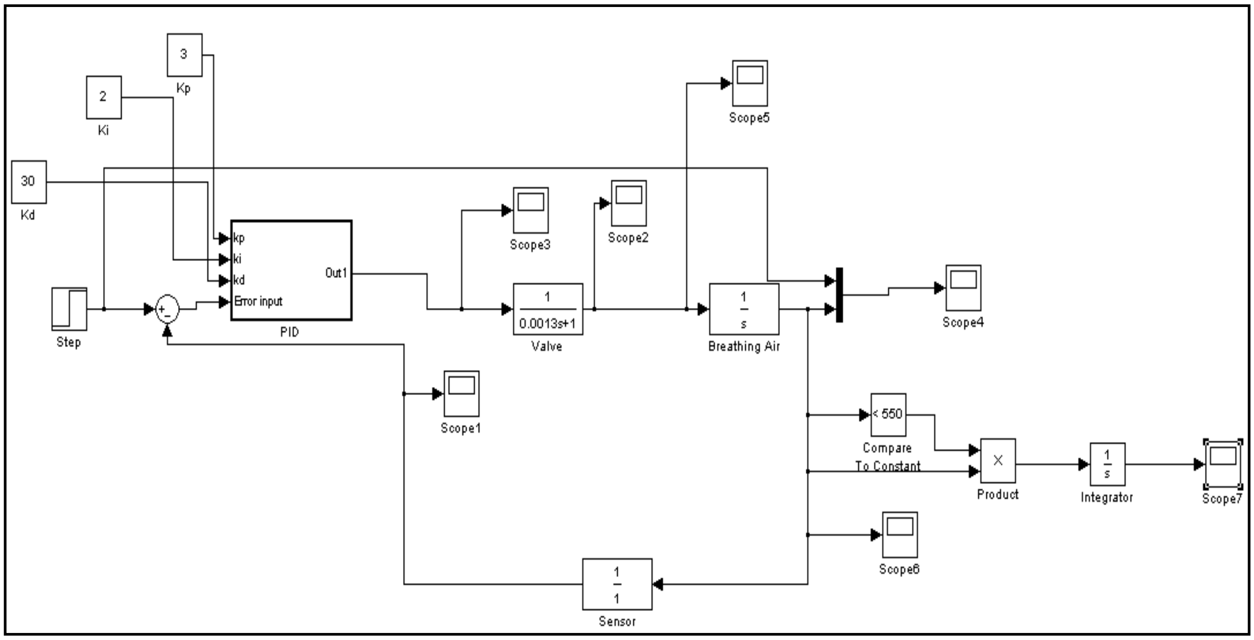

Figure 13: Model of breathing air using conventional PID controller in Simulink

4.3.1 Building fuzzy rules

Fuzzy rules were incorporated to fuzzy logic in order to develop Simulink model (Figure 12). This involves using fuzzy controller to support simulation of control model (Figure 14).

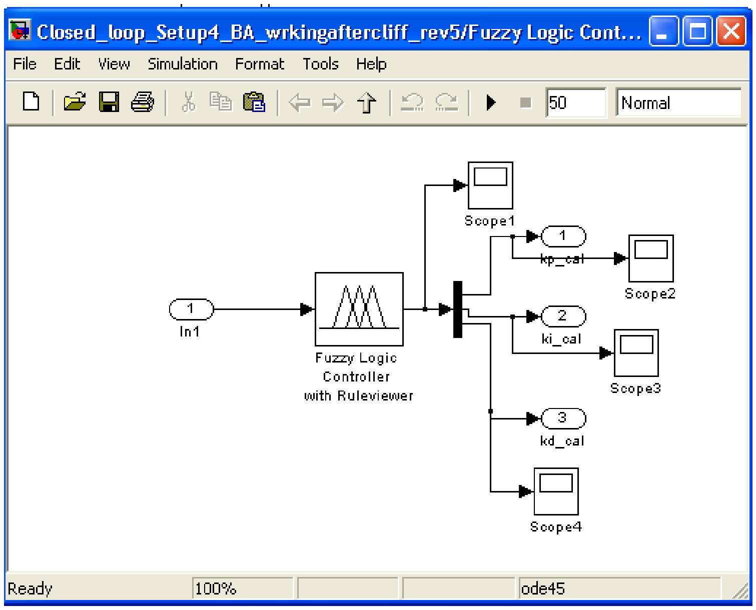

Figure 14: Fuzzy Logic Controller with Rule-viewer in Simulink

Input to the fuzzy model was kept as error (i.e. SP-actual pressure) to the breathing air system.

Furthermore, PID controller (Figure 15) was built to control positioning of the valves involving Kp, Ki and Kd parameters driven by the fuzzy controller (Figure 14).

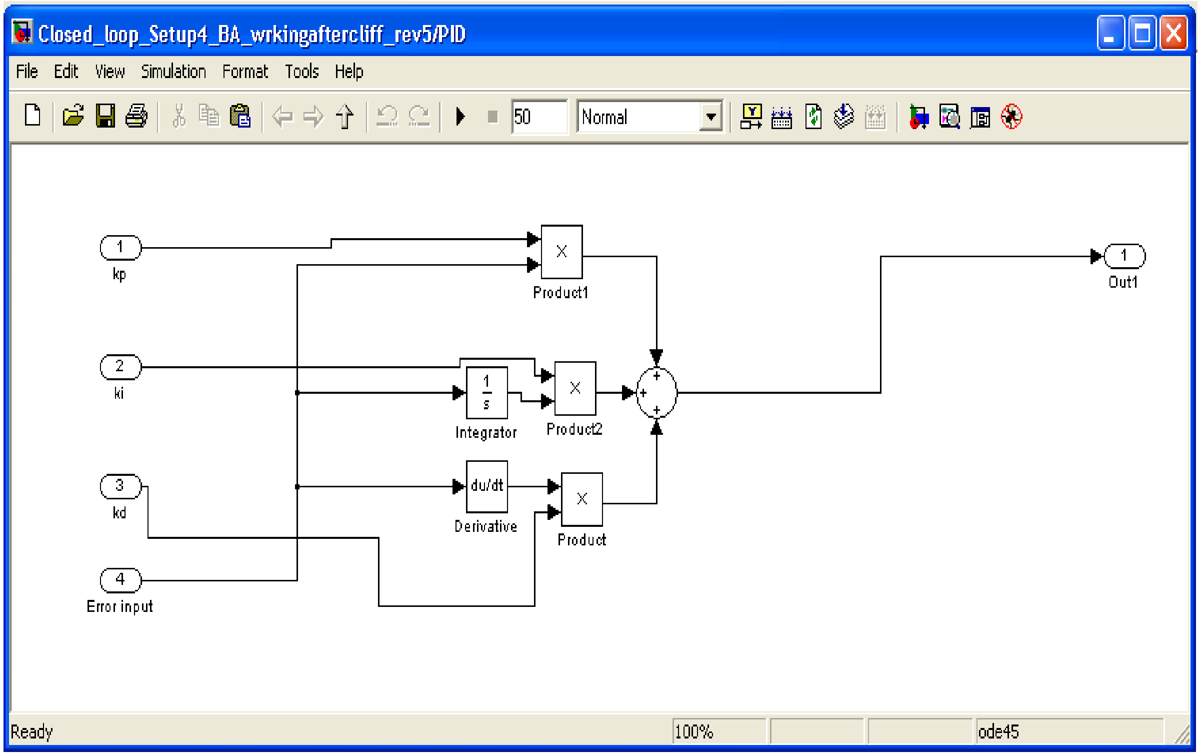

Figure 15: Conventional PID controller developed in Simulink

To formulate fuzzy rules, simple PID controller was run numerous times to understand the pattern of Kp, Ki and Kd in relation with system pressure and steady state error (Figure 16). Using this data, ranges to error, Kp ,Ki and Kd were established in fuzzy controller to reach optimum results. (Refer to Figure 17, 18, 19, 20)

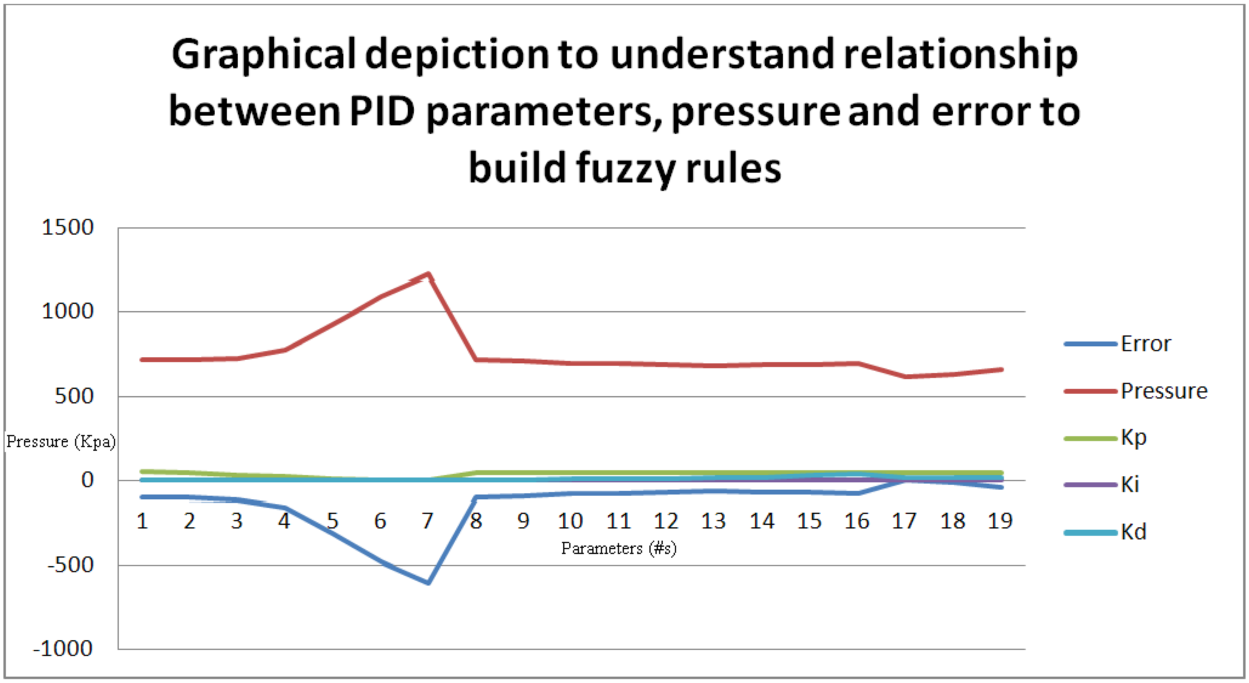

Figure 16: Graph showing relationship between PID parameters, system pressure and error

Error range was chosen from -620 to 620.

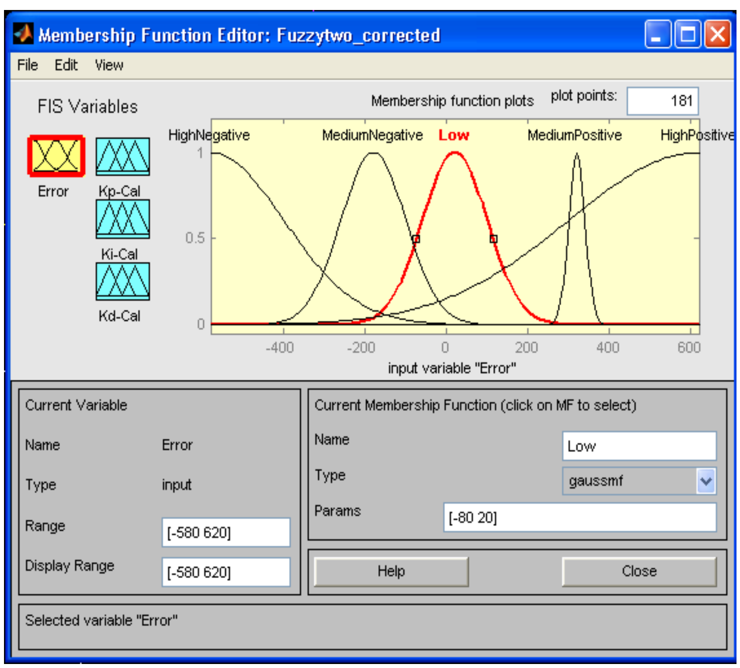

Figure 17: Error range for fuzzy logic controller

Kp range was chosen from 0 to 50.

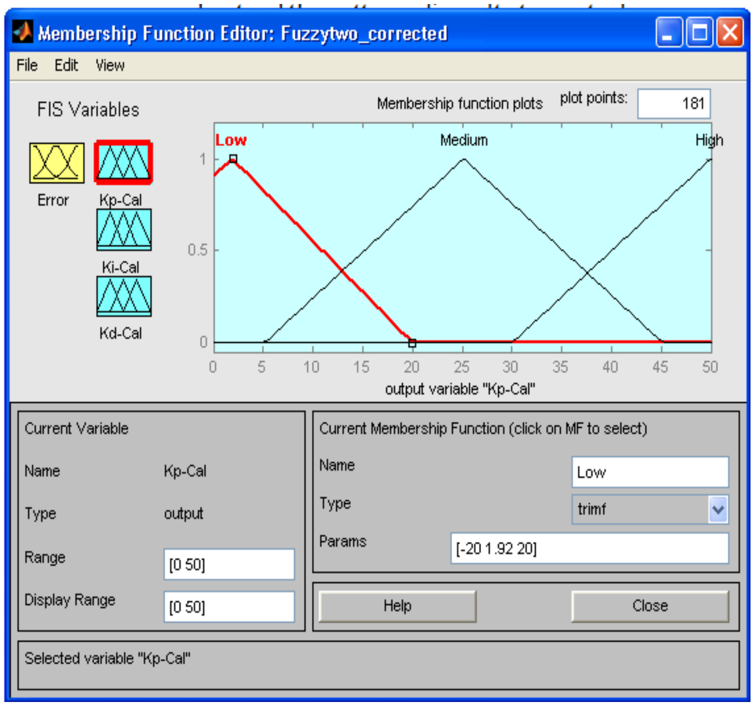

Figure 18: Kp range for fuzzy logic controller to drive PID controller

Ki range was chosen from 1 to 3.

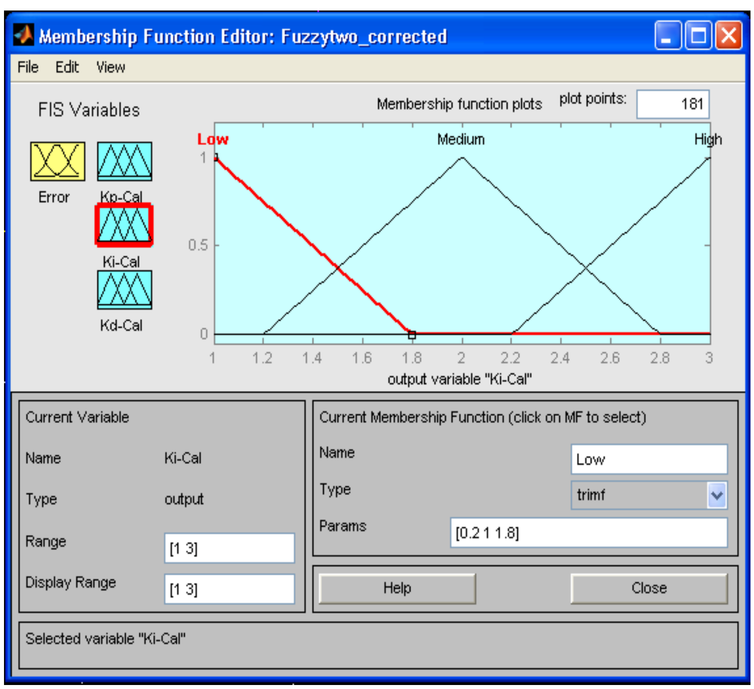

Figure 19: Ki range for fuzzy logic controller to drive PID controller

Kd range was chosen from 4 to 40.

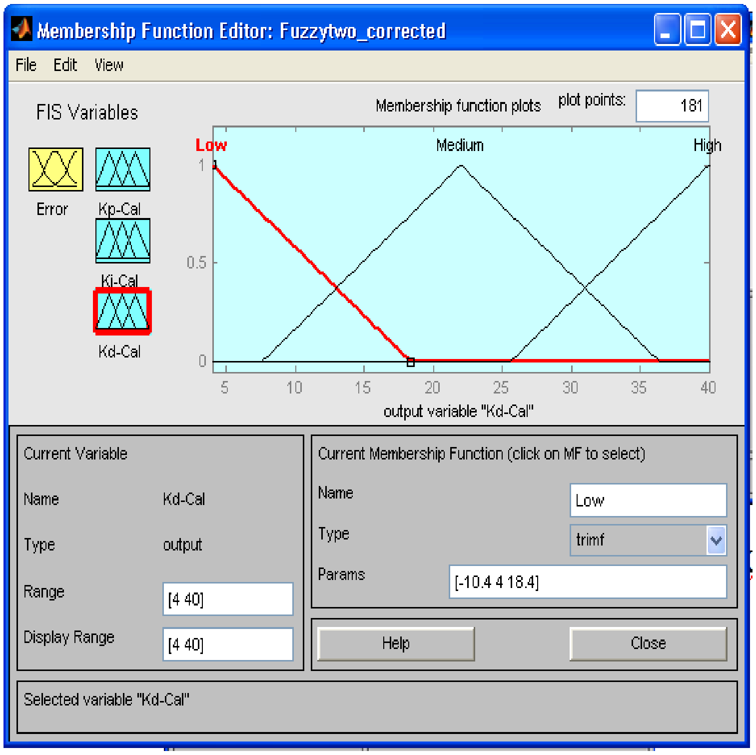

Figure 20: Kd range for fuzzy logic controller to drive PID controller

Rules used for building fuzzy logic are as follows:

If (Error is Low) then (Kd-Cal is Low)
If (Error is MediumNegative) then (Kp-Cal is High)(Ki-Cal is Medium)(Kd-Cal is Medium)
If (Error is MediumPositive) then (Kp-Cal is High)(Ki-Cal is Medium)(Kd-Cal is Medium)
If (Error is HighNegative) then (Kp-Cal is High)(Ki-Cal is High)(Kd-Cal is High)
If (Error is HighPositive) then (Kp-Cal is High)(Ki-Cal is High)(Kd-Cal is High)

4.4 Results

Simulation with Fuzzy+PID was run for 60 sec and demonstrated as per following results (Figure 21).

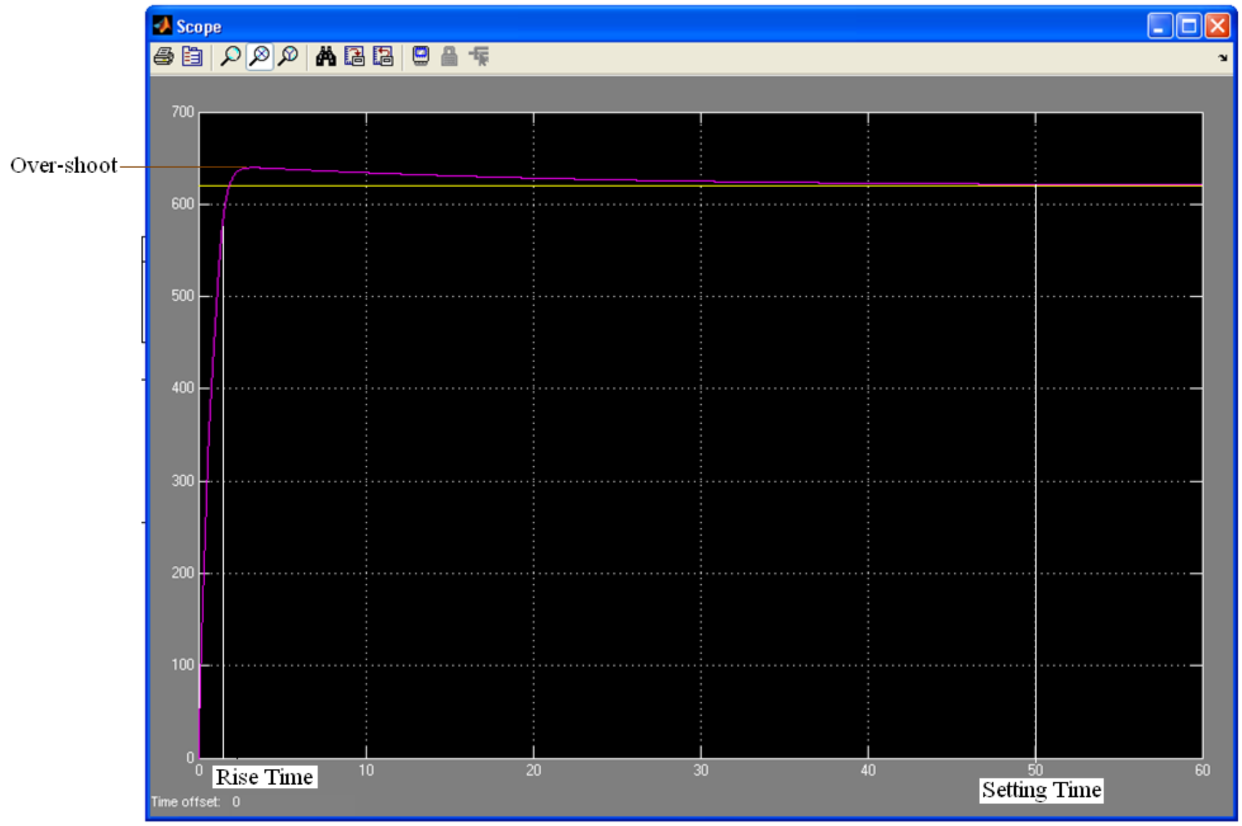

Figure 21: Simulation results for Fuzzy PID controller for 60 secs

To prove the model's effectiveness, simulation was run with simple PID controller (Kp=3, Ki=2, Kd=30) for 60 secs (Figure 22). Results demonstrate that pressure in the system didn't settle for 60 secs.

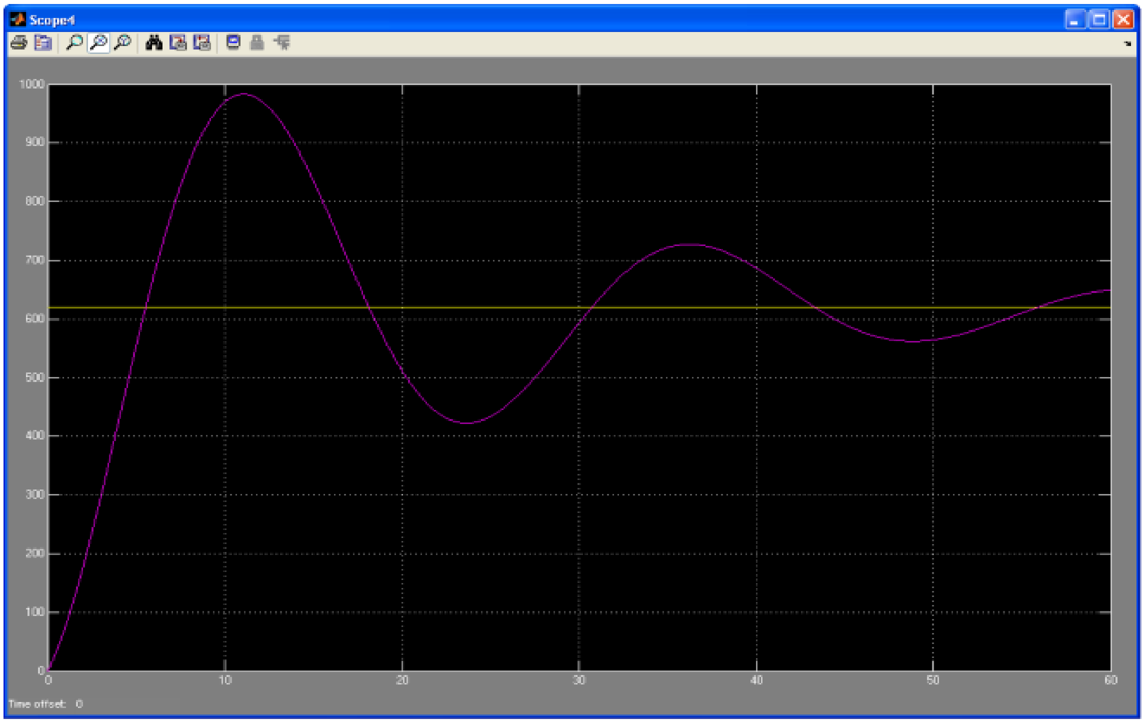

Figure 22: Simulation results for conventional PID controller for 60 secs

PID Simulation was repeated for 150sec with following results (Figure 23).

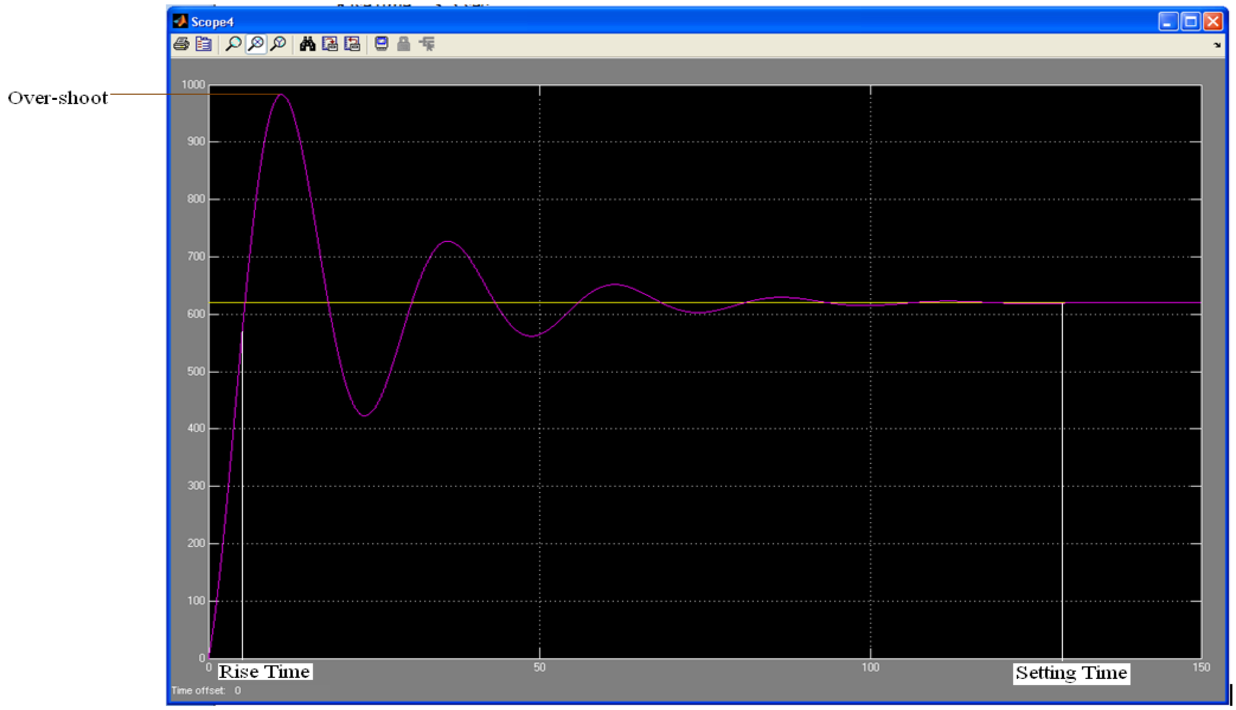

Figure 23: Simulation results for conventional PID controller for 120 secs to understand system stability

Simulation results for Fuzzy+PID controller	
Rise time	approx 2.5 sec (90% of 620kpa = 558kpa)
Over-shoot	640kpa
Settling time	50sec
S-S error	0.907

Table 1: Simulation results for Fuzzy+PID controller

Simulation results with simple PID controller	
Rise time	approx 6 sec (90% of 620kpa = 558kpa)
Over-shoot	990kpa
Settling time	130sec
S-S error	0.907 (approx)

Table 2: Simulation results with simple PID controller

As is evident from table 1 and 2, reduced rise-time, overshoot and settling time were noted with amalgamated Fuzzy PID controller.

4.5 Cost Comparison

Average cost for delaying an outage is estimated to be $20,000/hr (Canadian dollar) at a nuclear power plant in Ontario.

When pressure in breathing air system reaches below 550kpa, alarm is initiated in PNGS control room and personnel in the RB building are directed to evaluate building to restore system pressure back to 620kpa. Therefore, this results in delay in performing critical work in the Reactor Building.

A comparator was implemented to Simulink models to calculate area of running model beneath 550kpa.

Following were the results. Both controllers were run for 150sec to understand their cost relation with respect to model stability. Conventional PID Controller results are shown in Figure 24.

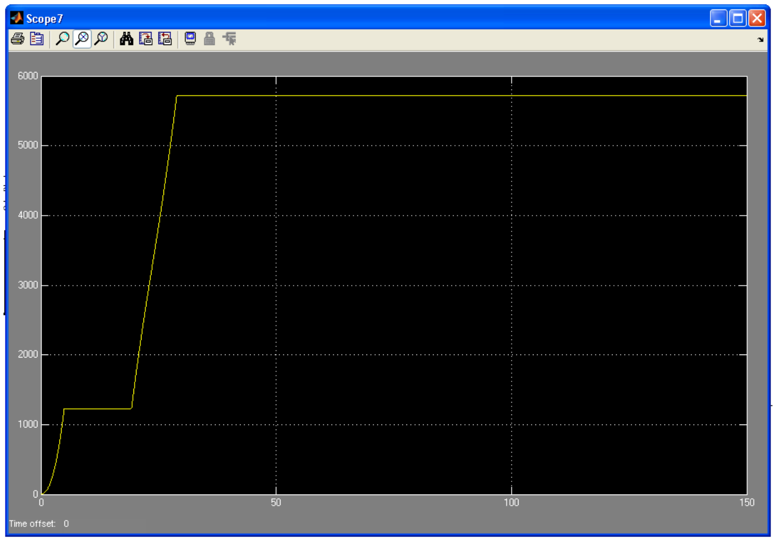

Figure 24: Graph depicting area underneath 550kpa using PID controller

Total area underneath 550kpa = 5750 (approx)

Judging by simulation graph, we know system pressure reached below 550kpa after rise time at 20sec. In running plant, this would initiate breathing air pressure low alarm and personnel will be asked to evacuate from RB building. All critical work be stopped resulting in outage delay of approx 3hrs.

Therefore, predicted outage delay cost with PID controller.

$$3hr \times \frac{\$20{,}000}{hr} = \$60{,}000\ (Canadian\ dollars)$$ Equation 17

Results for Fuzzy+PID controller for 150secs are shown in Figure 24.

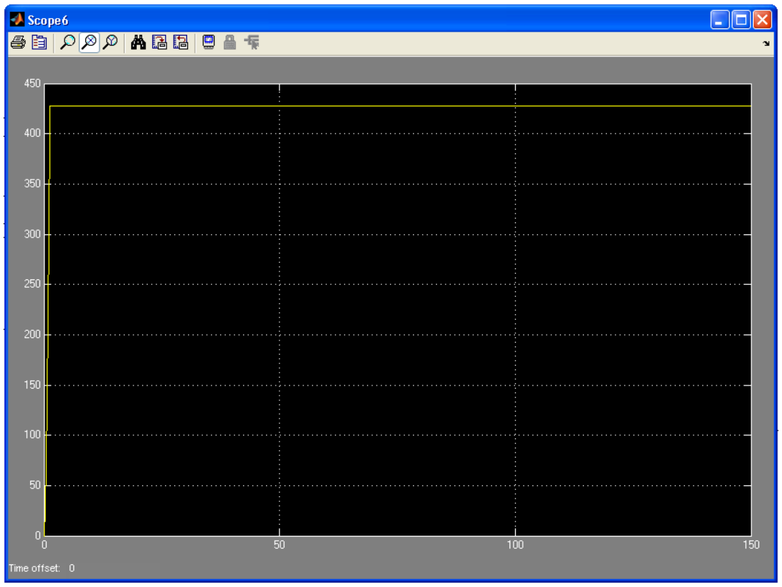

Figure 25: Graph depicting area underneath 550kpa using Fuzzy + PID controller

Total area underneath 550kpa = 428 (approx).

This area is only calculated during the initial rise time when operators would not be sending people into the RB building until system has reached pressure above 550kpa (i.e. no alarms be initiated into the control room). Confirmed by simulation results in Figure 25, the system pressure never reaches below 550kpa after the initial rise time.

Hence, there be no delay to outage schedule as predicted delay cost using Fuzzy + PID controller is zero dollars as shown in Equation 18.

$$0hr \times \frac{\$20,000}{hr} = \$0(Canadian\ dollars)$$ **Equation 18**

Results to system pressure using fuzzy + PID are shown in Figure 26 with rule-viewer in Figure 27.

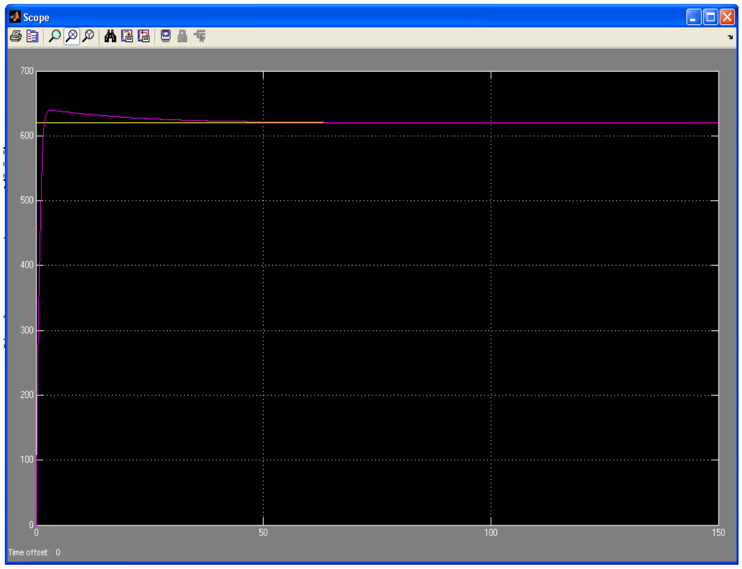

Figure 26: Simulation result of Fuzzy + PID controller for 120 secs to perform cost analysis

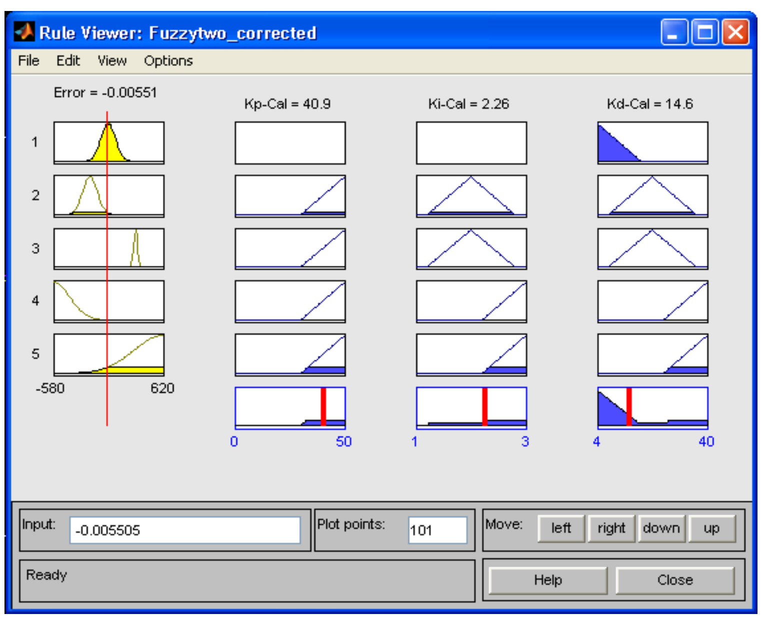

Figure 27: Rule-viewer for Fuzzy controller

NOTE: SS error = 0.00551 (with Fuzzy + PID controller)

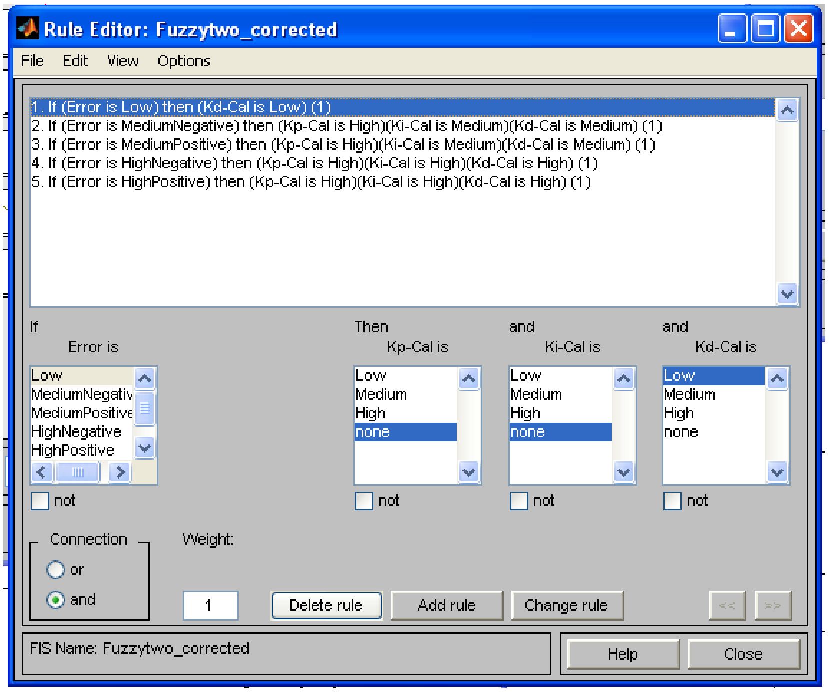

Figure 28: Please refer to Fuzzy rules built to run simulation

4.6 Conclusions

This thesis investigated using fuzzy PID controller to resolve control problem at PNGS station. Simulink was used to develop system model with fuzzy rules. Results were compared against conventional PID controller and demonstrated that fuzzy PID controller has superior control and precision in maintaining system design pressure with reduced rise-time, overshoot, settling time and steady-state error when compared against conventional PID. It was also shown that using PID conventional controller will cost extra $60,000 for losses incurred due to instability in the system. Hence, it is recommended that fuzzy PID controller be implemented to breathing air systems at nuclear power stations in Ontario for optimized system air pressure control.

NOTE: It possible that various plants might defer in breathing air system designs (such as Pickering 014 vs Pickering 058), but overall intent of research is to resolve control problems at relevant plants with limited compressor availability and similar designs wherein breathing air pressure is regulated by control valves installed downstream of compressors.

Chapter 5 - Design of Fuzzy Logic System to Detect Solenoid Valve Failures within Safety Related System

Note: System impacted is instrument air system (which is safety related system). Design of fuzzy logic system is studied to improve performance of instrument air compressors.

Safety of workers, environment and public is of paramount importance to Nuclear Sector. A Nuclear Power Plant produces energy that is used for various purposes in safe and reliable manner. As stated earlier, functionality of control systems and its instrumentation serves as a nervous system to the plant [9]. Various detection methods are available to act on problems before they become reactive. Instrument Air System is one such critical safety impacted system that is to be operationally available at all times.

Uses of instrument air system include running various air-operated devices (valves, air motors, dampers etc) and also used as pressurized cover gas for various systems.

Typical Instrument Air System at a Nuclear Generation Plant in Ontario consists of four "33%, 0.307 m^3/s (650scfm), 860kpa gauge (125psig), two stage, water cooled oil free rotary screw compressors each driven by a 150 kW (200hp) motor" [17]. All four compressors discharge air to four 7m^3 (250 cu ft) air receivers with parallel arrangement, connected downstream to four air driers via common header (Figure 29).

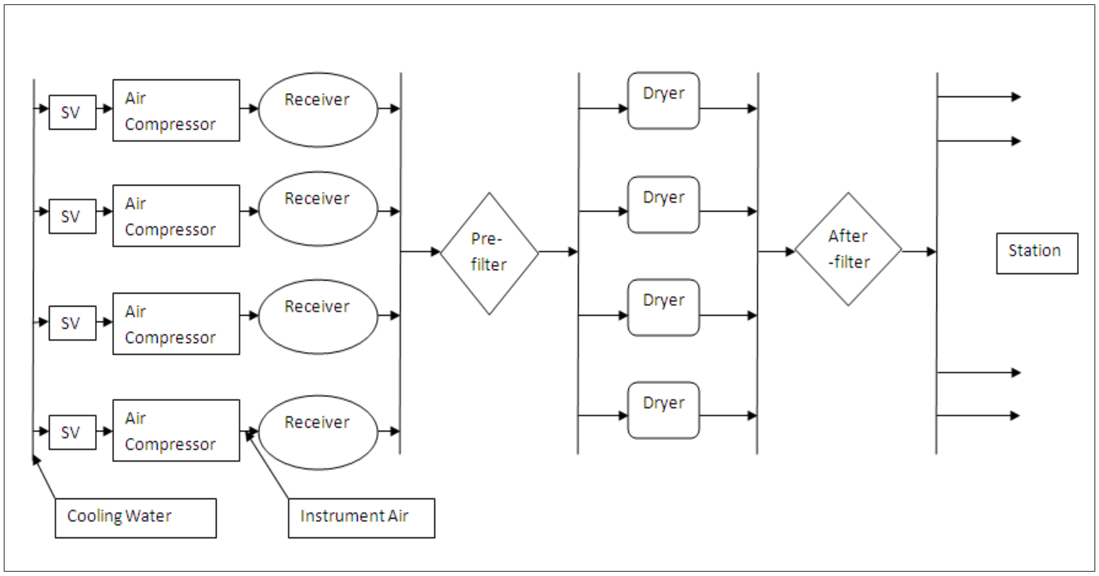

Figure 29: Schematic System Diagram

Four 33% heatless type air dryer units each has outlet capacity of 0.307 m^3/s (650scfm) are "twin tower, heatless, pressure swing solid activated alumina desiccant type giving an outlet dew point below minus 40C (-40F) at a rated gauge pressure of 860kpa (125psig)"[17] that provides dry air for station operation.

Once dry air is delivered, it is distributed via ring header to reactor buildings, reactor auxiliary bay, turbine auxiliary bay and turbine building. The headers consist of manual isolating valves for isolating the air for emergency purposes to avoid jeopardizing other air supply loads.

Individual compressed air stations consist of single/double manifolds with 8 outlets (1cm diameter) installed with isolating valves to feed downstream equipment. Please refer to Figure 30 describing layout of instrument air circuits at typical Nuclear Plant in Ontario.

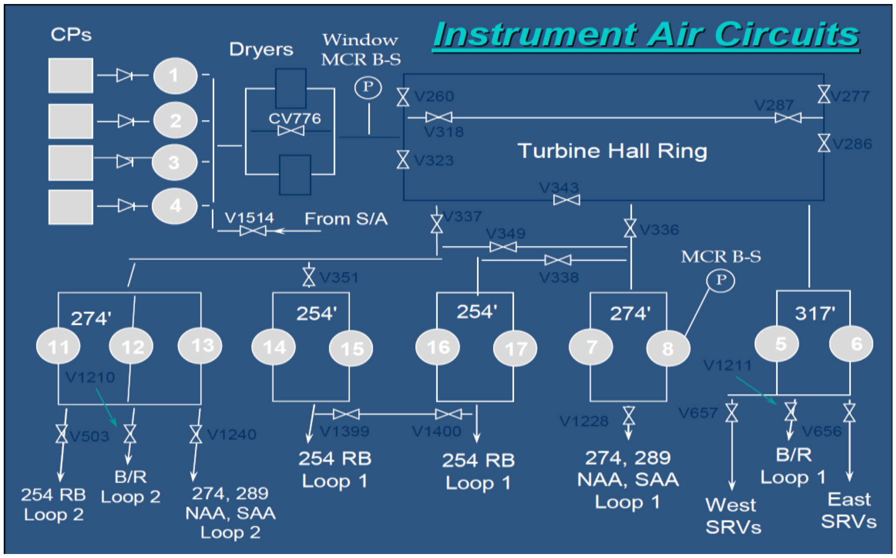

Figure 30: Pictorial view of instrument air circuits at typical Nuclear Power Plant in Ontario [15]

5.1 Problem Definition

The instrument air compressor internals (oil coolers, intercooler, and aftercoolers etc) are cooled using service water to maintain critical operational parameters under acceptable limits. These include inlet water pressure, intercooler air pressure, oil temperature, water temperature compressor out, discharge air pressure, air filter, oil pressure, discharge air temperature and water temperature aftercooler out [24].

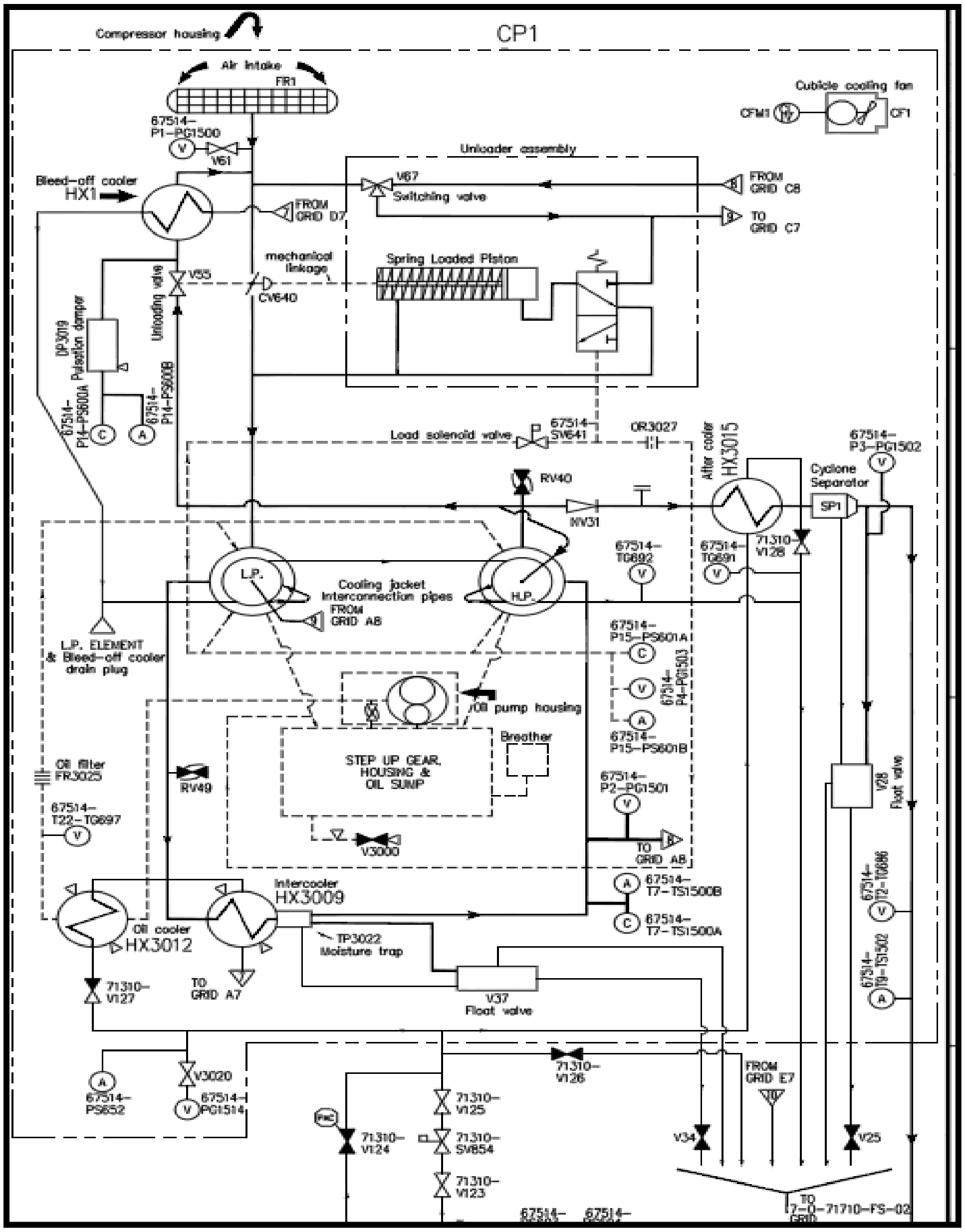

Figure 31: Flowsheet view of compressor [18]

Referring to Figure 31, service water enters compressor via solenoid valve (SV854) and distributes in two lines, one line goes to cool Oil Cooler (heat exchanger, HX3012 and intercooler HX3009) and other cools after-cooler (heat exchanger, HX3015).

Service water often could contain silt particulates that could plug compressor internals and damage solenoid valves (SV) that open/close to supply water to cool compressor. Typical damage of solenoid valve involves plugging its internal assembly, interrupting its operation by failing in same position. For example, if solenoid valve is open (and supplying cooling water to compressor) plugging of silt will keep it stuck in open position. This would mean even if compressor downstream is not running, supply of silted water will keep flowing through and continue to plug its internals (i.e. oil cooler, intercooler and aftercooler) affecting its heat transfer efficiency in long run.

Failure of this type doesn't trip compressor in short-timeframe but reduces its lifespan due to equipment degradation that includes overcooling of compressor internals, increasing oil viscosity and causing condensation within compressor that could lead to corrosion problems requiring part replacements. It also leads to service water system impairments due to increase of flow diversion (with SV stuck open) that could otherwise be used to cool other equipment in the plant.

Second mode of solenoid valve failure involves plugging its internals in closed position and preventing it to open when signaled by control system to cool compressor internals. This operation trips compressor within seconds and equipment is declared unavailable (reducing redundancy in the system with increased burden on other three compressors as four are dedicated to one unit). Maintenance resources will be required to clean SVs and heat exchangers inside compressors repeatedly to return compressor back in service.

This thesis investigates detecting SV failures proactively with a use of safety instrumented system (SIS) and fuzzy logic (also non-SIS systems) to increase air compressor performance, service water reliability and save maintenance resources. It uses all safety life cycle activities to investigate the problem [27]:

- Risk analysis – analyze [system] risks
- Risk reduction – assessing need for risk reduction
- Safety System Requirements – establish system performance requirements
- Safety System Implementation – implementing the system according to the required performance criteria.
- Safety Assurance – assure that system is always correctly operated and maintained

A case study is used to investigate this problem at Pickering Nuclear Generation Plant in Ontario (PNGS) wherein, silt (encompassing algae/debris/zebra mussels) in service water is causing constant degradation to instrument air compressors (installed beneath sea level and service water system (with flow diversion impairments) that needs resolution.

5.2 Proposed Solution

Hazard Analysis using FMEA and OPG Modification risk assessment methods were used to analyze system critical failure modes.

a) FMEA process

This method is a “Logical, structured analysis of a system, subsystem, device, or process” [27]. It is “used to identify possible failure modes, their causes, and the effects of these failures” [27]

Identification of critical system failures helps to investigate control measures and understand system gaps to be addressed.

Figure 32 lists common mode system failures (in a block diagram) for instrument air system. This analysis helped in producing FMEA results [24].

Case Study/Data Sets

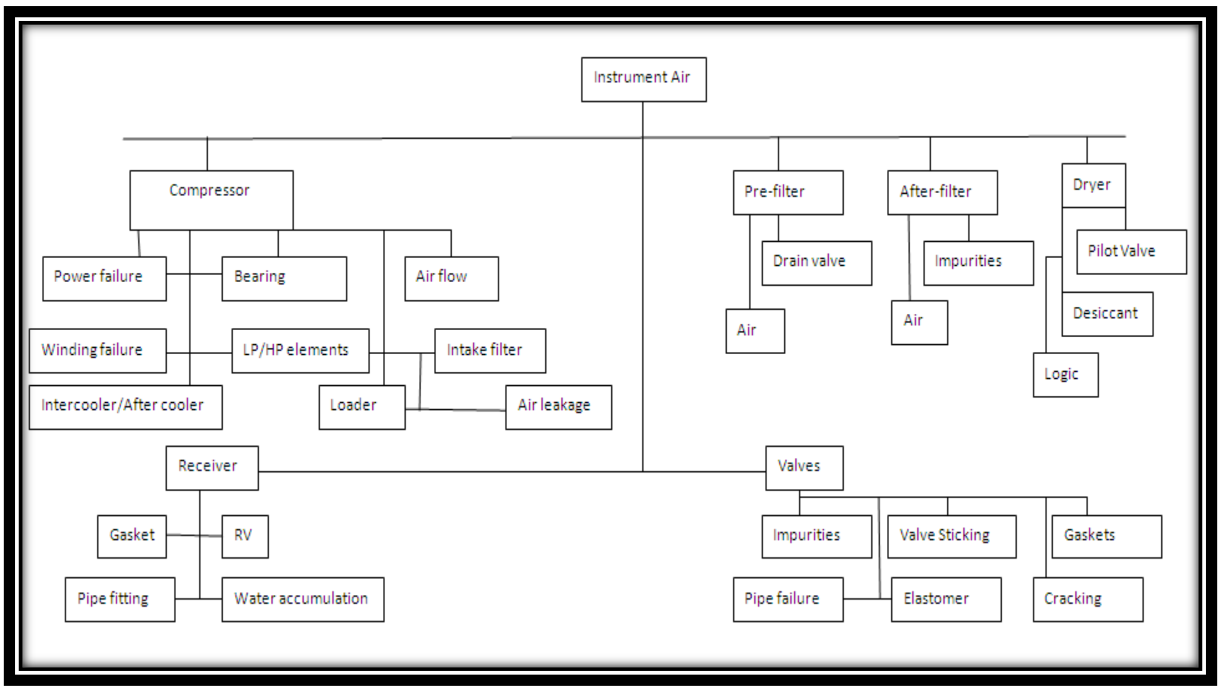

Figure 32: Instrument Air Block Diagram [24]

FMEA (Failure Mode and Effect Analysis) Results

Reviewing common failure modes for instrument air system, system reliability analysis was summarized by reviewing critical items, failure modes, failure causes, indications, consequences, severity, probability, criticality and control measures. These indications aid in daily system performance and monitoring to help reduce functional failures.

Item	**Failure Mode**	**Failure Causes**	**Indications/Parameters**	**Consequence**	**Severity**	**Probabili**	**Criticali**	**Control Measures/Remarks**
Compressor	**Unable to supply 650 scfm at 860kpa to station**	Power supply failure		**Inability to supply air loads to safety related systems and maintain 3Cs. Reliance on Back-up supply**	4	1	4	As specified by Electrical SE
		Bearing failure	Temperature/Smell/Noise		3	2	6	Work Order to be initiated to investigate abnormalities
			Vibrations		3	2	6	As specified by Vibe Engineer
		Winding failure due to insulation breakdown	Temperature/Smell/Sparking		3	2	6	Work Order to be initiated to investigate abnormalities
			Hi Motor Current		3	2	6	Perform megger test
		LP/HP compressor element failure	Discharge air temp, pressure, Atm. Pressure		3	2	6	Consider replacement of HP/LP element
			Vibration		3	2	6	
			Lubrication Oil		3	2	6	
		Flow blockage	Noise		2	2	4	Ensure no blockages of air passages
			Compressor intake filter DP		3	3	9	Replace intake filter element
			Discharge air pressure		3	3	9	
		Failure of intake filter	Noise		2	2	4	
		loader failure	Loading pressure, visual inspection		3	2	6	Check throttle valve operation
		Air leakage	Noise		2	2	4	Perform air leakage search through CP to RC
			Run time Vs Load time plot		3	4	12	Perform system leak search
		Rotary component bearing degradation	Oil Pressure		2	3	6	Check oil filter, oil pump, or lube oil injectors and system for blockage. Check cooling water supply temp, pressure. Check for overheating bearing / extreme friction
			Oil temperature		2	3	6	
			Lube oil sample		2	3	6	As Specified by Lube Engineer
			Outlet water temp		2	3	6	Adjust cooling water flow
		Inter-cooler/After-cooler Degradation	Discharge air temp. Aftercooler air temp Inlet water pressure, Service water leak from piping due to excessive vibration/ abnormal pipe movement.		3	2	6	Plan to clean HXs Check water SVs
			Aftercooler outlet water temp		3	2	6	Adjust cooling water flow
			Intercooler outlet air pressure temp		3	2	6	Plan to clean and overhaul HXs
		Solenoid Valve Failure	**Inlet water pressure/tundish/CP operating mode**		**3**	**3**	**9**	**System walkdowns/PG indications**
Service Water	**Unable to supply sufficient service water to station**	**Four or more bypass valves open to adjacent units**	**System walkdowns/System Pressure**	**Reduction in component redudancy or Margin of Safety**	**3**	**3**	**9**	**System walkdowns/PG indications**

Figure 33: FMEA results

Item	Failure Mode	Failure Causes	Indications/Parameters	Consequences	Severity	Probability	Criticality	Control Measures/Remarks
Receiver	Failure of Air Receiver to contain 7.0 m3 of compressed air to provide a total flow of air of 0.47 m3/s for at least 8 minutes while the compressors are transferred from class IV to class III power and maintain a minimum supply pressure of 550 kPag in the system	Receiver leakage due to gasket / pipe fitting degradation, RV leaking, or wall thinning due to corrosion.	Chattering/ Leakage Set pressure Removal and Replacement of RV's done per PM. Air leak due to gasket/pipe fitting degradation. Wall thinning. Pipe degradation due to loose hangers, bending, loads not supported uniformly.	Inability to supply Instrument Air Loads to safety related systems required to maintain effective Control / Cooling capability.	4	2	8	RV rebuild/replacement as specified by RV Engineer Initiate WR to report abnormalities in piping/loose hangers.
		Reduction in Air Receiver capacity due to water accumulation resulting from trap failure			4	2	8	
Pre-Filter	Failure of a pre-filter to allow design compressed air flow while removing impurities	Plugging of filter due to impure supply of air	High Delta Pressure	Inability to supply normal Instrument Air loads, degradation of loads due to	3	2	6	Replace filter element
		Water accumulation due to automatic drain valve failure	Observation		3	2	6	
After-Filter	Failure of an after-filter to allow design compressed air flow while removing impurities	Element plugging due to accumulation of impurities and inadequate maintenance.	High Delta Pressure	Plugging of air flow or instrument passages due to impurities	3	2	6	Replace filter element
Dryer	Failure of a dryer to dry compressed air to an outlet dew point of -40C) (-40F) at a	Pilot valve failure	Switching failure	Increased moisture content of air supplied to loads, increased pipe and	4	2	6	Issue WO to repair
		Logic failure	Switching failure		4	2	6	
		Desiccant Degradation	Air moisture high, breakdown due to aging/vit		4	2	6	
		Desiccant Breakdown	High Delta pressure, aging or vibration		4	2	6	

Figure 34: FMEA results

Item	Failure Mode	Failure Causes	Indications/Parameters	Consequences	Severity	Probability	Criticality	Control Measures/Remarks
Inter-unit tie valve	Failure of inter-unit tie valve to allow back-up instrument air to be supplied to a unit suffering a loss of Instrument Air	Accumulation of impurities leading to valve sticking	Tests performed for freedom of movement. Pipe degradation overtime causing leak due to loose hangers, bending or loads not supported uniformly.	Reduction in redundancy of supply air to a unit suffering a loss of instrument air loads	4	2	6	Investigate/repair as required
Diaphragm Valves	Inability to maintain instrument air system header pressure and flow due to insufficient leak tightness of pipes and valves	Gaskets degradation	Run time Vs. Load Time Plot	Loss of instrument air loads / excessive IA inleakage into RB	4	2	8	Conduct Air leak search and repair as required
		Isolation valves fail to provide satisfactory leak tightness			4	2	8	
		Pipe failure	Leak Check (audible)		4	2	8	
		Elastomer failure			4	2	8	
		Diaphragm fatigue	Diaphragm valves cracking to high cycle fatigue/vibration. Air leaks developed overtime due to excessive pipe movement, interference or loading.		4	2	8	
		Diaphragm/valve cracking from high cycle fatigue (due to vibration)			4	2	8	
		Wear or damage from moisture or other contamination in inlet air			4	2	8	
		Clearance between valve body and seals/gaskets/stem			4	2	8	

Figure 35: FMEA results

Risk matrices (Figure 33, 34, 35) were developed using semi-quantitative analysis to identify accidental events and potential hazards, rank their severity & probabilities and identify control measures to understand system gaps to recognize accidental events proactively for resolution.

Following criteria was used to classify severity rankings for the effects of failure modes [27]. Numbers were chosen based on system design and its impact on other systems.

Catastrophic: 1, Death, system loss, or severe environmental damage

Critical: 2, Severe injury, severe occupational illness, major system or environmental damage

Marginal: 3, Minor injury, minor occupational illness, or minor system or environmental damage

Negligible: 4, Less than minor injury, occupational illness, or less than minor system or environmental damage

Following criteria was used to estimate probabilities for identified failure modes [27]. Numbers were chosen based on System failure backlog (tracked per System Health Reports for the station as Work Order backlogs).

1-Extremely Remote (Unlikely to occur)

2-Remote (Possible to occur in time)

3-Reasonably Probable (Probably will occur in time)

4-Probable (Likely to occur immediately or within a short period of time)

Severity and Probabilities were multiplied to get Criticality values. Two hazardous scenarios (SV failure and Service water degradation) are rated Severity (3) X Probability (3) = 9 (Criticality) for investigation.

a) Risk Assessment – Ontario Power Generation [22][23]

Risk assessment at any plant must be carried out by understanding safeguards at the station.

Following are safeguards at PNGS to catch, detect and resolve station deficiencies proactively. These processes help personnel in making decisions to manage station risks for safe operation.

<u>Safeguards at station</u>

System Performance Monitoring

System Health Teams

System Health Reports

Operator Rounds (daily)

System Walkdowns (weekly)

System Performance Monitoring Plans (SPMPs)

Compressor Alarms (in MCR)

Preventative Maintenance (PMs)

Predictive Maintenance (PdMs)

Corrective Maintenance (CMs)

Engineering Review Meetings (weekly)

Probability Risk Assessment (PRAs) Models – Reactor Safety

Equipment Criticality

Station Condition Record (SCRs)

Corrective Action Plan (CAP)

Air Leak searches performed every outage inside RBCP loading monitored by System Engineer

Relief Valve Program

Replacement every 5 years

Risk assessment process was carried out for installing pressure transmitter on service water downstream of SV. Process not laid out here to maintain company privacy.

After risk summation was performed to understand if modification be standard or reduced (to define permitry, approvals and resources needed for field execution), engineering change request (ECR) initiated for installing pressure transmitters downstream of solenoid valves will be reduced risk modification and easier to implement on field as it requires reduced permitry and approvals.

NOTE: Mod Preference at PNGS is not based on resources but based on maintaining safety & reliability of systems. Standard MOD or Reduced Risk MOD only helps modification team leaders to align stakeholders for review/approvals on timely basis.

Path forward: Problem of silt (encompassing algae/debris/zebra mussels) in service water causing instrument air compressor & service water degradation at Pickering Nuclear will be resolved by installing pressure transmitters (PTs) downstream of SVs to measure line pressure feeding compressors (as reduced risk modification at PNGS).

5.3 Proposed System Design/Algorithms

Figure 36 describes the embedded new design of solenoid assembly feeding compressors monitored by Safety instrumented system (with Fuzzy Logic and annunciations).

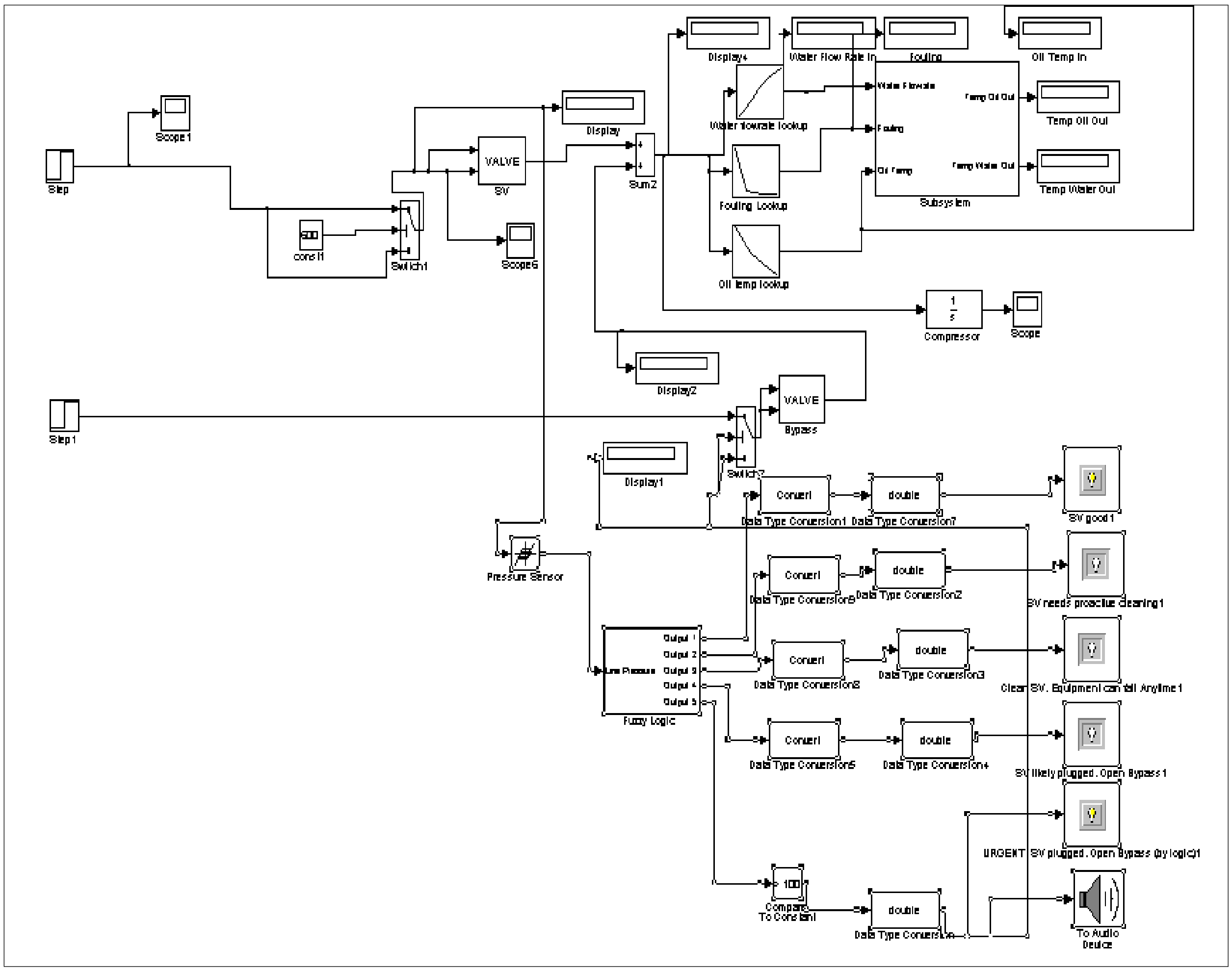

Figure 36: Proposed new Design of SV assembly with Fuzzy Logic and SIS.

Two constant values 'Service Water Pressure' and 'Clean bypass line' were used for simulation purposes. Water feeds into Switch1 that turns on and supplies water to 'SV'. There is 'Display' above to monitor pressure in the line that is linked with a 'Pressure Sensor' that feeds this live data to 'Fuzzy Logic'. Logic has ranges developed to take action and provide relevant annunciations.

Annunciation of ‘SV good’ is to be given when line pressure is reading between 400kpa to 600kpa.

‘SV needs proactive cleaning’ signaled when line pressure reads 350kpa to 400kpa.

‘Clean SV. Equipment can fail Anytime’ is provided between 200kpa to 350kpa.

‘SV likely plugged. Open Bypass’ is given between 100kpa to 200kpa.

Lastly, ‘URGENT: SV plugged. Open Bypass (by logic)’ is signaled between 0 to 100kpa. This will force bypass valve to open (with an Alarm) to alert operators to expeditiously bring maintenance to clean SV and restore cooling water supply via normal SV line to the compressor.

Fuzzy Logic rules were made to provide proactive annunciations for operators (during daily rounds) to act and file work requests (WR) based on SV annunciations and allow work assessing and maintenance time to schedule repair of SV as deficient maintenance rather than corrective maintenance (which is in reactive mode). This will also prevent compressor trips in long run.

5.3.1 Methodology

NOTE: For purposes of modeling in SIMULINK [35], heat transfer was studied for Oil Cooler as heat exchanger using properties of water and oil at average system temperature, 35C and 60C respectively to demonstrate how fuzzy logic based safety design can increase air compressor performance. [26]

Once service water passes the solenoid valve (SV), it is fed to ‘Air Compressor’ modeled as

1. heat exchanger to calculate water and oil temperatures going out based on inlet temperatures.

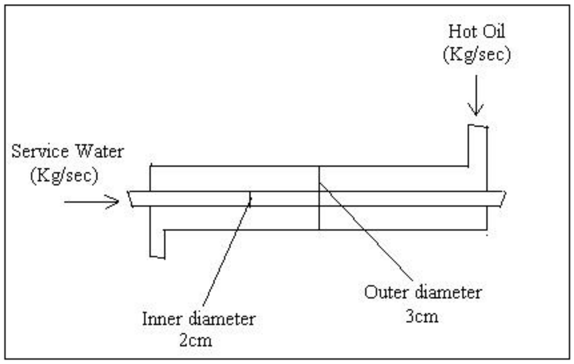

Figure 37: Schematic of Oil Cooler (heat exchanger inside air compressor)

2. Firstly, $\dot{m}_w$ and $\dot{m}_o$ are used as water and oil flow rates to start modeling process.

3. Temperature of service water going in ($T_{water,in}$) is kept constant at 35C for modeling purposes as its coming directly from lake.

4. Oil temperature going in ($T_{oil,in}$), Fouling rate (R_f) and water flowrate (m_w) is looked up to move the iterations. Data collected is qualitative based on system surveillance, monitoring and experience.

5. Hydraulic diameter of inner tube $D_{h,w} = 0.02m$.

$$Average\ velocity\ of\ service\ water = V_w$$ **Equation 19**

$$V_w = \frac{\dot{m_w}}{\rho_w \cdot A_{c,inner\ tube}}\ ({}^{m}/_{sec})$$ **Equation 20**

Using Eq (20), Reynolds number is determined

$$Re_{water} = \frac{V_w D_{h,w}}{v_w}$$ **Equation 21**

$$Note: v_w\ is\ the\ Kinematic\ viscosity$$

6. If Re number is turbulent, Nusselt number is determined using Eq (22) and water convection heat transfer coefficient (h_i) is calculated.

$$Nu_{water} = \frac{h_i D_{h,w}}{k_{water}} = 0.023 \cdot Re_{water}^{0.8} \cdot Pr_{water}^{0.4}$$ **Equation 22**

7. Same process is repeated for Oil.

Hydraulic diameter of annular space $D_{h,o} = 0.01m$.

Average velocity of Oil

$$V_o = \frac{\dot{m}_o}{\rho_o \cdot A_{c,outer\ tube}} \ (m/_{sec}) \quad \text{Equation 23}$$

Reynolds Number (for oil)

$$Re_{oil} = \frac{V_o D_{h,o}}{v_o} \quad \text{Equation 24}$$

8. If Re_{oil} for oil is determined as laminar flow in Eq 24, then Table 3 is used to get Nusselt number.

D(i)/D(o)	Nu
0	
0.05	17.46
0.1	11.56
0.25	7.37
0.5	5.74
1	4.86

Table 3: Nusselt number for fully developed laminar flow in a circular annulus with one surface insulated and the other isothermal [26]

9. After Nu_{oil} is determined, convection heat transfer coefficient is determined using Eq (25).

$$h_o = \frac{k_o}{D_{h,oil}} \cdot Nu_{oil} (W/m^2 \cdot °\mathrm{C}) \quad \text{Equation 25}$$

10. Using specific heat rates for water ($c_{p,water}$, 4.18 ($^{KJ}/_{Kg} \cdot$ ℃)) and oil ($c_{p,oil}$, 2.13 ($^{KJ}/_{Kg} \cdot$ ℃)), heat capacity rates for both water and oil are calculated.

$$C_{water} = \dot{m_w} \cdot c_{p,water} \; (^{KW}/_{C})$$ **Equation 26**

$$C_{oil} = \dot{m_o} \cdot c_{p,oil} \; (^{KW}/_{C})$$ **Equation 27**

Minimum heat capacity is divided with maximum heat capacity (Eq 28) to calculate ratio C.

$$c = \frac{C_{min}}{C_{max}}$$ **Equation 28**

Maximum heat transfer in Oil Cooler is calculated

$$\dot{Q}_{max}(KW) = C_{min}(T_{oil,in} - T_{water,in})$$ **Equation 29**

11. Afterwards, surface area of Oil Cooler (inner tube) is modeled in Simulink using Eq (30).

$$A_s = \pi * D * L$$ **Equation 30**

$$A_s \rightarrow Total\ surface\ area\ of\ inner\ tube\ (m^2)$$

$$D \rightarrow diameter\ (m)$$

$$L \rightarrow length\ (m)$$

12. Overall heat transfer rate is calculated as follows:

$$U = \frac{1}{1/_{hi} + 1/_{ho} + R_f}$$ **Equation 31**

Since we are interested in calculating heat transfer rate ($\dot{Q}$) and outlet temperatures ($T_{water,out}$, $T_{oil,out}$), log mean temperature difference method was reviewed and required tedious iterations to reach results that may not be practical. Kays and London in 1955 developed a method known as "effectiveness – NTU [number of transfer units] method" [26], which was modeled in Simulink to reach results.

<u>Effectiveness – NTU method</u>

NTU is first calculated with Eq (32).

$$NTU = \frac{UA_s}{C_{min}}$$ **Equation 32**

Referring to Figure 38, both C and NTU values are used to interpolate ∈ factor value

Actual heat transfer rate is calculated using (Eq 33).

$$\dot{Q}(KW) = \in \dot{Q}_{max}$$ **Equation 33**

Finally, $T_{water,out}$ and $T_{oil,out}$ are calculated as follows:

$$T_{water,out} = T_{water,in} + \frac{\dot{Q}}{C_{water}}$$ Equation 34

$$T_{oil,out} = T_{oil,in} - \frac{\dot{Q}}{C_{oil}}$$ Equation 35

Line Pressure (kpa)	Fouling factors R(f)
150	0.1
200	0.05
250	0.01
300	0.005
400	0.001
500	0.0009

Table 4: Fouling factors used for Simulation based on Line pressure

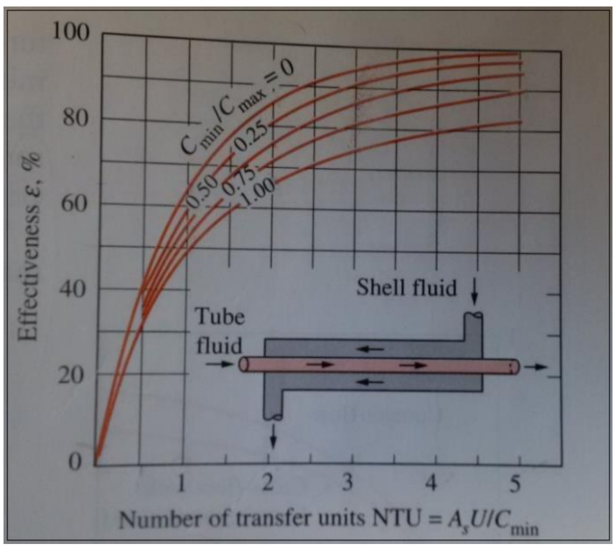

Figure 38: Effectiveness for Heat Exchangers [26]

5.4 Simulation/Results

a) Simulation without SIS (Figure 39).

Scenario considered: Inlet water pressure at 80kpa and SV stuck open → worst case for Air Compressor. This represents compressors acquiring cooling capacity with solenoid valve failed at open position with water flow maintained at 80kpa (pressure) flowing through compressors and analyzing its consequences without SIS.

As evident from Figure 39, Oil temperature of air compressor ONLY reduced from 72.8C to 57.45C with water temperature increased from 35C to 47.51C. Fouling factor is 0.17 (Table 4) based on line pressure. An increasing fouling number is sign of more silt/particulates to negatively impact heat transfer rate and degrade equipment performance.

In long run, re-circulating Oil temperature will continue to rise due to SV stuck in open position and Oil cooler will continue to plug up further decreasing the heat transfer efficiency and eventually trip the compressor.

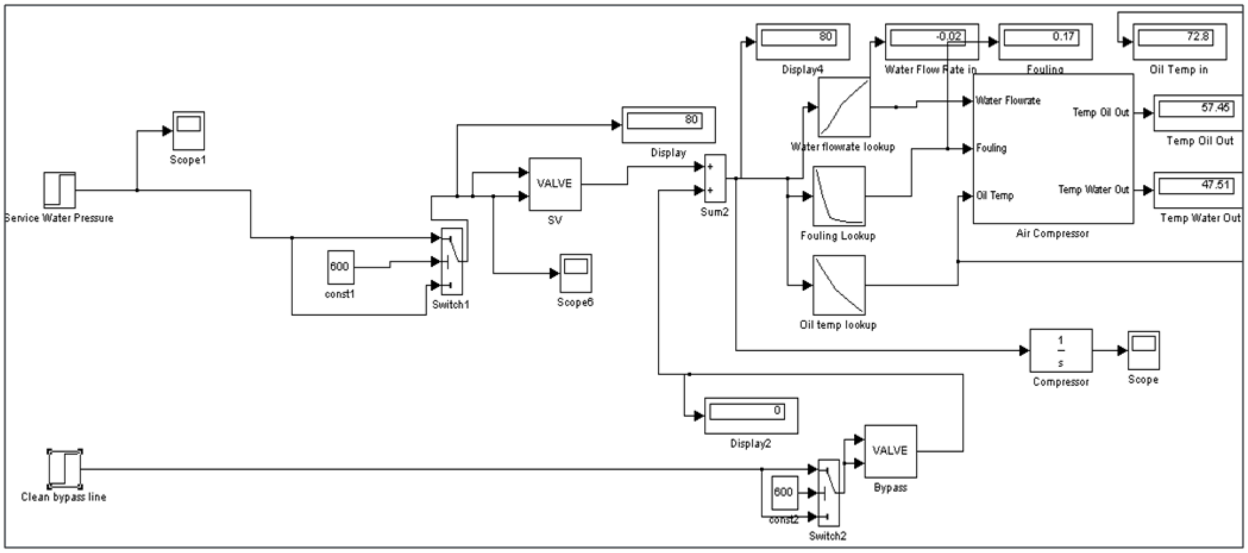

Figure 39: Simulink model results without SIS.

b) Simulation with SIS (Figure 40)

Scenario considered: Inlet water pressure at 80kpa and SV stuck open → worse case for Air Compressor. This represents compressors acquiring cooling capacity with solenoid valve failed at open position with water flow maintained at 80kpa (pressure) flowing through compressors and analyzing its consequences with SIS.

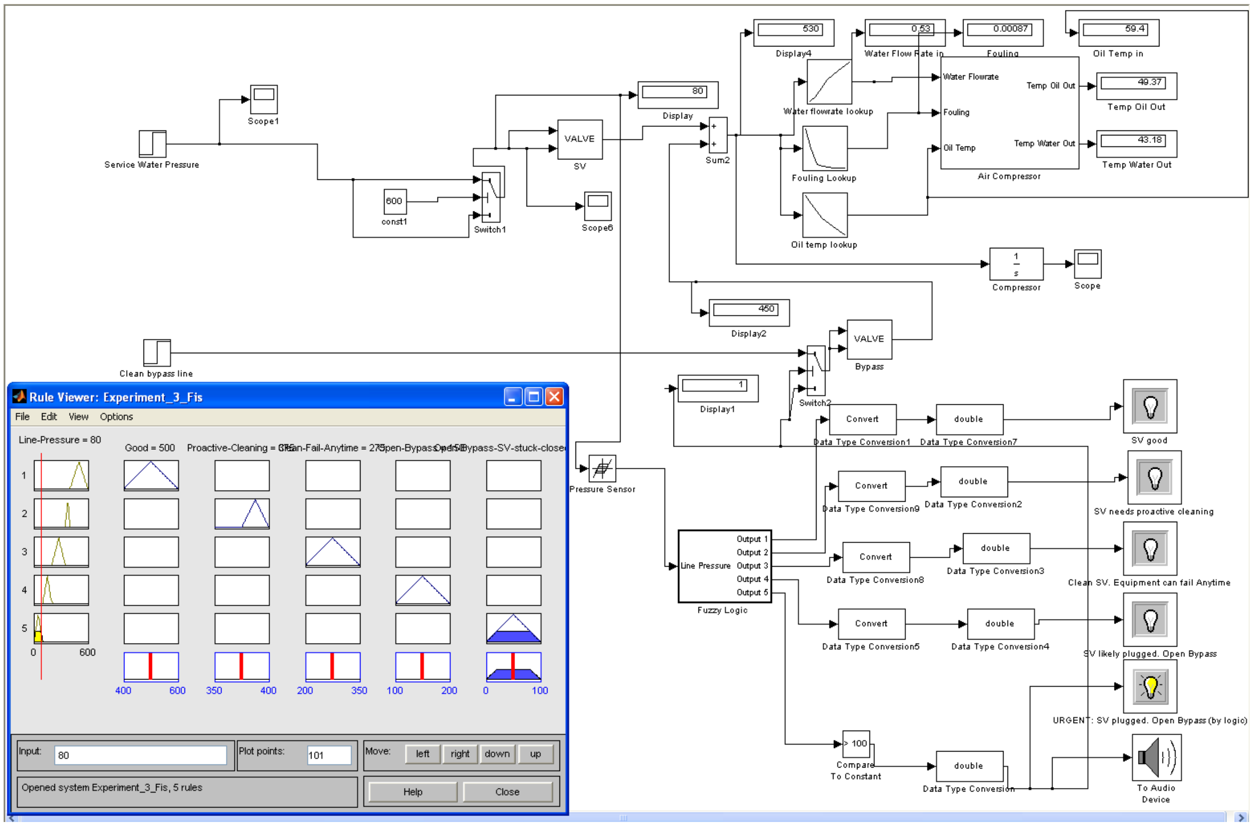

Figure 40: Simulink model results with SIS

Referring to Figure 40, Oil temperature of air compressor reduced from 59.4C to 49.37C.

NOTE: inlet Oil Temp is lower compared to 72.8C (without SIS) since rule based Fuzzy logic has opened bypass line (reading 450kpa) and line pressure feeding compressor is 530kpa (80kpa with SV plugged + 450kpa from bypass line).

In addition, annunciation 'URGENT. SV plugged. Open Bypass (by logic)' is also turned ON with an ALARM to notify operators that maintenance is needed to clean SV urgent before silt deposits are excessively fed to compressor heat exchangers (i.e. oil cooler, intercooler and aftercooler).

COMPARISION (No SIS Vs SIS):

No SIS:

Oil Temp (in) → 72.8C (bypass not open)

Oil Temp (out) → 57.45C (bypass not open)

Water Temp (in) → 35C (bypass not open)

Water Temp (out) → 47.51C (bypass not open)

Therefore, with no SIS, Oil temperature saw reduction of 15.35C and water temperature saw increase of 12.51C. Re-circulation of high temperature of Oil will eventually trip the compressor since the bypass line is not open and heat transfer efficiency will decrease.

With SIS (bypass line opened by fuzzy logic)

Oil Temp (in) → 59.4C

Oil Temp (out) → 49.37C

Water Temp (in) → 35C

Water Temp (out) → 43.18C

Therefore, with SIS installed, Oil temperature saw reduction of 10.03C and water temperature saw increase of 8.18C. Re-circulation of Oil temperature will remain low and not trip the compressor as the bypass line is opened by fuzzy logic and extra mass flowrate of water will maintain effective heat transfer rate.

NOTE: Operators also notified (with annunciation and alarm) of SV requiring urgent cleaning with SIS results modeled in SIMULINK that acquire less cleaning resources than cleaning compressor internals.

Cleaning SV requires less resources than cleaning compressor internals for which, more operation alignment is required and compressor is unavailable for service for longer duration reducing system redundancy (as only three compressors (out of four) be available to support unit station loads). Cleaning SV is few hours of work whereas cleaning compressor could take days or weeks.

In essence, using SIS with Fuzzy Logic results in reduced fouling factor, oil temperature and effective heat transfer rate to maintain high performance of air compressors.

Risk Matrix

			Risk Matrix		
			Consequences		
Probability	Insignificant	Minor	Moderate	Major	Catastrophic
Almost Certain					
Likely				Compressors/ Service Water degradation	
Moderate					
Unlikely					
Rare					
	Acceptable Risk	Low Risk	Medium Risk	High Risk	High Risk

Table 5: Risk Matrix [27]

Risk = Probability X Consequence.

Problem investigated using the case study is categorized as 'Likely' probability based on system failure rate of SVs over the years (Table 6). Data for Table 6 is collected from service water backlog tracked by work orders.

Count of WO	Year														
Row Labels	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	Grand Total
058-71310-SV854-								2		2			2	2	3	11
058-71310-SV855-						1		3		1			3	2	1	11
058-71310-SV856-					1			1		1	1		1	3	4	12
058-71310-SV857-		1	1					1	1		1	1		2	2	10
058-71310-SV858-		2			1			1	1		2	1	2	1	1	12
058-71310-SV859-					1		1		1					1	3	7
5-71310-SV850-				1				1	1					5	1	9
5-71310-SV851-		1					1	2	1		1			1		7
5-71310-SV852-					1	1		1	1	1		1	2	3	2	13
5-71310-SV853-					1			2	1	1	1		2	2	5	15
6-71310-SV850-			1	1					1	1	1	1		4	1	11
6-71310-SV851-		2					3	1	1		1		1	1	2	12
6-71310-SV852-		1					1		1			1	1	2	6	13
6-71310-SV853-		1						3		1			4	2	2	13
7-71310-SV850-															1	1
7-71310-SV851-						1					1		3	3	5	13
7-71310-SV852-		1			1						1		2	2	1	8
7-71310-SV853-					1	2		1	1	2	1		3	5	3	19
8-71310-SV850-	1	2					1			1			1	3	2	11
8-71310-SV851-		3	1			1	3	1	2	1	1			2	4	19
8-71310-SV852-		1					1				1	1	2	3	3	12
8-71310-SV853-		1					1		1				1	5	2	11
Grand Total	1	16	3	2	7	6	12	20	14	12	13	6	30	54	54	250

Table 6: SV failure rate on air compressors since 1998 to 2012 [34]

Consequences are judged as 'Major' due to problems pertaining to compressor degradation (requiring overhauls for repair tracked by system health reports [34]) and service water impairments.

Using SIS amalgamated results with fuzzy logic for condition based SV maintenance, it is concluded that installation of pressure transmitter as SIS will save huge cost incurs for PNGS in repairs and improve reliability of instrument air compressors/service water by detecting SV failures proactively. It ensures proper functioning of SV that will prevent compressor internal plugging by maintaining its operation per system design (i.e. open when signaled to open and close when signaled to close)

Example of Safety Instrumented Function:

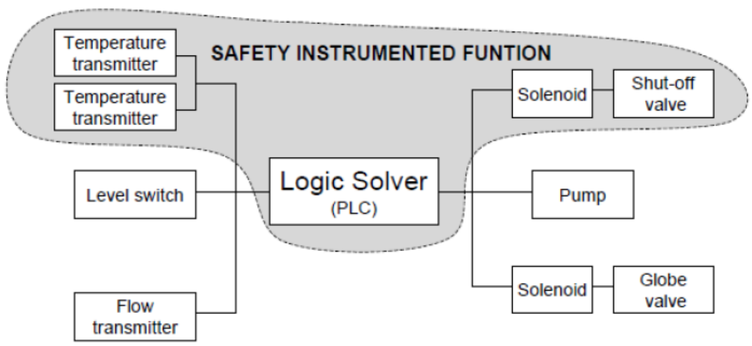

Figure 41: Pictorial depiction of SIF [27]

SIF of SIS is to "maintain safe state for the process industry in respect to hazardous event" [27]. In the case study, SIS provides operator annunciations (on timely basis) to avoid equipment degradation and flow diversion to service water.

Safety Integrity Level (SIL) "sets the performance target for the implementation in the form of the probability of Failure on Demand (PFD)" [27].

Performance target chosen for instrument air system and service water using SVs is 3. (SIL is 3). Probability of failure on demand is

$$(PFD) = 10^{-3}$$ Equation 36

Risk Reduction Factor:

$$RRF = \frac{1}{PFD} = \frac{1}{10^{-3}} = 1000$$ Equation 37

Without SIS.

SIL is low = a1

PFD = 10^{-1}

RRF = 10

Hence, SIS lowers the risk to the system (as also evident by the case study presented).

SIL	Required PFD	IPS AK	Old IPF Class
a1	$>10^{-1}$	-	I
a2	$>10^{-1}$	1	II
1	$<10^{-1}$	2/3	III
2	$<10^{-2}$	4	IV
3	$<10^{-3}$	5/6	V/VI
4	$<10^{-4}$	7	X

Table 7: PFD Requirements [27]

5.5 Analysis of Solenoid Valve (SV) Failures on Air Compressor Performance

Results of the case study demonstrated SV failure contributes to reduced air compressor performance (Modeled by SIMULINK) and require repetitive expensive repairs.

Based on OPEX, costs of repairs (without SIS) are as follows:

- Air compressor maintenance (approx)
- Labour + Parts → Up to $105,000 (Canadian $$)
- Burden to service water (approx)
- Labour + Parts → Up to $50,000 (Canadian $$)

NOTE: dollar values estimated at the discretion of System Engineers at PNGS.

Costs of repairs (with SIS) only involve proactive cleaning of Solenoid valves (to avoid long term compressor problems).

- Proactive SV cleaning (approx)
- Labour + Parts → Up to $1000 (Canadian $$)

Performance target of reporting SV condition and annunciations is given ‘Safety Integrity Level three (SIL 3) to improve risk reduction factor on timely basis and save huge cost incurs for the company. Using SIS with Fuzzy Logic results in reduced fouling factor, oil temperature and effective heat transfer rate to maintain high performance of air compressors. Installation of pressure transmitters with fuzzy logic will reduce maintenance burden, save company costs (in range of thousands of dollars) and improve system reliability by detecting proactive failures (via operator annunciation).

Please note pressure transmitter and associated logic will identify when silt collecting inside solenoid valve may potentially reduce flow of water below minimum required for compressor operation. Solution to this thesis is limited to prediction of early silt detection inside solenoid valve but not inside compressor internals.

Following principals of maintaining 'Value for Money and station safety, it is recommended that SIS (with fuzzy logic) be installed across Nuclear stations facing similar scenarios of silted water.

5.6 Reference Calculations

This section describes scenario calculation (modeled into Simulink) to describe how outlet water and oil temperatures were determined.

Hot oil inside air compressor is to be cooled in double tube counter flow heat exchanger (Oil Cooler, HX). HX has inner copper tubes (with diameter 2cm) and negligible thickness. Inner diameter of outer tube (shell) is 3cm.

Water flow through inner tube ($\dot{m}_w$) is

$\dot{m}_w \quad \rightarrow \quad Water\ mass\ flow\ rate\ \left({}^{kg}/_{sec}\right) = 0.5\ {}^{kg}/_{sec}$

$\dot{m}_o \quad \rightarrow \quad Oil\ mass\ flow\ rate\ \left({kg}/{sec}\right) = 0.8\ {kg}/{sec}$

$T_{water,in} \rightarrow$ 35℃ (This value is kept constant as lake water going into compressor is same for all modeling iterations)

$T_{oil,in} \rightarrow$ 60℃ (Value taken to start iterations of Oil temp. Other iterations will depend on heat transfer coefficient and how particulates on silt will impact heat transfer efficiency)

Properties of Water at 35℃ [26]

$\rho_w \rightarrow density\ of\ water = 994\ {Kg}/{m^3}$

$\mathrm{Pr_w} \rightarrow Prandtl\ number = 4.83$

$k_w \rightarrow Thermal\ Conductivity = 0.623\ W/m \cdot ℃$

$v_w \rightarrow Kinematic\ Viscosity = 0.724 * 10^{-6}\ {m^2}/{s}$

Please refer to Figure 37 for schematic of Oil Cooler (i.e. heat exchanger evaluated for case study)

Overall heat transfer coefficient be determined by:

$$\frac{1}{U} = \frac{1}{h_i} + \frac{1}{h_o}$$ **Equation 38**

Hydraulic diameter of circular tube is diameter of inner tube itself.

$D_{h,w} = 0.02m$

Now, we calculate average velocity of service water in inner copper tube and Reynolds number.

$$V_w = \frac{m_w}{\rho_w \cdot A_{c,inner\ tube}}$$ **Equation 39**

$$V_w = \frac{0.5\ kg/sec}{(994\ kg/m^3) \cdot (\frac{1}{4}\pi(0.02m)^2} = 1.6\ {}^{m}/_{sec}$$

$$Re_{water} = \frac{V_w D_{h,w}}{v_w}$$ **Equation 40**

$$Re_{water} = \frac{(1.6\ (m/\text{sec})) \cdot (0.02m)}{0.724 * 10^{-6}\ m^2/sec} = 44198.895$$

Reynolds number for service water is greater than 10,000 and termed as turbulent. Assuming this flow is fully developed, Nusselt number is next calculated.

$$Nu_{water} = \frac{h_i D_{h,w}}{k_{water}} = 0.023 \cdot Re_{water}^{0.8} \cdot Pr_{water}^{0.4}(22)$$

$$Nu_{water} = 0.023 \cdot (44198.89)^{0.8} \cdot (4.83)^{0.4} = 224.71$$

Therefore,

$$h_i = \frac{224.71 * 0.623(W/m \cdot ℃)}{0.02m}$$

$h_i = 6999.72 W / m^2 \cdot ℃$

Now, we repeat analysis for Oil.

Properties of Oil at 60□ [26]

$\rho_o \rightarrow density\ of\ oil = 863.9\ {}^{Kg}/_{m^3}$

$\mathrm{Pr_o} \rightarrow Prandtl\ number = 1080$

$k_o \rightarrow Thermal\ Conductivity = 0.1404\ W/m \cdot ℃$

$v_o \rightarrow Kinematic\ Viscosity = 8.565 * 10^{-5}\ {}^{m^2}/_{s}$

Hydraulic Diameter for the annular space is

$D_{h,o} = 0.03 - 0.02 = 0.01m$

$$V_o = \frac{\dot{m}_o}{\rho_o \cdot A_{c,outer\ tube}} \quad \text{Equation 41}$$

$$V_o = \frac{0.8\ kg/sec}{(863.9\ kg/m^3) \cdot (\frac{1}{4}\pi(0.03 - 0.02)m^2} = 2.36\ {}^{m}/_{sec}$$

$$Re_{oil} = \frac{V_o D_{h,o}}{v_o}$$ **Equation 42**

$$Re_{oil} = \frac{(2.36\ (m/\text{sec})) \cdot (0.01m)}{8.565 * 10^{-5}\ m^2/sec} = 275.54$$

Reynolds number for oil is less than 2300 and termed as laminar. Referring to Table 8 and the oil flow is fully developed, Nusselt number is next calculated.

D(i)/D(o)	Nu
0	
0.05	17.46
0.1	11.56
0.25	7.37
0.5	5.74
1	4.86

Table 8: Nusselt number for fully developed laminar flow in a circular annulus with one surface insulated and the other isothermal [26]

$$\frac{D_i}{D_o} = \frac{0.02}{0.03} = 0.667$$ **Equation 43**

Referring to Table 8 and interpolating for D(i)/D(o) as 0.667,

$$Nu_{oil} = 5.45$$

Oil convection heat transfer coefficient is

$$h_o = \frac{k_o}{D_{h,oil}} \cdot Nu_{oil}$$ **Equation 44**

$$h_o = \frac{0.1404(W/m \cdot ℃) \cdot (5.45)}{0.01m}$$

$$h_o = 76.52\ (W/m^2 \cdot ℃)$$

Calculating Oil and water temperature going out

$c_{p,water}$ → 4.18 $(^{KJ}/_{Kg} \cdot ℃)$

$c_{p,oil}$ → 2.13 $(^{KJ}/_{Kg} \cdot ℃)$

Calculating Heat Capacity rates for water and oil

$$C_{water} = \dot{m_w} \cdot c_{p,water}$$ **Equation 45**

$$= 0.5(^{kg}/_{sec}) X\ 4.18\ (^{KJ}/_{Kg} \cdot ℃) = 2.09\ ^{KW}/_{C}$$

$$C_{oil} = \dot{m_o} \cdot c_{p,oil}$$ **Equation 46**

$= 0.8(^{kg}/_{sec}) X\ 2.13\ (^{KJ}/_{Kg} \cdot °C) = 1.704\ ^{KW}/_{C}$

$C_{min} = C_{oil} = 1.704\ ^{KW}/_{C}$

$$c = \frac{C_{min}}{C_{max}}$$ **Equation 47**

$= \frac{1.704(^{KW}/_{C})}{2.09(^{KW}/_{C})} = 0.815$

Maximum possible heat transfer rate in the oil cooler.

$$\dot{Q}_{max} = C_{min}(T_{oil,in} - T_{water,in})$$ **Equation 48**

$\dot{Q}_{max} = 1.704 \frac{KW}{C}\ (60°C - 35°C) =\ 42.6\ KW$

$$A_s = \pi * D * L$$ **Equation 49**

$= \pi * (0.02) * (10) = 0.628m^2$

Now, we calculate overall heat transfer coefficient (U). Fouling factor (R_f) used is 0.0009 for this case wherein, line service water pressure is 500kpa (Table 4).

$$U = \frac{1}{1/_{hi} + 1/_{ho} + R_f} \qquad \text{Equation 50}$$

$$U = \frac{1}{1/_{6999.72} + 1/_{76.52} + 0.0009}$$

$$U = 70.86\ (\frac{W}{m^2}℃)$$

Using Effectives NTU (number of transfer units) method [26], Oil cooler efficiency can be calculated.

$$NTU = \frac{UA_s}{C_{min}} \qquad \text{Equation 51}$$

$$NTU = \frac{70.86\ (W/m^2 \cdot ℃) * 0.628m^2}{1704\ (W/℃)}$$

$$NTU = 0.026$$

With c = 0.815, NTU = 0.026 and referring to Figure 38, ∈ is calculated to be 0.04

Therefore, actual heat transfer rate is:

$$\dot{Q} = \in \dot{Q}_{max} \qquad \text{Equation 52}$$

$\dot{Q} = Q_{max} = 0.04 * 4\dot{2}.6 = 1.704\ KW$

Finally, outlet Oil and Water temperature are as follows:

$$T_{water,out} = T_{water,in} + \frac{\dot{Q}}{C_{water}}$$ **Equation 53**

$$T_{water,out} = 35℃ + \frac{1.704\ KW}{2.09\ KW}$$

$$T_{water,out} = 35.81\ ℃$$

$$T_{oil,out} = T_{oil,in} - \frac{\dot{Q}}{C_{oil}}$$ **Equation 54**

$$T_{oil,out} = 60℃ + \frac{1.704\ KW}{1.704\ KW}$$

$$T_{oil,out} = 61\ ℃$$

Chapter 6 – Results and Discussions

6.1 Fuzzy Logic Vs PID Control

First case study investigated system parameters with and without fuzzy logic.

Simulation in (Simulink) was run with PID Controller for 60 secs. Results demonstrate system pressure did not settle for 60 secs.

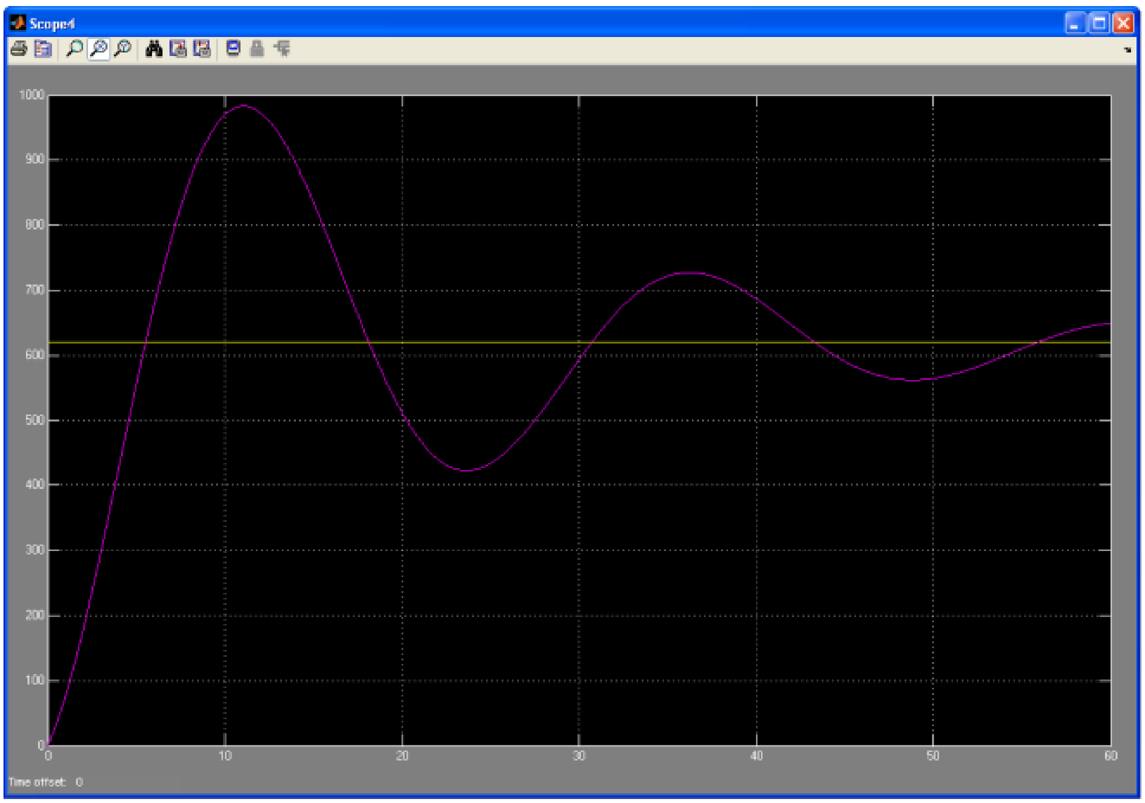

Figure 42: Simulation results for conventional PID controller for 60 secs

To further understand results, PID Simulation was repeated for 150 secs with following results.

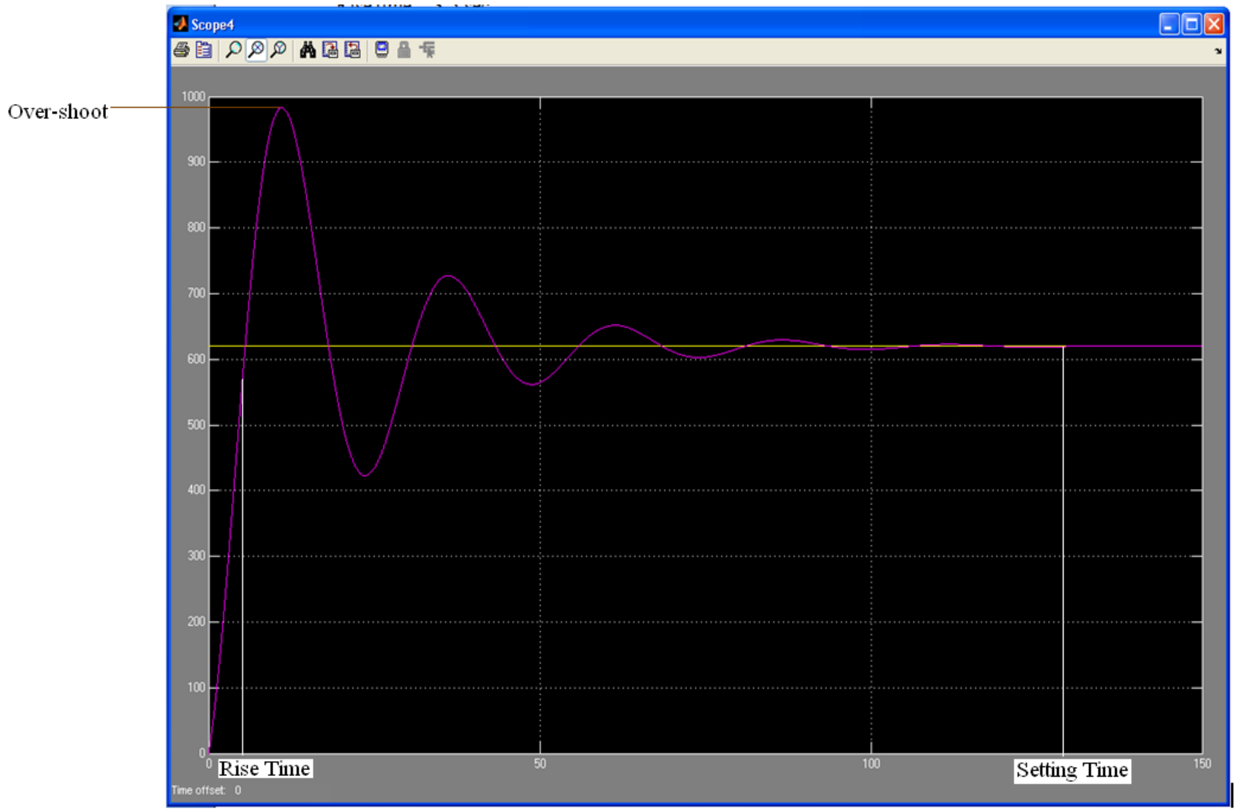

Figure 43: Simulation results for conventional PID controller for 120 secs to understand system stability

To compare results with fuzzy logic, simulation was repeated with similar scenario using Fuzzy logic driven PID controller for 60 secs with following results.

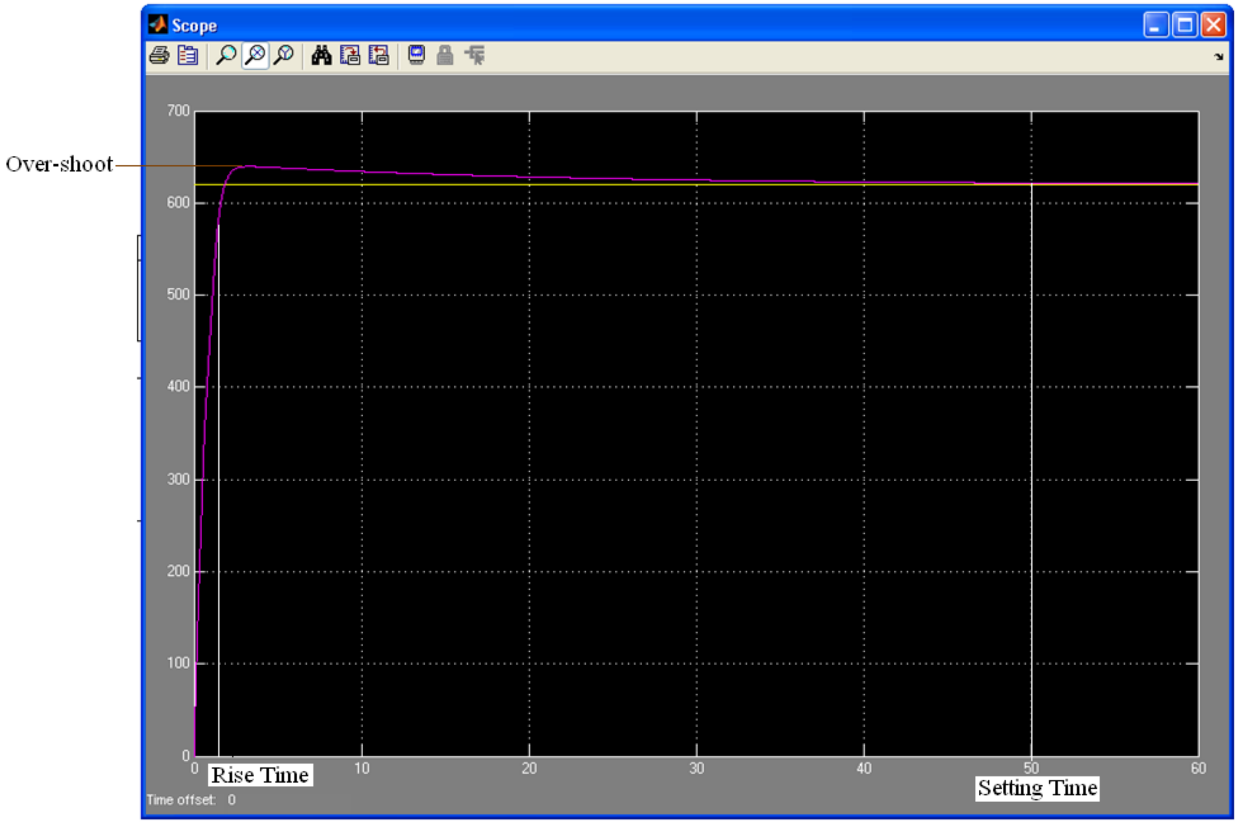

Figure 44: Simulation results for Fuzzy PID controller for 60 secs

Rise time	approx 6 sec (90% of 620kpa = 558kpa)
Over-shoot	990kpa
Settling time	130sec
S-S error	0.907 (approx)

Table 9: Simulation results with simple PID controller

Rise time	approx 2.5 sec (90% of 620kpa = 558kpa)
Over-shoot	640kpa
Settling time	50sec
S-S error	0.907

Table 10: Simulation results for Fuzzy+PID controller

It is evident comparing Table 9 & 10 that fuzzy logic driven PID controller produces reduced rise-time, reduced overshoot and setting time compared against conventional PID controller.

It is also critical to note per procedures, anytime breathing air pressure inside reactor building (RB) reaches below 550kpa, emergency alarm is initiated in control room and all personnel working inside RB building are directed to evaluate to restore system pressure back to 620kpa (i.e. system design pressure). Evacuation of personnel also results in delay to maintenance activities (inside planned reactor shutdown schedule), which leads to cost of $60,000 (approx) loss to the company.

To demonstrate results using Simulink, a comparator was implemented to calculate area of running model beneath 550kpa. Both controllers were run for 150sec to understand their cost relation with respect to model stability. Conventional PID Controller results were as follows:

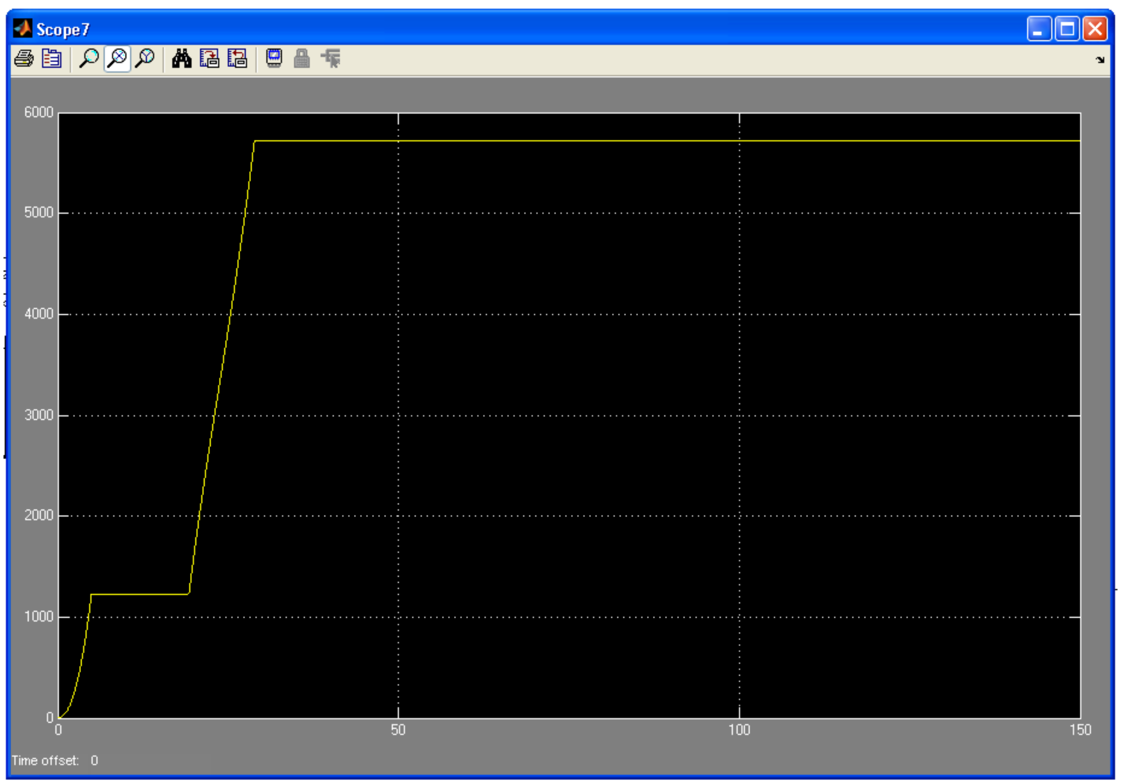

Figure 45: Graph depicting area underneath 550kpa using PID controller

Total area underneath 550kpa = 5750 (approx)

Judging by simulation graph, it is evident system pressure reached below 550kpa after rise time at 20sec. In Nuclear Power Plant, this would initiate breathing air pressure low emergency alarm and personnel will be directed to evacuate RB building. All critical work be stopped resulting in outage delay of approx 3hrs (till system pressure restores and stabilizes).

Predicted outage delay cost with PID controller.

$$3hr \times \frac{\$20,000}{hr} = \$60,000\ (Canadian\ dollars)$$ Equation 55

Simulink was run again with Fuzzy+PID controller for 150secs are shown

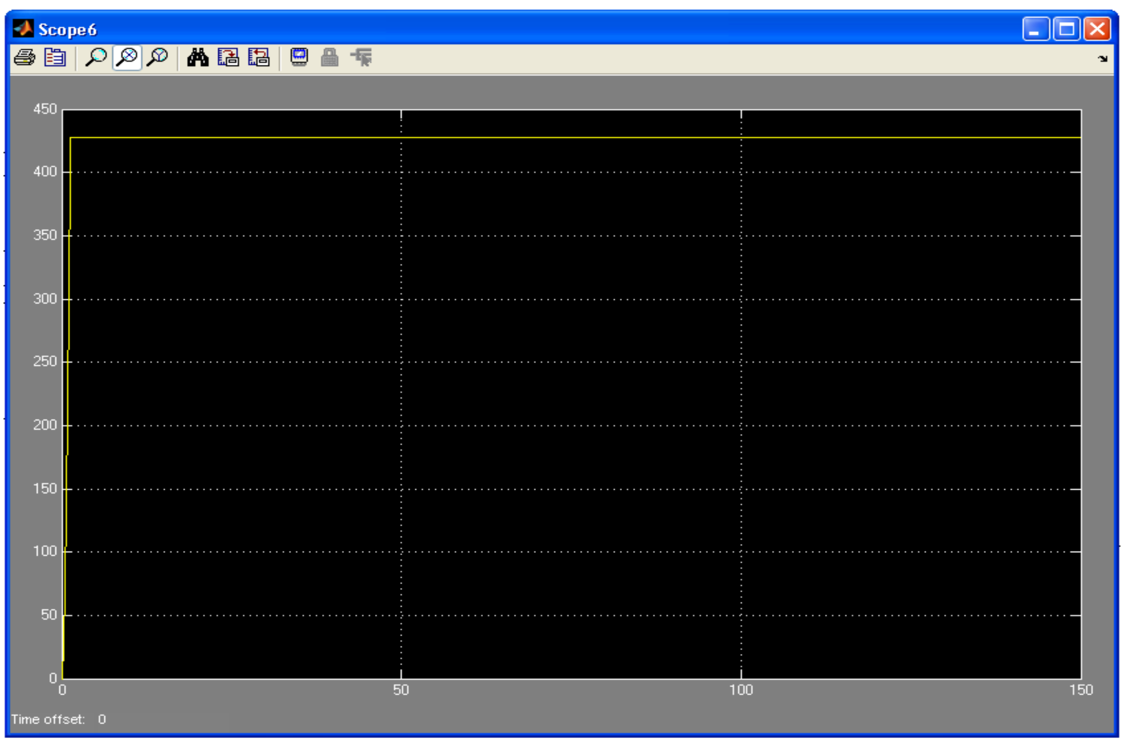

Figure 46: Graph depicting area underneath 550kpa using Fuzzy + PID controller

Total area underneath 550kpa = 428 (approx).

Area calculated during the initial rise time is neglected since operators would not be sending personnel into RB building until system has reached pressure above 550kpa (i.e. no alarms be initiated into the control room). Confirmed per Simulink simulation results, system pressure never reaches below 550kpa after initial rise time (using fuzzy driven PID controller).

Therefore, there be no outage schedule delay and predicted delay cost using Fuzzy + PID controller be zero dollars.

$$0hr \times \frac{\$20{,}000}{hr} = \$0(Canadian\ dollars)$$

Equation 56

Results demonstrate fuzzy PID controller has superior control and precision in maintaining system design pressure with reduced rise-time, overshoot, settling time and steady-state error compared against conventional PID. Furthermore, using PID conventional controller will cost extra $60,000 for losses incurred due to instability in the system (Table 11).

	PID Controller	Fuzzy PID Controller
Rise Time (sec)	6	2.5
Over-shoot (kpa)	990	640
Setting Time (sec)	130	50
Damage Cost ($$)	60,000	0

Table 11: PID Vs Fuzzy PID Controller

6.2 Performance Improvement Of Instrument Air Compressors

Simulink application was used again to compare results with and without fuzzy logic.

6.2.1 Simulation Without Safety Instrumented System (SIS)

Scenario considered for Air Compressor: Inlet water pressure at 80kpa and SV stuck open → worst case for Air Compressor. This represents compressors acquiring cooling capacity with solenoid valve failed at open position with water flow maintained at 80kpa (pressure) flowing through compressors and analyzing its consequences without SIS.

As evident, Oil temperature of air compressor ONLY reduced from 72.8C to 57.45C with water temperature increased from 35C to 47.51C. Fouling factor is 0.17 (Table 4) based on line pressure. An increasing fouling number is a sign of more silt/particulates to negatively impact heat transfer rate and degrade compressor performance.

In long run, re-circulating Oil temperature will continue to rise due to SV stuck in open position and Oil cooler will continue to plug up further decreasing heat transfer efficiency and will eventually trip the compressor.

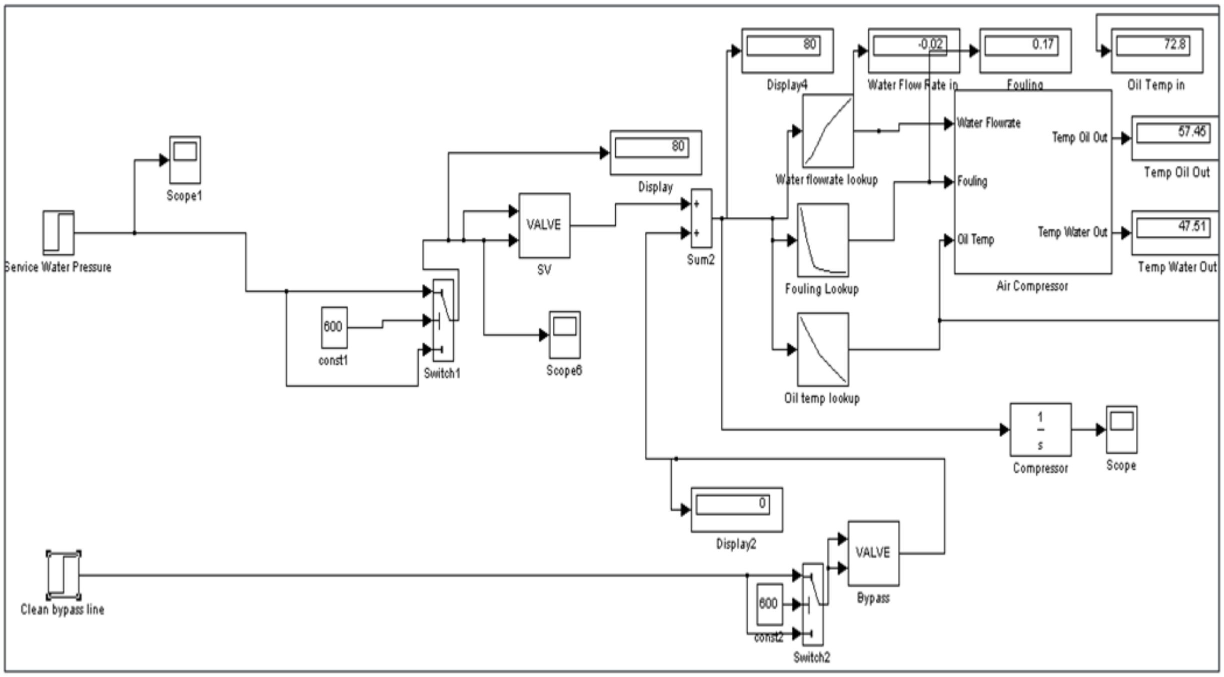

Figure 47: Simulink model results without SIS

6.2.2 Simulation With Safety Instrumented System (SIS) using Fuzzy logic

Scenario considered for Air Compressor: Inlet water pressure at 80kpa and SV stuck open → worse case for Air Compressor. This represents compressors acquiring cooling capacity with solenoid valve failed at open position with water flow maintained at 80kpa (pressure) flowing through compressors and analyzing its consequences with SIS.

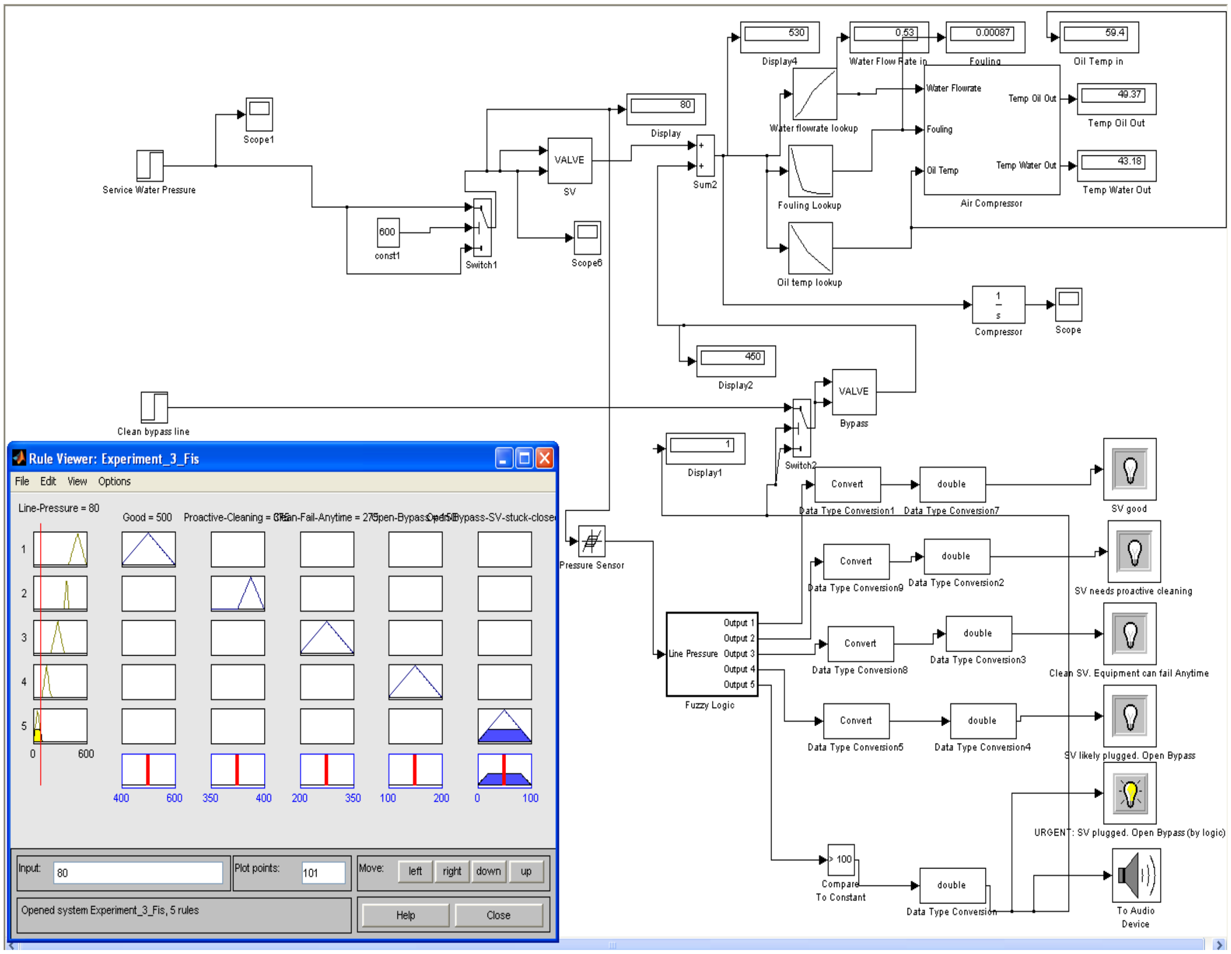

Figure 48: Simulink model results with SIS

Oil temperature of air compressor reduced from 59.4C to 49.37C

NOTE: inlet Oil Temp is lower compared to 72.8C (without SIS) since rule based Fuzzy logic has opened bypass line (reading 450kpa) and line pressure feeding compressor is 530kpa (80kpa with SV plugged + 450kpa from bypass line).

In addition, annunciation 'URGENT. SV plugged. Open Bypass (by logic)' is also turned ON with an ALARM to notify operators that maintenance is needed to clean SV urgent before silt deposits are excessively fed to compressor heat exchangers (i.e. oil cooler, intercooler and aftercooler).

COMPARISION (No SIS Vs SIS):

No SIS:

Oil Temp (in) → 72.8C (bypass not open)

Oil Temp (out) → 57.45C (bypass not open)

Water Temp (in) → 35C (bypass not open)

Water Temp (out) → 47.51C (bypass not open)

Therefore, with no SIS, Oil temperature saw reduction of 15.35C and water temperature saw increase of 12.51C. Re-circulation of high temperature of Oil will eventually trip the compressor since the bypass line is not open and heat transfer efficiency will decrease.

With SIS (bypass line opened by fuzzy logic)

Oil Temp (in) → 59.4C

Oil Temp (out) → 49.37C

Water Temp (in) → 35C

Water Temp (out) → 43.18C

Therefore, with SIS installed, Oil temperature saw reduction of 10.03C and water temperature saw increase of 8.18C. Re-circulation of Oil temperature will remain low and not trip the compressor as the bypass line is opened by fuzzy logic and extra mass flowrate of water will maintain effective heat transfer rate.

NOTE: Operators also notified (with annunciation and alarm) of SV requiring urgent cleaning with SIS results modeled in SIMULINK that acquire less cleaning resources than cleaning compressor internals.

Cleaning SV requires fewer resources than cleaning compressor internals for which, more operation alignment is required and compressor is unavailable for service for longer duration reducing system redundancy (as only three compressors (out of four) be available to support station loads). Cleaning SV is few hours of work whereas cleaning compressor can take weeks.

In essence, using SIS with Fuzzy Logic results in reduced fouling factor, oil temperature and effective heat transfer rate to maintain high performance of air compressors.

6.2.3 Analysis Of Solenoid Valve (SV) Failures On Air Compressor Performance

Results of the case study demonstrated SV failure contributes to reduced air compressor performance (Modeled by SIMULINK) and require repetitive expensive repairs.

Based on OPEX, costs of repairs (without SIS) are as follows:

- Air compressor maintenance (approx)
- Labour + Parts → Up to $105,000
- Burden to service water (approx)
- Labour + Parts → Up to $50,000

NOTE: dollar values estimated at the discretion of System Engineers at PNGS.

Costs of repairs (with SIS) only involve proactive cleaning of Solenoid valves (to avoid long term compressor problems).

- Proactive SV cleaning (approx)
- Labour + Parts → Up to $1000

Performance target of reporting SV condition and annunciations is given ‘Safety Integrity Level three (SIL 3) to improve risk reduction factor on timely basis and save huge cost incurs for the company. Using SIS with Fuzzy Logic results in reduced fouling factor, oil temperature and effective heat transfer rate to maintain high performance of air compressors. Installation of pressure transmitters with fuzzy logic will reduce maintenance burden, save company costs (in range of thousands of dollars) and improve system reliability by detecting proactive failures (via operator annunciation).

Please note pressure transmitter and associated logic will identify when silt collecting inside solenoid valve may potentially reduce flow of water below minimum required for compressor operation. Solution to this thesis is limited to prediction of early silt detection inside solenoid valve but not inside compressor internals.

Following principals of maintaining ‘Value for Money and station safety, it is recommended that SIS (with fuzzy logic) be installed across Nuclear stations facing similar scenarios of silted water.

6.3 Discussions

Results discussed in Sec 6.1 demonstrate fuzzy PID controller has superior precision and control in maintaining system design pressure with reduced rise time (2.5 sec Vs 6 sec in PID controller), overshoot (640kpa Vs 990kpa in PID controller) and settling time (50 sec Vs 130 sec in PID controller). Fuzzy PID controller further saves up to $60,000 (approx) losses, which be prevented due to system stability. In addition, better breathing air system control by implementing fuzzy driven PID controller will also improve safety of personnel in reactor building and operational performance of breathing air compressors. With proper control regulation in-place, compressor burden will reduce and help increase compressor equipment life, mean time between failures and system availability. It is recommended that fuzzy PID controller be implemented to breathing air systems at nuclear power stations in Ontario for optimized pressure control.

Results discussed in Sec 6.2 demonstrate fuzzy logic driven System instrumented system (SIS) results in reduced fouling factor (0.17 based on line pressure), oil temperature (reduction of 10.03C Vs 15.35C without SIS) and effective heat transfer rate (water temperature saw increase of 8.18C Vs 12.51 without SIS) to maintain high performance of instrument air compressors. Fuzzy logic driven SIS further demonstrates reduce maintenance burden, which results in savings up to $150,000 (approx) cost incurs for the company by detecting proactive valve failures. In addition, better instrument air compressor operation by implementing fuzzy logic will improve Nuclear Safety with increased system availability and redundancy. With proper control detection system to initiate valve cleaning, compressor equipment life and mean time between failures will increase. Hence, it is recommended that fuzzy logic be implemented to instrument air system at nuclear power stations for optimized pressure control.

Aligned with principals of Nuclear Safety and Value for money, fuzzy logic is a must implementation for breathing air system and instrument air compressors at Nuclear Power Plants. Its installation should be further considered for other safety related systems to make use of logic output and performance.

Examples of critical systems for consideration could include:

- Reactivity Control Units
- Moderator Main and Helium Cover Gas System
- Moderator D2O Collection System
- Moderator Liquid Poison System
- Heat Transport System
- Emergency Coolant Injection System
- Liquid Zone Control System
- Boiler Steam and Water System
- Turbine Generator Governing System
- Boiler Feed System
- Common Water Supply System
- Condenser Cooling Water System
- Sediment Suction System
- Sewage System
- Powerhouse Ventilation System
- Screenhouse Heating and Ventilation System

Chapter 7 - Conclusions and Future Work

7.1 Conclusion

This thesis studied three objectives. Firstly, to design self-tuning control system applied on breathing air system to enhance its performance. Case study 1 concluded with self-tuning fuzzy logic driven PID controller design, which was applied on breathing air system and enhanced its performance by superior precision and control in maintaining system design pressure with reduced rise time (2.5 sec Vs 6 sec in PID controller), overshoot (640kpa Vs 990kpa in PID controller) and settling time (50 sec Vs 130 sec in PID controller). Fuzzy PID controller further saves up to $60,000 (approx) losses, which be prevented due to system stability.

Second objective was to design fuzzy logic system to detect solenoid valve failures proactively to improve instrument air compressor performance. Case Study 2 concluded with safety integrated system designed to impact instrument air compressors which resulted in reduced fouling factor (0.17 based on line pressure), oil temperature (reduction of 10.03C Vs 15.35C without SIS) and effective heat transfer rate (water temperature saw increase of 8.18C Vs 12.51 without SIS) to maintain high performance of instrument air compressors. Fuzzy logic driven SIS further demonstrated reducing maintenance burden and savings up to $150,000 (approx) cost incurs for the company by detecting proactive solenoid valve failures early to prevent long term compressor failures and increase system reliability.

Third objective was to determine feasibility of fuzzy logic implementation. Based on results of two case studies studied on two safety related systems that provided logic applicability, improved system performance and operating costs, fuzzy logic is highly recommended for installation at other Nuclear utilities. Examples of related systems would include Reactivity Control Units, Moderator Main and Helium Cover Gas System, Moderator D2O Collection System etc.

Furthermore, impact of fuzzy logic is also good consideration to justify how potential rule based scenarios can be used to control disasters like Chernobyl (1986), Three Mile Island (1979) and Fukushima Daiichi (2011) incidents. Machines cannot make mistakes and more enhanced rules (programmed into fuzzy logic) can further improve safety of Nuclear Power Plants in preventing accidents.

7.2 Contribution and Innovation

Successful research in studying fuzzy logic and its applicability also led to two successful journal publications as follows:

Deol, Harsh, and Hossam A. Gabbar. "Self-tuning Fuzzy Logic PID Controller, Applications in Nuclear Power Plants." *IJISTA International Journal of Intelligent Systems Technologies and Applications* 14.1 (2015): 70. Web.

Deol, Harsh, and Hossam A. Gabbar. "Fuzzy Logic-based Safety Design for High Performance Air Compressors." *Progress in Nuclear Energy* 80 (2015): 136-50. Web.

Operating Nuclear Power Plants around the world are highly encouraged to consider fuzzy logic for respective control systems to ensure better performance and precision. It is an opportunity to improve safety performance throughout the world with a known application that has proved consistently to produce great results (supported by research). Candu Owners Group (COG), Nuclear utilities (connected via OPEX program), International nuclear conferences, System Engineers, Vendors etc are encouraged to review applicability of fuzzy logic for related systems.

7.3 Future Work

Performance of fuzzy logic intelligent control system improves operation and operability costs of safety related systems in Nuclear power plant. But its functionality is not used widely in the Nuclear sector and needs attention. However, these days few plants are starting to take note of intelligent control and slowly making process transition to utilize its benefits. Ex: Pickering Nuclear is using robots to conduct radiation surveys and other related jobs inside reactor buildings. Darlington Nuclear is also reviewing usage of robotic application. Intelligent System applicability and usage is highly encouraged for review and consideration. To further keep sensitivity of Nuclear industry towards preciseness, it is recommended applicability of ‘fuzzy’ termed logic be replaced with word ‘Rule-Based Learning algorithms’ for better perception.

I would further like to continue this research for selection into PhD related thesis and would like to investigate how beyond design based events (such as Fukushima Daiichi (2011) incident) can be prevented using fuzzy logic. Severe Accident Mitigation guidelines (SAMGs) addresses steps to take involving accidents that are beyond design based and it is recommended that study and investigation be conducted to investigate preventing scenarios wherein, safe stating equipment is necessary to shut down reactor in emergencies with fuzzy logic.

Chapter 8 - SIMULINK Algorithms

8.1 – Algorithm for Fuzzy Logic Based Self-Tuning Control System for Breathing Air System

```
Model {
  Name            "Closed_loop_Setup4_BA_wrkingaftercliff_rev5"
  Version         7.0
  MdlSubVersion       0
  GraphicalInterface {
    NumRootInports      0
    NumRootOutports     0
    ParameterArgumentNames  ""
    ComputedModelVersion    "1.43"
    NumModelReferences      0
    NumTestPointedSignals   0
  }
  SavedCharacterEncoding  "windows-1252"
  SaveDefaultBlockParams  on
  SampleTimeColors    off
  LibraryLinkDisplay      "none"
  WideLines       off
  ShowLineDimensions      off
  ShowPortDataTypes   off
  ShowLoopsOnError    on
  IgnoreBidirectionalLines off
  ShowStorageClass    off
  ShowTestPointIcons      on
  ShowViewerIcons     on
  SortedOrder         off
  ExecutionContextIcon    off
  ShowLinearizationAnnotations on
  ScopeRefreshTime    0.035000
  OverrideScopeRefreshTime on
  DisableAllScopes    off
  DataTypeOverride    "UseLocalSettings"
  MinMaxOverflowLogging   "UseLocalSettings"
  MinMaxOverflowArchiveMode "Overwrite"
  BlockNameDataTip    off
  BlockParametersDataTip  off
  BlockDescriptionStringDataTip off
  ToolBar         on
  StatusBar       on
  BrowserShowLibraryLinks off
  BrowserLookUnderMasks   off
  Created         "Tue Apr 03 12:34:39 2012"
  Creator         "Owner"
  UpdateHistory       "UpdateHistoryNever"
  ModifiedByFormat    "%<Auto>"
  LastModifiedBy      "Owner"
  ModifiedDateFormat      "%<Auto>"
  LastModifiedDate    "Sat Apr 28 15:32:48 2012"
  RTWModifiedTimeStamp    0
  ModelVersionFormat      "1.%<AutoIncrement:43>"
  ConfigurationManager    "None"
```

```
SimulationMode      "normal"
LinearizationMsg    "none"
Profile         off
ParamWorkspaceSource    "MATLABWorkspace"
AccelSystemTargetFile   "accel.tlc"
AccelTemplateMakefile   "accel_default_tmf"
AccelMakeCommand    "make_rtw"
TryForcingSFcnDF    off
RecordCoverage      off
CovPath         "/"
CovSaveName         "covdata"
CovMetricSettings   "dw"
CovNameIncrementing     off
CovHtmlReporting    on
covSaveCumulativeToWorkspaceVar on
CovSaveSingleToWorkspaceVar on
CovCumulativeVarName    "covCumulativeData"
CovCumulativeReport     off
CovReportOnPause    on
ExtModeBatchMode    off
ExtModeEnableFloating   on
ExtModeTrigType     "manual"
ExtModeTrigMode     "normal"
ExtModeTrigPort     "1"
ExtModeTrigElement      "any"
ExtModeTrigDuration     1000
ExtModeTrigDurationFloating "auto"
ExtModeTrigHoldOff      0
ExtModeTrigDelay    0
ExtModeTrigDirection    "rising"
ExtModeTrigLevel    0
ExtModeArchiveMode      "off"
ExtModeAutoIncOneShot   off
ExtModeIncDirWhenArm    off
ExtModeAddSuffixToVar   off
ExtModeWriteAllDataToWs off
ExtModeArmWhenConnect   on
ExtModeSkipDownloadWhenConnect off
ExtModeLogAll       on
ExtModeAutoUpdateStatusClock on
BufferReuse         on
ShowModelReferenceBlockVersion off
ShowModelReferenceBlockIO off
Array {
  Type            "Handle"
  Dimension           1
  Simulink.ConfigSet {
    $ObjectID           1
    Version             "1.3.0"
    Array {
  Type            "Handle"
  Dimension       8
  Simulink.SolverCC {
    $ObjectID       2
    Version         "1.3.0"
    StartTime       "0.0"
    StopTime        "150"
```

```
  AbsTol          "auto"
  FixedStep       "auto"
  InitialStep         "auto"
  MaxNumMinSteps      "-1"
  MaxOrder        5
  ConsecutiveZCsStepRelTol "10*128*eps"
  MaxConsecutiveZCs   "1000"
  ExtrapolationOrder      4
  NumberNewtonIterations  1
  MaxStep         "auto"
  MinStep         "auto"
  MaxConsecutiveMinStep   "1"
  RelTol          "1e-3"
  SolverMode          "Auto"
  Solver          "ode45"
  SolverName          "ode45"
  ZeroCrossControl    "UseLocalSettings"
  AlgebraicLoopSolver     "TrustRegion"
  SolverResetMethod   "Fast"
  PositivePriorityOrder   off
  AutoInsertRateTranBlk   off
  SampleTimeConstraint    "Unconstrained"
  RateTranMode        "Deterministic"
}
Simulink.DataIOCC {
  $ObjectID       3
  Version         "1.3.0"
  Decimation          "1"
  ExternalInput       "[t, u]"
  FinalStateName      "xFinal"
  InitialState        "xInitial"
  LimitDataPoints     on
  MaxDataPoints       "1000"
  LoadExternalInput   off
  LoadInitialState    off
  SaveFinalState      off
  SaveFormat          "Array"
  SaveOutput          on
  SaveState       off
  SignalLogging       on
  InspectSignalLogs   off
  SaveTime        on
  StateSaveName       "xout"
  TimeSaveName        "tout"
  OutputSaveName      "yout"
  SignalLoggingName   "logsout"
  OutputOption        "RefineOutputTimes"
  OutputTimes         "[]"
  Refine          "1"
}
Simulink.OptimizationCC {
  $ObjectID       4
  Array {
    Type            "Cell"
    Dimension           5
    Cell            "ZeroExternalMemoryAtStartup"
    Cell            "ZeroInternalMemoryAtStartup"
```

```
    Cell            "InitFltsAndDblsToZero"
    Cell            "OptimizeModelRefInitCode"
    Cell            "NoFixptDivByZeroProtection"
    PropName            "DisabledProps"
  }
  Version         "1.3.0"
  BlockReduction      on
  BooleanDataType     on
  ConditionallyExecuteInputs on
  InlineParams        off
  InlineInvariantSignals  off
  OptimizeBlockIOStorage  on
  BufferReuse         on
  EnforceIntegerDowncast  on
  ExpressionFolding   on
  ExpressionDepthLimit    2147483647
  FoldNonRolledExpr   on
  LocalBlockOutputs   on
  RollThreshold       5
  SystemCodeInlineAuto    off
  StateBitsets        off
  DataBitsets         off
  UseTempVars         off
  ZeroExternalMemoryAtStartup on
  ZeroInternalMemoryAtStartup on
  InitFltsAndDblsToZero   on
  NoFixptDivByZeroProtection off
  EfficientFloat2IntCast  off
  OptimizeModelRefInitCode off
  LifeSpan        "inf"
  BufferReusableBoundary  on
  SimCompilerOptimization "Off"
  AccelVerboseBuild   off
}
Simulink.DebuggingCC {
  $ObjectID       5
  Version         "1.3.0"
  RTPrefix        "error"
  ConsistencyChecking     "none"
  ArrayBoundsChecking     "none"
  SignalInfNanChecking    "none"
  SignalRangeChecking     "none"
  ReadBeforeWriteMsg      "UseLocalSettings"
  WriteAfterWriteMsg      "UseLocalSettings"
  WriteAfterReadMsg   "UseLocalSettings"
  AlgebraicLoopMsg    "warning"
  ArtificialAlgebraicLoopMsg "warning"
  SaveWithDisabledLinksMsg "warning"
  SaveWithParameterizedLinksMsg "warning"
  CheckSSInitialOutputMsg on
  CheckExecutionContextPreStartOutputMsg off
  CheckExecutionContextRuntimeOutputMsg off
  SignalResolutionControl "UseLocalSettings"
  BlockPriorityViolationMsg "warning"
  MinStepSizeMsg      "warning"
  TimeAdjustmentMsg   "none"
  MaxConsecutiveZCsMsg    "error"
```

```
  SolverPrmCheckMsg   "warning"
  InheritedTsInSrcMsg     "warning"
  DiscreteInheritContinuousMsg "warning"
  MultiTaskDSMMsg     "error"
  MultiTaskCondExecSysMsg "error"
  MultiTaskRateTransMsg   "error"
  SingleTaskRateTransMsg  "none"
  TasksWithSamePriorityMsg "warning"
  SigSpecEnsureSampleTimeMsg "warning"
  CheckMatrixSingularityMsg "none"
  IntegerOverflowMsg      "warning"
  Int32ToFloatConvMsg     "warning"
  ParameterDowncastMsg    "error"
  ParameterOverflowMsg    "error"
  ParameterUnderflowMsg   "none"
  ParameterPrecisionLossMsg "warning"
  ParameterTunabilityLossMsg "warning"
  UnderSpecifiedDataTypeMsg "none"
  UnnecessaryDatatypeConvMsg "none"
  VectorMatrixConversionMsg "none"
  InvalidFcnCallConnMsg   "error"
  FcnCallInpInsideContextMsg "Use local settings"
  SignalLabelMismatchMsg  "none"
  UnconnectedInputMsg     "warning"
  UnconnectedOutputMsg    "warning"
  UnconnectedLineMsg      "warning"
  SFcnCompatibilityMsg    "none"
  UniqueDataStoreMsg      "none"
  BusObjectLabelMismatch  "warning"
  RootOutportRequireBusObject "warning"
  AssertControl       "UseLocalSettings"
  EnableOverflowDetection off
  ModelReferenceIOMsg     "none"
  ModelReferenceVersionMismatchMessage "none"
  ModelReferenceIOMismatchMessage "none"
  ModelReferenceCSMismatchMessage "none"
  ModelReferenceSimTargetVerbose off
  UnknownTsInhSupMsg      "warning"
  ModelReferenceDataLoggingMessage "warning"
  ModelReferenceSymbolNameMessage "warning"
  ModelReferenceExtraNoncontSigs "error"
  StateNameClashWarn      "warning"
  StrictBusMsg        "Warning"
  LoggingUnavailableSignals "error"
}
Simulink.HardwareCC {
  $ObjectID       6
  Version         "1.3.0"
  ProdBitPerChar      8
  ProdBitPerShort     16
  ProdBitPerInt       32
  ProdBitPerLong      32
  ProdIntDivRoundTo   "Undefined"
  ProdEndianess       "Unspecified"
  ProdWordSize        32
  ProdShiftRightIntArith  on
  ProdHWDeviceType    "32-bit Generic"
```

```
  TargetBitPerChar    8
  TargetBitPerShort   16
  TargetBitPerInt     32
  TargetBitPerLong    32
  TargetShiftRightIntArith on
  TargetIntDivRoundTo     "Undefined"
  TargetEndianess     "Unspecified"
  TargetWordSize      32
  TargetTypeEmulationWarnSuppressLevel 0
  TargetPreprocMaxBitsSint 32
  TargetPreprocMaxBitsUint 32
  TargetHWDeviceType      "Specified"
  TargetUnknown       off
  ProdEqTarget        on
}
Simulink.ModelReferenceCC {
  $ObjectID       7
  Version         "1.3.0"
  UpdateModelReferenceTargets "IfOutOfDateOrStructuralChange"
  CheckModelReferenceTargetMessage "error"
  ModelReferenceNumInstancesAllowed "Multi"
  ModelReferencePassRootInputsByReference on
  ModelReferenceMinAlgLoopOccurrences off
}
Simulink.RTWCC {
  $BackupClass        "Simulink.RTWCC"
  $ObjectID       8
  Array {
    Type            "Cell"
    Dimension           2
    Cell            "IncludeHyperlinkInReport"
    Cell            "GenerateTraceInfo"
    PropName            "DisabledProps"
  }
  Version         "1.3.0"
  SystemTargetFile    "grt.tlc"
  GenCodeOnly         off
  MakeCommand         "make_rtw"
  GenerateMakefile    on
  TemplateMakefile    "grt_default_tmf"
  GenerateReport      off
  SaveLog         off
  RTWVerbose          on
  RetainRTWFile       off
  ProfileTLC          off
  TLCDebug        off
  TLCCoverage         off
  TLCAssert       off
  ProcessScriptMode   "Default"
  ConfigurationMode   "Optimized"
  ConfigAtBuild       off
  IncludeHyperlinkInReport off
  LaunchReport        off
  TargetLang          "C"
  IncludeBusHierarchyInRTWFileBlockHierarchyMap off
  IncludeERTFirstTime     off
  GenerateTraceInfo   off
```

```
RTWCompilerOptimization "Off"
Array {
  Type             "Handle"
  Dimension            2
  Simulink.CodeAppCC {
    $ObjectID            9
    Array {
  Type             "Cell"
  Dimension        16
  Cell             "IgnoreCustomStorageClasses"
  Cell             "InsertBlockDesc"
  Cell             "SFDataObjDesc"
  Cell             "SimulinkDataObjDesc"
  Cell             "DefineNamingRule"
  Cell             "SignalNamingRule"
  Cell             "ParamNamingRule"
  Cell             "InlinedPrmAccess"
  Cell             "CustomSymbolStr"
  Cell             "CustomSymbolStrGlobalVar"
  Cell             "CustomSymbolStrType"
  Cell             "CustomSymbolStrField"
  Cell             "CustomSymbolStrFcn"
  Cell             "CustomSymbolStrBlkIO"
  Cell             "CustomSymbolStrTmpVar"
  Cell             "CustomSymbolStrMacro"
  PropName         "DisabledProps"
    }
    Version              "1.3.0"
    ForceParamTrailComments off
    GenerateComments        on
    IgnoreCustomStorageClasses on
    IncHierarchyInIds       off
    MaxIdLength          31
    PreserveName         off
    PreserveNameWithParent  off
    ShowEliminatedStatement off
    IncAutoGenComments      off
    SimulinkDataObjDesc     off
    SFDataObjDesc        off
    IncDataTypeInIds        off
    MangleLength         1
    CustomSymbolStrGlobalVar "$R$N$M"
    CustomSymbolStrType     "$N$R$M"
    CustomSymbolStrField    "$N$M"
    CustomSymbolStrFcn      "$R$N$M$F"
    CustomSymbolStrBlkIO    "rtb_$N$M"
    CustomSymbolStrTmpVar   "$N$M"
    CustomSymbolStrMacro    "$R$N$M"
    DefineNamingRule        "None"
    ParamNamingRule         "None"
    SignalNamingRule        "None"
    InsertBlockDesc         off
    SimulinkBlockComments   on
    EnableCustomComments    off
    InlinedPrmAccess        "Literals"
    ReqsInCode           off
  }
```

```
Simulink.GRTTargetCC {
  $BackupClass        "Simulink.TargetCC"
  $ObjectID           10
  Array {
Type            "Cell"
Dimension       15
Cell            "IncludeMdlTerminateFcn"
Cell            "CombineOutputUpdateFcns"
Cell            "SuppressErrorStatus"
Cell            "ERTCustomFileBanners"
Cell            "GenerateSampleERTMain"
Cell            "GenerateTestInterfaces"
Cell            "ModelStepFunctionPrototypeControlCompliant"
Cell            "MultiInstanceERTCode"
Cell            "PurelyIntegerCode"
Cell            "SupportNonFinite"
Cell            "SupportComplex"
Cell            "SupportAbsoluteTime"
Cell            "SupportContinuousTime"
Cell            "SupportNonInlinedSFcns"
Cell            "PortableWordSizes"
PropName        "DisabledProps"
  }
  Version             "1.3.0"
  TargetFcnLib        "ansi_tfl_table_tmw.mat"
  TargetLibSuffix         ""
  TargetPreCompLibLocation ""
  GenFloatMathFcnCalls    "ANSI_C"
  UtilityFuncGeneration   "Auto"
  GenerateFullHeader      on
  GenerateSampleERTMain   off
  GenerateTestInterfaces  off
  IsPILTarget         off
  ModelReferenceCompliant on
  CompOptLevelCompliant   on
  IncludeMdlTerminateFcn  on
  CombineOutputUpdateFcns off
  SuppressErrorStatus     off
  IncludeFileDelimiter    "Auto"
  ERTCustomFileBanners    off
  SupportAbsoluteTime     on
  LogVarNameModifier      "rt_"
  MatFileLogging          on
  MultiInstanceERTCode    off
  SupportNonFinite        on
  SupportComplex          on
  PurelyIntegerCode       off
  SupportContinuousTime   on
  SupportNonInlinedSFcns  on
  EnableShiftOperators    on
  ParenthesesLevel        "Nominal"
  PortableWordSizes       off
  ModelStepFunctionPrototypeControlCompliant off
  ExtMode             off
  ExtModeStaticAlloc      off
  ExtModeTesting          off
  ExtModeStaticAllocSize  1000000
```

```
        ExtModeTransport        0
        ExtModeMexFile          "ext_comm"
        ExtModeIntrfLevel       "Level1"
        RTWCAPISignals          off
        RTWCAPIParams       off
        RTWCAPIStates       off
        GenerateASAP2       off
      }
      PropName            "Components"
    }
  }
  hdlcoderui.hdlcc {
    $ObjectID       11
    Description         "HDL Coder custom configuration component"
    Version         "1.3.0"
    Name            "HDL Coder"
    Array {
      Type            "Cell"
      Dimension           1
      Cell            ""
      PropName            "HDLConfigFile"
    }
    HDLCActiveTab       "0"
  }
  PropName        "Components"
    }
    Name            "Configuration"
    CurrentDlgPage          "Solver"
  }
  PropName            "ConfigurationSets"
}
Simulink.ConfigSet {
  $PropName           "ActiveConfigurationSet"
  $ObjectID           1
}
BlockDefaults {
  Orientation         "right"
  ForegroundColor     "black"
  BackgroundColor     "white"
  DropShadow          off
  NamePlacement       "normal"
  FontName            "Arial"
  FontSize            10
  FontWeight          "normal"
  FontAngle           "normal"
  ShowName            on
}
BlockParameterDefaults {
  Block {
    BlockType           ActionPort
    InitializeStates        "held"
    ActionType          "unset"
  }
  Block {
    BlockType           DataTypeConversion
    OutMin              "[]"
    OutMax              "[]"
```

```
  OutDataTypeMode         "Inherit via back propagation"
  OutDataType         "fixdt(1,16,0)"
  OutScaling          "[]"
  LockScale           off
  ConvertRealWorld        "Real World Value (RWV)"
  RndMeth             "Zero"
  SaturateOnIntegerOverflow on
  SampleTime          "-1"
}
Block {
  BlockType           Demux
  Outputs             "4"
  DisplayOption       "none"
  BusSelectionMode        off
}
Block {
  BlockType           Derivative
  LinearizePole       "inf"
}
Block {
  BlockType           Gain
  Gain            "1"
  Multiplication          "Element-wise(K.*u)"
  ParamMin            "[]"
  ParamMax            "[]"
  ParameterDataTypeMode   "Same as input"
  ParameterDataType       "fixdt(1,16,0)"
  ParameterScalingMode    "Best Precision: Matrix-wise"
  ParameterScaling        "[]"
  OutMin              "[]"
  OutMax              "[]"
  OutDataTypeMode         "Same as input"
  OutDataType         "fixdt(1,16,0)"
  OutScaling          "[]"
  LockScale           off
  RndMeth             "Floor"
  SaturateOnIntegerOverflow on
  SampleTime          "-1"
}
Block {
  BlockType           If
  NumInputs           "1"
  IfExpression        "u1 > 0"
  ShowElse            on
  ZeroCross           on
  SampleTime          "-1"
}
Block {
  BlockType           Inport
  Port            "1"
  UseBusObject        off
  BusObject           "BusObject"
  BusOutputAsStruct       off
  PortDimensions          "-1"
  SampleTime          "-1"
  OutMin              "[]"
  OutMax              "[]"
```

```
    DataType             "auto"
    OutDataType          "fixdt(1,16,0)"
    OutScaling           "[]"
    SignalType           "auto"
    SamplingMode         "auto"
    LatchByDelayingOutsideSignal off
    LatchByCopyingInsideSignal off
    Interpolate          on
  }
  Block {
    BlockType            Integrator
    ExternalReset        "none"
    InitialConditionSource  "internal"
    InitialCondition        "0"
    LimitOutput          off
    UpperSaturationLimit    "inf"
    LowerSaturationLimit    "-inf"
    ShowSaturationPort      off
    ShowStatePort        off
    AbsoluteTolerance       "auto"
    IgnoreLimit          off
    ZeroCross            on
    ContinuousStateAttributes "''"
  }
  Block {
    BlockType            Math
    Operator             "exp"
    OutputSignalType        "auto"
    SampleTime           "-1"
    OutMin               "[]"
    OutMax               "[]"
    OutDataTypeMode         "Same as first input"
    OutDataType          "fixdt(1,16,0)"
    OutScaling           "[]"
    LockScale            off
    RndMeth              "Floor"
    SaturateOnIntegerOverflow on
  }
  Block {
    BlockType            Mux
    Inputs               "4"
    DisplayOption        "none"
    UseBusObject         off
    BusObject            "BusObject"
    NonVirtualBus        off
  }
  Block {
    BlockType            Outport
    Port             "1"
    UseBusObject         off
    BusObject            "BusObject"
    BusOutputAsStruct       off
    PortDimensions          "-1"
    SampleTime           "-1"
    OutMin               "[]"
    OutMax               "[]"
    DataType             "auto"
```

```
    OutDataType          "fixdt(1,16,0)"
    OutScaling           "[]"
    SignalType           "auto"
    SamplingMode         "auto"
    OutputWhenDisabled       "held"
    InitialOutput        "[]"
  }
  Block {
    BlockType            Product
    Inputs               "2"
    Multiplication           "Element-wise(.*)"
    CollapseMode         "All dimensions"
    CollapseDim          "1"
    InputSameDT          on
    OutMin               "[]"
    OutMax               "[]"
    OutDataTypeMode          "Same as first input"
    OutDataType          "fixdt(1,16,0)"
    OutScaling           "[]"
    LockScale            off
    RndMeth              "Zero"
    SaturateOnIntegerOverflow on
    SampleTime           "-1"
  }
  Block {
    BlockType            Scope
    ModelBased           off
    TickLabels           "OneTimeTick"
    ZoomMode             "on"
    Grid             "on"
    TimeRange            "auto"
    YMin             "-5"
    YMax             "5"
    SaveToWorkspace          off
    SaveName             "ScopeData"
    LimitDataPoints          on
    MaxDataPoints        "5000"
    Decimation           "1"
    SampleInput          off
    SampleTime           "-1"
  }
  Block {
    BlockType            "S-Function"
    FunctionName         "system"
    SFunctionModules         "''"
    PortCounts           "[]"
  }
  Block {
    BlockType            Step
    Time             "1"
    Before               "0"
    After            "1"
    SampleTime           "-1"
    VectorParams1D           on
    ZeroCross            on
  }
  Block {
```

```
    BlockType             SubSystem
    ShowPortLabels            "FromPortIcon"
    Permissions           "ReadWrite"
    PermitHierarchicalResolution "All"
    TreatAsAtomicUnit         off
    SystemSampleTime          "-1"
    RTWFcnNameOpts            "Auto"
    RTWFileNameOpts           "Auto"
    RTWMemSecFuncInitTerm     "Inherit from model"
    RTWMemSecFuncExecute      "Inherit from model"
    RTWMemSecDataConstants    "Inherit from model"
    RTWMemSecDataInternal     "Inherit from model"
    RTWMemSecDataParameters "Inherit from model"
    SimViewingDevice          off
    DataTypeOverride          "UseLocalSettings"
    MinMaxOverflowLogging     "UseLocalSettings"
  }
  Block {
    BlockType             Sum
    IconShape             "rectangular"
    Inputs                "++"
    CollapseMode          "All dimensions"
    CollapseDim           "1"
    InputSameDT           on
    OutMin                "[]"
    OutMax                "[]"
    OutDataTypeMode           "Same as first input"
    OutDataType           "fixdt(1,16,0)"
    OutScaling            "[]"
    LockScale             off
    RndMeth               "Floor"
    SaturateOnIntegerOverflow on
    SampleTime            "-1"
  }
  Block {
    BlockType             Switch
    Criteria              "u2 >= Threshold"
    Threshold             "0"
    InputSameDT           on
    OutMin                "[]"
    OutMax                "[]"
    OutDataTypeMode           "Inherit via internal rule"
    OutDataType           "fixdt(1,16,0)"
    OutScaling            "[]"
    LockScale             off
    RndMeth               "Floor"
    SaturateOnIntegerOverflow on
    ZeroCross             on
    SampleTime            "-1"
  }
  Block {
    BlockType             Terminator
  }
  Block {
    BlockType             TransferFcn
    Numerator             "[1]"
    Denominator           "[1 2 1]"
```

```
      AbsoluteTolerance       "auto"
      ContinuousStateAttributes "''"
      Realization         "auto"
    }
    Block {
      BlockType           ZeroOrderHold
      SampleTime          "1"
    }
    Block {
      BlockType           Merge
      Inputs              "2"
      InitialOutput       "[]"
      AllowUnequalInputPortWidths off
      InputPortOffsets        "[]"
    }
    Block {
      BlockType           Constant
      Value           "1"
      VectorParams1D          on
      SamplingMode        "Sample based"
      OutMin              "[]"
      OutMax              "[]"
      OutDataTypeMode         "Inherit from 'Constant value'"
      OutDataType         "fixdt(1,16,0)"
      ConRadixGroup       "Use specified scaling"
      OutScaling          "[]"
      SampleTime          "inf"
      FramePeriod         "inf"
    }
    Block {
      BlockType           MinMax
      Function            "min"
      Inputs              "1"
      InputSameDT         on
      OutMin              "[]"
      OutMax              "[]"
      OutDataTypeMode         "Inherit via internal rule"
      OutDataType         "fixdt(1,16,0)"
      OutScaling          "[]"
      LockScale           off
      RndMeth             "Floor"
      SaturateOnIntegerOverflow on
      ZeroCross           on
      SampleTime          "-1"
    }
    Block {
      BlockType           RelationalOperator
      Operator            ">="
      InputSameDT         on
      LogicOutDataTypeMode    "Logical (see Configuration Parameters:
Optimization)"
      LogicDataType       "uint(8)"
      ZeroCross           on
      SampleTime          "-1"
    }
  }
  AnnotationDefaults {
```

```
    HorizontalAlignment     "center"
    VerticalAlignment       "middle"
    ForegroundColor     "black"
    BackgroundColor     "white"
    DropShadow          off
    FontName            "Arial"
    FontSize            10
    FontWeight          "normal"
    FontAngle           "normal"
    UseDisplayTextAsClickCallback off
  }
  LineDefaults {
    FontName            "Arial"
    FontSize            9
    FontWeight          "normal"
    FontAngle           "normal"
  }
  System {
    Name            "Closed_loop_Setup4_BA_wrkingaftercliff_rev5"
    Location            [2, 82, 1670, 1004]
    Open            on
    ModelBrowserVisibility  off
    ModelBrowserWidth       200
    ScreenColor         "white"
    PaperOrientation        "landscape"
    PaperPositionMode       "auto"
    PaperType           "usletter"
    PaperUnits          "inches"
    TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
    TiledPageScale      1
    ShowPageBoundaries      off
    ZoomFactor          "100"
    ReportName          "simulink-default.rpt"
    Block {
      BlockType           TransferFcn
      Name            "Breathing Air"
      Position            [815, 517, 875, 553]
      Denominator         "[1 0]"
    }
    Block {
      BlockType           Reference
      Name            "Compare\nTo Constant"
      Ports           [1, 1]
      Position            [970, 595, 1000, 625]
      SourceBlock         "simulink/Logic and Bit\nOperations/Compare\nTo
Constant"
      SourceType          "Compare To Constant"
      ShowPortLabels          "FromPortIcon"
      SystemSampleTime        "-1"
      FunctionWithSeparateData off
      RTWMemSecFuncInitTerm   "Inherit from model"
      RTWMemSecFuncExecute    "Inherit from model"
      RTWMemSecDataConstants  "Inherit from model"
      RTWMemSecDataInternal   "Inherit from model"
      RTWMemSecDataParameters "Inherit from model"
      relop           "<"
      const           "550"
```

```
      LogicOutDataTypeMode    "boolean"
      ZeroCross           off
    }
    Block {
      BlockType           SubSystem
      Name             "Fuzzy Logic Controller"
      Ports            [1, 3]
      Position             [175, 304, 275, 346]
      TreatAsAtomicUnit       on
      MinAlgLoopOccurrences   off
      RTWSystemCode       "Auto"
      FunctionWithSeparateData off
      System {
    Name             "Fuzzy Logic Controller"
    Location         [209, 573, 715, 878]
    Open             off
    ModelBrowserVisibility  off
    ModelBrowserWidth   200
    ScreenColor      "white"
    PaperOrientation    "landscape"
    PaperPositionMode   "auto"
    PaperType        "usletter"
    PaperUnits       "inches"
    TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
    TiledPageScale      1
    ShowPageBoundaries  off
    ZoomFactor       "100"
    Block {
      BlockType        Inport
      Name             "In1"
      Position         [110, 118, 140, 132]
      IconDisplay          "Port number"
      OutDataType          "sfix(16)"
      OutScaling           "2^0"
    }
    Block {
      BlockType        Demux
      Name             "Demux"
      Ports            [1, 3]
      Position         [305, 106, 310, 144]
      BackgroundColor      "black"
      ShowName         off
      Outputs          "3"
      DisplayOption        "bar"
    }
    Block {
      BlockType        Reference
      Name             "Fuzzy Logic \nController \nwith Ruleviewer"
      Ports            [1, 1]
      Position         [210, 100, 270, 150]
      SourceBlock          "fuzblock/Fuzzy Logic \nController \nwith
Ruleviewer"
      SourceType           "FIS"
      ShowPortLabels       "FromPortIcon"
      SystemSampleTime     "-1"
      FunctionWithSeparateData off
      RTWMemSecFuncInitTerm   "Inherit from model"
```

```
    RTWMemSecFuncExecute    "Inherit from model"
    RTWMemSecDataConstants  "Inherit from model"
    RTWMemSecDataInternal   "Inherit from model"
    RTWMemSecDataParameters "Inherit from model"
    fismatrix       "Fuzzytwo_corrected"
    Ts              "2"
  }
  Block {
    BlockType       Scope
    Name            "Scope1"
    Ports           [1]
    Position        [315, 24, 345, 56]
    Floating        off
    Location        [1, 52, 1681, 1019]
    Open            off
    NumInputPorts       "1"
    List {
      ListType            AxesTitles
      axes1           "%<SignalLabel>"
    }
    SaveName        "ScopeData4"
    DataFormat          "StructureWithTime"
    SampleTime          "0"
  }
  Block {
    BlockType       Scope
    Name            "Scope2"
    Ports           [1]
    Position        [425, 74, 455, 106]
    Floating        off
    Location        [1, 52, 1681, 1019]
    Open            off
    NumInputPorts       "1"
    List {
      ListType            AxesTitles
      axes1           "%<SignalLabel>"
    }
    SaveName        "ScopeData5"
    DataFormat          "StructureWithTime"
    SampleTime          "0"
  }
  Block {
    BlockType       Scope
    Name            "Scope3"
    Ports           [1]
    Position        [415, 139, 445, 171]
    Floating        off
    Location        [1, 52, 1681, 1019]
    Open            off
    NumInputPorts       "1"
    List {
      ListType            AxesTitles
      axes1           "%<SignalLabel>"
    }
    SaveName        "ScopeData6"
    DataFormat          "StructureWithTime"
    SampleTime          "0"
```

```
}
Block {
  BlockType       Scope
  Name            "Scope4"
  Ports           [1]
  Position        [360, 234, 390, 266]
  Floating        off
  Location        [1, 52, 1681, 1019]
  Open            off
  NumInputPorts       "1"
  List {
    ListType            AxesTitles
    axes1           "%<SignalLabel>"
  }
  SaveName        "ScopeData7"
  DataFormat          "StructureWithTime"
  SampleTime          "0"
}
Block {
  BlockType       Outport
  Name            "kp_cal"
  Position        [365, 68, 395, 82]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Outport
  Name            "ki_cal"
  Position        [365, 123, 395, 137]
  Port            "2"
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Outport
  Name            "kd_cal"
  Position        [365, 193, 395, 207]
  Port            "3"
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Line {
  SrcBlock        "In1"
  SrcPort         1
  DstBlock        "Fuzzy Logic \nController \nwith Ruleviewer"
  DstPort         1
}
Line {
  SrcBlock        "Demux"
  SrcPort         1
  Points          [15, 0; 0, -40; 15, 0]
  Branch {
    DstBlock            "kp_cal"
    DstPort         1
```

```
    }
    Branch {
      Points          [0, 15]
      DstBlock            "Scope2"
      DstPort         1
    }
  }
  Line {
    SrcBlock        "Demux"
    SrcPort         2
    Points          [0, 5; 30, 0]
    Branch {
      DstBlock            "ki_cal"
      DstPort         1
    }
    Branch {
      Points          [0, 25]
      DstBlock            "Scope3"
      DstPort         1
    }
  }
  Line {
    SrcBlock        "Demux"
    SrcPort         3
    Points          [15, 0; 0, 60]
    Branch {
      Points          [0, 5]
      DstBlock            "kd_cal"
      DstPort         1
    }
    Branch {
      Points          [0, 55]
      DstBlock            "Scope4"
      DstPort         1
    }
  }
  Line {
    SrcBlock        "Fuzzy Logic \nController \nwith Ruleviewer"
    SrcPort         1
    Points          [10, 0]
    Branch {
      DstBlock            "Demux"
      DstPort         1
    }
    Branch {
      Points          [0, -85]
      DstBlock            "Scope1"
      DstPort         1
    }
  }
    }
  }
  Block {
    BlockType           Integrator
    Name            "Integrator"
    Ports           [1, 1]
    Position            [1160, 630, 1190, 660]
```

```
}
Block {
  BlockType           Mux
  Name            "Mux"
  Ports           [2, 1]
  Position            [925, 506, 930, 544]
  ShowName            off
  Inputs              "2"
  DisplayOption       "bar"
}
Block {
  BlockType           SubSystem
  Name            "PID"
  Ports           [4, 1]
  Position            [400, 472, 505, 543]
  TreatAsAtomicUnit       on
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  System {
Name            "PID"
Location        [650, 136, 1517, 468]
Open            off
ModelBrowserVisibility  off
ModelBrowserWidth   200
ScreenColor     "white"
PaperOrientation    "landscape"
PaperPositionMode   "auto"
PaperType       "usletter"
PaperUnits      "inches"
TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
TiledPageScale      1
ShowPageBoundaries  off
ZoomFactor      "100"
Block {
  BlockType       Inport
  Name            "kp"
  Position        [65, 53, 95, 67]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Inport
  Name            "ki"
  Position        [65, 123, 95, 137]
  Port            "2"
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Inport
  Name            "kd"
  Position        [65, 193, 95, 207]
  Port            "3"
  IconDisplay         "Port number"
```

```
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType       Inport
    Name            "Error input"
    Position        [65, 248, 95, 262]
    Port            "4"
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType       Derivative
    Name            "Derivative"
    Position        [335, 185, 365, 215]
  }
  Block {
    BlockType       Integrator
    Name            "Integrator"
    Ports           [1, 1]
    Position        [335, 130, 365, 160]
  }
  Block {
    BlockType       Product
    Name            "Product"
    Ports           [2, 1]
    Position        [405, 192, 435, 223]
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       Product
    Name            "Product1"
    Ports           [2, 1]
    Position        [390, 47, 420, 78]
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       Product
    Name            "Product2"
    Ports           [2, 1]
    Position        [400, 127, 430, 158]
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
```

```
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Sum
  Name            "Sum"
  Ports           [3, 1]
  Position        [460, 130, 490, 160]
  ShowName        off
  IconShape       "round"
  Inputs          "+|+|+"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Outport
  Name            "Out1"
  Position        [805, 58, 835, 72]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Line {
  SrcBlock        "Sum"
  SrcPort         1
  Points          [145, 0; 0, -80]
  DstBlock        "Out1"
  DstPort         1
}
Line {
  SrcBlock        "Error input"
  SrcPort         1
  Points          [110, 0; 0, -55]
  Branch {
    DstBlock            "Derivative"
    DstPort         1
  }
  Branch {
    Points          [0, -55]
    Branch {
      DstBlock            "Integrator"
      DstPort             1
    }
    Branch {
      Points              [0, -75]
      DstBlock            "Product1"
      DstPort             2
    }
  }
}
Line {
  SrcBlock        "Derivative"
  SrcPort         1
  DstBlock        "Product"
```

```
    DstPort         1
  }
  Line {
    SrcBlock        "kd"
    SrcPort         1
    Points          [0, 30; 170, 0; 0, 40; 115, 0; 0, -55]
    DstBlock        "Product"
    DstPort         2
  }
  Line {
    SrcBlock        "Product"
    SrcPort         1
    Points          [35, 0]
    DstBlock        "Sum"
    DstPort         3
  }
  Line {
    SrcBlock        "kp"
    SrcPort         1
    Points          [135, 0; 0, -5]
    DstBlock        "Product1"
    DstPort         1
  }
  Line {
    SrcBlock        "Product1"
    SrcPort         1
    Points          [50, 0]
    DstBlock        "Sum"
    DstPort         1
  }
  Line {
    SrcBlock        "Product2"
    SrcPort         1
    DstBlock        "Sum"
    DstPort         2
  }
  Line {
    SrcBlock        "Integrator"
    SrcPort         1
    Points          [5, 0; 0, 5]
    DstBlock        "Product2"
    DstPort         2
  }
  Line {
    SrcBlock        "ki"
    SrcPort         1
    Points          [230, 0; 0, -5; 55, 0]
    DstBlock        "Product2"
    DstPort         1
  }
    }
  }
  Block {
    BlockType           Product
    Name            "Product"
    Ports           [2, 1]
    Position            [1065, 627, 1095, 658]
```

```
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Scope
    Name            "Scope"
    Ports           [1]
    Position            [1035, 509, 1065, 541]
    Floating            off
    Location            [277, 258, 1285, 917]
    Open            off
    NumInputPorts           "1"
    ZoomMode            "xonly"
    List {
  ListType        AxesTitles
  axes1           "%<SignalLabel>"
    }
    TimeRange           "150             "
    YMin            "620"
    YMax            "620.053"
    DataFormat          "StructureWithTime"
    LimitDataPoints         off
    SampleTime          "0"
  }
  Block {
    BlockType           Scope
    Name            "Scope1"
    Ports           [1]
    Position            [585, 579, 615, 611]
    Floating            off
    Location            [5, 52, 1685, 1019]
    Open            off
    NumInputPorts           "1"
    List {
  ListType        AxesTitles
  axes1           "%<SignalLabel>"
    }
    YMin            "696.951"
    YMax            "696.951"
    SaveName            "ScopeData1"
    DataFormat          "StructureWithTime"
    SampleTime          "0"
  }
  Block {
    BlockType           Scope
    Name            "Scope2"
    Ports           [1]
    Position            [730, 444, 760, 476]
    Floating            off
    Location            [1, 52, 1681, 1019]
    Open            off
    NumInputPorts           "1"
    List {
```

```
ListType        AxesTitles
axes1           "%<SignalLabel>"
  }
  SaveName            "ScopeData2"
  DataFormat          "StructureWithTime"
  SampleTime          "0"
}
Block {
  BlockType           Scope
  Name            "Scope3"
  Ports           [1]
  Position            [645, 449, 675, 481]
  Floating            off
  Location            [5, 52, 1685, 1019]
  Open            off
  NumInputPorts       "1"
  List {
ListType        AxesTitles
axes1           "%<SignalLabel>"
  }
  SaveName            "ScopeData3"
  DataFormat          "StructureWithTime"
  SampleTime          "0"
}
Block {
  BlockType           Scope
  Name            "Scope4"
  Ports           [1]
  Position            [945, 689, 975, 721]
  Floating            off
  Location            [5, 52, 1685, 1019]
  Open            off
  NumInputPorts       "1"
  ZoomMode            "xonly"
  List {
ListType        AxesTitles
axes1           "%<SignalLabel>"
  }
  TimeRange           "400"
  YMin            "696.943"
  YMax            "696.951"
  SaveName            "ScopeData8"
  DataFormat          "StructureWithTime"
  LimitDataPoints         off
  SampleTime          "0"
}
Block {
  BlockType           Scope
  Name            "Scope5"
  Ports           [1]
  Position            [835, 359, 865, 391]
  Floating            off
  Location            [5, 52, 1685, 1019]
  Open            off
  NumInputPorts       "1"
  ZoomMode            "xonly"
  List {
```

```
ListType          AxesTitles
axes1             "%<SignalLabel>"
  }
  TimeRange            "400"
  YMin             "667.418"
  YMax             "667.442"
  SaveName             "ScopeData9"
  DataFormat           "StructureWithTime"
  LimitDataPoints          off
  SampleTime           "0"
}
Block {
  BlockType            Scope
  Name             "Scope6"
  Ports            [1]
  Position             [1260, 629, 1290, 661]
  Floating             off
  Location             [305, 320, 1139, 888]
  Open             off
  NumInputPorts        "1"
  ZoomMode             "xonly"
  List {
ListType          AxesTitles
axes1             "%<SignalLabel>"
  }
  TimeRange            "400"
  YMin             "140000"
  YMax             "197500"
  SaveName             "ScopeData10"
  DataFormat           "StructureWithTime"
  LimitDataPoints          off
  SampleTime           "0"
}
Block {
  BlockType            Scope
  Name             "Scope7"
  Ports            [1]
  Position             [450, 374, 480, 406]
  Floating             off
  Location             [5, 52, 1685, 1019]
  Open             off
  NumInputPorts        "1"
  ZoomMode             "xonly"
  List {
ListType          AxesTitles
axes1             "%<SignalLabel>"
  }
  TimeRange            "400"
  YMin             "667.418"
  YMax             "667.442"
  SaveName             "ScopeData11"
  DataFormat           "StructureWithTime"
  LimitDataPoints          off
  SampleTime           "0"
}
Block {
  BlockType            Scope
```

```
    Name            "Scope8"
    Ports           [1]
    Position            [1075, 559, 1105, 591]
    Floating            off
    Location            [5, 52, 1685, 1019]
    Open            off
    NumInputPorts       "1"
    ZoomMode            "xonly"
    List {
  ListType        AxesTitles
  axes1           "%<SignalLabel>"
    }
    TimeRange           "400"
    YMin            "140000"
    YMax            "197500"
    SaveName            "ScopeData12"
    DataFormat          "StructureWithTime"
    LimitDataPoints         off
    SampleTime          "0"
  }
  Block {
    BlockType           TransferFcn
    Name            "Sensor"
    Position            [705, 712, 765, 748]
    Orientation         "left"
    Denominator         "[0 1]"
  }
  Block {
    BlockType           Step
    Name            "Step"
    Position            [245, 520, 275, 550]
    Time            "0"
    After           "620"
    SampleTime          "0.1"
  }
  Block {
    BlockType           Sum
    Name            "Sum"
    Ports           [2, 1]
    Position            [335, 525, 355, 545]
    ShowName            off
    IconShape           "round"
    Inputs              "|+-"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           TransferFcn
    Name            "Valve"
    Position            [645, 517, 705, 553]
    Denominator         "[0.0013 1]"
  }
  Line {
```

```
  SrcBlock            "Step"
  SrcPort             1
  Points              [10, 0]
  Branch {
DstBlock        "Sum"
DstPort         1
  }
  Branch {
Points          [0, -100; 575, 0; 0, 80]
DstBlock        "Mux"
DstPort         1
  }
}
Line {
  SrcBlock            "Valve"
  SrcPort             1
  Points              [5, 0]
  Branch {
DstBlock        "Scope2"
DstPort         1
  }
  Branch {
Points          [80, 0]
Branch {
  DstBlock        "Breathing Air"
  DstPort         1
}
Branch {
  Points          [0, -160]
  DstBlock        "Scope5"
  DstPort         1
}
  }
}
Line {
  SrcBlock            "Mux"
  SrcPort             1
  DstBlock            "Scope"
  DstPort             1
}
Line {
  SrcBlock            "Sum"
  SrcPort             1
  Points              [0, 0]
  Branch {
Points          [10, 0; 0, -5]
DstBlock        "PID"
DstPort         4
  }
  Branch {
Points          [0, -115; -35, 0]
Branch {
  Points          [-165, 0]
  DstBlock        "Fuzzy Logic Controller"
  DstPort         1
}
Branch {
```

```
  Points          [0, -30]
  DstBlock        "Scope7"
  DstPort         1
}
  }
}
Line {
  SrcBlock            "PID"
  SrcPort             1
  Points              [0, 10; 90, 0]
  Branch {
Points          [30, 0]
DstBlock        "Valve"
DstPort         1
  }
  Branch {
Points          [0, -55]
DstBlock        "Scope3"
DstPort         1
  }
}
Line {
  SrcBlock            "Breathing Air"
  SrcPort             1
  Points              [25, 0]
  Branch {
DstBlock        "Mux"
DstPort         2
  }
  Branch {
Points          [0, 75]
Branch {
  DstBlock        "Compare\nTo Constant"
  DstPort         1
}
Branch {
  Points          [0, 40]
  Branch {
    Points          [0, 50]
    Branch {
      Points              [0, 30]
      DstBlock            "Sensor"
      DstPort             1
    }
    Branch {
      Points              [0, 5]
      DstBlock            "Scope4"
      DstPort             1
    }
  }
  Branch {
    DstBlock            "Product"
    DstPort         2
  }
}
  }
}
```

```
Line {
  SrcBlock              "Fuzzy Logic Controller"
  SrcPort               1
  Points                [0, 15; 105, 0]
  DstBlock              "PID"
  DstPort               1
}
Line {
  Labels                [0, 0]
  SrcBlock              "Fuzzy Logic Controller"
  SrcPort               2
  Points                [0, 65; 100, 0; 0, 110]
  DstBlock              "PID"
  DstPort               2
}
Line {
  Labels                [0, 0]
  SrcBlock              "Fuzzy Logic Controller"
  SrcPort               3
  Points                [0, -35; 100, 0; 0, 210]
  DstBlock              "PID"
  DstPort               3
}
Line {
  SrcBlock              "Sensor"
  SrcPort               1
  Points                [-150, 0; 0, -135]
  Branch {
Points          [0, -35]
DstBlock        "Sum"
DstPort         2
  }
  Branch {
DstBlock        "Scope1"
DstPort         1
  }
}
Line {
  SrcBlock              "Compare\nTo Constant"
  SrcPort               1
  Points                [10, 0]
  Branch {
Points          [10, 0; 0, 25]
DstBlock        "Product"
DstPort         1
  }
  Branch {
Points          [0, -35]
DstBlock        "Scope8"
DstPort         1
  }
}
Line {
  SrcBlock              "Product"
  SrcPort               1
  DstBlock              "Integrator"
  DstPort               1
```

```
    }
    Line {
      SrcBlock		      "Integrator"
      SrcPort		      1
      DstBlock		      "Scope6"
      DstPort		      1
    }
  }
}
```

8.2 – Algorithm for Fuzzy Logic System to Detect Solenoid Valve Failures within Safety Related System

```
Model {
  Name            "Experiment_6_matlab5"
  Version         7.0
  MdlSubVersion       0
  GraphicalInterface {
    NumRootInports      0
    NumRootOutports     0
    ParameterArgumentNames  ""
    ComputedModelVersion    "1.137"
    NumModelReferences      0
    NumTestPointedSignals   0
  }
  SavedCharacterEncoding  "windows-1252"
  SaveDefaultBlockParams  on
  SampleTimeColors    off
  LibraryLinkDisplay      "none"
  WideLines       off
  ShowLineDimensions      off
  ShowPortDataTypes   off
  ShowLoopsOnError    on
  IgnoreBidirectionalLines off
  ShowStorageClass    off
  ShowTestPointIcons      on
  ShowViewerIcons     on
  SortedOrder         off
  ExecutionContextIcon    off
  ShowLinearizationAnnotations on
  ScopeRefreshTime    0.035000
  OverrideScopeRefreshTime on
  DisableAllScopes    off
  DataTypeOverride    "UseLocalSettings"
  MinMaxOverflowLogging   "UseLocalSettings"
  MinMaxOverflowArchiveMode "Overwrite"
  BlockNameDataTip    off
  BlockParametersDataTip  off
  BlockDescriptionStringDataTip off
  ToolBar         on
  StatusBar       on
  BrowserShowLibraryLinks off
  BrowserLookUnderMasks   off
  Created         "Tue Apr 03 12:34:39 2012"
  Creator         "Owner"
  UpdateHistory       "UpdateHistoryNever"
  ModifiedByFormat    "%<Auto>"
  LastModifiedBy      "Owner"
  ModifiedDateFormat      "%<Auto>"
  LastModifiedDate    "Wed Nov 12 23:33:57 2014"
  RTWModifiedTimeStamp    0
  ModelVersionFormat      "1.%<AutoIncrement:137>"
  ConfigurationManager    "None"
  SimulationMode      "normal"
  LinearizationMsg    "none"
```

```
Profile         off
ParamWorkspaceSource    "MATLABWorkspace"
AccelSystemTargetFile   "accel.tlc"
AccelTemplateMakefile   "accel_default_tmf"
AccelMakeCommand    "make_rtw"
TryForcingSFcnDF    off
RecordCoverage      off
CovPath         "/"
CovSaveName         "covdata"
CovMetricSettings   "dw"
CovNameIncrementing     off
CovHtmlReporting    on
covSaveCumulativeToWorkspaceVar on
CovSaveSingleToWorkspaceVar on
CovCumulativeVarName    "covCumulativeData"
CovCumulativeReport     off
CovReportOnPause    on
ExtModeBatchMode    off
ExtModeEnableFloating   on
ExtModeTrigType     "manual"
ExtModeTrigMode     "normal"
ExtModeTrigPort     "1"
ExtModeTrigElement      "any"
ExtModeTrigDuration     1000
ExtModeTrigDurationFloating "auto"
ExtModeTrigHoldOff      0
ExtModeTrigDelay    0
ExtModeTrigDirection    "rising"
ExtModeTrigLevel    0
ExtModeArchiveMode      "off"
ExtModeAutoIncOneShot   off
ExtModeIncDirWhenArm    off
ExtModeAddSuffixToVar   off
ExtModeWriteAllDataToWs off
ExtModeArmWhenConnect   on
ExtModeSkipDownloadWhenConnect off
ExtModeLogAll       on
ExtModeAutoUpdateStatusClock on
BufferReuse         on
ShowModelReferenceBlockVersion off
ShowModelReferenceBlockIO off
Array {
  Type            "Handle"
  Dimension           1
  Simulink.ConfigSet {
    $ObjectID           1
    Version             "1.3.0"
    Array {
  Type            "Handle"
  Dimension       8
  Simulink.SolverCC {
    $ObjectID       2
    Version         "1.3.0"
    StartTime       "0.0"
    StopTime        "500"
    AbsTol          "auto"
    FixedStep       "auto"
```

```
  InitialStep         "auto"
  MaxNumMinSteps      "-1"
  MaxOrder        5
  ConsecutiveZCsStepRelTol "10*128*eps"
  MaxConsecutiveZCs   "1000"
  ExtrapolationOrder      4
  NumberNewtonIterations  1
  MaxStep         "auto"
  MinStep         "auto"
  MaxConsecutiveMinStep   "1"
  RelTol          "1e-3"
  SolverMode          "Auto"
  Solver          "ode45"
  SolverName          "ode45"
  ZeroCrossControl    "UseLocalSettings"
  AlgebraicLoopSolver     "TrustRegion"
  SolverResetMethod   "Fast"
  PositivePriorityOrder   off
  AutoInsertRateTranBlk   off
  SampleTimeConstraint    "Unconstrained"
  RateTranMode        "Deterministic"
}
Simulink.DataIOCC {
  $ObjectID       3
  Version         "1.3.0"
  Decimation          "1"
  ExternalInput       "[t, u]"
  FinalStateName      "xFinal"
  InitialState        "xInitial"
  LimitDataPoints     on
  MaxDataPoints       "1000"
  LoadExternalInput   off
  LoadInitialState    off
  SaveFinalState      off
  SaveFormat          "Array"
  SaveOutput          on
  SaveState       off
  SignalLogging       on
  InspectSignalLogs   off
  SaveTime        on
  StateSaveName       "xout"
  TimeSaveName        "tout"
  OutputSaveName      "yout"
  SignalLoggingName   "logsout"
  OutputOption        "RefineOutputTimes"
  OutputTimes         "[]"
  Refine          "1"
}
Simulink.OptimizationCC {
  $ObjectID       4
  Array {
    Type            "Cell"
    Dimension           5
    Cell            "ZeroExternalMemoryAtStartup"
    Cell            "ZeroInternalMemoryAtStartup"
    Cell            "InitFltsAndDblsToZero"
    Cell            "OptimizeModelRefInitCode"
```

```
    Cell            "NoFixptDivByZeroProtection"
    PropName            "DisabledProps"
  }
  Version         "1.3.0"
  BlockReduction      on
  BooleanDataType     on
  ConditionallyExecuteInputs on
  InlineParams        off
  InlineInvariantSignals  off
  OptimizeBlockIOStorage  on
  BufferReuse         on
  EnforceIntegerDowncast  on
  ExpressionFolding   on
  ExpressionDepthLimit    2147483647
  FoldNonRolledExpr   on
  LocalBlockOutputs   on
  RollThreshold       5
  SystemCodeInlineAuto    off
  StateBitsets        off
  DataBitsets         off
  UseTempVars         off
  ZeroExternalMemoryAtStartup on
  ZeroInternalMemoryAtStartup on
  InitFltsAndDblsToZero   on
  NoFixptDivByZeroProtection off
  EfficientFloat2IntCast  off
  OptimizeModelRefInitCode off
  LifeSpan        "inf"
  BufferReusableBoundary  on
  SimCompilerOptimization "Off"
  AccelVerboseBuild   off
}
Simulink.DebuggingCC {
  $ObjectID       5
  Version         "1.3.0"
  RTPrefix        "error"
  ConsistencyChecking     "none"
  ArrayBoundsChecking     "none"
  SignalInfNanChecking    "none"
  SignalRangeChecking     "none"
  ReadBeforeWriteMsg      "UseLocalSettings"
  WriteAfterWriteMsg      "UseLocalSettings"
  WriteAfterReadMsg   "UseLocalSettings"
  AlgebraicLoopMsg    "warning"
  ArtificialAlgebraicLoopMsg "warning"
  SaveWithDisabledLinksMsg "warning"
  SaveWithParameterizedLinksMsg "warning"
  CheckSSInitialOutputMsg on
  CheckExecutionContextPreStartOutputMsg off
  CheckExecutionContextRuntimeOutputMsg off
  SignalResolutionControl "UseLocalSettings"
  BlockPriorityViolationMsg "warning"
  MinStepSizeMsg      "warning"
  TimeAdjustmentMsg   "none"
  MaxConsecutiveZCsMsg    "error"
  SolverPrmCheckMsg   "warning"
  InheritedTsInSrcMsg     "warning"
```

```
    DiscreteInheritContinuousMsg "warning"
    MultiTaskDSMMsg     "error"
    MultiTaskCondExecSysMsg "error"
    MultiTaskRateTransMsg   "error"
    SingleTaskRateTransMsg  "none"
    TasksWithSamePriorityMsg "warning"
    SigSpecEnsureSampleTimeMsg "warning"
    CheckMatrixSingularityMsg "none"
    IntegerOverflowMsg      "warning"
    Int32ToFloatConvMsg     "warning"
    ParameterDowncastMsg    "error"
    ParameterOverflowMsg    "error"
    ParameterUnderflowMsg   "none"
    ParameterPrecisionLossMsg "warning"
    ParameterTunabilityLossMsg "warning"
    UnderSpecifiedDataTypeMsg "none"
    UnnecessaryDatatypeConvMsg "none"
    VectorMatrixConversionMsg "none"
    InvalidFcnCallConnMsg   "error"
    FcnCallInpInsideContextMsg "Use local settings"
    SignalLabelMismatchMsg  "none"
    UnconnectedInputMsg     "warning"
    UnconnectedOutputMsg    "warning"
    UnconnectedLineMsg      "warning"
    SFcnCompatibilityMsg    "none"
    UniqueDataStoreMsg      "none"
    BusObjectLabelMismatch  "warning"
    RootOutportRequireBusObject "warning"
    AssertControl       "UseLocalSettings"
    EnableOverflowDetection off
    ModelReferenceIOMsg     "none"
    ModelReferenceVersionMismatchMessage "none"
    ModelReferenceIOMismatchMessage "none"
    ModelReferenceCSMismatchMessage "none"
    ModelReferenceSimTargetVerbose off
    UnknownTsInhSupMsg      "warning"
    ModelReferenceDataLoggingMessage "warning"
    ModelReferenceSymbolNameMessage "warning"
    ModelReferenceExtraNoncontSigs "error"
    StateNameClashWarn      "warning"
    StrictBusMsg        "Warning"
    LoggingUnavailableSignals "error"
  }
  Simulink.HardwareCC {
    $ObjectID       6
    Version         "1.3.0"
    ProdBitPerChar      8
    ProdBitPerShort     16
    ProdBitPerInt       32
    ProdBitPerLong      32
    ProdIntDivRoundTo   "Undefined"
    ProdEndianess       "Unspecified"
    ProdWordSize        32
    ProdShiftRightIntArith  on
    ProdHWDeviceType    "32-bit Generic"
    TargetBitPerChar    8
    TargetBitPerShort   16
```

```
  TargetBitPerInt     32
  TargetBitPerLong    32
  TargetShiftRightIntArith on
  TargetIntDivRoundTo     "Undefined"
  TargetEndianess     "Unspecified"
  TargetWordSize      32
  TargetTypeEmulationWarnSuppressLevel 0
  TargetPreprocMaxBitsSint 32
  TargetPreprocMaxBitsUint 32
  TargetHWDeviceType      "Specified"
  TargetUnknown       off
  ProdEqTarget        on
}
Simulink.ModelReferenceCC {
  $ObjectID       7
  Version         "1.3.0"
  UpdateModelReferenceTargets "IfOutOfDateOrStructuralChange"
  CheckModelReferenceTargetMessage "error"
  ModelReferenceNumInstancesAllowed "Multi"
  ModelReferencePassRootInputsByReference on
  ModelReferenceMinAlgLoopOccurrences off
}
Simulink.RTWCC {
  $BackupClass        "Simulink.RTWCC"
  $ObjectID       8
  Array {
    Type            "Cell"
    Dimension           2
    Cell            "IncludeHyperlinkInReport"
    Cell            "GenerateTraceInfo"
    PropName            "DisabledProps"
  }
  Version         "1.3.0"
  SystemTargetFile    "grt.tlc"
  GenCodeOnly         off
  MakeCommand         "make_rtw"
  GenerateMakefile    on
  TemplateMakefile    "grt_default_tmf"
  GenerateReport      off
  SaveLog         off
  RTWVerbose          on
  RetainRTWFile       off
  ProfileTLC          off
  TLCDebug        off
  TLCCoverage         off
  TLCAssert       off
  ProcessScriptMode   "Default"
  ConfigurationMode   "Optimized"
  ConfigAtBuild       off
  IncludeHyperlinkInReport off
  LaunchReport        off
  TargetLang          "C"
  IncludeBusHierarchyInRTWFileBlockHierarchyMap off
  IncludeERTFirstTime     off
  GenerateTraceInfo   off
  RTWCompilerOptimization "Off"
  Array {
```

```
Type             "Handle"
Dimension            2
Simulink.CodeAppCC {
  $ObjectID            9
  Array {
Type             "Cell"
Dimension        16
Cell             "IgnoreCustomStorageClasses"
Cell             "InsertBlockDesc"
Cell             "SFDataObjDesc"
Cell             "SimulinkDataObjDesc"
Cell             "DefineNamingRule"
Cell             "SignalNamingRule"
Cell             "ParamNamingRule"
Cell             "InlinedPrmAccess"
Cell             "CustomSymbolStr"
Cell             "CustomSymbolStrGlobalVar"
Cell             "CustomSymbolStrType"
Cell             "CustomSymbolStrField"
Cell             "CustomSymbolStrFcn"
Cell             "CustomSymbolStrBlkIO"
Cell             "CustomSymbolStrTmpVar"
Cell             "CustomSymbolStrMacro"
PropName         "DisabledProps"
  }
  Version              "1.3.0"
  ForceParamTrailComments off
  GenerateComments        on
  IgnoreCustomStorageClasses on
  IncHierarchyInIds       off
  MaxIdLength          31
  PreserveName         off
  PreserveNameWithParent  off
  ShowEliminatedStatement off
  IncAutoGenComments      off
  SimulinkDataObjDesc     off
  SFDataObjDesc        off
  IncDataTypeInIds        off
  MangleLength         1
  CustomSymbolStrGlobalVar "$R$N$M"
  CustomSymbolStrType     "$N$R$M"
  CustomSymbolStrField    "$N$M"
  CustomSymbolStrFcn      "$R$N$M$F"
  CustomSymbolStrBlkIO    "rtb_$N$M"
  CustomSymbolStrTmpVar   "$N$M"
  CustomSymbolStrMacro    "$R$N$M"
  DefineNamingRule        "None"
  ParamNamingRule         "None"
  SignalNamingRule        "None"
  InsertBlockDesc         off
  SimulinkBlockComments   on
  EnableCustomComments    off
  InlinedPrmAccess        "Literals"
  ReqsInCode           off
}
Simulink.GRTTargetCC {
  $BackupClass         "Simulink.TargetCC"
```

```
  $ObjectID           10
  Array {
Type            "Cell"
Dimension       15
Cell            "IncludeMdlTerminateFcn"
Cell            "CombineOutputUpdateFcns"
Cell            "SuppressErrorStatus"
Cell            "ERTCustomFileBanners"
Cell            "GenerateSampleERTMain"
Cell            "GenerateTestInterfaces"
Cell            "ModelStepFunctionPrototypeControlCompliant"
Cell            "MultiInstanceERTCode"
Cell            "PurelyIntegerCode"
Cell            "SupportNonFinite"
Cell            "SupportComplex"
Cell            "SupportAbsoluteTime"
Cell            "SupportContinuousTime"
Cell            "SupportNonInlinedSFcns"
Cell            "PortableWordSizes"
PropName        "DisabledProps"
  }
  Version             "1.3.0"
  TargetFcnLib        "ansi_tfl_table_tmw.mat"
  TargetLibSuffix         ""
  TargetPreCompLibLocation ""
  GenFloatMathFcnCalls    "ANSI_C"
  UtilityFuncGeneration   "Auto"
  GenerateFullHeader      on
  GenerateSampleERTMain   off
  GenerateTestInterfaces  off
  IsPILTarget         off
  ModelReferenceCompliant on
  CompOptLevelCompliant   on
  IncludeMdlTerminateFcn  on
  CombineOutputUpdateFcns off
  SuppressErrorStatus     off
  IncludeFileDelimiter    "Auto"
  ERTCustomFileBanners    off
  SupportAbsoluteTime     on
  LogVarNameModifier      "rt_"
  MatFileLogging          on
  MultiInstanceERTCode    off
  SupportNonFinite        on
  SupportComplex          on
  PurelyIntegerCode       off
  SupportContinuousTime   on
  SupportNonInlinedSFcns  on
  EnableShiftOperators    on
  ParenthesesLevel        "Nominal"
  PortableWordSizes       off
  ModelStepFunctionPrototypeControlCompliant off
  ExtMode             off
  ExtModeStaticAlloc      off
  ExtModeTesting          off
  ExtModeStaticAllocSize  1000000
  ExtModeTransport        0
  ExtModeMexFile          "ext_comm"
```

```
        ExtModeIntrfLevel       "Level1"
        RTWCAPISignals          off
        RTWCAPIParams       off
        RTWCAPIStates       off
        GenerateASAP2       off
      }
      PropName            "Components"
    }
  }
  hdlcoderui.hdlcc {
    $ObjectID       11
    Description         "HDL Coder custom configuration component"
    Version         "1.3.0"
    Name            "HDL Coder"
    Array {
      Type            "Cell"
      Dimension           1
      Cell            ""
      PropName            "HDLConfigFile"
    }
    HDLCActiveTab       "0"
  }
  PropName        "Components"
    }
    Name            "Configuration"
    CurrentDlgPage          "Solver"
  }
  PropName            "ConfigurationSets"
}
Simulink.ConfigSet {
  $PropName           "ActiveConfigurationSet"
  $ObjectID           1
}
BlockDefaults {
  Orientation         "right"
  ForegroundColor     "black"
  BackgroundColor     "white"
  DropShadow          off
  NamePlacement       "normal"
  FontName            "Arial"
  FontSize            10
  FontWeight          "normal"
  FontAngle           "normal"
  ShowName            on
}
BlockParameterDefaults {
  Block {
    BlockType           ActionPort
    InitializeStates        "held"
    ActionType          "unset"
  }
  Block {
    BlockType           Backlash
    BacklashWidth       "1"
    InitialOutput       "0"
    ZeroCross           on
    SampleTime          "-1"
```

```
    }
    Block {
      BlockType             DataTypeConversion
      OutMin                "[]"
      OutMax                "[]"
      OutDataTypeMode           "Inherit via back propagation"
      OutDataType           "fixdt(1,16,0)"
      OutScaling            "[]"
      LockScale             off
      ConvertRealWorld          "Real World Value (RWV)"
      RndMeth               "Zero"
      SaturateOnIntegerOverflow on
      SampleTime            "-1"
    }
    Block {
      BlockType             Demux
      Outputs               "4"
      DisplayOption         "none"
      BusSelectionMode          off
    }
    Block {
      BlockType             Display
      Format                "short"
      Decimation            "10"
      Floating              off
      SampleTime            "-1"
    }
    Block {
      BlockType             Fcn
      Expr              "sin(u[1])"
      SampleTime            "-1"
    }
    Block {
      BlockType             If
      NumInputs             "1"
      IfExpression          "u1 > 0"
      ShowElse              on
      ZeroCross             on
      SampleTime            "-1"
    }
    Block {
      BlockType             Inport
      Port              "1"
      UseBusObject          off
      BusObject             "BusObject"
      BusOutputAsStruct         off
      PortDimensions            "-1"
      SampleTime            "-1"
      OutMin                "[]"
      OutMax                "[]"
      DataType              "auto"
      OutDataType           "fixdt(1,16,0)"
      OutScaling            "[]"
      SignalType            "auto"
      SamplingMode          "auto"
      LatchByDelayingOutsideSignal off
      LatchByCopyingInsideSignal off
```

```
    Interpolate         on
  }
  Block {
    BlockType           Integrator
    ExternalReset       "none"
    InitialConditionSource  "internal"
    InitialCondition        "0"
    LimitOutput         off
    UpperSaturationLimit    "inf"
    LowerSaturationLimit    "-inf"
    ShowSaturationPort      off
    ShowStatePort       off
    AbsoluteTolerance       "auto"
    IgnoreLimit         off
    ZeroCross           on
    ContinuousStateAttributes "''"
  }
  Block {
    BlockType           Lookup2D
    RowIndex            "[0 1]"
    ColumnIndex         "[0 1]"
    Table           "[0 0;0 0]"
    LookUpMeth          "Interpolation-Extrapolation"
    InputSameDT         on
    OutMin              "[]"
    OutMax              "[]"
    OutDataTypeMode         "Same as first input"
    OutDataType         "fixdt(1,16,0)"
    OutScaling          "[]"
    LockScale           off
    RndMeth             "Floor"
    SaturateOnIntegerOverflow on
    SampleTime          "-1"
    LUTDesignTableMode      "Redesign Table"
    LUTDesignDataSource     "Block Dialog"
    LUTDesignFunctionName   "sqrt(x)"
    LUTDesignUseExistingBP  on
    LUTDesignRelError       "0.01"
    LUTDesignAbsError       "1e-6"
  }
  Block {
    BlockType           Math
    Operator            "exp"
    OutputSignalType        "auto"
    SampleTime          "-1"
    OutMin              "[]"
    OutMax              "[]"
    OutDataTypeMode         "Same as first input"
    OutDataType         "fixdt(1,16,0)"
    OutScaling          "[]"
    LockScale           off
    RndMeth             "Floor"
    SaturateOnIntegerOverflow on
  }
  Block {
    BlockType           Mux
    Inputs              "4"
```

```
    DisplayOption       "none"
    UseBusObject        off
    BusObject           "BusObject"
    NonVirtualBus       off
  }
  Block {
    BlockType           Outport
    Port            "1"
    UseBusObject        off
    BusObject           "BusObject"
    BusOutputAsStruct       off
    PortDimensions          "-1"
    SampleTime          "-1"
    OutMin              "[]"
    OutMax              "[]"
    DataType            "auto"
    OutDataType         "fixdt(1,16,0)"
    OutScaling          "[]"
    SignalType          "auto"
    SamplingMode        "auto"
    OutputWhenDisabled      "held"
    InitialOutput       "[]"
  }
  Block {
    BlockType           Product
    Inputs              "2"
    Multiplication          "Element-wise(.*)"
    CollapseMode        "All dimensions"
    CollapseDim         "1"
    InputSameDT         on
    OutMin              "[]"
    OutMax              "[]"
    OutDataTypeMode         "Same as first input"
    OutDataType         "fixdt(1,16,0)"
    OutScaling          "[]"
    LockScale           off
    RndMeth             "Zero"
    SaturateOnIntegerOverflow on
    SampleTime          "-1"
  }
  Block {
    BlockType           Scope
    ModelBased          off
    TickLabels          "OneTimeTick"
    ZoomMode            "on"
    Grid            "on"
    TimeRange           "auto"
    YMin            "-5"
    YMax            "5"
    SaveToWorkspace         off
    SaveName            "ScopeData"
    LimitDataPoints         on
    MaxDataPoints       "5000"
    Decimation          "1"
    SampleInput         off
    SampleTime          "-1"
  }
```

```
Block {
  BlockType           "S-Function"
  FunctionName        "system"
  SFunctionModules        "''"
  PortCounts          "[]"
}
Block {
  BlockType           Step
  Time            "1"
  Before              "0"
  After           "1"
  SampleTime          "-1"
  VectorParams1D          on
  ZeroCross           on
}
Block {
  BlockType           SubSystem
  ShowPortLabels          "FromPortIcon"
  Permissions         "ReadWrite"
  PermitHierarchicalResolution "All"
  TreatAsAtomicUnit       off
  SystemSampleTime        "-1"
  RTWFcnNameOpts          "Auto"
  RTWFileNameOpts         "Auto"
  RTWMemSecFuncInitTerm   "Inherit from model"
  RTWMemSecFuncExecute    "Inherit from model"
  RTWMemSecDataConstants  "Inherit from model"
  RTWMemSecDataInternal   "Inherit from model"
  RTWMemSecDataParameters "Inherit from model"
  SimViewingDevice        off
  DataTypeOverride        "UseLocalSettings"
  MinMaxOverflowLogging   "UseLocalSettings"
}
Block {
  BlockType           Sum
  IconShape           "rectangular"
  Inputs              "++"
  CollapseMode        "All dimensions"
  CollapseDim         "1"
  InputSameDT         on
  OutMin              "[]"
  OutMax              "[]"
  OutDataTypeMode         "Same as first input"
  OutDataType         "fixdt(1,16,0)"
  OutScaling          "[]"
  LockScale           off
  RndMeth             "Floor"
  SaturateOnIntegerOverflow on
  SampleTime          "-1"
}
Block {
  BlockType           Switch
  Criteria            "u2 >= Threshold"
  Threshold           "0"
  InputSameDT         on
  OutMin              "[]"
  OutMax              "[]"
```

```
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "fixdt(1,16,0)"
    OutScaling          "[]"
    LockScale           off
    RndMeth             "Floor"
    SaturateOnIntegerOverflow on
    ZeroCross           on
    SampleTime          "-1"
  }
  Block {
    BlockType           Terminator
  }
  Block {
    BlockType           TransferFcn
    Numerator           "[1]"
    Denominator         "[1 2 1]"
    AbsoluteTolerance       "auto"
    ContinuousStateAttributes "''"
    Realization         "auto"
  }
  Block {
    BlockType           ZeroOrderHold
    SampleTime          "1"
  }
  Block {
    BlockType           Merge
    Inputs              "2"
    InitialOutput       "[]"
    AllowUnequalInputPortWidths off
    InputPortOffsets        "[]"
  }
  Block {
    BlockType           Constant
    Value           "1"
    VectorParams1D          on
    SamplingMode        "Sample based"
    OutMin              "[]"
    OutMax              "[]"
    OutDataTypeMode         "Inherit from 'Constant value'"
    OutDataType         "fixdt(1,16,0)"
    ConRadixGroup       "Use specified scaling"
    OutScaling          "[]"
    SampleTime          "inf"
    FramePeriod         "inf"
  }
  Block {
    BlockType           Lookup
    InputValues         "[-4:5]"
    Table           " rand(1,10)-0.5"
    LookUpMeth          "Interpolation-Extrapolation"
    OutMin              "[]"
    OutMax              "[]"
    OutDataTypeMode         "Same as input"
    OutDataType         "fixdt(1,16,0)"
    OutScaling          "[]"
    LockScale           off
    RndMeth             "Floor"
```

```
      SaturateOnIntegerOverflow on
      SampleTime          "-1"
      LUTDesignTableMode      "Redesign Table"
      LUTDesignDataSource     "Block Dialog"
      LUTDesignFunctionName   "sqrt(x)"
      LUTDesignUseExistingBP  on
      LUTDesignRelError       "0.01"
      LUTDesignAbsError       "1e-6"
    }
    Block {
      BlockType           MinMax
      Function            "min"
      Inputs              "1"
      InputSameDT         on
      OutMin              "[]"
      OutMax              "[]"
      OutDataTypeMode         "Inherit via internal rule"
      OutDataType         "fixdt(1,16,0)"
      OutScaling          "[]"
      LockScale           off
      RndMeth             "Floor"
      SaturateOnIntegerOverflow on
      ZeroCross           on
      SampleTime          "-1"
    }
    Block {
      BlockType           RelationalOperator
      Operator            ">="
      InputSameDT         on
      LogicOutDataTypeMode    "Logical (see Configuration Parameters:
Optimization)"
      LogicDataType       "uint(8)"
      ZeroCross           on
      SampleTime          "-1"
    }
    Block {
      BlockType           Saturate
      UpperLimit          "0.5"
      LowerLimit          "-0.5"
      LinearizeAsGain         on
      ZeroCross           on
      SampleTime          "-1"
      OutMin              "[]"
      OutMax              "[]"
      OutDataTypeMode         "Same as input"
      OutDataType         "fixdt(1,16,0)"
      OutScaling          "[]"
      LockScale           off
      RndMeth             "Floor"
    }
  }
  AnnotationDefaults {
    HorizontalAlignment     "center"
    VerticalAlignment       "middle"
    ForegroundColor     "black"
    BackgroundColor     "white"
    DropShadow          off
```

```
  FontName            "Arial"
  FontSize            10
  FontWeight          "normal"
  FontAngle           "normal"
  UseDisplayTextAsClickCallback off
}
LineDefaults {
  FontName            "Arial"
  FontSize            9
  FontWeight          "normal"
  FontAngle           "normal"
}
System {
  Name            "Experiment_6_matlab5"
  Location            [2, 78, 1398, 1000]
  Open            on
  ModelBrowserVisibility  off
  ModelBrowserWidth       200
  ScreenColor         "white"
  PaperOrientation        "landscape"
  PaperPositionMode       "auto"
  PaperType           "usletter"
  PaperUnits          "inches"
  TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
  TiledPageScale      1
  ShowPageBoundaries      off
  ZoomFactor          "100"
  ReportName          "simulink-default.rpt"
  Block {
    BlockType           SubSystem
    Name            "Air Compressor"
    Ports           [3, 2]
    Position            [975, 57, 1130, 183]
    MinAlgLoopOccurrences   off
    RTWSystemCode       "Auto"
    FunctionWithSeparateData off
    System {
  Name            "Air Compressor"
  Location        [6, 82, 1274, 746]
  Open            off
  ModelBrowserVisibility  off
  ModelBrowserWidth   200
  ScreenColor     "white"
  PaperOrientation    "landscape"
  PaperPositionMode   "auto"
  PaperType       "usletter"
  PaperUnits      "inches"
  TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
  TiledPageScale      1
  ShowPageBoundaries  off
  ZoomFactor      "100"
  Block {
    BlockType       Inport
    Name            "Water Flowrate"
    Position        [40, 63, 70, 77]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
```

```
    OutScaling          "2^0"
  }
  Block {
    BlockType       Inport
    Name            "Fouling"
    Position        [40, 123, 70, 137]
    Port            "2"
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType       Inport
    Name            "Oil Temp"
    Position        [45, 318, 75, 332]
    Port            "3"
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType       Display
    Name            "Display2"
    Ports           [1]
    Position        [615, 290, 705, 320]
    Decimation          "1"
    Lockdown        off
  }
  Block {
    BlockType       Product
    Name            "Divide1"
    Ports           [2, 1]
    Position        [780, 447, 810, 478]
    Inputs          "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth         "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       Product
    Name            "Divide2"
    Ports           [2, 1]
    Position        [775, 197, 805, 228]
    Inputs          "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth         "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
```

```
      BlockType       Lookup2D
      Name            "Lookup\nTable (2-D)1"
      Position        [465, 138, 520, 187]
      RowIndex        "[1,2,3,4,5]"
      ColumnIndex         "[0,0.25,0.5,0.75,1]"
      Table
"reshape([0.5,0.65,0.75,0.8,0.83,0.52,0.71,0.81,0.86,0.9,0.55,0.75,0.86,0.92,
0.94,0.6,0.81,0.91,0.96,0.97,0.65,0.85,0.94,0.97,0.99],5,5)"
      InputSameDT         off
      OutDataType         "sfix(16)"
      OutScaling          "2^0"
      SaturateOnIntegerOverflow off
    }
    Block {
      BlockType       SubSystem
      Name            "NTU value"
      Ports           [2, 1]
      Position        [110, 49, 300, 156]
      TreatAsAtomicUnit   on
      MinAlgLoopOccurrences   off
      RTWSystemCode       "Auto"
      FunctionWithSeparateData off
      System {
        Name            "NTU value"
        Location            [6, 82, 1274, 746]
        Open            off
        ModelBrowserVisibility  off
        ModelBrowserWidth       200
        ScreenColor         "white"
        PaperOrientation        "landscape"
        PaperPositionMode       "auto"
        PaperType           "usletter"
        PaperUnits          "inches"
        TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
        TiledPageScale      1
        ShowPageBoundaries      off
        ZoomFactor          "100"
        Block {
          BlockType           Inport
          Name            "Water Flowrate"
          Position            [15, 23, 45, 37]
          IconDisplay         "Port number"
          OutDataType         "sfix(16)"
          OutScaling          "2^0"
        }
        Block {
          BlockType           Inport
          Name            "Fouling"
          Position            [25, 563, 55, 577]
          Port            "2"
          IconDisplay         "Port number"
          OutDataType         "sfix(16)"
          OutScaling          "2^0"
        }
        Block {
          BlockType           Constant
          Name            "C_min"
```

```
    Position            [920, 366, 945, 394]
    Value           "1704"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType           Product
    Name            "Divide"
    Ports           [2, 1]
    Position            [335, 87, 365, 118]
    Inputs              "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth             "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Product
    Name            "Divide1"
    Ports           [2, 1]
    Position            [320, 337, 350, 368]
    Inputs              "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth             "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Product
    Name            "Divide2"
    Ports           [2, 1]
    Position            [705, 167, 735, 198]
    Inputs              "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth             "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Product
    Name            "Divide3"
    Ports           [2, 1]
    Position            [990, 297, 1020, 328]
    Inputs              "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
```

```
  OutScaling          "2^-10"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           Display
  Name            "NTU"
  Ports           [1]
  Position            [1060, 190, 1150, 220]
  Decimation          "1"
  Lockdown            off
}
Block {
  BlockType           Product
  Name            "Product1"
  Ports           [2, 1]
  Position            [885, 241, 925, 299]
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode         "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           Sum
  Name            "Sum"
  Ports           [3, 1]
  Position            [550, 185, 590, 235]
  ShowName            off
  IconShape           "round"
  Inputs              "|+++"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode         "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           SubSystem
  Name            "Surface Area"
  Ports           [0, 1]
  Position            [655, 269, 755, 311]
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  System {
Name            "Surface Area"
Location        [433, 403, 931, 703]
Open            off
ModelBrowserVisibility  off
ModelBrowserWidth   200
ScreenColor     "white"
PaperOrientation    "landscape"
PaperPositionMode   "auto"
PaperType       "usletter"
```

```
PaperUnits      "inches"
TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
TiledPageScale      1
ShowPageBoundaries  off
ZoomFactor      "100"
Block {
  BlockType       Constant
  Name            "D"
  Position        [115, 121, 140, 149]
  Value           "0.02"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Display
  Name            "Display2"
  Ports           [1]
  Position        [310, 60, 400, 90]
  Decimation          "1"
  Lockdown        off
}
Block {
  BlockType       Constant
  Name            "L"
  Position        [120, 186, 145, 214]
  Value           "10"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "Pi"
  Position        [115, 61, 140, 89]
  Value           "3.14"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Product
  Name            "Product1"
  Ports           [3, 1]
  Position        [240, 112, 270, 148]
  Inputs          "3"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Outport
  Name            "Out1"
  Position        [355, 123, 385, 137]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
```

```
}
Line {
  SrcBlock        "Pi"
  SrcPort         1
  Points          [40, 0; 0, 45]
  DstBlock        "Product1"
  DstPort         1
}
Line {
  SrcBlock        "D"
  SrcPort         1
  Points          [40, 0; 0, -5]
  DstBlock        "Product1"
  DstPort         2
}
Line {
  SrcBlock        "L"
  SrcPort         1
  Points          [35, 0; 0, -60]
  DstBlock        "Product1"
  DstPort         3
}
Line {
  SrcBlock        "Product1"
  SrcPort         1
  Points          [20, 0]
  Branch {
    DstBlock            "Out1"
    DstPort         1
  }
  Branch {
    DstBlock            "Display2"
    DstPort         1
  }
}
  }
}
Block {
  BlockType           Constant
  Name            "const1"
  Position            [220, 321, 245, 349]
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType           Constant
  Name            "const2"
  Position            [615, 106, 640, 134]
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType           Constant
  Name            "const7"
  Position            [245, 46, 270, 74]
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
```

```
}
Block {
  BlockType           SubSystem
  Name            "hi"
  Ports           [1, 1]
  Position            [75, 141, 265, 249]
  TreatAsAtomicUnit       on
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  System {
Name            "hi"
Location        [6, 82, 1274, 746]
Open            off
ModelBrowserVisibility  off
ModelBrowserWidth   200
ScreenColor     "white"
PaperOrientation    "landscape"
PaperPositionMode   "auto"
PaperType       "usletter"
PaperUnits      "inches"
TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
TiledPageScale      1
ShowPageBoundaries  off
ZoomFactor      "100"
Block {
  BlockType       Inport
  Name            "In1"
  Position        [15, 68, 45, 82]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Display
  Name            "Display1"
  Ports           [1]
  Position        [525, 155, 615, 185]
  Decimation          "1"
  Lockdown        off
}
Block {
  BlockType       Display
  Name            "Display2"
  Ports           [1]
  Position        [1025, 215, 1115, 245]
  Decimation          "1"
  Lockdown        off
}
Block {
  BlockType       Display
  Name            "Display3"
  Ports           [1]
  Position        [870, 170, 960, 200]
  Decimation          "1"
  Lockdown        off
}
```

```
Block {
  BlockType       Product
  Name            "Divide"
  Ports           [2, 1]
  Position        [325, 212, 355, 243]
  Inputs          "*/"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^-10"
  RndMeth         "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Product
  Name            "Divide1"
  Ports           [2, 1]
  Position        [565, 237, 595, 268]
  Inputs          "*/"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^-10"
  RndMeth         "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Product
  Name            "Divide2"
  Ports           [2, 1]
  Position        [840, 252, 870, 283]
  Inputs          "*/"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^-10"
  RndMeth         "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       SubSystem
  Name            "Embedded\nMATLAB Function"
  Ports           [1, 1]
  Position        [1080, 14, 1120, 46]
  PermitHierarchicalResolution "ExplicitOnly"
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  Array {
    Type            "Handle"
    Dimension           0
    PropName            "AvailSigsLoadSave"
  }
  MaskType        "Stateflow"
```

```
MaskDescription     "Embedded MATLAB block"
MaskDisplay         "disp('fcn');"
MaskSelfModifiable      on
MaskIconFrame       on
MaskIconOpaque      off
MaskIconRotate      "none"
MaskIconUnits       "autoscale"
System {
  Name            "Embedded\nMATLAB Function"
  Location            [257, 457, 812, 717]
  Open            off
  ModelBrowserVisibility  off
  ModelBrowserWidth       200
  ScreenColor         "white"
  PaperOrientation        "landscape"
  PaperPositionMode       "auto"
  PaperType           "usletter"
  PaperUnits          "inches"
  TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
  TiledPageScale      1
  ShowPageBoundaries      off
  ZoomFactor          "100"
  Block {
  BlockType           Inport
  Name            "u"
  Position            [20, 101, 40, 119]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  }
  Block {
  BlockType           Demux
  Name            " Demux "
  Ports           [1, 1]
  Position            [270, 160, 320, 200]
  Outputs         "1"
  }
  Block {
  BlockType           "S-Function"
  Name            " SFunction "
  Tag             "Stateflow S-Function Experiment_6_matlab5 2"
  Ports           [1, 2]
  Position            [180, 100, 230, 160]
  FunctionName        "sf_sfun"
  PortCounts          "[1 2]"
  EnableBusSupport        on
  Port {
  PortNumber          2
  Name            "y"
  RTWStorageClass     "Auto"
  DataLoggingNameMode     "SignalName"
  }
  }
  Block {
  BlockType           Terminator
  Name            " Terminator "
  Position            [460, 171, 480, 189]
```

```
    }
    Block {
    BlockType           Outport
    Name            "y"
    Position            [460, 101, 480, 119]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    }
    Line {
    SrcBlock            " SFunction "
    SrcPort         1
    Points          [0, 65]
    DstBlock            " Demux "
    DstPort         1
    }
    Line {
    SrcBlock            " Demux "
    SrcPort         1
    DstBlock            " Terminator "
    DstPort         1
    }
    Line {
    SrcBlock            "u"
    SrcPort         1
    DstBlock            " SFunction "
    DstPort         1
    }
    Line {
    Name            "y"
    Labels          [0, 0]
    SrcBlock            " SFunction "
    SrcPort         2
    DstBlock            "y"
    DstPort         1
    }
  }
}
Block {
  BlockType       SubSystem
  Name            "Embedded\nMATLAB Function1"
  Ports           [1, 1]
  Position        [1175, 14, 1215, 46]
  PermitHierarchicalResolution "ExplicitOnly"
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  Array {
    Type            "Handle"
    Dimension           0
    PropName            "AvailSigsLoadSave"
  }
  MaskType        "Stateflow"
  MaskDescription     "Embedded MATLAB block"
  MaskDisplay         "disp('fcn');"
  MaskSelfModifiable      on
  MaskIconFrame       on
```

```
MaskIconOpaque      off
MaskIconRotate      "none"
MaskIconUnits       "autoscale"
System {
  Name            "Embedded\nMATLAB Function1"
  Location            [257, 457, 812, 717]
  Open            off
  ModelBrowserVisibility  off
  ModelBrowserWidth       200
  ScreenColor         "white"
  PaperOrientation        "landscape"
  PaperPositionMode       "auto"
  PaperType           "usletter"
  PaperUnits          "inches"
  TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
  TiledPageScale      1
  ShowPageBoundaries      off
  ZoomFactor          "100"
  Block {
  BlockType           Inport
  Name            "u"
  Position            [20, 101, 40, 119]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  }
  Block {
  BlockType           Demux
  Name            " Demux "
  Ports           [1, 1]
  Position            [270, 160, 320, 200]
  Outputs         "1"
  }
  Block {
  BlockType           "S-Function"
  Name            " SFunction "
  Tag             "Stateflow S-Function Experiment_6_matlab5 3"
  Ports           [1, 2]
  Position            [180, 100, 230, 160]
  FunctionName        "sf_sfun"
  PortCounts          "[1 2]"
  EnableBusSupport        on
  Port {
  PortNumber          2
  Name            "y"
  RTWStorageClass     "Auto"
  DataLoggingNameMode     "SignalName"
  }
  }
  Block {
  BlockType           Terminator
  Name            " Terminator "
  Position            [460, 171, 480, 189]
  }
  Block {
  BlockType           Outport
  Name            "y"
```

```
    Position            [460, 101, 480, 119]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    }
    Line {
    SrcBlock            " SFunction "
    SrcPort         1
    Points          [0, 65]
    DstBlock            " Demux "
    DstPort         1
    }
    Line {
    SrcBlock            " Demux "
    SrcPort         1
    DstBlock            " Terminator "
    DstPort         1
    }
    Line {
    SrcBlock            "u"
    SrcPort         1
    DstBlock            " SFunction "
    DstPort         1
    }
    Line {
    Name            "y"
    Labels          [0, 0]
    SrcBlock            " SFunction "
    SrcPort         2
    DstBlock            "y"
    DstPort         1
    }
  }
}
Block {
  BlockType       Math
  Name            "Math\nFunction"
  Ports           [2, 1]
  Position        [630, 237, 660, 268]
  Operator        "pow"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Math
  Name            "Math\nFunction1"
  Ports           [2, 1]
  Position        [635, 352, 665, 383]
  Operator        "pow"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Product
  Name            "Product"
  Ports           [4, 1]
  Position        [245, 253, 275, 287]
```

```
    Inputs            "4"
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Product
    Name              "Product1"
    Ports             [2, 1]
    Position          [455, 215, 485, 250]
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Product
    Name              "Product2"
    Ports             [4, 1]
    Position          [745, 222, 775, 258]
    Inputs            "4"
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Constant
    Name              "const1"
    Position          [565, 296, 590, 324]
    Value             "0.8"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Constant
    Name              "const10"
    Position          [650, 300, 690, 330]
    Value             "0.623"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Constant
    Name              "const11"
    Position          [775, 316, 800, 344]
    Value             "0.02"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
```

```
Block {
  BlockType       Constant
  Name            "const12"
  Position        [580, 401, 605, 429]
  Value           "0.4"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const2"
  Position        [140, 231, 165, 259]
  Value           "994"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const3"
  Position        [135, 286, 160, 314]
  Value           "0.25"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const4"
  Position        [130, 341, 155, 369]
  Value           "3.14"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const5"
  Position        [190, 368, 260, 392]
  Value           "0.0004"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const6"
  Position        [375, 251, 400, 279]
  Value           "0.02"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const7"
  Position        [430, 288, 510, 322]
  Value           "0.000000724"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
```

```
    BlockType         Constant
    Name              "const8"
    Position          [625, 177, 685, 203]
    Value             "0.023"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Constant
    Name              "const9"
    Position          [580, 351, 605, 379]
    Value             "4.83"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Outport
    Name              "Out1"
    Position          [930, 258, 960, 272]
    IconDisplay           "Port number"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Line {
    SrcBlock          "const2"
    SrcPort           1
    Points            [30, 0; 0, 10]
    DstBlock          "Product"
    DstPort           1
  }
  Line {
    SrcBlock          "const3"
    SrcPort           1
    Points            [30, 0; 0, -35]
    DstBlock          "Product"
    DstPort           2
  }
  Line {
    SrcBlock          "const4"
    SrcPort           1
    Points            [45, 0; 0, -80]
    DstBlock          "Product"
    DstPort           3
  }
  Line {
    SrcBlock          "const5"
    SrcPort           1
    Points            [0, -75; -35, 0]
    DstBlock          "Product"
    DstPort           4
  }
  Line {
    SrcBlock          "In1"
    SrcPort           1
    Points            [195, 0; 0, 100; 30, 0; 0, 45]
    DstBlock          "Divide"
    DstPort           1
```

```
}
Line {
  SrcBlock        "Product"
  SrcPort         1
  Points          [30, 0]
  DstBlock        "Divide"
  DstPort         2
}
Line {
  SrcBlock        "Divide"
  SrcPort         1
  Points          [40, 0; 0, -5]
  DstBlock        "Product1"
  DstPort         1
}
Line {
  SrcBlock        "const6"
  SrcPort         1
  Points          [15, 0; 0, -25]
  DstBlock        "Product1"
  DstPort         2
}
Line {
  SrcBlock        "Product1"
  SrcPort         1
  Points          [15, 0]
  Branch {
    Points          [15, 0; 0, 10]
    DstBlock            "Divide1"
    DstPort         1
  }
  Branch {
    Points          [0, -65]
    DstBlock            "Display1"
    DstPort         1
  }
}
Line {
  SrcBlock        "const7"
  SrcPort         1
  Points          [5, 0; 0, -45]
  DstBlock        "Divide1"
  DstPort         2
}
Line {
  SrcBlock        "const8"
  SrcPort         1
  Points          [20, 0; 0, 35]
  DstBlock        "Product2"
  DstPort         1
}
Line {
  SrcBlock        "const10"
  SrcPort         1
  Points          [10, 0; 0, -70]
  DstBlock        "Product2"
  DstPort         3
```

```
}
Line {
  SrcBlock        "Product2"
  SrcPort         1
  Points          [20, 0; 0, 20; 10, 0]
  Branch {
    DstBlock            "Divide2"
    DstPort         1
  }
  Branch {
    Points          [0, -75]
    DstBlock            "Display3"
    DstPort         1
  }
}
Line {
  SrcBlock        "const11"
  SrcPort         1
  Points          [10, 0; 0, -55]
  DstBlock        "Divide2"
  DstPort         2
}
Line {
  SrcBlock        "Divide2"
  SrcPort         1
  Points          [20, 0; 0, -5; 5, 0]
  Branch {
    DstBlock            "Out1"
    DstPort         1
  }
  Branch {
    Points          [0, -35]
    DstBlock            "Display2"
    DstPort         1
  }
}
Line {
  SrcBlock        "Divide1"
  SrcPort         1
  Points          [5, 0; 0, -10]
  DstBlock        "Math\nFunction"
  DstPort         1
}
Line {
  SrcBlock        "Math\nFunction"
  SrcPort         1
  Points          [30, 0; 0, -20]
  DstBlock        "Product2"
  DstPort         2
}
Line {
  SrcBlock        "const1"
  SrcPort         1
  Points          [10, 0; 0, -50]
  DstBlock        "Math\nFunction"
  DstPort         2
}
```

```
        Line {
          SrcBlock        "const9"
          SrcPort         1
          Points          [5, 0; 0, -5]
          DstBlock        "Math\nFunction1"
          DstPort         1
        }
        Line {
          SrcBlock        "const12"
          SrcPort         1
          Points          [5, 0; 0, -40]
          DstBlock        "Math\nFunction1"
          DstPort         2
        }
        Line {
          SrcBlock        "Math\nFunction1"
          SrcPort         1
          Points          [60, 0]
          DstBlock        "Product2"
          DstPort         4
        }
          }
        }
        Block {
          BlockType           SubSystem
          Name            "ho"
          Ports           [0, 1]
          Position            [70, 406, 260, 514]
          TreatAsAtomicUnit       on
          MinAlgLoopOccurrences   off
          RTWSystemCode       "Auto"
          FunctionWithSeparateData off
          System {
        Name            "ho"
        Location        [2, 82, 1270, 754]
        Open            off
        ModelBrowserVisibility  off
        ModelBrowserWidth   200
        ScreenColor     "white"
        PaperOrientation    "landscape"
        PaperPositionMode   "auto"
        PaperType       "usletter"
        PaperUnits      "inches"
        TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
        TiledPageScale      1
        ShowPageBoundaries  off
        ZoomFactor      "100"
        Block {
          BlockType       Reference
          Name            "Compare\nTo Constant"
          Ports           [1, 1]
          Position        [595, 195, 640, 235]
          SourceBlock         "simulink/Logic and Bit\nOperations/Compare\nTo
Constant"
          SourceType          "Compare To Constant"
          ShowPortLabels      "FromPortIcon"
          SystemSampleTime    "-1"
```

```
      FunctionWithSeparateData off
      RTWMemSecFuncInitTerm   "Inherit from model"
      RTWMemSecFuncExecute    "Inherit from model"
      RTWMemSecDataConstants  "Inherit from model"
      RTWMemSecDataInternal   "Inherit from model"
      RTWMemSecDataParameters "Inherit from model"
      relop           "<"
      const           "2300"
      LogicOutDataTypeMode    "uint8"
      ZeroCross       off
    }
    Block {
      BlockType       Reference
      Name            "Compare\nTo Constant1"
      Ports           [1, 1]
      Position        [595, 285, 640, 325]
      SourceBlock         "simulink/Logic and Bit\nOperations/Compare\nTo
Constant"
      SourceType          "Compare To Constant"
      ShowPortLabels      "FromPortIcon"
      SystemSampleTime    "-1"
      FunctionWithSeparateData off
      RTWMemSecFuncInitTerm   "Inherit from model"
      RTWMemSecFuncExecute    "Inherit from model"
      RTWMemSecDataConstants  "Inherit from model"
      RTWMemSecDataInternal   "Inherit from model"
      RTWMemSecDataParameters "Inherit from model"
      relop           ">"
      const           "2300"
      LogicOutDataTypeMode    "boolean"
      ZeroCross       off
    }
    Block {
      BlockType       Display
      Name            "Display1"
      Ports           [1]
      Position        [1125, 195, 1215, 225]
      Decimation          "1"
      Lockdown        off
    }
    Block {
      BlockType       Display
      Name            "Display2"
      Ports           [1]
      Position        [995, 475, 1085, 505]
      Decimation          "1"
      Lockdown        off
    }
    Block {
      BlockType       Display
      Name            "Display3"
      Ports           [1]
      Position        [970, 415, 1060, 445]
      Decimation          "1"
      Lockdown        off
    }
    Block {
```

```
    BlockType       Display
    Name            "Display4"
    Ports           [1]
    Position        [830, 120, 920, 150]
    Decimation          "1"
    Lockdown        off
  }
  Block {
    BlockType       Product
    Name            "Divide"
    Ports           [2, 1]
    Position        [250, 162, 280, 193]
    Inputs          "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth         "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       Product
    Name            "Divide1"
    Ports           [2, 1]
    Position        [490, 187, 520, 218]
    Inputs          "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth         "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       Product
    Name            "Divide2"
    Ports           [2, 1]
    Position        [930, 477, 960, 508]
    Inputs          "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode     "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^-10"
    RndMeth         "Floor"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       Product
    Name            "Divide3"
    Ports           [2, 1]
    Position        [1065, 237, 1095, 268]
    Inputs          "*/"
    CollapseMode        "All dimensions"
    InputSameDT         off
```

```
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^-10"
  RndMeth         "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       SubSystem
  Name            "Embedded\nMATLAB Function"
  Ports           [1, 1]
  Position        [755, 324, 795, 356]
  PermitHierarchicalResolution "ExplicitOnly"
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  Array {
    Type            "Handle"
    Dimension           0
    PropName            "AvailSigsLoadSave"
  }
  MaskType        "Stateflow"
  MaskDescription     "Embedded MATLAB block"
  MaskDisplay         "disp('fcn');"
  MaskSelfModifiable      on
  MaskIconFrame       on
  MaskIconOpaque      off
  MaskIconRotate      "none"
  MaskIconUnits       "autoscale"
  System {
    Name            "Embedded\nMATLAB Function"
    Location            [257, 457, 812, 717]
    Open            off
    ModelBrowserVisibility  off
    ModelBrowserWidth       200
    ScreenColor         "white"
    PaperOrientation        "landscape"
    PaperPositionMode       "auto"
    PaperType           "usletter"
    PaperUnits          "inches"
    TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
    TiledPageScale      1
    ShowPageBoundaries      off
    ZoomFactor          "100"
    Block {
    BlockType           Inport
    Name            "u"
    Position            [20, 101, 40, 119]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    }
    Block {
    BlockType           Demux
    Name            " Demux "
    Ports           [1, 1]
    Position            [270, 160, 320, 200]
    Outputs         "1"
```

```
}
Block {
BlockType           "S-Function"
Name            " SFunction "
Tag             "Stateflow S-Function Experiment_6_matlab5 1"
Ports           [1, 2]
Position            [180, 100, 230, 160]
FunctionName        "sf_sfun"
PortCounts          "[1 2]"
EnableBusSupport        on
Port {
PortNumber          2
Name            "y"
RTWStorageClass     "Auto"
DataLoggingNameMode     "SignalName"
}
}
Block {
BlockType           Terminator
Name            " Terminator "
Position            [460, 171, 480, 189]
}
Block {
BlockType           Outport
Name            "y"
Position            [460, 101, 480, 119]
IconDisplay         "Port number"
OutDataType         "sfix(16)"
OutScaling          "2^0"
}
Line {
SrcBlock            " SFunction "
SrcPort         1
Points          [0, 65]
DstBlock            " Demux "
DstPort         1
}
Line {
SrcBlock            " Demux "
SrcPort         1
DstBlock            " Terminator "
DstPort         1
}
Line {
SrcBlock            "u"
SrcPort         1
DstBlock            " SFunction "
DstPort         1
}
Line {
Name            "y"
Labels          [0, 0]
SrcBlock            " SFunction "
SrcPort         2
DstBlock            "y"
DstPort         1
}
```

```
    }
  }
  Block {
    BlockType       SubSystem
    Name            "Embedded\nMATLAB Function1"
    Ports           [1, 1]
    Position        [760, 434, 800, 466]
    PermitHierarchicalResolution "ExplicitOnly"
    MinAlgLoopOccurrences   off
    RTWSystemCode       "Auto"
    FunctionWithSeparateData off
    Array {
      Type            "Handle"
      Dimension           0
      PropName            "AvailSigsLoadSave"
    }
    MaskType        "Stateflow"
    MaskDescription     "Embedded MATLAB block"
    MaskDisplay         "disp('fcn');"
    MaskSelfModifiable      on
    MaskIconFrame       on
    MaskIconOpaque      off
    MaskIconRotate      "none"
    MaskIconUnits       "autoscale"
    System {
      Name            "Embedded\nMATLAB Function1"
      Location            [257, 457, 812, 717]
      Open            off
      ModelBrowserVisibility  off
      ModelBrowserWidth       200
      ScreenColor         "white"
      PaperOrientation        "landscape"
      PaperPositionMode       "auto"
      PaperType           "usletter"
      PaperUnits          "inches"
      TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
      TiledPageScale      1
      ShowPageBoundaries      off
      ZoomFactor          "100"
      Block {
      BlockType           Inport
      Name            "u"
      Position            [20, 101, 40, 119]
      IconDisplay         "Port number"
      OutDataType         "sfix(16)"
      OutScaling          "2^0"
      }
      Block {
      BlockType           Demux
      Name            " Demux "
      Ports           [1, 1]
      Position            [270, 160, 320, 200]
      Outputs         "1"
      }
      Block {
      BlockType           "S-Function"
      Name            " SFunction "
```

```
    Tag               "Stateflow S-Function Experiment_6_matlab5 4"
    Ports             [1, 2]
    Position              [180, 100, 230, 160]
    FunctionName          "sf_sfun"
    PortCounts            "[1 2]"
    EnableBusSupport          on
    Port {
    PortNumber            2
    Name              "y"
    RTWStorageClass       "Auto"
    DataLoggingNameMode       "SignalName"
    }
    }
    Block {
    BlockType             Terminator
    Name              " Terminator "
    Position              [460, 171, 480, 189]
    }
    Block {
    BlockType             Outport
    Name              "y"
    Position              [460, 101, 480, 119]
    IconDisplay           "Port number"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    }
    Line {
    SrcBlock              " SFunction "
    SrcPort           1
    Points            [0, 65]
    DstBlock              " Demux "
    DstPort           1
    }
    Line {
    SrcBlock              " Demux "
    SrcPort           1
    DstBlock              " Terminator "
    DstPort           1
    }
    Line {
    SrcBlock              "u"
    SrcPort           1
    DstBlock              " SFunction "
    DstPort           1
    }
    Line {
    Name              "y"
    Labels            [0, 0]
    SrcBlock              " SFunction "
    SrcPort           2
    DstBlock              "y"
    DstPort           1
    }
  }
}
Block {
  BlockType        Lookup
```

```
    Name              "Lookup Table"
    Position          [835, 185, 885, 235]
    InputValues           "[0,0.05,0.1,0.25,0.5,1]"
    Table             "[0,17.46,11.56,7.37,5.74,4.86]"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Product
    Name              "Product"
    Ports             [4, 1]
    Position          [170, 203, 200, 237]
    Inputs            "4"
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Product
    Name              "Product1"
    Ports             [2, 1]
    Position          [380, 165, 410, 200]
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Product
    Name              "Product2"
    Ports             [4, 1]
    Position          [855, 312, 885, 348]
    Inputs            "4"
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType         Product
    Name              "Product3"
    Ports             [2, 1]
    Position          [735, 200, 765, 235]
    CollapseMode          "All dimensions"
    InputSameDT           off
    OutDataTypeMode       "Inherit via internal rule"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
    SaturateOnIntegerOverflow off
```

```
}
Block {
  BlockType       Product
  Name            "Product4"
  Ports           [2, 1]
  Position        [670, 322, 700, 353]
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Product
  Name            "Product5"
  Ports           [2, 1]
  Position        [965, 200, 995, 235]
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Constant
  Name            "const1"
  Position        [140, 111, 165, 139]
  Value           "0.8"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const10"
  Position        [775, 386, 800, 414]
  Value           "0.62"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const11"
  Position        [865, 541, 890, 569]
  Value           "0.02"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const13"
  Position        [660, 251, 685, 279]
  Value           "0.667"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
```

```
Block {
  BlockType       Constant
  Name            "const14"
  Position        [905, 231, 930, 259]
  Value           "0.1404"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const15"
  Position        [985, 286, 1010, 314]
  Value           "0.01"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const2"
  Position        [65, 181, 90, 209]
  Value           "863.9"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const3"
  Position        [60, 236, 85, 264]
  Value           "0.25"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const4"
  Position        [55, 291, 80, 319]
  Value           "3.14"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const5"
  Position        [140, 276, 165, 304]
  Value           "0.0005"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Constant
  Name            "const6"
  Position        [300, 201, 325, 229]
  Value           "0.01"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
```

```
    BlockType        Constant
    Name             "const7"
    Position         [390, 241, 415, 269]
    Value            "0.00008565"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Block {
    BlockType        Constant
    Name             "const8"
    Position         [770, 266, 795, 294]
    Value            "0.023"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Block {
    BlockType        Constant
    Name             "const9"
    Position         [675, 446, 700, 474]
    Value            "4.83"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Block {
    BlockType        Outport
    Name             "Out1"
    Position         [1130, 248, 1160, 262]
    IconDisplay          "Port number"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Line {
    SrcBlock         "const2"
    SrcPort          1
    Points           [30, 0; 0, 10]
    DstBlock         "Product"
    DstPort          1
  }
  Line {
    SrcBlock         "const3"
    SrcPort          1
    Points           [30, 0; 0, -35]
    DstBlock         "Product"
    DstPort          2
  }
  Line {
    SrcBlock         "const4"
    SrcPort          1
    Points           [45, 0; 0, -80]
    DstBlock         "Product"
    DstPort          3
  }
  Line {
    SrcBlock         "const5"
    SrcPort          1
    Points           [0, -35; -15, 0]
    DstBlock         "Product"
```

```
    DstPort          4
  }
  Line {
    SrcBlock         "const1"
    SrcPort          1
    Points           [30, 0; 0, 45]
    DstBlock         "Divide"
    DstPort          1
  }
  Line {
    SrcBlock         "Product"
    SrcPort          1
    Points           [30, 0]
    DstBlock         "Divide"
    DstPort          2
  }
  Line {
    SrcBlock         "Divide"
    SrcPort          1
    Points           [40, 0; 0, -5]
    DstBlock         "Product1"
    DstPort          1
  }
  Line {
    SrcBlock         "const6"
    SrcPort          1
    Points           [15, 0; 0, -25]
    DstBlock         "Product1"
    DstPort          2
  }
  Line {
    SrcBlock         "Product1"
    SrcPort          1
    Points           [30, 0; 0, 10]
    DstBlock         "Divide1"
    DstPort          1
  }
  Line {
    SrcBlock         "const7"
    SrcPort          1
    Points           [25, 0; 0, -45]
    DstBlock         "Divide1"
    DstPort          2
  }
  Line {
    SrcBlock         "const8"
    SrcPort          1
    Points           [20, 0; 0, 35]
    DstBlock         "Product2"
    DstPort          1
  }
  Line {
    SrcBlock         "const10"
    SrcPort          1
    Points           [35, 0]
    DstBlock         "Product2"
    DstPort          3
```

```
}
Line {
  SrcBlock        "Embedded\nMATLAB Function1"
  SrcPort         1
  Points          [35, 0]
  DstBlock        "Product2"
  DstPort         4
}
Line {
  SrcBlock        "Product2"
  SrcPort         1
  Points          [0, 155; 10, 0]
  Branch {
    DstBlock            "Divide2"
    DstPort         1
  }
  Branch {
    Points          [0, -55]
    DstBlock            "Display3"
    DstPort         1
  }
}
Line {
  SrcBlock        "const11"
  SrcPort         1
  Points          [20, 0]
  DstBlock        "Divide2"
  DstPort         2
}
Line {
  SrcBlock        "Divide2"
  SrcPort         1
  Points          [0, -5]
  DstBlock        "Display2"
  DstPort         1
}
Line {
  SrcBlock        "Embedded\nMATLAB Function"
  SrcPort         1
  Points          [5, 0; 0, -15]
  DstBlock        "Product2"
  DstPort         2
}
Line {
  SrcBlock        "const9"
  SrcPort         1
  Points          [40, 0]
  DstBlock        "Embedded\nMATLAB Function1"
  DstPort         1
}
Line {
  SrcBlock        "Divide1"
  SrcPort         1
  Points          [25, 0; 0, 10; 5, 0]
  Branch {
    DstBlock            "Compare\nTo Constant"
    DstPort         1
```

```
    }
    Branch {
      Points          [0, 90; 10, 0]
      Branch {
      DstBlock            "Compare\nTo Constant1"
      DstPort         1
      }
      Branch {
      Points          [0, 40]
      DstBlock            "Product4"
      DstPort         2
      }
    }
  }
  Line {
    SrcBlock        "Compare\nTo Constant"
    SrcPort         1
    Points          [15, 0; 0, -5]
    DstBlock        "Product3"
    DstPort         1
  }
  Line {
    SrcBlock        "const13"
    SrcPort         1
    Points          [30, 0]
    DstBlock        "Product3"
    DstPort         2
  }
  Line {
    SrcBlock        "Product3"
    SrcPort         1
    Points          [25, 0; 0, -10; 10, 0]
    Branch {
      DstBlock            "Lookup Table"
      DstPort         1
    }
    Branch {
      Points          [0, -75]
      DstBlock            "Display4"
      DstPort         1
    }
  }
  Line {
    SrcBlock        "Compare\nTo Constant1"
    SrcPort         1
    Points          [5, 0; 0, 25]
    DstBlock        "Product4"
    DstPort         1
  }
  Line {
    SrcBlock        "Product4"
    SrcPort         1
    DstBlock        "Embedded\nMATLAB Function"
    DstPort         1
  }
  Line {
    SrcBlock        "Lookup Table"
```

```
    SrcPort         1
    DstBlock        "Product5"
    DstPort         1
  }
  Line {
    SrcBlock        "const14"
    SrcPort         1
    Points          [5, 0; 0, -20]
    DstBlock        "Product5"
    DstPort         2
  }
  Line {
    SrcBlock        "Product5"
    SrcPort         1
    Points          [25, 0; 0, 25]
    DstBlock        "Divide3"
    DstPort         1
  }
  Line {
    SrcBlock        "const15"
    SrcPort         1
    Points          [15, 0; 0, -40]
    DstBlock        "Divide3"
    DstPort         2
  }
  Line {
    SrcBlock        "Divide3"
    SrcPort         1
    Points          [5, 0]
    Branch {
      DstBlock            "Out1"
      DstPort         1
    }
    Branch {
      Points          [0, -45]
      DstBlock            "Display1"
      DstPort         1
    }
  }
    }
  }
  Block {
    BlockType           Outport
    Name            "Out1"
    Position            [1080, 308, 1110, 322]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Line {
    SrcBlock            "const7"
    SrcPort             1
    Points              [20, 0; 0, 35]
    DstBlock            "Divide"
    DstPort             1
  }
  Line {
```

```
    SrcBlock                "hi"
    SrcPort                 1
    Points                  [25, 0; 0, -85]
    DstBlock                "Divide"
    DstPort                 2
  }
  Line {
    SrcBlock                "ho"
    SrcPort                 1
    Points                  [20, 0; 0, -100]
    DstBlock                "Divide1"
    DstPort                 2
  }
  Line {
    SrcBlock                "const1"
    SrcPort                 1
    Points                  [25, 0; 0, 10]
    DstBlock                "Divide1"
    DstPort                 1
  }
  Line {
    SrcBlock                "Divide"
    SrcPort                 1
    Points                  [169, 0]
    DstBlock                "Sum"
    DstPort                 1
  }
  Line {
    SrcBlock                "Divide1"
    SrcPort                 1
    Points                  [90, 0; 0, -130]
    DstBlock                "Sum"
    DstPort                 2
  }
  Line {
    SrcBlock                "Fouling"
    SrcPort                 1
    Points                  [430, 0; 0, -155; 80, 0]
    DstBlock                "Sum"
    DstPort                 3
  }
  Line {
    SrcBlock                "const2"
    SrcPort                 1
    Points                  [20, 0; 0, 55]
    DstBlock                "Divide2"
    DstPort                 1
  }
  Line {
    SrcBlock                "Sum"
    SrcPort                 1
    Points                  [45, 0; 0, -20]
    DstBlock                "Divide2"
    DstPort                 2
  }
  Line {
    SrcBlock                "Divide2"
```

```
      SrcPort             1
      Points              [65, 0; 0, 70]
      DstBlock            "Product1"
      DstPort             1
    }
    Line {
      SrcBlock            "Surface Area"
      SrcPort             1
      Points              [55, 0; 0, -5]
      DstBlock            "Product1"
      DstPort             2
    }
    Line {
      SrcBlock            "Product1"
      SrcPort             1
      Points              [20, 0; 0, 35]
      DstBlock            "Divide3"
      DstPort             1
    }
    Line {
      SrcBlock            "C_min"
      SrcPort             1
      Points              [10, 0; 0, -60]
      DstBlock            "Divide3"
      DstPort             2
    }
    Line {
      SrcBlock            "Divide3"
      SrcPort             1
      Points              [25, 0]
      Branch {
    DstBlock        "Out1"
    DstPort         1
      }
      Branch {
    Points          [-5, 0]
    DstBlock        "NTU"
    DstPort         1
      }
    }
    Line {
      SrcBlock            "Water Flowrate"
      SrcPort             1
      Points              [5, 0; 0, 165]
      DstBlock            "hi"
      DstPort             1
    }
  }
}
Block {
  BlockType       Product
  Name            "Product1"
  Ports           [2, 1]
  Position        [635, 171, 675, 229]
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
```

```
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType       SubSystem
    Name            "Q_max"
    Ports           [2, 1]
    Position        [390, 303, 520, 367]
    TreatAsAtomicUnit   on
    MinAlgLoopOccurrences   off
    RTWSystemCode       "Auto"
    FunctionWithSeparateData off
    System {
      Name            "Q_max"
      Location            [2, 82, 1270, 754]
      Open            off
      ModelBrowserVisibility  off
      ModelBrowserWidth       200
      ScreenColor         "white"
      PaperOrientation        "landscape"
      PaperPositionMode       "auto"
      PaperType           "usletter"
      PaperUnits          "inches"
      TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
      TiledPageScale      1
      ShowPageBoundaries      off
      ZoomFactor          "100"
      Block {
        BlockType           Inport
        Name            "T_Oil_in"
        Position            [320, 253, 350, 267]
        IconDisplay         "Port number"
        OutDataType         "sfix(16)"
        OutScaling          "2^0"
      }
      Block {
        BlockType           Inport
        Name            "T_Water_in"
        Position            [325, 328, 355, 342]
        Port            "2"
        IconDisplay         "Port number"
        OutDataType         "sfix(16)"
        OutScaling          "2^0"
      }
      Block {
        BlockType           Constant
        Name            "C_min"
        Position            [370, 141, 395, 169]
        Value           "1.704"
        OutDataType         "sfix(16)"
        OutScaling          "2^0"
      }
      Block {
        BlockType           Display
        Name            "Max_Q"
        Ports           [1]
```

```
    Position            [700, 95, 790, 125]
    Decimation          "1"
    Lockdown            off
  }
  Block {
    BlockType           Product
    Name            "Product1"
    Ports           [2, 1]
    Position            [580, 151, 620, 209]
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Sum
    Name            "Sum"
    Ports           [2, 1]
    Position            [475, 245, 515, 295]
    ShowName            off
    IconShape           "round"
    Inputs              "|+-"
    CollapseMode        "All dimensions"
    InputSameDT         off
    OutDataTypeMode         "Inherit via internal rule"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Outport
    Name            "Out1"
    Position            [690, 173, 720, 187]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Line {
    SrcBlock            "C_min"
    SrcPort             1
    Points              [15, 0; 0, 10]
    DstBlock            "Product1"
    DstPort             1
  }
  Line {
    SrcBlock            "Sum"
    SrcPort             1
    Points              [20, 0; 0, -75]
    DstBlock            "Product1"
    DstPort             2
  }
  Line {
    SrcBlock            "Product1"
    SrcPort             1
    Points              [25, 0]
```

```
      Branch {
    DstBlock        "Out1"
    DstPort         1
      }
      Branch {
    Points          [0, -70]
    DstBlock        "Max_Q"
    DstPort         1
      }
    }
    Line {
      SrcBlock            "T_Oil_in"
      SrcPort             1
      Points              [50, 0; 0, 10]
      DstBlock            "Sum"
      DstPort             1
    }
    Line {
      SrcBlock            "T_Water_in"
      SrcPort             1
      Points              [135, 0]
      DstBlock            "Sum"
      DstPort             2
    }
  }
}
Block {
  BlockType       Sum
  Name            "Sum"
  Ports           [2, 1]
  Position        [840, 202, 880, 258]
  ShowName        off
  Inputs          "|++"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Sum
  Name            "Sum1"
  Ports           [2, 1]
  Position        [855, 467, 895, 523]
  ShowName        off
  Inputs          "|-+"
  CollapseMode        "All dimensions"
  InputSameDT         off
  OutDataTypeMode     "Inherit via internal rule"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType       Constant
  Name            "T_Water_in"
```

```
    Position          [265, 381, 290, 409]
    Value             "35"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Display
    Name              "Temp_Oil_Out"
    Ports             [1]
    Position          [1010, 215, 1100, 245]
    Decimation            "1"
    Lockdown          off
  }
  Block {
    BlockType         Display
    Name              "Temp_Water_Out"
    Ports             [1]
    Position          [1025, 520, 1115, 550]
    Decimation            "1"
    Lockdown          off
  }
  Block {
    BlockType         Constant
    Name              "c1"
    Position          [720, 221, 745, 249]
    Value             "2.09"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Constant
    Name              "c2"
    Position          [375, 196, 400, 224]
    Value             "0.815"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Constant
    Name              "c3"
    Position          [730, 501, 755, 529]
    Value             "1.704"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Outport
    Name              "Temp Oil Out"
    Position          [1000, 163, 1030, 177]
    IconDisplay           "Port number"
    OutDataType           "sfix(16)"
    OutScaling            "2^0"
  }
  Block {
    BlockType         Outport
    Name              "Temp Water Out"
    Position          [1025, 458, 1055, 472]
```

```
  Port              "2"
  IconDisplay           "Port number"
  OutDataType           "sfix(16)"
  OutScaling            "2^0"
}
Line {
  SrcBlock          "c2"
  SrcPort           1
  Points            [20, 0; 0, -35]
  DstBlock          "Lookup\nTable (2-D)1"
  DstPort           2
}
Line {
  SrcBlock          "NTU value"
  SrcPort           1
  Points            [35, 0; 0, 45]
  DstBlock          "Lookup\nTable (2-D)1"
  DstPort           1
}
Line {
  SrcBlock          "Lookup\nTable (2-D)1"
  SrcPort           1
  Points            [45, 0; 0, 20]
  DstBlock          "Product1"
  DstPort           1
}
Line {
  SrcBlock          "Q_max"
  SrcPort           1
  Points            [45, 0; 0, -35]
  Branch {
    Points            [0, -85]
    DstBlock              "Product1"
    DstPort           2
  }
  Branch {
    Points            [0, 5]
    DstBlock              "Display2"
    DstPort           1
  }
}
Line {
  SrcBlock          "Oil Temp"
  SrcPort           1
  Points            [235, 0]
  Branch {
    Points            [20, 0; 0, -5]
    DstBlock              "Q_max"
    DstPort           1
  }
  Branch {
    Points            [0, 190]
    DstBlock              "Sum1"
    DstPort           2
  }
}
Line {
```

```
  SrcBlock        "T_Water_in"
  SrcPort         1
  Points          [40, 0; 0, -5]
  Branch {
    Points          [0, -40]
    DstBlock            "Q_max"
    DstPort         2
  }
  Branch {
    Points          [480, 0; 0, -140]
    DstBlock            "Sum"
    DstPort         2
  }
}
Line {
  SrcBlock        "Product1"
  SrcPort         1
  Points          [25, 0]
  Branch {
    Points          [15, 0; 0, 5]
    DstBlock            "Divide2"
    DstPort         1
  }
  Branch {
    Points          [0, 255]
    DstBlock            "Divide1"
    DstPort         1
  }
}
Line {
  SrcBlock        "c1"
  SrcPort         1
  Points          [5, 0; 0, -15]
  DstBlock        "Divide2"
  DstPort         2
}
Line {
  SrcBlock        "Divide2"
  SrcPort         1
  Points          [15, 0]
  DstBlock        "Sum"
  DstPort         1
}
Line {
  SrcBlock        "c3"
  SrcPort         1
  Points          [0, -45]
  DstBlock        "Divide1"
  DstPort         2
}
Line {
  SrcBlock        "Divide1"
  SrcPort         1
  Points          [10, 0; 0, 30]
  DstBlock        "Sum1"
  DstPort         1
}
```

```
Line {
  SrcBlock        "Water Flowrate"
  SrcPort         1
  Points          [10, 0; 0, 5]
  DstBlock        "NTU value"
  DstPort         1
}
Line {
  SrcBlock        "Fouling"
  SrcPort         1
  DstBlock        "NTU value"
  DstPort         2
}
Line {
  SrcBlock        "Sum1"
  SrcPort         1
  Points          [60, 0; 0, -265]
  Branch {
    Points          [0, -60]
    DstBlock            "Temp Oil Out"
    DstPort         1
  }
  Branch {
    DstBlock            "Temp_Oil_Out"
    DstPort         1
  }
}
Line {
  SrcBlock        "Sum"
  SrcPort         1
  Points          [30, 0; 0, 235; 70, 0]
  Branch {
    DstBlock            "Temp Water Out"
    DstPort         1
  }
  Branch {
    Points          [0, 70]
    DstBlock            "Temp_Water_Out"
    DstPort         1
  }
}
  }
}
Block {
  BlockType           SubSystem
  Name            "Bypass"
  Ports           [2, 1]
  Position            [870, 353, 920, 397]
  ShowPortLabels          "none"
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  MaskType            "Valve"
  MaskDescription         "Valve"
  MaskPromptString        "Initial position [0=closed 1=open]:"
  MaskStyleString         "edit"
  MaskTunableValueString  "on"
```

```
      MaskEnableString        "on"
      MaskVisibilityString    "on"
      MaskToolTipString       "on"
      MaskInitialization      "InitialPosition = @1;"
      MaskDisplay         "disp('VALVE')"
      MaskIconFrame       on
      MaskIconOpaque          on
      MaskIconRotate          "none"
      MaskIconUnits       "autoscale"
      MaskValueString         "0"
      System {
    Name            "Bypass"
    Location        [408, 498, 736, 676]
    Open            off
    ModelBrowserVisibility  off
    ModelBrowserWidth   200
    ScreenColor     "white"
    PaperOrientation    "landscape"
    PaperPositionMode   "auto"
    PaperType       "usletter"
    PaperUnits      "inches"
    TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
    TiledPageScale      1
    ShowPageBoundaries  off
    ZoomFactor      "100"
    Block {
      BlockType       Inport
      Name            "control"
      Position        [20, 20, 40, 40]
      IconDisplay         "Port number"
      OutDataType         "sfix(16)"
      OutScaling          "2^0"
    }
    Block {
      BlockType       Inport
      Name            "source flow"
      Position        [20, 110, 40, 130]
      Port            "2"
      IconDisplay         "Port number"
      OutDataType         "sfix(16)"
      OutScaling          "2^0"
    }
    Block {
      BlockType       SubSystem
      Name            "Limited Integrator"
      Ports           [1, 1]
      Position        [80, 9, 130, 51]
      ShowPortLabels      "none"
      MinAlgLoopOccurrences   off
      RTWSystemCode       "Auto"
      FunctionWithSeparateData off
      MaskType        "Limited integrator."
      MaskDescription     "Limited integrator"
      MaskHelp        "Implements a limited integrator with \\nan expression
of the form:\\n\\nif(x<=lb and u<0) or (x>=ub and
u>0)\\n\\txdot=0\\nelse\\n\\txdot=u."
      MaskPromptString    "Lower bound:|Upper bound:|Initial condition"
```

```
      MaskStyleString     "edit,edit,edit"
      MaskTunableValueString  "on,on,on"
      MaskCallbackString      "||"
      MaskEnableString    "on,on,on"
      MaskVisibilityString    "on,on,on"
      MaskToolTipString   "on,on,on"
      MaskVarAliasString      ",,"
      MaskInitialization      "lb = @1; ub = @2; xi = @3;"
      MaskDisplay         "plot(-1,-
0.2,3.5,1.2,[0.05,1,2,2.9],[0,0,1,1]);disp(' 1/s    ')"
      MaskIconFrame       on
      MaskIconOpaque      on
      MaskIconRotate      "none"
      MaskIconUnits       "autoscale"
      MaskValueString     "0|1|InitialPosition"
      MaskTabNameString   ",,"
      System {
        Name            "Limited Integrator"
        Location            [0, 82, 791, 315]
        Open            off
        ModelBrowserVisibility  off
        ModelBrowserWidth       200
        ScreenColor         "white"
        PaperOrientation        "landscape"
        PaperPositionMode       "auto"
        PaperType           "usletter"
        PaperUnits          "inches"
        TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
        TiledPageScale      1
        ShowPageBoundaries      off
        ZoomFactor          "100"
        Block {
          BlockType           Inport
          Name            "In_1"
          Position            [20, 130, 40, 150]
          IconDisplay         "Port number"
          OutDataType         "sfix(16)"
          OutScaling          "2^0"
        }
        Block {
          BlockType           Fcn
          Name            "Fcn"
          Position            [150, 116, 520, 154]
          Expr            "u[2]*((((u[1]>lb)+(u[2]>=0))>0) *
(((u[1]<ub)+(u[2]<=0))>0))"
        }
        Block {
          BlockType           Integrator
          Name            "Integrator"
          Ports           [1, 1]
          Position            [540, 126, 585, 144]
          InitialCondition        "xi"
        }
        Block {
          BlockType           Mux
          Name            "Mux"
          Ports           [2, 1]
```

```
  Position            [85, 116, 115, 149]
  Inputs              "2"
}
Block {
  BlockType           Saturate
  Name            "Saturation"
  Position            [630, 121, 675, 149]
  UpperLimit          "ub"
  LowerLimit          "lb"
}
Block {
  BlockType           Outport
  Name            "Out_1"
  Position            [720, 125, 740, 145]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  InitialOutput       "0"
}
Line {
  SrcBlock            "Integrator"
  SrcPort             1
  Points              [10, 0]
  Branch {
DstBlock        "Saturation"
DstPort         1
  }
  Branch {
Points          [0, -100; -545, 0; 0, 90]
DstBlock        "Mux"
DstPort         1
  }
}
Line {
  SrcBlock            "Fcn"
  SrcPort             1
  DstBlock            "Integrator"
  DstPort             1
}
Line {
  SrcBlock            "Mux"
  SrcPort             1
  DstBlock            "Fcn"
  DstPort             1
}
Line {
  SrcBlock            "In_1"
  SrcPort             1
  DstBlock            "Mux"
  DstPort             2
}
Line {
  SrcBlock            "Saturation"
  SrcPort             1
  DstBlock            "Out_1"
  DstPort             1
}
```

```
    }
  }
  Block {
    BlockType       Product
    Name            "Product"
    Ports           [2, 1]
    Position        [180, 76, 215, 134]
    CollapseMode        "All dimensions"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType       Outport
    Name            "out flow"
    Position        [250, 95, 270, 115]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    InitialOutput       "0"
  }
  Line {
    SrcBlock        "source flow"
    SrcPort         1
    DstBlock        "Product"
    DstPort         2
  }
  Line {
    SrcBlock        "Product"
    SrcPort         1
    DstBlock        "out flow"
    DstPort         1
  }
  Line {
    SrcBlock        "control"
    SrcPort         1
    DstBlock        "Limited Integrator"
    DstPort         1
  }
  Line {
    SrcBlock        "Limited Integrator"
    SrcPort         1
    Points          [25, 0; 0, 60]
    DstBlock        "Product"
    DstPort         1
  }
    }
  }
  Block {
    BlockType           Reference
    Name            "Clean SV. Equipment can fail Anytime"
    Ports           [1]
    Position            [1180, 580, 1240, 640]
    UserDataPersistent      on
    UserData            "DataTag0"
    SourceBlock         "gaugeslibv2/ActiveX\nControl"
    SourceType          "ActiveX Block"
    progid              "mwtoggle.togglectrl.1"
```

```
      connect             "input"
      input           "Value"
      init            "hActx.configuration='Toggle Switch\\Bitmap 
Toggles\\Light Bulb';"
      inblock             on
      border              on
      updateParam         "0"
    }
    Block {
      BlockType           Step
      Name            "Clean bypass line"
      Position            [80, 375, 110, 405]
      Time            "0"
      After           "450"
      SampleTime          "0.1"
    }
    Block {
      BlockType           Reference
      Name            "Compare\nTo Constant"
      Ports           [1, 1]
      Position            [865, 815, 895, 845]
      SourceBlock         "simulink/Logic and Bit\nOperations/Compare\nTo 
Constant"
      SourceType          "Compare To Constant"
      ShowPortLabels          "FromPortIcon"
      SystemSampleTime        "-1"
      FunctionWithSeparateData off
      RTWMemSecFuncInitTerm   "Inherit from model"
      RTWMemSecFuncExecute    "Inherit from model"
      RTWMemSecDataConstants  "Inherit from model"
      RTWMemSecDataInternal   "Inherit from model"
      RTWMemSecDataParameters "Inherit from model"
      relop           ">"
      const           "100"
      LogicOutDataTypeMode    "boolean"
      ZeroCross           off
    }
    Block {
      BlockType           TransferFcn
      Name            "Compressor"
      Position            [1030, 272, 1090, 308]
      Denominator         "[1 0]"
    }
    Block {
      BlockType           DataTypeConversion
      Name            "Data Type Conversion"
      Position            [970, 838, 1045, 872]
      OutDataTypeMode         "double"
      OutDataType         "sfix(16)"
      OutScaling          "2^0"
      RndMeth             "Floor"
      SaturateOnIntegerOverflow off
    }
    Block {
      BlockType           DataTypeConversion
      Name            "Data Type Conversion1"
      Position            [850, 448, 925, 482]
```

```
  OutDataTypeMode         "Specify via dialog"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           DataTypeConversion
  Name            "Data Type Conversion2"
  Position            [975, 518, 1050, 552]
  OutDataTypeMode         "double"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           DataTypeConversion
  Name            "Data Type Conversion3"
  Position            [1000, 593, 1075, 627]
  OutDataTypeMode         "double"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           DataTypeConversion
  Name            "Data Type Conversion4"
  Position            [1005, 678, 1080, 712]
  OutDataTypeMode         "double"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           DataTypeConversion
  Name            "Data Type Conversion5"
  Position            [860, 678, 935, 712]
  OutDataTypeMode         "Specify via dialog"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           DataTypeConversion
  Name            "Data Type Conversion7"
  Position            [970, 448, 1045, 482]
  OutDataTypeMode         "double"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType           DataTypeConversion
  Name            "Data Type Conversion8"
  Position            [865, 603, 940, 637]
  OutDataTypeMode         "Specify via dialog"
  RndMeth             "Floor"
  SaturateOnIntegerOverflow off
}
```

```
Block {
  BlockType            DataTypeConversion
  Name             "Data Type Conversion9"
  Position             [860, 523, 935, 557]
  OutDataTypeMode          "Specify via dialog"
  RndMeth              "Floor"
  SaturateOnIntegerOverflow off
}
Block {
  BlockType            Display
  Name             "Display"
  Ports            [1]
  Position             [635, 85, 725, 115]
  Decimation           "1"
  Lockdown             off
}
Block {
  BlockType            Display
  Name             "Display1"
  Ports            [1]
  Position             [680, 415, 770, 445]
  Decimation           "1"
  Lockdown             off
}
Block {
  BlockType            Display
  Name             "Display2"
  Ports            [1]
  Position             [715, 330, 805, 360]
  Decimation           "1"
  Lockdown             off
}
Block {
  BlockType            Display
  Name             "Display4"
  Ports            [1]
  Position             [765, 15, 855, 45]
  Decimation           "1"
  Lockdown             off
}
Block {
  BlockType            Display
  Name             "Fouling"
  Ports            [1]
  Position             [1000, 15, 1090, 45]
  Decimation           "1"
  Lockdown             off
}
Block {
  BlockType            Lookup
  Name             "Fouling Lookup"
  Position             [820, 135, 870, 185]
  InputValues          "[150,200,250,300,400,500]"
  Table            "[0.1,0.05,0.01,0.005,0.001,0.0009]"
  OutDataType          "sfix(16)"
  OutScaling           "2^0"
  SaturateOnIntegerOverflow off
```

```
    }
    Block {
      BlockType           SubSystem
      Name            "Fuzzy Logic "
      Ports           [1, 5]
      Position            [680, 589, 785, 671]
      TreatAsAtomicUnit       on
      MinAlgLoopOccurrences   off
      RTWSystemCode       "Auto"
      FunctionWithSeparateData off
      System {
    Name            "Fuzzy Logic "
    Location        [177, 212, 1205, 779]
    Open            on
    ModelBrowserVisibility  off
    ModelBrowserWidth   200
    ScreenColor     "white"
    PaperOrientation    "landscape"
    PaperPositionMode   "auto"
    PaperType       "usletter"
    PaperUnits      "inches"
    TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
    TiledPageScale      1
    ShowPageBoundaries  off
    ZoomFactor      "100"
    Block {
      BlockType       Inport
      Name            "Line Pressure"
      Position        [110, 118, 140, 132]
      IconDisplay         "Port number"
      OutDataType         "sfix(16)"
      OutScaling          "2^0"
    }
    Block {
      BlockType       Demux
      Name            "Demux"
      Ports           [1, 5]
      Position        [405, 22, 415, 228]
      BackgroundColor     "black"
      ShowName        off
      Outputs         "5"
      DisplayOption       "bar"
    }
    Block {
      BlockType       Reference
      Name            "Fuzzy Logic \nController \nwith Ruleviewer"
      Ports           [1, 1]
      Position        [210, 100, 270, 150]
      SourceBlock         "fuzblock/Fuzzy Logic \nController \nwith
Ruleviewer"
      SourceType          "FIS"
      ShowPortLabels      "FromPortIcon"
      SystemSampleTime    "-1"
      FunctionWithSeparateData off
      RTWMemSecFuncInitTerm   "Inherit from model"
      RTWMemSecFuncExecute    "Inherit from model"
      RTWMemSecDataConstants  "Inherit from model"
```

```
    RTWMemSecDataInternal   "Inherit from model"
    RTWMemSecDataParameters "Inherit from model"
    fismatrix       "Experiment_3_Fis"
    Ts              "2"
  }
  Block {
    BlockType       Scope
    Name            "Scope1"
    Ports           [1]
    Position        [345, 19, 375, 51]
    Floating        off
    Location        [5, 52, 1285, 769]
    Open            off
    NumInputPorts       "1"
    List {
      ListType            AxesTitles
      axes1           "%<SignalLabel>"
    }
    SaveName        "ScopeData5"
    DataFormat          "StructureWithTime"
    SampleTime          "0"
  }
  Block {
    BlockType       Scope
    Name            "Scope7"
    Ports           [1]
    Position        [195, 39, 225, 71]
    Floating        off
    Location        [5, 52, 1285, 769]
    Open            off
    NumInputPorts       "1"
    List {
      ListType            AxesTitles
      axes1           "%<SignalLabel>"
    }
    SaveName        "ScopeData3"
    DataFormat          "StructureWithTime"
    SampleTime          "0"
  }
  Block {
    BlockType       Outport
    Name            "Output 1"
    Position        [535, 23, 565, 37]
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
    BlockType       Outport
    Name            "Output 2"
    Position        [535, 63, 565, 77]
    Port            "2"
    IconDisplay         "Port number"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
  }
  Block {
```

```
    BlockType        Outport
    Name             "Output 3"
    Position         [540, 103, 570, 117]
    Port             "3"
    IconDisplay          "Port number"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Block {
    BlockType        Outport
    Name             "Output 4"
    Position         [540, 153, 570, 167]
    Port             "4"
    IconDisplay          "Port number"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Block {
    BlockType        Outport
    Name             "Output 5"
    Position         [535, 198, 565, 212]
    Port             "5"
    IconDisplay          "Port number"
    OutDataType          "sfix(16)"
    OutScaling           "2^0"
  }
  Line {
    SrcBlock         "Line Pressure"
    SrcPort          1
    Points           [30, 0]
    Branch {
      DstBlock             "Fuzzy Logic \nController \nwith Ruleviewer"
      DstPort          1
    }
    Branch {
      Points           [0, -70]
      DstBlock             "Scope7"
      DstPort          1
    }
  }
  Line {
    SrcBlock         "Fuzzy Logic \nController \nwith Ruleviewer"
    SrcPort          1
    Points           [50, 0]
    Branch {
      DstBlock             "Demux"
      DstPort          1
    }
    Branch {
      Points           [0, -90]
      DstBlock             "Scope1"
      DstPort          1
    }
  }
  Line {
    SrcBlock         "Demux"
    SrcPort          1
```

```
    Points          [50, 0; 0, -15]
    DstBlock        "Output 1"
    DstPort         1
  }
  Line {
    SrcBlock        "Demux"
    SrcPort         2
    Points          [50, 0; 0, -15]
    DstBlock        "Output 2"
    DstPort         1
  }
  Line {
    SrcBlock        "Demux"
    SrcPort         3
    Points          [50, 0; 0, -15]
    DstBlock        "Output 3"
    DstPort         1
  }
  Line {
    SrcBlock        "Demux"
    SrcPort         4
    Points          [50, 0; 0, -5]
    DstBlock        "Output 4"
    DstPort         1
  }
  Line {
    SrcBlock        "Demux"
    SrcPort         5
    DstBlock        "Output 5"
    DstPort         1
  }
    }
  }
  Block {
    BlockType           Display
    Name            "Oil Temp in"
    Ports           [1]
    Position            [1130, 15, 1220, 45]
    Decimation          "1"
    Lockdown            off
  }
  Block {
    BlockType           Lookup
    Name            "Oil temp lookup"
    Position            [820, 210, 870, 260]
    InputValues         "[150,200,250,300,400,500]"
    Table           "[70,68,66,64,62,60]"
    OutDataType         "sfix(16)"
    OutScaling          "2^0"
    SaturateOnIntegerOverflow off
  }
  Block {
    BlockType           Backlash
    Name            "Pressure Sensor"
    Position            [580, 505, 610, 535]
    BacklashWidth       "0.01"
  }
```

```
Block {
  BlockType           SubSystem
  Name            "SV"
  Ports           [2, 1]
  Position            [545, 128, 595, 172]
  ShowPortLabels          "none"
  MinAlgLoopOccurrences   off
  RTWSystemCode       "Auto"
  FunctionWithSeparateData off
  MaskType            "Valve"
  MaskDescription         "Valve"
  MaskPromptString        "Initial position [0=closed 1=open]:"
  MaskStyleString         "edit"
  MaskTunableValueString  "on"
  MaskEnableString        "on"
  MaskVisibilityString    "on"
  MaskToolTipString       "on"
  MaskInitialization      "InitialPosition = @1;"
  MaskDisplay         "disp('VALVE')"
  MaskIconFrame       on
  MaskIconOpaque          on
  MaskIconRotate          "none"
  MaskIconUnits       "autoscale"
  MaskValueString         "0"
  System {
Name            "SV"
Location        [406, 466, 738, 696]
Open            off
ModelBrowserVisibility  off
ModelBrowserWidth   200
ScreenColor     "white"
PaperOrientation    "landscape"
PaperPositionMode   "auto"
PaperType       "usletter"
PaperUnits      "inches"
TiledPaperMargins   [0.500000, 0.500000, 0.500000, 0.500000]
TiledPageScale      1
ShowPageBoundaries  off
ZoomFactor      "100"
Block {
  BlockType       Inport
  Name            "control"
  Position        [20, 20, 40, 40]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Inport
  Name            "source flow"
  Position        [20, 110, 40, 130]
  Port            "2"
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
```

```
      BlockType       SubSystem
      Name            "Limited Integrator"
      Ports           [1, 1]
      Position        [80, 9, 130, 51]
      ShowPortLabels      "none"
      MinAlgLoopOccurrences   off
      RTWSystemCode       "Auto"
      FunctionWithSeparateData off
      MaskType        "Limited integrator."
      MaskDescription     "Limited integrator"
      MaskHelp        "Implements a limited integrator with \\nan expression
of the form:\\n\\nif(x<=lb and u<0) or (x>=ub and
u>0)\\n\\txdot=0\\nelse\\n\\txdot=u."
      MaskPromptString    "Lower bound:|Upper bound:|Initial condition"
      MaskStyleString     "edit,edit,edit"
      MaskTunableValueString  "on,on,on"
      MaskCallbackString      "||"
      MaskEnableString    "on,on,on"
      MaskVisibilityString    "on,on,on"
      MaskToolTipString   "on,on,on"
      MaskVarAliasString      ",,"
      MaskInitialization      "lb = @1; ub = @2; xi = @3;"
      MaskDisplay         "plot(-1,-
0.2,3.5,1.2,[0.05,1,2,2.9],[0,0,1,1]);disp(' 1/s    ')"
      MaskIconFrame       on
      MaskIconOpaque      on
      MaskIconRotate      "none"
      MaskIconUnits       "autoscale"
      MaskValueString     "0|1|InitialPosition"
      MaskTabNameString   ",,"
      System {
        Name            "Limited Integrator"
        Location            [0, 82, 791, 315]
        Open            off
        ModelBrowserVisibility  off
        ModelBrowserWidth       200
        ScreenColor         "white"
        PaperOrientation        "landscape"
        PaperPositionMode       "auto"
        PaperType           "usletter"
        PaperUnits          "inches"
        TiledPaperMargins       [0.500000, 0.500000, 0.500000, 0.500000]
        TiledPageScale      1
        ShowPageBoundaries      off
        ZoomFactor          "100"
        Block {
          BlockType           Inport
          Name            "In_1"
          Position            [20, 130, 40, 150]
          IconDisplay         "Port number"
          OutDataType         "sfix(16)"
          OutScaling          "2^0"
        }
        Block {
          BlockType           Fcn
          Name            "Fcn"
          Position            [150, 116, 520, 154]
```

```
      Expr             "u[2]*((((u[1]>lb)+(u[2]>=0))>0) *
(((u[1]<ub)+(u[2]<=0))>0))"
    }
    Block {
      BlockType            Integrator
      Name             "Integrator"
      Ports            [1, 1]
      Position             [540, 126, 585, 144]
      InitialCondition         "xi"
    }
    Block {
      BlockType            Mux
      Name             "Mux"
      Ports            [2, 1]
      Position             [85, 116, 115, 149]
      Inputs               "2"
    }
    Block {
      BlockType            Saturate
      Name             "Saturation"
      Position             [630, 121, 675, 149]
      UpperLimit           "ub"
      LowerLimit           "lb"
    }
    Block {
      BlockType            Outport
      Name             "Out_1"
      Position             [720, 125, 740, 145]
      IconDisplay          "Port number"
      OutDataType          "sfix(16)"
      OutScaling           "2^0"
      InitialOutput        "0"
    }
    Line {
      SrcBlock             "Integrator"
      SrcPort              1
      Points               [10, 0]
      Branch {
    DstBlock         "Saturation"
    DstPort          1
      }
      Branch {
    Points           [0, -100; -545, 0; 0, 90]
    DstBlock         "Mux"
    DstPort          1
      }
    }
    Line {
      SrcBlock             "Fcn"
      SrcPort              1
      DstBlock             "Integrator"
      DstPort              1
    }
    Line {
      SrcBlock             "Mux"
      SrcPort              1
      DstBlock             "Fcn"
```

```
      DstPort             1
    }
    Line {
      SrcBlock            "In_1"
      SrcPort             1
      DstBlock            "Mux"
      DstPort             2
    }
    Line {
      SrcBlock            "Saturation"
      SrcPort             1
      DstBlock            "Out_1"
      DstPort             1
    }
  }
}
Block {
  BlockType       Product
  Name            "Product"
  Ports           [2, 1]
  Position        [180, 76, 215, 134]
  CollapseMode        "All dimensions"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
}
Block {
  BlockType       Outport
  Name            "out flow"
  Position        [250, 95, 270, 115]
  IconDisplay         "Port number"
  OutDataType         "sfix(16)"
  OutScaling          "2^0"
  InitialOutput       "0"
}
Line {
  SrcBlock        "source flow"
  SrcPort         1
  DstBlock        "Product"
  DstPort         2
}
Line {
  SrcBlock        "Product"
  SrcPort         1
  DstBlock        "out flow"
  DstPort         1
}
Line {
  SrcBlock        "control"
  SrcPort         1
  DstBlock        "Limited Integrator"
  DstPort         1
}
Line {
  SrcBlock        "Limited Integrator"
  SrcPort         1
  Points          [25, 0; 0, 60]
  DstBlock        "Product"
```

```
      DstPort         1
    }
      }
    }
    Block {
      BlockType           Reference
      Name            "SV good"
      Ports           [1]
      Position            [1180, 420, 1240, 480]
      UserDataPersistent      on
      UserData            "DataTag1"
      SourceBlock         "gaugeslibv2/ActiveX\nControl"
      SourceType          "ActiveX Block"
      progid              "mwtoggle.togglectrl.1"
      connect             "input"
      input           "Value"
      init            "hActx.configuration='Toggle Switch\\Bitmap
Toggles\\Light Bulb';"
      inblock             on
      border              on
      updateParam         "0"
    }
    Block {
      BlockType           Reference
      Name            "SV likely plugged. Open Bypass"
      Ports           [1]
      Position            [1180, 660, 1240, 720]
      UserDataPersistent      on
      UserData            "DataTag2"
      SourceBlock         "gaugeslibv2/ActiveX\nControl"
      SourceType          "ActiveX Block"
      progid              "mwtoggle.togglectrl.1"
      connect             "input"
      input           "Value"
      init            "hActx.configuration='Toggle Switch\\Bitmap
Toggles\\Light Bulb';"
      inblock             on
      border              on
      updateParam         "0"
    }
    Block {
      BlockType           Reference
      Name            "SV needs proactive cleaning"
      Ports           [1]
      Position            [1185, 500, 1245, 560]
      UserDataPersistent      on
      UserData            "DataTag3"
      SourceBlock         "gaugeslibv2/ActiveX\nControl"
      SourceType          "ActiveX Block"
      progid              "mwtoggle.togglectrl.1"
      connect             "input"
      input           "Value"
      init            "hActx.configuration='Toggle Switch\\Bitmap
Toggles\\Light Bulb';"
      inblock             on
      border              on
      updateParam         "0"
```

```
}
Block {
  BlockType           Scope
  Name             "Scope"
  Ports            [1]
  Position            [1125, 274, 1155, 306]
  Floating            off
  Location            [5, 52, 1285, 769]
  Open             off
  NumInputPorts        "1"
  ZoomMode            "xonly"
  List {
ListType        AxesTitles
axes1           "%<SignalLabel>"
  }
  TimeRange           "150              "
  YMin             "620"
  YMax             "620.053"
  DataFormat          "StructureWithTime"
  LimitDataPoints         off
  SampleTime          "0"
}
Block {
  BlockType           Scope
  Name             "Scope1"
  Ports            [1]
  Position            [200, 89, 230, 121]
  Floating            off
  Location            [5, 52, 1285, 769]
  Open             off
  NumInputPorts        "1"
  ZoomMode            "xonly"
  List {
ListType        AxesTitles
axes1           "%<SignalLabel>"
  }
  TimeRange           "150              "
  YMin             "620"
  YMax             "620.053"
  SaveName            "ScopeData1"
  DataFormat          "StructureWithTime"
  LimitDataPoints         off
  SampleTime          "0"
}
Block {
  BlockType           Scope
  Name             "Scope6"
  Ports            [1]
  Position            [575, 209, 605, 241]
  Floating            off
  Location            [5, 52, 1285, 769]
  Open             off
  NumInputPorts        "1"
  ZoomMode            "xonly"
  List {
ListType        AxesTitles
axes1           "%<SignalLabel>"
```

```
  }
  TimeRange           "150              "
  YMin            "620"
  YMax            "620.053"
  SaveName            "ScopeData12"
  DataFormat          "StructureWithTime"
  LimitDataPoints         off
  SampleTime          "0"
}
Block {
  BlockType           Step
  Name            "Service Water Pressure"
  Position            [75, 140, 105, 170]
  Time            "0"
  After           "80"
  SampleTime          "0.1"
}
Block {
  BlockType           Sum
  Name            "Sum2"
  Ports           [2, 1]
  Position            [715, 125, 735, 165]
  CollapseMode        "All dimensions"
  OutDataTypeMode         "Inherit via internal rule"
}
Block {
  BlockType           Switch
  Name            "Switch1"
  Position            [460, 187, 480, 243]
  Threshold           "200"
}
Block {
  BlockType           Switch
  Name            "Switch2"
  Position            [825, 382, 845, 438]
  Threshold           "1"
}
Block {
  BlockType           Display
  Name            "Temp Oil Out"
  Ports           [1]
  Position            [1150, 75, 1240, 105]
  Decimation          "1"
  Lockdown            off
}
Block {
  BlockType           Display
  Name            "Temp Water Out"
  Ports           [1]
  Position            [1150, 140, 1240, 170]
  Decimation          "1"
  Lockdown            off
}
Block {
  BlockType           Reference
  Name            "To Audio\nDevice"
  Ports           [1]
```

```
      Position             [1190, 819, 1245, 871]
      DialogController        "dspDDGCreate"
      DialogControllerArgs    "DataTag4"
      SourceBlock          "dspsnks4/To Audio\nDevice"
      SourceType           "To Audio Device"
      deviceName           "Default"
      inheritSampleRate       off
      sampleRate           "8000"
      deviceDatatype          "Determine from input data type"
      autoBufferSize          on
      bufferSize           "4096"
      queueDuration        "4"
    }
    Block {
      BlockType            Reference
      Name             "URGENT: SV plugged. Open Bypass (by logic)"
      Ports            [1]
      Position             [1180, 735, 1240, 795]
      UserDataPersistent      on
      UserData             "DataTag5"
      SourceBlock          "gaugeslibv2/ActiveX\nControl"
      SourceType           "ActiveX Block"
      progid               "mwtoggle.togglectrl.1"
      connect              "input"
      input            "Value"
      init             "hActx.configuration='Toggle Switch\\Bitmap
Toggles\\Light Bulb';"
      inblock              on
      border               on
      updateParam          "0"
    }
    Block {
      BlockType            Display
      Name             "Water Flow Rate in"
      Ports            [1]
      Position             [890, 15, 980, 45]
      Decimation           "1"
      Lockdown             off
    }
    Block {
      BlockType            Lookup
      Name             "Water flowrate lookup"
      Position             [825, 60, 875, 110]
      InputValues          "[150,200,250,300,400,500]"
      Table            "[0.05,0.1,0.2,0.3,0.4,0.5]"
      OutDataType          "sfix(16)"
      OutScaling           "2^0"
      SaturateOnIntegerOverflow off
    }
    Block {
      BlockType            Constant
      Name             "const1"
      Position             [350, 206, 375, 234]
      Value            "600"
      OutDataType          "sfix(16)"
      OutScaling           "2^0"
    }
```

```
Line {
  SrcBlock              "Compressor"
  SrcPort               1
  DstBlock              "Scope"
  DstPort               1
}
Line {
  SrcBlock              "const1"
  SrcPort               1
  Points                [30, 0; 0, -5]
  DstBlock              "Switch1"
  DstPort               2
}
Line {
  SrcBlock              "Switch1"
  SrcPort               1
  Points                [0, -35; -35, 0; 0, -20; 25, 0]
  Branch {
Points          [15, 0]
Branch {
  Points          [20, 0]
  Branch {
    DstBlock              "SV"
    DstPort         2
  }
  Branch {
    Points          [0, 65]
    DstBlock              "Scope6"
    DstPort         1
  }
}
Branch {
  Points          [0, -20]
  DstBlock        "SV"
  DstPort         1
}
  }
  Branch {
Points          [0, -60; 140, 0]
Branch {
  DstBlock        "Display"
  DstPort         1
}
Branch {
  Points          [0, 390; -50, 0]
  DstBlock        "Pressure Sensor"
  DstPort         1
}
  }
}
Line {
  SrcBlock              "Fuzzy Logic "
  SrcPort               5
  Points                [10, 0; 0, 170]
  DstBlock              "Compare\nTo Constant"
  DstPort               1
}
```

```
Line {
  SrcBlock              "SV"
  SrcPort               1
  Points                [40, 0; 0, -15]
  DstBlock              "Sum2"
  DstPort               1
}
Line {
  SrcBlock              "Bypass"
  SrcPort               1
  Points                [40, 0; 0, -65; -265, 0]
  Branch {
Points          [-30, 0; 0, -155]
DstBlock        "Sum2"
DstPort         2
  }
  Branch {
Points          [-5, 0; 0, 35]
DstBlock        "Display2"
DstPort         1
  }
}
Line {
  SrcBlock              "Service Water Pressure"
  SrcPort               1
  Points                [55, 0]
  Branch {
Points          [0, -50]
DstBlock        "Scope1"
DstPort         1
  }
  Branch {
Points          [110, 0; 0, 40; 40, 0]
Branch {
  DstBlock        "Switch1"
  DstPort         1
}
Branch {
  Points          [0, 65; 130, 0]
  DstBlock        "Switch1"
  DstPort         3
}
  }
}
Line {
  SrcBlock              "Compare\nTo Constant"
  SrcPort               1
  Points                [0, 25]
  DstBlock              "Data Type Conversion"
  DstPort               1
}
Line {
  SrcBlock              "Fuzzy Logic "
  SrcPort               1
  Points                [45, 0]
  DstBlock              "Data Type Conversion1"
  DstPort               1
```

```
}
Line {
  SrcBlock            "Switch2"
  SrcPort             1
  Points              [0, -25]
  Branch {
Points          [0, -20]
DstBlock        "Bypass"
DstPort         1
  }
  Branch {
DstBlock        "Bypass"
DstPort         2
  }
}
Line {
  SrcBlock            "Data Type Conversion"
  SrcPort             1
  Points              [0, 25; 25, 0]
  Branch {
Points          [25, 0]
Branch {
  Points          [0, -35]
  DstBlock        "To Audio\nDevice"
  DstPort         1
}
Branch {
  Points          [40, 0; 0, -385; -345, 0]
  Branch {
    Points          [0, -30; 10, 0]
    Branch {
      DstBlock            "Switch2"
      DstPort             3
    }
    Branch {
      Points              [0, -55]
      DstBlock            "Switch2"
      DstPort             2
    }
  }
  Branch {
    Points          [-120, 0; 0, -65]
    DstBlock            "Display1"
    DstPort         1
  }
}
  }
  Branch {
Points          [0, -115]
DstBlock        "URGENT: SV plugged. Open Bypass (by logic)"
DstPort         1
  }
}
Line {
  SrcBlock            "Fuzzy Logic "
  SrcPort             4
  Points              [25, 0; 0, 50]
```

```
    DstBlock              "Data Type Conversion5"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion5"
    SrcPort               1
    DstBlock              "Data Type Conversion4"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion4"
    SrcPort               1
    Points                [0, -5]
    DstBlock              "SV likely plugged. Open Bypass"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion1"
    SrcPort               1
    DstBlock              "Data Type Conversion7"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion7"
    SrcPort               1
    Points                [115, 0]
    DstBlock              "SV good"
    DstPort               1
  }
  Line {
    SrcBlock              "Fuzzy Logic "
    SrcPort               3
    Points                [60, 0]
    DstBlock              "Data Type Conversion8"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion8"
    SrcPort               1
    Points                [40, 0]
    DstBlock              "Data Type Conversion3"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion3"
    SrcPort               1
    DstBlock              "Clean SV. Equipment can fail Anytime"
    DstPort               1
  }
  Line {
    SrcBlock              "Fuzzy Logic "
    SrcPort               2
    Points                [55, 0]
    DstBlock              "Data Type Conversion9"
    DstPort               1
  }
  Line {
```

```
    SrcBlock              "Data Type Conversion9"
    SrcPort               1
    Points                [20, 0]
    DstBlock              "Data Type Conversion2"
    DstPort               1
  }
  Line {
    SrcBlock              "Data Type Conversion2"
    SrcPort               1
    Points                [115, 0]
    DstBlock              "SV needs proactive cleaning"
    DstPort               1
  }
  Line {
    SrcBlock              "Air Compressor"
    SrcPort               1
    DstBlock              "Temp Oil Out"
    DstPort               1
  }
  Line {
    SrcBlock              "Air Compressor"
    SrcPort               2
    DstBlock              "Temp Water Out"
    DstPort               1
  }
  Line {
    SrcBlock              "Water flowrate lookup"
    SrcPort               1
    Points                [20, 0]
    Branch {
  Points          [20, 0; 0, -5]
  DstBlock        "Air Compressor"
  DstPort         1
    }
    Branch {
  Points          [-25, 0]
  DstBlock        "Water Flow Rate in"
  DstPort         1
    }
  }
  Line {
    SrcBlock              "Fouling Lookup"
    SrcPort               1
    Points                [40, 0; 0, -40; 35, 0]
    Branch {
  DstBlock        "Air Compressor"
  DstPort         2
    }
    Branch {
  Points          [0, -90]
  DstBlock        "Fouling"
  DstPort         1
    }
  }
  Line {
    SrcBlock              "Oil temp lookup"
    SrcPort               1
```

```
  Points              [85, 0; 0, -20]
  Branch {
DstBlock        "Air Compressor"
DstPort         3
  }
  Branch {
Points          [360, 0; 0, -210; -205, 0]
DstBlock        "Oil Temp in"
DstPort         1
  }
}
Line {
  SrcBlock            "Sum2"
  SrcPort             1
  Points              [5, 0]
  Branch {
Points          [50, 0]
Branch {
  Points          [15, 0]
  DstBlock        "Water flowrate lookup"
  DstPort         1
}
Branch {
  Points          [0, 15]
  Branch {
    DstBlock            "Fouling Lookup"
    DstPort         1
  }
  Branch {
    Points          [0, 75]
    DstBlock            "Oil temp lookup"
    DstPort         1
  }
}
Branch {
  Points          [-45, 0]
  DstBlock        "Display4"
  DstPort         1
}
  }
  Branch {
Points          [0, 145]
DstBlock        "Compressor"
DstPort         1
  }
}
Line {
  SrcBlock            "Pressure Sensor"
  SrcPort             1
  Points              [50, 0]
  DstBlock            "Fuzzy Logic "
  DstPort             1
}
Line {
  SrcBlock            "Clean bypass line"
  SrcPort             1
  DstBlock            "Switch2"
```

```
      DstPort             1
    }
  }
}
# Finite State Machines
#
#    Stateflow Version 6.7 (R2007b) dated Aug  7 2007, 16:48:14
#
#


Stateflow {
  machine {
    id                1
    name              "Experiment_6_matlab5"
    created           "19-Apr-2013 16:20:55"
    isLibrary             0
    firstTarget           30
    sfVersion             67014000.000001
  }
  chart {
    id                2
    name              "Air Compressor/NTU value/ho/Embedded\nMATLAB Function"
    windowPosition        [341.813 294 200.25 189.75]
    viewLimits            [0 156.75 0 153.75]
    screen            [1 1 1280 800 1.333333333333333]
    treeNode              [0 3 0 0]
    firstTransition       5
    firstJunction         4
    viewObj           2
    machine           1
    decomposition         CLUSTER_CHART
    type              EML_CHART
    firstData             6
    chartFileNumber       1
    disableImplicitCasting  1
    eml {
      name            "fcn"
    }
  }
  state {
    id                3
    labelString           "eML_blk_kernel()"
    position              [18 64.5 118 66]
    fontSize              12
    chart             2
    treeNode              [2 0 0 0]
    superState            SUBCHART
    subviewer             2
    type              FUNC_STATE
    decomposition         CLUSTER_STATE
    eml {
      isEML           1
      script              "function y = fcn(u)\n% This block supports the
Embedded MATLAB subset.\n% See the help menu for details. \n\ny = u^.8;"
      editorLayout        "100 M4x1[146 214 671 364]"
```

```
    }
  }
  junction {
    id              4
    position            [23.5747 49.5747 7]
    chart           2
    linkNode            [2  0  0]
    subviewer           2
    type            CONNECTIVE_JUNCTION
  }
  transition {
    id              5
    labelString         "{eML_blk_kernel();}"
    labelPosition       [32.125 19.875 102.544 14.964]
    fontSize            12
    src {
      intersection        [0 0 1 0 23.5747 14.625 0 0]
    }
    dst {
      id              4
      intersection        [7 0 -1 -1 23.5747 42.5747 0 0]
    }
    midPoint            [23.5747 24.9468]
    chart           2
    linkNode            [2  0  0]
    dataLimits          [23.575 23.575 14.625 34.575]
    subviewer           2
    drawStyle           SMART
    executionOrder      1
  }
  data {
    id              6
    name            "u"
    linkNode            [2  0  7]
    scope           INPUT_DATA
    machine         1
    props {
      array {
    size            "-1"
      }
      type {
    method          SF_INHERITED_TYPE
      }
      complexity          SF_COMPLEX_INHERITED
    }
    dataType            "inherited"
    fixptType {
      slope           1
      baseType            SF_INT16_TYPE
    }
    complexity          SF_COMPLEX_INHERITED
  }
  data {
    id              7
    name            "y"
    linkNode            [2  6  0]
    scope           OUTPUT_DATA
```

```
  machine         1
  props {
    array {
  size            "-1"
    }
    type {
  method          SF_INHERITED_TYPE
    }
    complexity          SF_COMPLEX_INHERITED
    frame           SF_FRAME_NO
  }
  dataType            "inherited"
  fixptType {
    slope           1
    baseType            SF_INT16_TYPE
  }
  complexity          SF_COMPLEX_INHERITED
}
instance {
  id              8
  name            "Air Compressor/NTU value/ho/Embedded\nMATLAB Function"
  machine         1
  chart           2
}
chart {
  id              9
  name            "Air Compressor/NTU value/hi/Embedded\nMATLAB Function"
  windowPosition      [341.813 294 200.25 189.75]
  viewLimits          [0 156.75 0 153.75]
  screen          [1 1 1280 800 1.333333333333333]
  treeNode            [0 10 0 0]
  firstTransition     12
  firstJunction       11
  viewObj         9
  machine         1
  decomposition       CLUSTER_CHART
  type            EML_CHART
  firstData           13
  chartFileNumber     2
  disableImplicitCasting  1
  eml {
    name            "fcn"
  }
}
state {
  id              10
  labelString         "eML_blk_kernel()"
  position            [18 64.5 118 66]
  fontSize            12
  chart           9
  treeNode            [9 0 0 0]
  superState          SUBCHART
  subviewer           9
  type            FUNC_STATE
  decomposition       CLUSTER_STATE
  eml {
    isEML           1
```

```
      script               "function y = fcn(u)\n% This block supports the
Embedded MATLAB subset.\n% See the help menu for details. \n\ny = u^.8;"
      editorLayout        "100 M4x1[146 214 671 364]"
    }
  }
  junction {
    id              11
    position            [23.5747 49.5747 7]
    chart           9
    linkNode            [9 0 0]
    subviewer           9
    type            CONNECTIVE_JUNCTION
  }
  transition {
    id              12
    labelString         "{eML_blk_kernel();}"
    labelPosition       [32.125 19.875 102.544 14.964]
    fontSize            12
    src {
      intersection        [0 0 1 0 23.5747 14.625 0 0]
    }
    dst {
      id              11
      intersection        [7 0 -1 -1 23.5747 42.5747 0 0]
    }
    midPoint            [23.5747 24.9468]
    chart           9
    linkNode            [9 0 0]
    dataLimits          [23.575 23.575 14.625 34.575]
    subviewer           9
    drawStyle           SMART
    executionOrder      1
  }
  data {
    id              13
    name            "u"
    linkNode            [9 0 14]
    scope           INPUT_DATA
    machine         1
    props {
      array {
    size            "-1"
      }
      type {
    method          SF_INHERITED_TYPE
      }
      complexity          SF_COMPLEX_INHERITED
    }
    dataType            "inherited"
    fixptType {
      slope           1
      baseType            SF_INT16_TYPE
    }
    complexity          SF_COMPLEX_INHERITED
  }
  data {
    id              14
```

```
    name            "y"
    linkNode            [9 13 0]
    scope           OUTPUT_DATA
    machine         1
    props {
      array {
    size            "-1"
      }
      type {
    method          SF_INHERITED_TYPE
      }
      complexity          SF_COMPLEX_INHERITED
      frame           SF_FRAME_NO
    }
    dataType            "inherited"
    fixptType {
      slope           1
      baseType            SF_INT16_TYPE
    }
    complexity          SF_COMPLEX_INHERITED
  }
  instance {
    id              15
    name            "Air Compressor/NTU value/hi/Embedded\nMATLAB Function"
    machine         1
    chart           9
  }
  chart {
    id              16
    name            "Air Compressor/NTU value/hi/Embedded\nMATLAB Function1"
    windowPosition      [356.813 279 200.25 189.75]
    viewLimits          [0 156.75 0 153.75]
    screen          [1 1 1280 800 1.333333333333333]
    treeNode            [0 17 0 0]
    firstTransition     19
    firstJunction       18
    viewObj         16
    machine         1
    decomposition       CLUSTER_CHART
    type            EML_CHART
    firstData           20
    chartFileNumber     3
    disableImplicitCasting  1
    eml {
      name            "fcn"
    }
  }
  state {
    id              17
    labelString         "eML_blk_kernel()"
    position            [18 64.5 118 66]
    fontSize            12
    chart           16
    treeNode            [16 0 0 0]
    superState          SUBCHART
    subviewer           16
    type            FUNC_STATE
```

```
    decomposition       CLUSTER_STATE
    eml {
      isEML           1
      script              "function y = fcn(u)\n% This block supports the
Embedded MATLAB subset.\n% See the help menu for details. \n\ny = u^.4;"
      editorLayout        "100 M4x1[146 214 671 364]"
    }
  }
  junction {
    id                18
    position              [23.5747 49.5747 7]
    chart             16
    linkNode              [16 0 0]
    subviewer             16
    type              CONNECTIVE_JUNCTION
  }
  transition {
    id                19
    labelString           "{eML_blk_kernel();}"
    labelPosition         [32.125 19.875 102.544 14.964]
    fontSize              12
    src {
      intersection        [0 0 1 0 23.5747 14.625 0 0]
    }
    dst {
      id              18
      intersection        [7 0 -1 -1 23.5747 42.5747 0 0]
    }
    midPoint              [23.5747 24.9468]
    chart             16
    linkNode              [16 0 0]
    dataLimits            [23.575 23.575 14.625 34.575]
    subviewer             16
    drawStyle             SMART
    executionOrder        1
  }
  data {
    id                20
    name              "u"
    linkNode              [16 0 21]
    scope             INPUT_DATA
    machine           1
    props {
      array {
    size              "-1"
      }
      type {
    method            SF_INHERITED_TYPE
      }
      complexity          SF_COMPLEX_INHERITED
    }
    dataType              "inherited"
    fixptType {
      slope           1
      baseType            SF_INT16_TYPE
    }
    complexity            SF_COMPLEX_INHERITED
```

```
}
data {
  id              21
  name            "y"
  linkNode            [16 20 0]
  scope           OUTPUT_DATA
  machine         1
  props {
    array {
  size            "-1"
    }
    type {
  method          SF_INHERITED_TYPE
    }
    complexity          SF_COMPLEX_INHERITED
    frame           SF_FRAME_NO
  }
  dataType            "inherited"
  fixptType {
    slope           1
    baseType            SF_INT16_TYPE
  }
  complexity          SF_COMPLEX_INHERITED
}
instance {
  id              22
  name            "Air Compressor/NTU value/hi/Embedded\nMATLAB Function1"
  machine         1
  chart           16
}
chart {
  id              23
  name            "Air Compressor/NTU value/ho/Embedded\nMATLAB Function1"
  windowPosition      [356.813 279 200.25 189.75]
  viewLimits          [0 156.75 0 153.75]
  screen          [1 1 1280 800 1.333333333333333]
  treeNode            [0 24 0 0]
  firstTransition     26
  firstJunction       25
  viewObj         23
  machine         1
  decomposition       CLUSTER_CHART
  type            EML_CHART
  firstData           27
  chartFileNumber     4
  disableImplicitCasting  1
  eml {
    name            "fcn"
  }
}
state {
  id              24
  labelString         "eML_blk_kernel()"
  position            [18 64.5 118 66]
  fontSize            12
  chart           23
  treeNode            [23 0 0 0]
```

```
    superState           SUBCHART
    subviewer            23
    type             FUNC_STATE
    decomposition        CLUSTER_STATE
    eml {
      isEML            1
      script               "function y = fcn(u)\n% This block supports the
Embedded MATLAB subset.\n% See the help menu for details. \n\ny = u^.4;"
      editorLayout         "100 M4x1[146 214 671 364]"
    }
  }
  junction {
    id               25
    position             [23.5747 49.5747 7]
    chart            23
    linkNode             [23 0 0]
    subviewer            23
    type             CONNECTIVE_JUNCTION
  }
  transition {
    id               26
    labelString          "{eML_blk_kernel();}"
    labelPosition        [32.125 19.875 102.544 14.964]
    fontSize             12
    src {
      intersection         [0 0 1 0 23.5747 14.625 0 0]
    }
    dst {
      id               25
      intersection         [7 0 -1 -1 23.5747 42.5747 0 0]
    }
    midPoint             [23.5747 24.9468]
    chart            23
    linkNode             [23 0 0]
    dataLimits           [23.575 23.575 14.625 34.575]
    subviewer            23
    drawStyle            SMART
    executionOrder       1
  }
  data {
    id               27
    name             "u"
    linkNode             [23 0 28]
    scope            INPUT_DATA
    machine          1
    props {
      array {
    size             "-1"
      }
      type {
    method           SF_INHERITED_TYPE
      }
      complexity           SF_COMPLEX_INHERITED
    }
    dataType             "inherited"
    fixptType {
      slope            1
```

```
      baseType            SF_INT16_TYPE
    }
    complexity          SF_COMPLEX_INHERITED
  }
  data {
    id                28
    name              "y"
    linkNode              [23 27 0]
    scope             OUTPUT_DATA
    machine           1
    props {
      array {
    size              "-1"
      }
      type {
    method            SF_INHERITED_TYPE
      }
      complexity          SF_COMPLEX_INHERITED
      frame             SF_FRAME_NO
    }
    dataType              "inherited"
    fixptType {
      slope             1
      baseType              SF_INT16_TYPE
    }
    complexity            SF_COMPLEX_INHERITED
  }
  instance {
    id                29
    name              "Air Compressor/NTU value/ho/Embedded\nMATLAB Function1"
    machine           1
    chart             23
  }
  target {
    id                30
    name              "sfun"
    description           "Default Simulink S-Function Target."
    machine           1
    linkNode              [1 0 0]
  }
}
```

References

[1] 'Breathing Air System Design Manual'. NK30-75140.*Pickering B Nuclear Generation Station*. Pickering: Ontario Power Generation, 1989. Print

[2] ‘Breathing Air System Flowsheet’. 30-058-O-75140-FS-01.*Pickering B Nuclear Generation Station*. Pickering: Ontario Power Generation. Print

[3] “The System Assessment – Comprehensive Compressor Air Audits – 5 Step Process." Compressed Air Best Practices. Atlas Copco Compressors LLC, Dec. 2011. Web. 22 Jan. 2012. <http://www.airbestpractices.com/sites/default/files/CABP_2011_12December_LR3.pdf>.

[4] Beitao, Guo, Liu Hongyi, Jiang Yang, Yang Cao, and TianHonghai. "Fuzzy Immune PID Control in VVVF Hydraulic System." *School of Mechanical Engineering & Automation, Northeastern University/Shenyang Institute Of Chemical Technology*. Web.

[5] LU, Zhihong, Zhiyong TANG, Hao LI, and Zhongcai PEI. "Fuzzy PID Control of Intelligent Pump."*School of Automation Science and Electrical Engineering*.Web.

[6] Zulfatman, and M. F. Rahmat. "APPLICATION OF SELF-TUNING FUZZY PID CONTROLLER ON INDUSTRIAL HYDRAULIC ACTUATOR USING SYSTEM IDENTIFICATION APPROACH."Faculty of Electrical Engineering.Web. 01 Apr. 2012.

[7] He, Lan, and Lepeng Song. "The Pump House Constant Pressure Fuzzy Selftuning PID Control System Simulation." School of Electronic and Information Engineering, Chongqing University of Science and Technology.Web. 29 Mar. 2012.

[8] Jialiang, Lu, Chen Guanrong, and Ying Hao. "Predictive Fuzzy PID Control: Theory, Design and Simulation." Department of Electrical and Computer Engineering, University of Houston.Web. 15 Mar. 2012.

[9] "Instrumentation and Control (I&C) Systems in Nuclear Power Plants: A Time of Transition." Instrumentation and Control (I&C) Systems in Nuclear Power Plants: A Time of Transition. IAEA. Web. 21 Jan. 2012. <http://www.iaea.org/About/Policy/GC/GC52/GC52InfDocuments/English/gc52inf-3-att5_en.pdf>.

[10] Fakhreddine O. Karray and Clarence de Silva, Soft Computing and Intelligent Systems Design-Theory, Tools and Applications, Pearson Education Limited Edinburgh Gate Harlow Essex, CM20 2JE, England, First published 2004.

[11] ONE, Team. "Pickering NGS: BUIA, SG & Silting Condition Assessment." Ontario Power Generation (n.d.): n. pag. Web.

[12] D, Raun. "Safety Regulations and Fuzzy- Logic Control to Nuclear Reactors." (n.d.): n. pag. Print.

[13] “Control (Wet) Storage – A misunderstood concept." Compressed Air Best Practices. Atlas Copco Compressors LLC, Dec. 2011. Web. 22 Jan. 2012. <http://www.airbestpractices.com/sites/default/files/CABP_2011_12December_LR3.pdf>.

[14] Zhong, Jinghua. "PID Controller Tuning: A Short Tutorial." Purdue University, 2006.Web. 3 Apr. 2012. <http://wwwdsa.uqac.ca/~rbeguena/Systemes_Asservis/PID.pdf>

[15] ‘AUTHORIZATION TRAINING - INSTRUMENT AIR - PART A & B’. N-OVH-24461-00001. Pickering B Nuclear Generation Station. Pickering: Ontario Power Generation, 2008.

[16] Ali Tarique1, Hossam A. Gabbar2*Tarique, Ali, and Hossam A. Gabbar. "Particle Swarm Optimization (PSO) Based Turbine Control." *ICA Intelligent Control and Automation* 04.02 (2013): 126-37. Web.

[17] ‘Instrument Air Design Manual’. NK30-DM-75120-00001. Pickering B Nuclear Generation Station. Pickering: Ontario Power Generation, 2010.

[18] ‘Instrument Air Compressor Flowsheet’. 30-5-O-75120-FS-01. Pickering B Nuclear Generation Station. Pickering: Ontario Power Generation.

[19] 'Instrument Air Design Manual'. NK38-DM-75120. Darlington Nuclear Generation Station. Darlington: Ontario Power Generation, 1991.

[20] Rastogi, Achint, and Hossam A. Gabbar. "Fuzzy-Logic-Based Safety Verification Framework for Nuclear Power Plants." *Risk Analysis* 33.6 (2012): 1128-145. Web.

[21] Hsu, Yin-Sung, Chi-Ma Wei, Yuan-Chi Ting, Shih-Yi Yuan, Chia-Ling Chang, and Kao-Chung Chang. "Capacitive Sensing Technique for Silt Suspended Sediment Concentration Monitoring." *International Journal of Sediment Research* 25.2 (2010): 175-84. Web.

[22] Chander, R. Modification Process [N-PROC-MP-0090]. Vol. 8. N.p.: Ontario Power Generation, 2012.

[23] Modification Outline [N-FORM-10958]. Vol. 12. N.p.: Ontario Power Generation, 2012.

[24] Deol, H. "System Performance Monitoring Plan Instrument Air System (P-SPM-75120-0443254)." Ontario Power Generation (2013)

[25] Smith, Michael, and Sukumar Kamalasadan. "Method for Improved Pressurizer System Knowledge Enabling Enhanced Pressure Control." (n.d.): n. pag. Web. 1 Feb. 2016.

[26] Cengel, Yunus A. *Heat and Mass Transfer: A Practical Approach*. 3rd ed. New York, NY 10020: McGraw-Hill, 2007.

[27] Gabbar, Dr. Hossam A. NUCL 5275G: Safety Instrumented Systems, Lec 1-9.

[28] Jiang, Dr Jin, and Jianping Ma. "Applications of Fault Detection and Diagnosis Methods in Nuclear Power Plants: A Review." *ScienceDirect* (2010): 255-64. Web. 15 Feb. 2013. <www.sciencedirect.com>.

[29] Gratton, Mary Ann. "Road Salt and Cars Produce Extreme Water Contamination in Frenchman's Bay." *Road Salt and Cars Produce Extreme Water Contamination in Frenchman's Bay*. N.p., 1 Mar. 2010. Web. 28 Apr. 2013. <https://ose.utsc.utoronto.ca/ose/story.php?id=2036>.

[30] Patwary, Masum A., William Thomas O'Hare, and Mosharraf H. Sarker. "Assessment of Occupational and Environmental Safety Associated with Medical Waste Disposal in Developing Countries: A Qualitative Approach." *ScienceDirect* (2011): 1200-205. Web. 20 Apr. 2013. <www.sciencedirect.com>.

[31] "Nuclear Power." Ontario Power Generation. Ontario Power Generation, n.d. Web. 21 Feb. 2015.

[32] Mark, O., C. Wennberg, T. Van Kalken, F. Rabbi, and B. Albinsson. "Risk Analyses for Sewer Systems Based on Numerical Modelling and GIS." *Safety Science* (1998): 99-106. *ScienceDirect*. Web. 28 Mar. 2013. <www.sciencedirect.com>.

[33] Mundra, Sanjay. "Advantages and Disadvantages of Predictive/Condition Based Maintenance." *Advantages and Disadvantages of Predictive/Condition Based Maintenance*. Preserve Articles, n.d. Web. 2 Apr. 2013. <http://www.preservearticles.com/2012020822922/advantages-and-disadvantages-of-predictivecondition-based-maintenance.html>.

[34] Deol, H. “Instrument Air System Health Report.” Ontario Power Generation (2013)

[35] MathWorks. *MATLAB [SIMULINK]: The Language of Technical Computing*. Computer software. Vers. 7.5.0.342 (R2007b). The MathWorks Inc., n.d. Web. 16 Jan. 2013.

[36] Boon C, Hwang. "Intelligent Control for a Nuclear Power Plant Using Artificial Neural Networks." (n.d.): n. pag. Web.

[37] **(**Kramer, B.j. "A Case Study in Developing Complex Safety Critical Systems."*Proceedings of the Thirtieth Hawaii International Conference on System Sciences* (n.d.): n. pag. Web.)

[38] **(**Wang, Lin, Yurong Zeng, Yanhui Li, and Hong Wang. "An Intelligent Decision Support System for Spare Parts Joint Replenishment." *2006 International Conference on Hybrid Information Technology* (2006): n. pag. Web.)

[39] (Andone, Daniela G., Ioana I. Fagarasan, and Matei R. Dobrescu. "Advanced Control of a Steam Generator." *2006 3rd International IEEE Conference Intelligent Systems* (2006): n. pag. Web.)

[40] (Shrikhande, S. V., V. K. Patil, G. Ganesh, B. B. Biswas, and R. K. Patil. "Hardware Reliability Prediction of Computer Based Safety Systems of Indian Nuclear Plants." *2010 2nd International Conference on Reliability, Safety and Hazard - Risk-Based Technologies and Physics-of-Failure Methods (ICRESH)* (2010): n. pag. Web.)

[41] (Guan, Da, Lei Yan, Yibo Yang, and Wenfu Xu. "A Small Climbing Robot for the Intelligent Inspection of Nuclear Power Plants." *2014 4th IEEE International Conference on Information Science and Technology*(2014): n. pag. Web)

[42] (Pan, Erzhen, Da Guan, Wenfu Xu, and Bingshan Hu. "Control System of a Small Intelligent Inspection Robot for Nuclear Power Plant Use." *2015 IEEE International Conference on Information and Automation* (2015): n. pag. Web.)

[43] (Kumar, Neeraj, I. Koley, P.r. Krishnamurthy, and S.n. Rao. "Regulatory Review of Computer Based Systems: Indian Perspectives." *2010 2nd International Conference on Reliability, Safety and Hazard - Risk-Based Technologies and Physics-of-Failure Methods (ICRESH)* (2010): n. pag. Web.)

[44] (Rodrigues, A. P., M. Correia, A. Batista, J. Sousa, B. Goncalves, C. M. B. Correia, and C. A. F. Varandas. "Intelligent Platform Management Controller for Nuclear Fusion Fast Plant System Controllers." *IEEE Trans. Nucl. Sci. IEEE Transactions on Nuclear Science* 58.4 (2011): 1733-737. Web.)

[45] Smith, Bax. ""Classical vs Intelligent Control."" (n.d.): n. pag. Print.

[46] Deol, Harsh, and Hossam A. Gabbar. "Self-tuning Fuzzy Logic PID Controller, Applications in Nuclear Power Plants." *IJISTA International Journal of Intelligent Systems Technologies and Applications* 14.1 (2015): 70. Web.

[47] Deol, Harsh, and Hossam A. Gabbar. "Fuzzy Logic-based Safety Design for High Performance Air Compressors." *Progress in Nuclear Energy* 80 (2015): 136-50. Web.