QATAR UNIVERSITY

COLLEGE OF ENGINEERING

CYBER ATTACK DETECTION IN NONLINEAR BINARY DISTILLATION COLUMN

BY

H.M. SABBIR AHMAD

A Thesis Submitted to

the College of Engineering

in Partial Fulfillment of the Requirements for the Degree of

Masters of Science in Electrical Engineering

January 2020

COMMITTEE PAGE

The members of the Committee approve the Thesis of
H. M. Sabbir Ahmad defended on 17/11/2019.

Dr. Nader Meskin
Thesis/Dissertation Supervisor

Dr. Reza Tafreshi
Committee Member

Dr. Khaled Khan
Committee Member

Dr. Hasan Mehrjerdi
Committee Member

Add Member

Approved:

Khalid Kamal Naji, Dean, College of Engineering

ABSTRACT

AHMAD, H.M., SABBIR., Masters : January : [2020]

Masters of Science in Electrical Engineering

Title: Cyber Attack Detection in Nonlinear Binary Distillation Column

Supervisor of Thesis : Nader Meskin.

This thesis focuses on addressing the issue of cyber security for Industrial Control Systems (ICS). Evolution in computing and internet technology has encouraged increasing number of Industrial Control Systems (ICS) to be linked to cyber world giving rise to a new class of systems called Cyber Physical system (CPS) making them more vulnerable to cyber threats. The effect of cyber-attacks differs in cyber physical critical ICS compared to traditional ICT systems as they can cause damage to physical infrastructure posing threats to human health and environment. The four main objectives of this thesis are as follows:

i. Mathematically model various malicious adversary attacks and intrusions.

ii. Mathematically model dynamics of a crude oil distillation column (DC) and design its control system.

iii. Analyze the effect of cyber-attacks on dynamic performance of the DC.

iv. Design attack detection techniques and validate their performance and effectiveness on the model of a DC.

DEDICATION

The thesis is dedicated to my parents who have supported me all along my way to pursue higher studies.

ACKNOWLEDGMENTS

This publication is made possible by the Graduate Sponsorship Research Award (GSRA) award (GSRA4-2- 0518-17083) from the Qatar National Research Fund (QNRF), a member of the Qatar Foundation. Its contents are solely the responsibility of the authors and do not necessarily represent the official views of QNRF.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Due to the continuous development of technology, an increasing number of electronic devices are being created with networking features suitable for connecting to IT networks. This technological evolution has also made its way to ICS where an increasing number of monitoring and controlling devices have been connected to computer networks facilitating supervisory level monitoring and control which provides CPS manifolds economic and performance enhancing benefits. However, it also makes Industrial Control Systems (ICS) more vulnerable to cyber-attacks. The envisaged complex CPS infrastructure more than ever requires the development of novel and proactive security technologies, as these systems are continuously targeted by attacks and intrusions by intelligent adversaries. Some typical examples of attacks in real systems are the Stuxnet worm attack, multiple recent power blackouts in Brazil, and the SQL Slammer worm attack on the Davis-Besse nuclear plant, to name a few, further justifying the need to address cyber security for ICS [1], [2].

Extensive research has been conducted on security issues from the prospective of network and communication technologies to securely defend network performance against adversaries. These researches have mainly concentrated on designing methodologies to secure communication networks in CPS ignoring interactions between the cyber and physical domain. Traditionally, cyber security for ICS has been dealt by IT engineers from the prospective of network security. Such approaches primarily aim to secure the communication network to protect the IT infrastructure without considering the physical behavior of the plant and how the ICS is affected by cyber-attack. This thesis considers the behavior of the plant and ICS as part of the study of cyber security.

ICS are characterized by feedback closed-loop control architecture and aim to optimize the system control performance, such as reducing state estimation errors, stabilizing an unstable plant, and enhancing the robustness against uncertainties and noise. It is important to optimize the system control performances and at the same time to guarantee the survivability and resiliency of cyber-physical ICS subject to multiple types of malicious attacks. A cyber-physical system (CPS) is characterized by the number of control loops containing controller, sensors, and actuators. Integrity, availability, and confidentiality are three essential characteristics that are required for data transmitted across communication channels in a closed-loop control in order to operate the plant as per the specifications. An attacker can launch an attack in order to violate one of these three requirements of data that are transmitted over the communication channels. Hence, the attacker can target the data from the plant sensors or the actuators.

The huge worldwide demand for crude oil can make them a target for attackers. Since distillation columns (DC) handle highly flammable chemicals, attackers can launch attacks with the motivation of causing damage to the plant infrastructure or upset the column operation by degrading the quality of the distilled products. In this thesis, a continuous binary DC is considered as a cyber-physical framework to investigate the effect of attacks on the performance of DC as well as the development of attack detection algorithms.

1.1 Literature review

Attack modelling is the first step which is required to generate attack vectors for performing cyber security study. As this thesis focuses on cyber security for ICS hence attacks which are relevant in this regard have been reviewed. Based on the existing literature, attacks on ICS can be classified into four categories: i) sensor attacks, ii) actuator attacks, iii) sensor and actuator attacks, and iv) controller attacks. In [3]- [4], [5], and [6] various different models for sensor attacks are presented. Various different actuator attack models have been presented by authors in [7]- [8]. In [1], [2], [9], [10]- [11], [12] authors present attacks which can be applied to both sensors and actuators. Finally, [7] presents cyber-attacks to be applied on controllers.

Understanding the effect of attacks is the first step to design attack diagnosis, detection and mitigation technologies. One of the key factors that needs to be identified is the parts of the ICS the attacker may target. Thus far, to the best of our knowledge, there is no literature available in the area of cyber security for a DC. Hence, motivation is drawn from the literatures available in the area of fault diagnosis, fault recovery, fault survivability, and fault tolerant control techniques. It will be a fair assumption to believe the attacker is limited to targeting the controllers, sensors and actuators as they are the only hardware which are part of the closed-loop control and are integrated to the cyber world. In [13]- [14], faults with the reboiler power and reflux flow rate have been studied. The effect of the top and bottom composition sensor faults have been studied in [15] and [14]. Fault in the feed flow rate has been considered and assessed in [16], [17], [18], [14]. Besides that, [17] and [18] also study faults in the feed composition and the feed preheater power, respectively and their effect on the DC performance. Finally, fault in the bottoms flow rate has been discussed in [19].

Attack detection is the first step towards designing attack mitigation, survivability, and recovery techniques for ICS against cyber threats. The focus is to design detection techniques following approaches in the field of control engineering based on the physical knowledge of the plant and ICS and validate their performance and effectiveness in terms of attack detection using the model of the DC. However, several different model-based techniques have been found in the literature for detection of cyber-attacks. Observer-based detection techniques have been presented in [20], [21], [22], [23], and [24]. In [5], and [25] Kalman Filter (KF) is used for securely estimating the system states for detecting cyber-attacks. The authors in [26] proposes Extended Kalman Filter (EKF) based algorithm for attack detection. Unscented Kalman Filter (UKF) has been used for attack detection in [27] - [28]. Comparison between Observer and Cubature KF (CKF) has been done in [19]. [23] compares between UKF, EKF and Enhanced EKF the ability of detecting cyber-attacks. In [29], authors provide a comparative study of observer, EKF, UKF, and CKF for detection of malicious cyber-attacks.

In terms of application areas, in [30], [31], and [32] authors explore the effect of cyber-attacks on Smart Grid whereas [33]- [34] study the effect of cyber-attacks on Power Systems. In [35] and [36], the effect of cyber-attacks on Smart Cities and Water Distribution System are investigated, respectively. In regards to cyber-attack detection [37] and [38] present machine learning algorithms for Water Distribution System. On the other hand, [39] proposes a model-based detection technique for water treatment plant. Authors in [40], and [41] present machine learning algorithms for detecting attacks in Smart Grid. Finally, detection methods for attacks on Power Systems are presented in [42], [43], and [44].

1.2 Thesis objectives

The first objective is to mathematically model cyber-attacks for ICS. As part of this thesis attacks which are commonly found in the literature for ICS are primarily modeled. All the existing literature provides attack models for the class of linear time invariant systems. However, most plants in real world have nonlinear dynamics. Hence in this thesis the attacks have been modeled for nonlinear systems.

The second objective is to model DC in order to investigate the effects of different cyber-attacks on the performance of DC. In our case we limit the study to a continuous binary DC. A continuous binary DC during its entire lifecycle takes a fixed raw crude stream and splits it into two product streams. The availability of a wide scale computer simulation tools makes it possible to simulate the dynamic behavior of a nonlinear plant based on their mathematical models, to facilitate safer tools for performing cyber security study and consequently, to save cost and time. A distillation column has a nonlinear dynamic behavior. There are two different DC models which are used for the study as part of this thesis. The first plant model is generated based on data found in [45] and [46] using which a DC is designed in Aspen Plus Dynamics. The first model has been derived based on grey box modelling approach where the known dynamics are generated using MESH equations and remaining dynamics are generated using Aspen Plus Dynamics. The second model is based on a hybrid simulation engine where the plant dynamics are simulated using Aspen Plus Dynamics and the ICS is implemented in Simulink.

The third objective is to study the effect of cyber-attacks on the control performance of a DC. Based on the literature survey, the attack surface for a DC is determined. Then, the modeled attacks are injected and their effect on the performance of the control system and consequently the DC is observed.

The final objective is to propose attack detection techniques for the DC. The thesis focuses on designing model-based detection techniques for cyber-attacks using secure estate estimation. There is no literature available for detection of cyber-attacks on a DC. As part of the thesis, three different detection techniques have been proposed. Two techniques have been designed based on nonlinear grey box model of the DC which relies on state estimation using Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF). EKF and UKF are well established algorithms used for estimating the plant states and consequently outputs for a nonlinear system based on the knowledge of the plant inputs and outputs. The third detection algorithm has been designed for the Aspen Plus Dynamics and Simulink based hybrid model using Luenberger observer which is a common technique for estimating states for linear systems based on the knowledge of plant inputs and outputs.

1.3 Thesis contribution

There are three main contributions of the thesis. The first contribution is modeling attacks for ICS for nonlinear CPS including sensor attacks, actuator attacks and controller attacks. The second contribution is about studying and identifying the behavior of a continuous binary DC under cyber-attacks. Lastly, this thesis provides three attack detection techniques that can be applied to detect attacks on DC.

1.4 Thesis organization

The thesis is organized into four chapters including introduction chapter. Chapter 2 concisely presents fundamentals of the distillation theory before presenting the details of the two DC models along with their control schemes. In Chapter 3, the mathematical models of the attacks modeled as part of this thesis is presented along with results for illustrating their effect on the behavior of the two DCs. Following that the details of the attack detection algorithms are presented. Moreover, the performance and effectiveness of the algorithms for detecting attacks on DC are illustrated using simulation results. Finally, the thesis ends with a short conclusion along with scope for future work in this field.

CHAPTER 2: DISTILLATION COLUMN MODELLING AND CONTROL

This chapter begins by introducing fundamentals of distillation theory along with the key essential physical and chemical properties used for characterization of raw crude. Following that the distillation column (DC) models along with their control scheme and closed loop simulation results are presented in separate subsections. The first subsection presents the grey box model of binary continuous DC. After that the hybrid DC model implemented using plant dynamics generated from Aspen Plus Dynamics based DC model and the ICS implemented in Simulink has been presented.

2.1 Distillation theory

Crude oil in its raw form is considered as a mixture of various chemical components. Distillation is an ancient technology which is commonly used in petroleum refineries to render various valuable product streams from crude feed stream. As raw crude is a multicomponent mixture hence it is essential to characterize the feed stream, intermediate, and final product streams prior to designing crude oil processing refinery.

Fundamentally a refinery consists of a combination of physical and chemical processes that are used to render final valuable products from raw feed stream. Distillation is a commonly found physical process used to separate components from their mixture based on their relative volatilities due to differences in their boiling points. Broadly the process in a petroleum refinery can be categorized into four groups which includes separation, finishing, conversion, and support used to generate valuable final product streams from crude feedstock. Therefore, characterization of crude feedstock is necessary to design the various processes in a refinery. Characterization of crude is done based on their chemical and physical properties. The chemical properties include identifying the presence of functional groups present in the mixture such as olefins, parrafins, napthenes, aromatics, and resins. On the other hand, properties such as density, viscosity, boiling points and so on are used for physical characterization of crude oil

2.1.1 Crude oil characterization

In this section, the physical and chemical properties of crude oil are reviewed.

2.1.1.a Physical properties:

i. **API Gravity:** It is a measure of the relative density of the crude stream in comparison to water. The API gravity is generated and measured at 60^{o}F, and can be expressed as:

API = 141.5/specific gravity – 131.5 (1)

where the specific gravity (SG) of a liquid material is defined as the density at 60°F (15.6°C) over density of water at 60°F. The SG of water at 15.6°C is 0.999 g/cm^3 (999 kg/m^3). Hence components heavier than water has SG greater than 1; thus, their API gravity is less than that of water. On the other hand, components lighter than water has higher API gravity than water. Hence lower SG indicates higher API gravity thus lighter crude fractions and vice-versa.

ii. **True Boiling Point (TBP) distillation curve** [47]**,** [48] **:** TBP curve offers an insight into a petroleum liquid mixture composition before it is processed and provides a means to characterize it. It generates boiling points distribution for a crude mixture, which can be used to determine the components that constitutes the mixture along with their volume percentage in the mixture. A TBP curve plots the boiling point as a function of the distilled volume percent. This is an ideal method which can achieve the best possible separation. The separation results obtained using this distillation method can be closely acquired from an actual distillation column which makes it a useful tool in the distillation column design process. Batch distillation method is used to generate the TBP curve for a mixture. The method incorporates an apparatus that includes well over 100 theoretical plates and involves maintaining a high reflux ratio (reflux to distillate ratio) during the distillation process to acquire the TBP curve.

iii. **American Society for Testing and Materials (ASTM) curve** [48]**:** Although TBP curves are ideal, however the required distillation method is expensive and time consuming. To remedy these issues, other quicker methods such as ASTM have been developed. ASTM methods provides another means to acquire information about the constituents of a crude oil mixture that can be used to characterize it. There are many ASTM distillation standards available that provide the boiling point distribution. ASTM D86-96 is one such commonly used method that involves finding out the boiling point distribution of lighter cuts in a crude mixture at atmospheric pressure. This process involves heating the sample at varying temperatures and collecting the vapor as liquid in a container after passing through a condenser. The curve is obtained by plotting the vapor temperature as a function of the cumulative volume of liquid collected. Unlike TBP, the reflux ratio in this process is kept to zero. There may be unintentional reflux flow from the container back to the flask while cooling.

iv. **Equilibrium Flash Vaporization (EFV Curve)** [49]**:** This is another method used to generate the boiling point distribution of a crude sample to identify its constituents and their compositions for characterization. In this type of separation method, the crude sample under pressure is flashed or vaporized by passing it through a heater. The two- phase liquid vapor mixture is then separated in a flash drum. The curve is generated by flashing the sample at different temperatures but at fixed pressure and measuring the volume of liquid collected at every temperature and then plotting the heater temperature against liquid volume collected.

A comparison of the three curves is provided in Figure 1. The key differences are the initial boiling temperature and final boiling temperature achieved using the three methods. As can be seen from Figure 1, TBP gives the lowest initial boiling point followed by ASTM and EFV curves. On the other hand, TBP gives the highest final boiling temperature followed by ASTM and EFV curves. The three curves intersect at fifty percent volume.

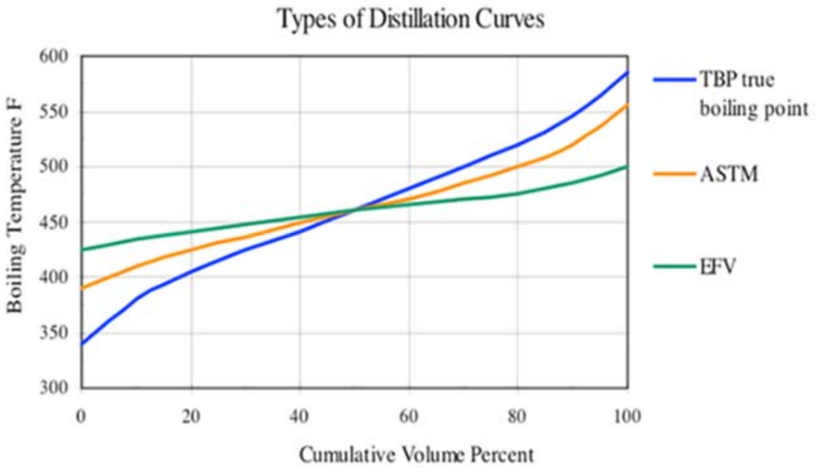


Figure 1. TBP vs ASTM vs EFV curves [50].

v. **Watson K factor:** Originally introduced by researchers at Universal Oil Product (U.O.P), the Watson K factor takes into consideration the boiling point of a hydrocarbon mixture varies with H/C ratio. The factor is expressed as follows

$$K = \frac{\sqrt[3]{T_B}}{S} \tag{2}$$

where T_B is the average boiling point of the mixture in Rankine (°R) and S is the specific gravity of the mixture at 60 Fahrenheit (°F).

vi. **Viscosity:** Viscosity is used to define the flow properties of crude oil. In industries it is generally expressed in centistokes or saybolt seconds or redwood seconds.

vii. Flash point, fire point and kindling point: These are properties of crude mixture which are related to safety and transmission of refinery products. It is well known that hydrocarbons are volatile mixtures. All liquids have vapor pressures which depends on temperature. As temperature increases, the vapor pressure also increases. Increase in the vapor pressure increases the concentration of flammable combustible liquid in air. Flash point of a volatile mixture is the lowest temperature at which there is enough concentration of flammable liquid in air to ignite/flash in the presence of an ignition source however not enough to sustain combustion. Fire point of a volatile mixture is the temperature at which the concentration of flammable fluid in air is high enough to continue burning for few seconds once ignited and the ignition source is removed. The fire point of a liquid is higher than the flash point. Kindling point of a mixture is the temperature at which it ignites spontaneously irrespective of the presence of an ignition source. At this temperature, the mixture is said to have the activation energy required to undergo combustion. All these parameters are dependent upon external pressure as the vapor pressure is a function of external pressure.

viii. Pour point: The pour point is the lowest temperature at which the oil will pour or flow under gravity. Below this temperature the fluid loses its flow characteristics. It is one of the important low temperature characteristics of high boiling point fractions as below this temperature the product cannot be transferred though pipeline. The pour point of crude oils relates to their paraffin content, the higher the pour point the higher the paraffin content.

ix. Sulphur content: The Sulphur content is the most important characteristics that defines the price of crude oil in the market [36]. Crude oil contains both organic and inorganic Sulphur. Categorically, the Sulphur content is used to determine how sweet or sour a crude mixture is. If the Sulphur content is high in crude oil then it is called as sour crude and if the Sulphur content is low then it is termed as sweet crude. Typically, crude oil Sulphur content consists of 0.5 – 5 wt % (weight percent) of Sulphur. Crude oil is considered sweet if its Sulfur content is at most 0.5 wt. %, while sour crude has Sulfur content of more than 0.5 wt. % [51]. On the other hand, crude oil ranges from sweet crude oil ≤ 0.05wt. % to sour crude oil ≥ 1 wt. % [51]. It is necessary to control the Sulphur content in crude oil for various reasons; it is poisonous, combustion of Sulphur releases oxides of Sulphur which are responsible for acid rain and so on.

2.1.1.b Chemical properties:

Tabulated below is a list of functional groups which are derived from chemical analysis which are found in crude oil.

Table 1. Organic functional groups found in crude oil.

Name	Chemical structure
Paraffins	Alkanes with straight Carbon chain.
Olefins	Alkenes with double bond between carbons.
Napthenes	Cyclic alkanes like cyclopropane, methyl cyclohexane.
Aromatics	Ring structured alkenes (Benzene, Xylene, Toulene).
Napthalenes	Poly-aromatic structure containing two or more aromatic rings.
Organic sulphur	Various organic compounds such as mercaptans, theophenes and so on are also found in crude oil.
Oxygen	Weak acids such as carboxylic acid, phenol, cresylic acid and napthenic acid also exist in crude oil.
Resins	Polynuclear aromatic rings contains branches of parrafins and other polar compounds.
Asphalt	The heaviest leftover fraction from crude oil containing large aromatic chain (> 20 rings) with branched paraffinic and naphthenic chains.

Crude petroleum is a mixture of compounds that can be separated into different generic boiling fractions: Light naphtha (boiling point (bp): –1 to 50◦C), gasoline (bp:

–1 to 180◦C), heavy naphtha (bp: 150–205◦C), kerosene (bp: 205–260◦C), light gas oil (bp: 260–315◦C), heavy gas oil (bp: 315–425◦C), lubricating oil (bp: >400◦C), vacuum gas oil (bp: 425–600◦C), and residuum (bp: >510◦C) [52]. These products can be categorized under various existing functional groups in organic chemistry. The chemical properties of crude oil are determined based on one or more of these functional groups present in the mixture.

2.2 Mathematical model, control and simulation of a continuous binary distillation column

A continuous binary distillation column separates a crude feedstock into two product streams i. distillate and ii. bottoms product. It is assumed that feed into the column is a two phase (liquid and vapor phase) pseudo-binary mixture of a light product and a heavy product which is separated in the column. The concentration of the light component in the feed in the liquid phase is denoted by x_F. The concentration of light component in the top product/distillate in liquid phase is denoted by x_D and the concentration of the lighter component in bottoms product collected from the bottom of the column in liquid phase denoted by x_B. For successful separation it is desired that the concentration of lighter product in distillate is close to 100% and in bottoms product (virtually) 0% which arises the necessity for control system in DC making it a CPS.

The column can be divided in two sections i. rectifying section and ii. Striping section. The rectifying section is located at the top just above the feed and the bottom section is called the stripping section. The original crude feedstock is passed through a preheater which heats the feed to a certain temperature to separate the feed in a two phase fluid before feeding to the distillation column. This is one of the energy input port of the distillation column. The top product is fed to a condenser which effectively condenses the vapor distillate to a cold liquid reflux. This allows for energy to be extracted off the column. Finally the bottom product stream is reheated using a partial re-boiler which allows for energy into the column. A flowsheet of a binary DC is presented in Figure 2.

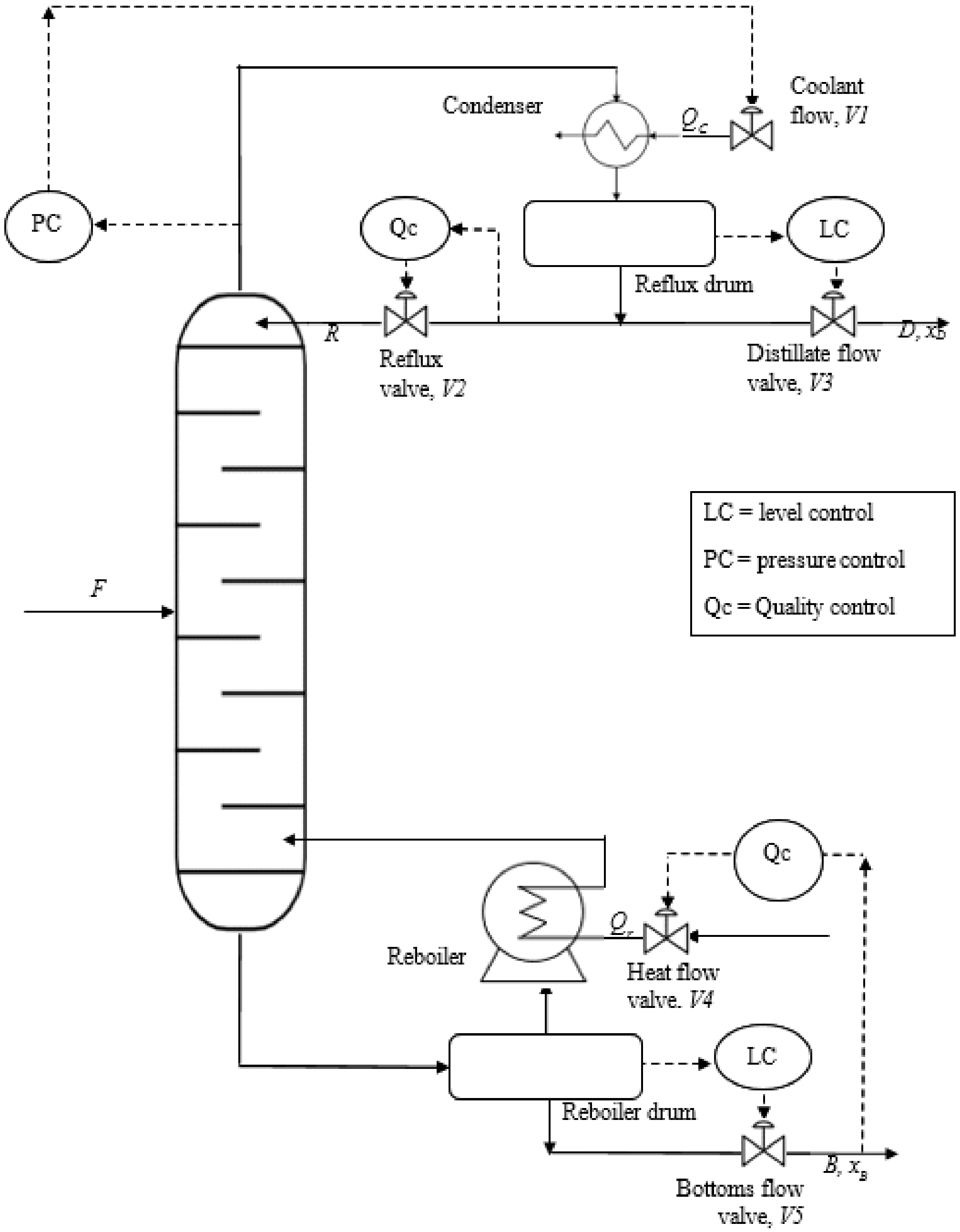


Figure 2. Flowsheet of a binary distillation column.

2.2.1 Plant data

The distillation column model presented is based on a real petroleum project to build a gas processing plant to raise the utility value of the raw condensate. The plant operates for 24 hours and 365 days a year during which it processes 130,000 tons of raw condensate. The operating specification is to maintain the quality of the distilled products; the purity of the distillate, x_D has to be higher than or equal to 98% and the impurity of the bottoms, x_B has to be less/equal than 2% [45]. The basic feed stock data and its actual compositions are based on the data given in [46].

The plant equipment has been designed to operate within 10% above the nominal capacity and 50% below the turndown ratio. Considering the plant processes 130,000 raw condensate a year, the feed rate is calculated as below:

$$Feed = \frac{130000}{24\,h\,.365\,working\ days} = 15.476 \text{ tons/hour}$$

Table 2, presents the true composition of the raw condensate. The actual composition of the raw condensate generally fluctuates around the true composition.

Table 2. Raw condensate composition.

Component	Mole fraction %	Component	Mole fraction %
Propane	0.00	n-C11H24	1.94
Normal Butane	19.00	n-C12H26	2.02
Iso-Butane	26.65	Cyclopentane	1.61
Iso-Pentane	20.95	Methylclopentane	2.02
Normal Pentane	10.05	Benzene	1.61
Hexane	7.26	Toulen	0.00
Heptane	3.23	O-xylene	0.00
Octane	1.21	E-benzen	0.00
Noname	0.00	124-Mbenzen	0.00
Normal Decane	0.00		

As can be observed from Table 2, the liquid feed consists multiple components. However since the aim is to use a binary DC hence it is considered as a pseudo binary mixture consisting of Ligas (iso-butane, n-butane and propane) and Napthas (iso-pentane, n- pentane, and heavier components) [45].

The column has been designed with 14 trays, i.e. $N = 14$. The model of the column is generated by lumping some components of the condensate together (pseudo-components) only [8]. The properties of the pseudo components are allowed to fluctuate within the range shown in Table 3, based on the fluctuation in the condensate composition.

Table 3. Properties of pseudo components.

Properties	Ligas	Napthas
Molar weight	54.5 – 55.6	84.1 – 86.3
Liquid density (kg/m^3)	570 – 575	725 – 735
Feed composition (vol %)	38-42	58 – 62

- ***Relative volatility:***

It is a measure of how volatile one component is compared to the other. This gives an indication of the degree of ease or difficulty these two components can be separated from a mixture.

Consider two components i and j; their relative volatility is given by:

$$\alpha_{ij} = \frac{\left[\frac{y_i}{x_i}\right]}{\left[\frac{y_j}{x_j}\right]} \tag{3}$$

where y_i, and y_j represents the mole fraction of the component i and j in the vapor phase, respectively; and x_i, and x_j represents the mole fraction of the component si and j, respectively in the liquid phase. From [34] the relative volatility of the pseudo-binary mixture is found to be α = 5.68 for the operating temperature and pressure.

2.2.2 Mathematical modeling of a distillation column

The column dynamics is mathematically modelled using MESH equations which are given below for a general *n^th^* tray.

Mass balance:

$$\frac{dM_n}{dt} = L_{n-1} - L_n + V_{n+1} - V_n \quad (4)$$

Component balance:

$$\frac{d(M_n x_{n,i})}{dt} = L_{n-1}x_{n-1,i} - L_n x_{n,i} + V_{n+1}y_{n+1,i} - V_n y_{n,i} \quad (5)$$

where $x_{n,i}$ and $y_{n,i}$ corresponds to the liquid and vapor mole fraction of the i^{th} component in tray *n*, respectively. L_n and h_n corresponds to flow rate of the liquid (in kmole/hr) flowing down the n^{th} tray and the amount of heat energy that is passed with the liquid from the n^{th} tray. Similarly V_n and H_n correspond to the vapor flow rate at the nth tray (in kmole/hr) and the energy carried by the vapor. M_n is the liquid hold up (in kmole) in the n^{th} tray.

Enthalpy balance:

$$\frac{d(M_n h_n)}{dt} = h_{n+1}L_{n+1} - h_n L_n + H_{n-1}V_{n-1} - H_n V_n \quad (6)$$

By differentiating (5) and substituting (4) into (5) it follows:

$$\frac{dx_{n,i}}{dt} = \frac{L_{n-1}x_{n-1,i} + V_{n+1}y_{n+1,i} - (L_{n-1} + V_{n+1})x_{n,i} - V_n(y_{n,i} - x_{n.i})}{M_n} \quad (7)$$

Equilibrium:

$$y_{n,i} = k\, x_{n,i} \quad (8)$$

Summation:

$$\sum_{i=1}^{N} x_i = 1$$

$$\sum_{i=1}^{N} y_i = 1 \quad (9)$$

The column is designed using Redfrac model available in ASPEN Plus as shown in Figure 3 using the plant data to determine the steady state operating point of the column. Aspen Plus is one of the many industrially used process simulation softwares provided by AspenTech. The parameter values for the column obtained from the simulation run in Aspen Plus are used in the mathematical model.

Redfrac provides two degrees of freedom for the column design. In our case, distillate rate (kmole/hr) and reflux rate (kmole/hr) are the two parameters selected for the column design. Table 4, presents the data used in ASPEN Plus for the column design.

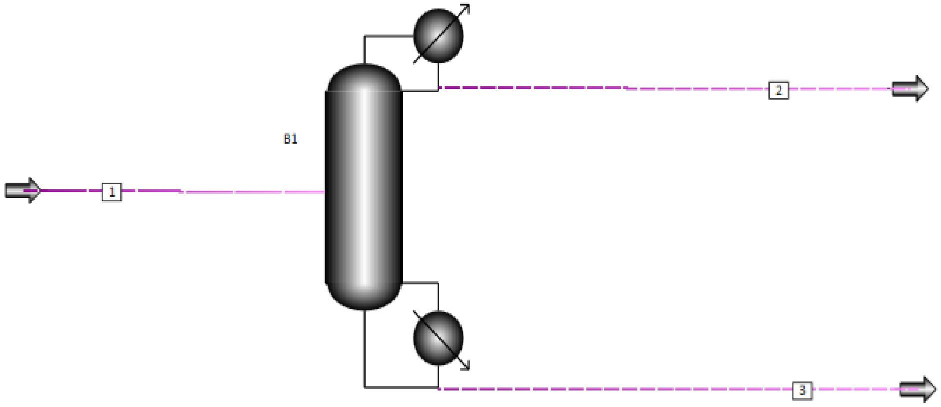


Figure 3. Redfrac binary distillation column design in Aspen Plus.

Table 4. Column design parameters used in Aspen Plus

Parameter	Value	Parameter	Value
Feed temperature (ºC)	118	Feed pressure (atm)	4.6
Condenser pressure (bar)	4	Stage pressure drop (bar)	.075
Distillate rate (kmole/hr)	93	Reflux rate (kmole/hr)	350

The mathematical model is simulated in Simulink, and the control objectives are to maintain the purity in the top product and impurity in the bottom product within a certain level. Therefore, only the mass and component balance equations are only used in simulation. The values obtained from Aspen Plus are used during open loop simulation of the distillation column. Aspen Plus also computes the values based on mass and energy balance. Hence using these values result in the column operating in the steady state unless perturbations are made to any of the energy ports of the column.

The model from Aspen Plus is also transported to Aspen Plus Dynamics for performing dynamic simulation and it is observed that the liquid and vapor flow rates in *n^th^* tray have transient dynamics observed during changes made to any of the energy ports which is due to the hydraulics of the tray. It is necessary to include this dynamical behavior since the time constants associated to the liquid and vapor flow rates are quite large which will affect the overall response time of the system. The molar flow rates of the vapor and liquid through the stripping and rectifying sections are constant in the steady state. Hence the effect of tray hydraulics is included in the simulation by adding a time constant to the liquid and vapor flow rates across each tray in the column and selecting their dynamics as follows:

$$dL_n(s) = \frac{1}{\tau_L s+1}\, dL(s) \tag{10}$$

$$dV_n(s) = \frac{1}{\tau_V s+1}\, dV(s) \tag{11}$$

where the time constants are determined from Aspen Plus Dynamics. $dL(s) = (L - L_{nominal})(s),\ L_n = L_{n(nominal)} + dL_n,\ dV(s) = (V - V_{nominal})(s), and\ V_n = V_{n(nominal)} + dV_n$, *L(s)* and *V(s)* corresponds to reflux rate (kmole/hr) and boil up rate (kmole/hr) in Laplace domain.

The initial molar holdup in each tray has been computed using the Francise-Wier formula presented in [31]. The following assumptions are made in simulation.

i. The relative volatility is constant across each tray of the column. Hence,

$$y_i = (\alpha x_i) / (1 + (\alpha - 1)x_i)$$

where, x_i and y_i are the liquid concentration and vapor concentration of the lighter component in the binary mixture, respectively, and α is the relative volatility.

ii. The overhead vapor is totally condensed in a condenser.

iii. The pressure remains constant at the top of the column and the differential pressure between trays remains constant.

iv. The holdup of vapor is negligible throughout the system.

v. Any change in the reflux flow rate by an dL_1 is reflected in the liquid flow rates by the same amount (i.e. in the steady state $dL_1 = dL_2 = ... = dL$)

2.2.3 Control of Distillation Column

To commence discussion on controlling a binary DC, at first, it is essential to determine its degree of freedom (DoF). There are several ways of identifying the DoF of a process. A simple way to determine the DoF of a process is by counting the number of independent variables which can be used to achieve the desired control objective of the process. Hence one of the approaches to determine the DoF for a distillation column is count the number of valves. The flowsheet shown in Figure 2 contains five control valves each of which can be used to independently vary the distillate flow rate, reflux flow rate, bottoms flow rate, condenser duty and reboiler duty. Therefore, this column has five DoF. However, it is assumed that the column is operated under constant pressure; hence an independent variable has to be allocated to control the column pressure which leaves the column with four independent variables.

The feed stream is considered being set by the upstream process; hence is considered to be a constant. In a process the inventories have to be regulated constantly. Inventory loops in the case of the distillation column presented in Figure 2 includes liquid levels in the reflux drum and column base which must be controlled. Out of the four DoF, allocating two independent variables one for controlling each liquid level leaves us with two independent variables which can be used for controlling two other variables.

The remaining two independent variables are manipulated to maintain the product qualities of the distillate and bottoms product. Generally, a column is designed to operate in the steady state at the values determined from design calculations during the normal operation. A column remains at energy and material balance (described by mesh equations) during the steady state condition. Material balance infers that the sum of products entering the column must be equal (approximately) to the sum of products leaving the column; and energy balance implies that the heat input to the column must be equal (approximately) to heat removed from the system. A column is said to be "stable" when it is under energy and material balance.

The column dynamics arises from the control loops, i.e. if value of a control variable fluctuates from its desired value then the corresponding manipulated variable is adjusted to bring the control variable back to its desired value. Such changes in value of control variables may occur due to various reasons including change in properties of the feed within the range mentioned in Table 2. As mentioned previously, the proposed distillation column in Figure 2 has five input and five output variables. There are several control configurations which can be used to achieve the desired control objective of the DC. Control configuration refers to the selection of independent variables for controlling the outputs of the process.

Amongst several available control configurations a widely used configuration for a binary DC summarized in Table 5, is selected to achieve the control objective of the binary DC. Out of the four outputs (excluding column pressure) two outputs are controlled using feedforward control while the remaining two outputs are controlled using feedback control containing PID controllers. The PID controllers for distillation and bottoms product composition control are tuned using model-based PID tuning tools available from Matlab. The controller performance is verified by implementing the model in Simulink.

Table 5. Control configuration details for distillation column.

Controlled variables	Manipulated variables	Control valve (Figure 2)
Concentration of distillate	Reflux flow rate	Flux flow V2
Liquid level	Distillate flow rate	Distillate flow D
Concentration of bottoms	Re-boiler duty	Heat flow V4
Liquid level in column base	Bottoms flow rate	Bottom flow V5

The mathematical expression of the PID controller.

$$u(t) = K_P + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \tag{12}$$

The liquid holdup in reflux drum and column base is kept constant using feedforward control by adjusting the distillate flow rate and bottoms flow rate respectively as follows

$$D(t) = V_{15}(t) - L_{16}(t) \tag{13}$$

$$B(t) = L_2(t) - V_1(t) \tag{14}$$

2.2.4 Simulation of distillation column in SIMULINK

Figure 4 shows the closed-loop simulation of the DC in Simulink.

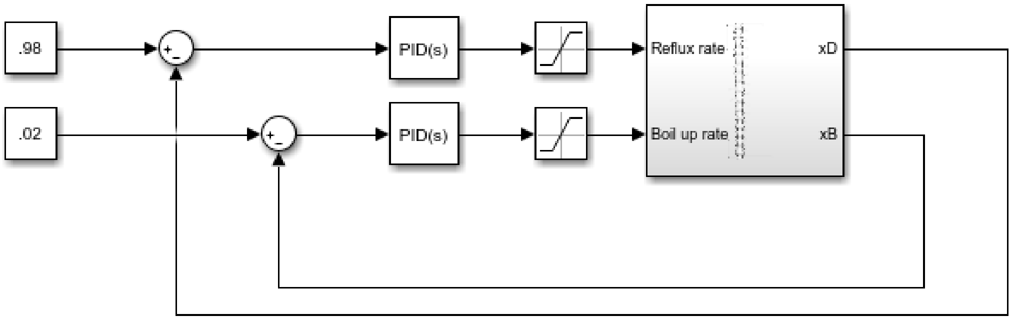


Figure 4. Simulation of distillation column in SIMULINK.

2.2.4.a Response of distillation column under the proposed control scheme:

The performance of the control scheme proposed for the column has been validated by observing if the distillate purity level and bottoms impurity level conform to the column performance requirements specified in section 2.2. Figure 5 and Figure 6 illustrates the closed loop response of the distillate purity and bottoms impurity level. As can be seen both the products meet their quality requirements thus validating the operation of the proposed control scheme.

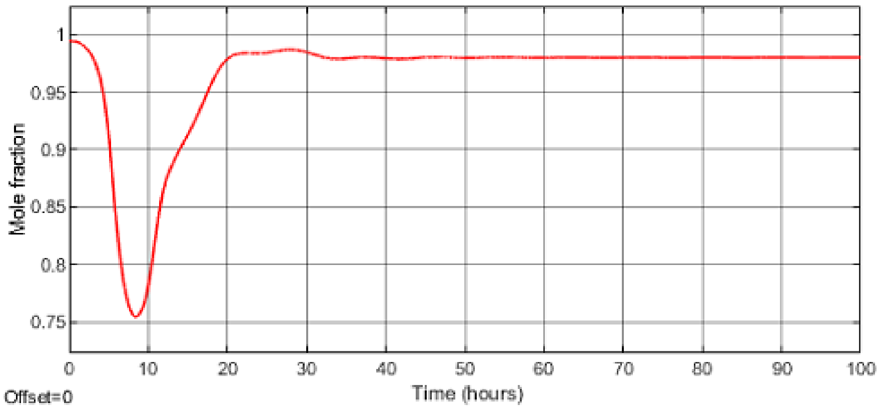


Figure 5. Plot of distillate purity against time in closed loop.

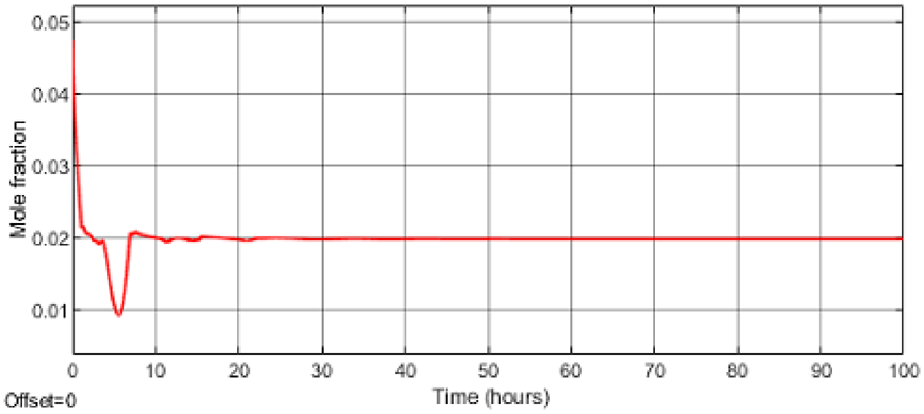


Figure 6. Plot of bottoms impurity against time in closed loop.

2.2.4.b Effect of feed disturbance on the column performance:

The effect of change in the feed rate is investigated by changing the flow rate by. 5% from its nominal value. This is done by setting the feed flow rate using a sine function with amplitude of 0.5% of the nominal feed flow rate and its effect on the distillate purity and bottoms impurity have been illustrated in Figure 7, and Figure 8, respectively. It is observed that adding a sinusoidal disturbance to the feed rate introduces sinusoidal disturbance in the distillate purity and bottoms impurity respectively.

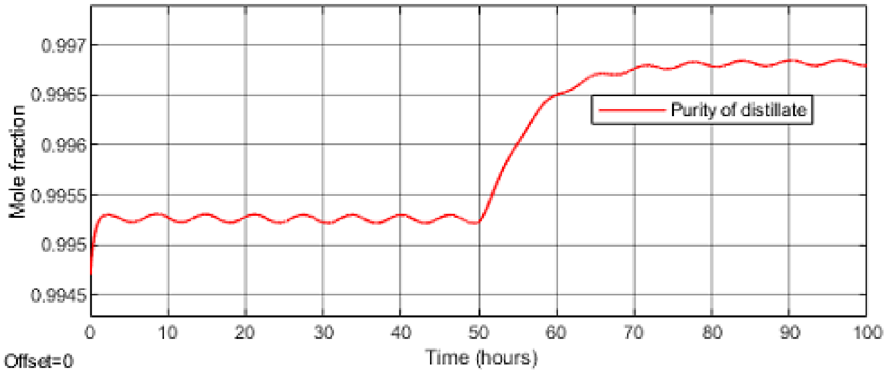


Figure 7. Plot of distillate purity for 0.5% change in feed flow rate. A step change of 5 mole/hr is applied to reflux rate at 50 hours during simulation.

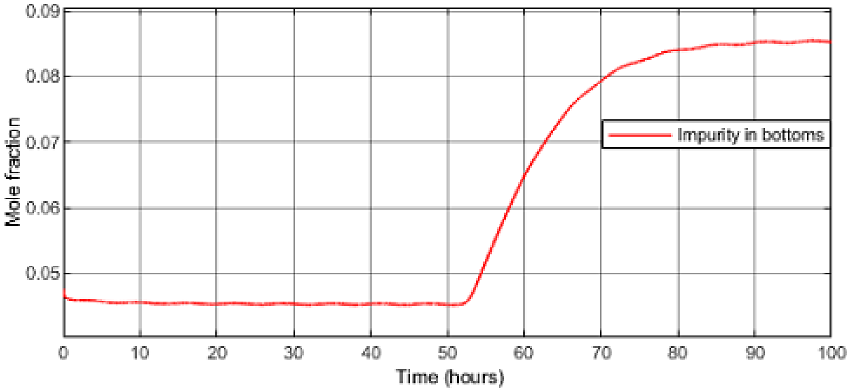


Figure 8. Plot of bottoms impurity for 0.5% change in feed flow rate. A step change of 5 mole/hr is applied to reflux rate at 50 hours during simulation.

2.3 Aspen Plus Dynamics based design, simulation and control of a binary distillation column

As part of this dynamic simulation scheme the column has been first designed using Aspen Plus which is then imported to Aspen Plus Dynamics in order to perform the dynamic simulation. Then closed loop control scheme has been designed to regulate the column output parameters in Simulink. Following that, the closed-loop control in Simulink integrated with the dynamic model in Aspen Plus Dynamics in order to perform dynamic simulation. The distillation column is design using Aspen Plus and then the dynamic model is derived using Aspen Plus dynamics. Table 6 summarizes the composition of the raw feed condensate used to design the DC in Aspen Plus.

Table 6. Composition of the raw condensate.

Component	Mole %
Propane	40.00
Iso Butane	60.00

There are 32 trays inside the column in total. The raw crude stock is fed to tray

15 in the distillation column. The column has been designed to operate at a pressure of 14 atm at the top, i.e. the pressure at the reflux drum is 14 atm. The differential pressure between the stages is set to 0.1 psi. The height and length of the both the reflux drum and the reboiler have been set to 1 meter respectively.

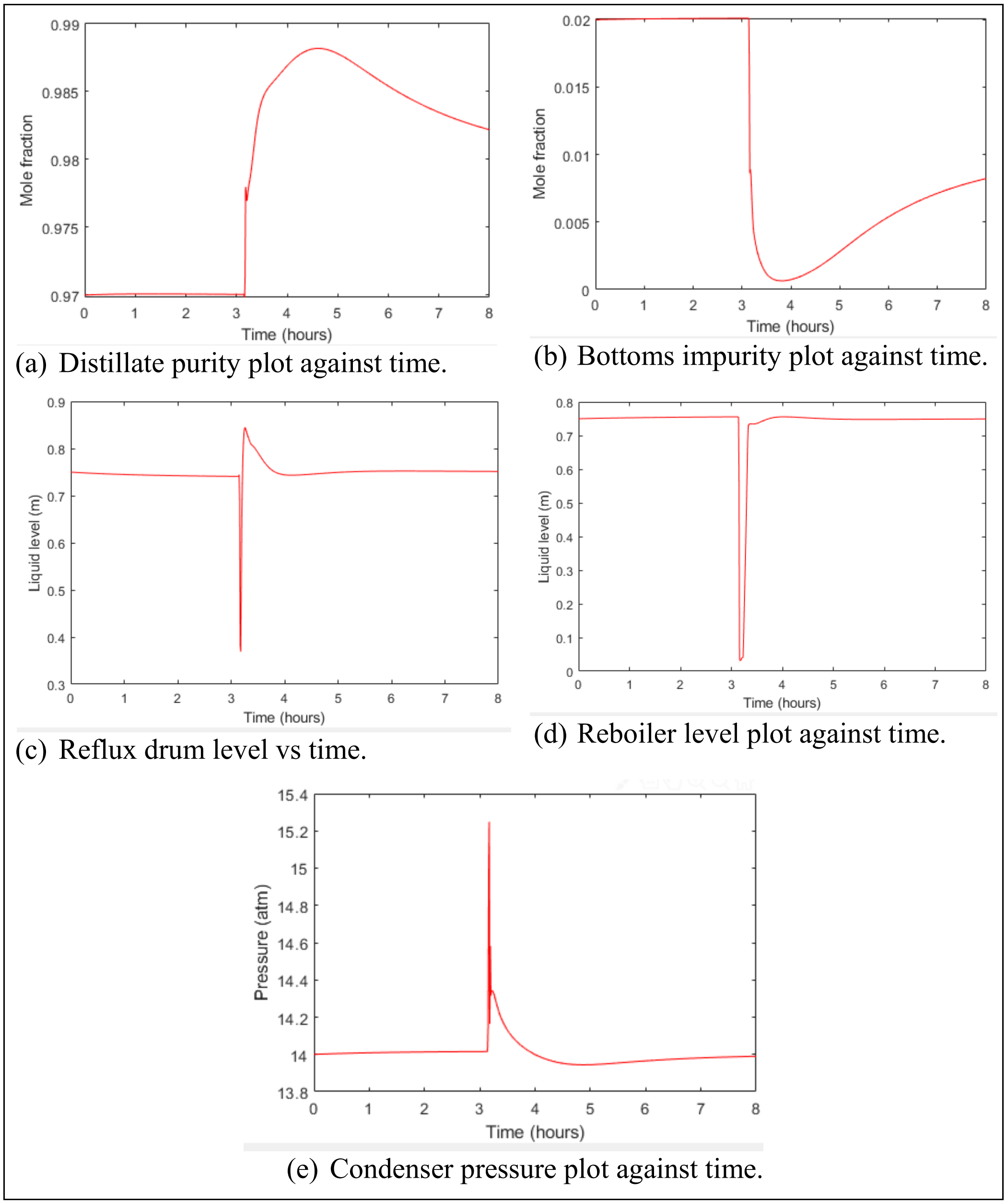
The aim is to collect more than 98% propane in the distillate and more than 99% isobutene (or less than 1% propane) in the bottoms product. There are two inventories in a binary DC which are the liquid level in the reflux drum and the sump (column base) which needs to be controlled. The level of liquid in these drum have to be maintained around a constant value to operate the column under mass balance. The column has been designed to operate under constant pressure hence the pressure has to be controlled as the excessive pressure can risk the mechanical integrity of the column. So in total there are five variables which have to be controlled using five inputs mentioned in the Table 7. In addition to that, feed flow rate is another input to the plant.

Table 7. Control configuration for a distillation column

Controlled variables	Manipulated variables	Control valve
Column pressure, P	Condenser duty, Q_C	Coolant flow
Concentration of distillate, x_D	Reflux flow rate, R	Flux flow
Liquid level, L_C	Distillate flow rate, B	Distillate flow
Concentration of bottoms, x_B	Re-boiler duty, Q_r	Heat flow
Liquid level in column base, L_R	Bottoms flow rate, B	Bottom flow

The column dynamic model is linked to Simulink and the controllers are designed and tuned. PID controller is used to control each variable. The performance of the closed-loop system along with the controllers are validated in simulation by running the model in open loop for 3.1 hours and then switching to the closed loop control scheme till 8 hours. The distillate and bottoms purity setpoints are selected as

0.98 and 0.01 respectively, the column pressure is set at 14 atm and the level set at the reflux drum and reboiler are selected at 0.75 m. Result of the simulation has been presented in Figure 9. The results shows with the introduction of feedback control the PID controllers are able to track the outputs around their desired set-points. Figure 10 presents the control loop diagram of the system.



(a) Distillate purity plot against time.

(b) Bottoms impurity plot against time.

(c) Reflux drum level vs time.

(d) Reboiler level plot against time.

(e) Condenser pressure plot against time.

Figure 9. Aspen Plus Dynamics DC Modeler with controllers in Simulink.

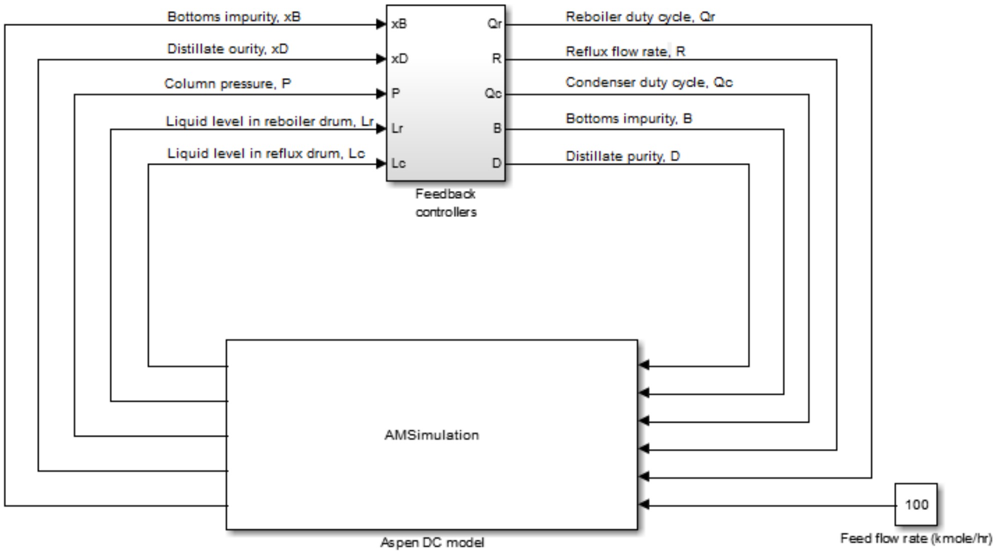


Figure 10. Simulink and Aspen Plus dynamics closed loop model of DC.

2.3.1 Emergency shutdown

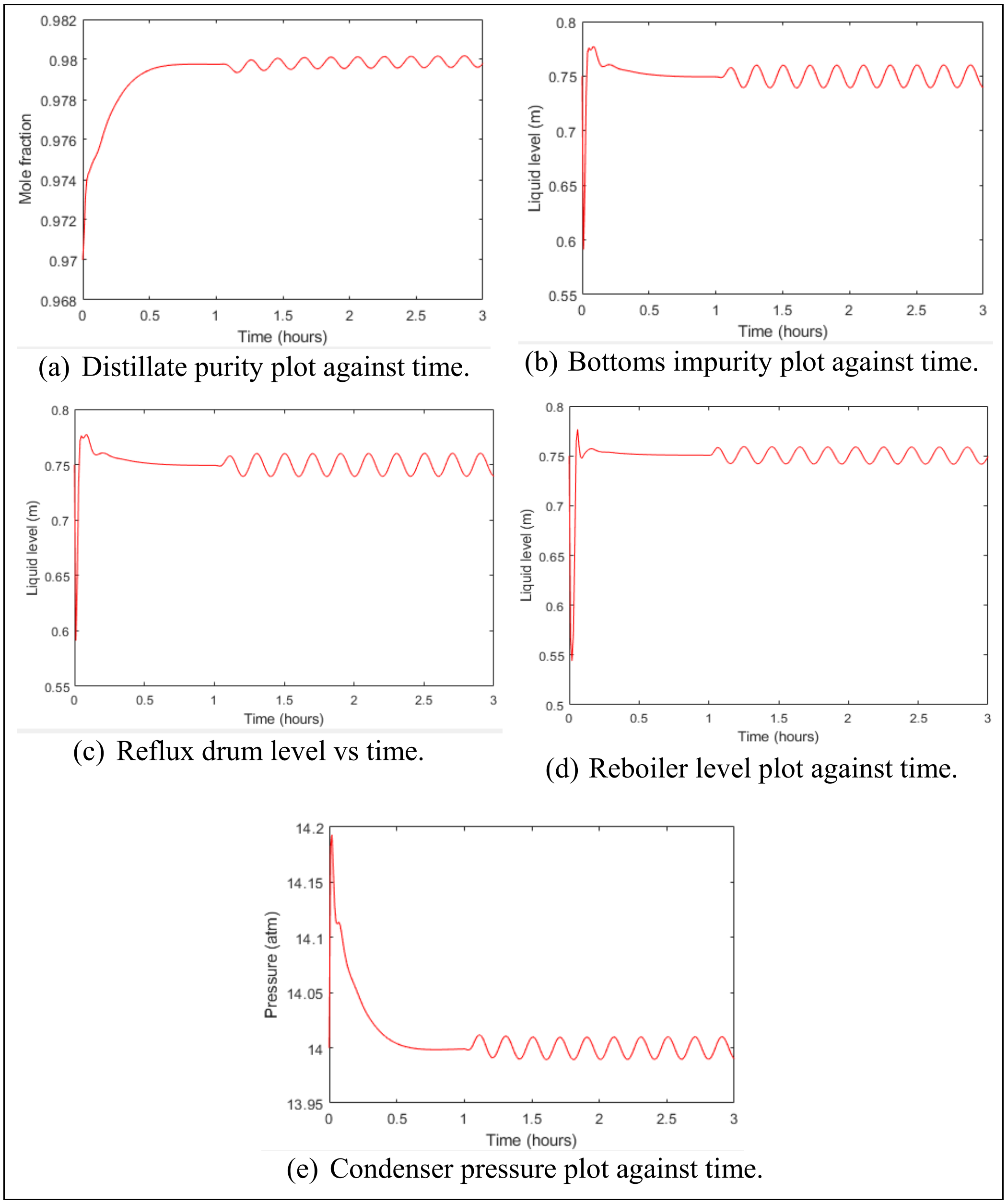
All five controlled variables are critical for the safe operation of the column following its performance specifications. Hence it is necessary to include shutdown thresholds for each of these parameters to avoid operating the column if their values violated the thresholds. Table 8 summarizes the thresholds set for each of the five parameters for emergency shutdown.

Table 8. Emergency shutdown threshold for each controlled variables.

Controlled variables	Maximum value	Minimum value
Column pressure (atm)	20	--
Concentration of distillate (mole fraction)	--	0.90
Liquid level in reboiler (m)	1.0	--
Concentration of bottoms (mole fraction)	0.10	.05
Liquid level in column base (m)	1.0	.05

2.3.2 Effect of feed disturbance of Distillation Column:

The effect of disturbance in the feed flow rate is observed using simulation by injecting a sinusoidal disturbance input with amplitude of 10% of the nominal feed rate and its effect on the system output is observed which has been presented in Figure 11.



(a) Distillate purity plot against time.

(b) Bottoms impurity plot against time.

(c) Reflux drum level vs time.

(d) Reboiler level plot against time.

(e) Condenser pressure plot against time.

Figure 11. Effect of feed disturbance on Distillation Column in closed loop.

The sinusoidal disturbance in feed introduces sinusoidal fluctuation in each output with same frequency as the frequency of the feed disturbance. This is expected as the feed rate is one of the inputs of the DC and thus has a gain with each output. However, the level of fluctuation is different for every output. This is attributed to the fact that the sensitivity of every output is different to the feed flow rate input.

CHAPTER 3: MODELLING CYBER ATTACKS AND THEIR BEHAVIOR ON DISTILLATION COLUMN

This chapter begins by enlisting the attack models under the four categories mentioned previously which are i. sensor attacks, ii. actuator attacks, iii. sensor and actuator attacks, and iv. controller attacks. Following that the effect of these attacks on the performance of the DC has been presented. In case of the grey box model, as there are two feedback control loops hence attacks are primarily injected in these loops. The Aspen Plus Dynamics based hybrid model had five control loops. However, since the column is operated under constant pressure hence attacks are injected in the remaining four control loops. The results of the attacks have been presented in this section.

3.1 Mathematical modelling of cyber attacks

Industrial control systems (ICS) for any physical plant consists of a number of control loops that are responsible for controlling various parameters related to the plant. Each control loop fundamentally contains a controller, sensors, and actuators. The sensors monitor and feed the output data to the controller. The controller evaluates the outputs and accordingly adjusts the actuators to achieve the desired control objective. The evolution in computing and internet technology has encouraged increasing number of these industrial systems to be linked to cyber world which gives rise to a new class of systems called Cyber Physical system (CPS). It is crucial to ensure security of these systems as attacks can not only potentially damage plant infrastructure but also cause financial damage and risk human lives. Figure 12 presents a diagram of a networked CPS under attack under the assumption that the attacker has managed to sneak through the IT security infrastructure to the control systems operating the plant and is capable of launching attack on these systems. This is the worst attack scenario possible.

This chapter will outline the different types of cyber-attacks which can be carried out on cyber physical systems along with their mathematical models. Mathematically modelling of a cyber-attack provides a means of evaluating its effect on the dynamics of a physical plant with the aid of computer simulation. Cyber threats can be categorized in the attack space illustrated in Figure 13, which depicts several attack scenarios as pointed out in [6], [53]. The attacks have been modelled for a nonlinear physical system.

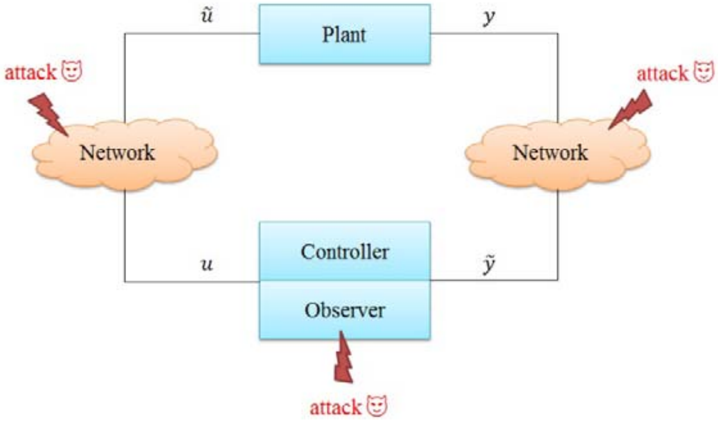


Figure 12. Block level illustration of Cyber Physical System under attack.

Consider the dynamics of a continuous time nonlinear plant as given,

$$\dot{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{t}), \tilde{\boldsymbol{u}}(\boldsymbol{t}), t) + \boldsymbol{w}(t)$$

$$\boldsymbol{y}(\boldsymbol{t}) = \boldsymbol{g}(\boldsymbol{x}(\boldsymbol{t}), t) + \boldsymbol{v}(t) \tag{15}$$

And the dynamics of a discrete time digital nonlinear controller described as below.

$$\dot{\boldsymbol{z}}(\boldsymbol{k}) = \boldsymbol{p}(\boldsymbol{z}(\boldsymbol{k}), \tilde{\boldsymbol{y}}(\boldsymbol{k}), k)$$

$$\boldsymbol{u}(\boldsymbol{k}) = \boldsymbol{q}(\boldsymbol{z}(\boldsymbol{k}), \tilde{\boldsymbol{y}}(\boldsymbol{k}), k) \tag{16}$$

where $\boldsymbol{x}(\boldsymbol{t}) \in R^n$, $\tilde{\boldsymbol{u}}(\boldsymbol{t}) \in R^m$, $\boldsymbol{u}(\boldsymbol{k}) \in R^m$, $\tilde{\boldsymbol{y}}(\boldsymbol{k}) \in R^o$ and $\boldsymbol{y}(\boldsymbol{t}) \in R^o$ represents the system states, input to the plant actuators, controller output, controller input and output from the plant sensors respectively with n states, m inputs and o outputs. Functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are nonlinear functions describing the system state dynamics and relationship between system output and states. $\boldsymbol{w(t)}$ and $\boldsymbol{v(t)}$ represents the process noise and measurement noise, respectively.

Similarly functions $\boldsymbol{p}$ and $\boldsymbol{q}$ represent the nonlinear controller dynamics and relationship between controller states and output. It should be noted that the plant is considered as a continuous time system and the controller as discrete time digital system as a controller can be viewed as a tailored computer en-tasked to control the plant.

The controller parameters includes the controller internal state $\boldsymbol{z(k)}$, the plant sensor input $\boldsymbol{\widetilde{y}(k)}$ and the output of the plant actuator input from the controller ***u(k).*** The sampling is done following zero order hold model. If the network integrity is maintained, at every sampling instant $\boldsymbol{k}$, the output from the plant should be available precisely at the input of the controller i.e. $\boldsymbol{\widetilde{y}(k) = y(t = k)}$ and similarly the output from the controller should be available to the actuators precisely i.e. $\boldsymbol{u(k) = \widetilde{u}(t = k)}$. The attacks can be modelled in a three-dimensional attack space each of which are described below:

I. **Adversary's a priori system model knowledge resource:** This represents the resource available to the attacker which is used to launch complex attacks which are hard to detect and can cause fatal consequences if not mitigated.

II. **Adversary's disclosure resource:** The attacks uses these resources to gather and accumulate sensitive plant related data. The resources accessible to the attacker are the real plant output measurements ***y(k)*** and the actuator control actions ***u(k)***. These attacks do not directly affect the plant dynamics but can be used to launch more deadly attacks like the replay attack which is the case in Stuxnet attack.

III. **Adversary's disrupting resource:** This resource can be used by the adversary to affect the system operation, which happens for instance when data integrity or availability properties are violated. These kinds of attacks modify the control actions $\boldsymbol{u}$ and sensor measurements $\boldsymbol{y}$ from their calculated or real values to the corrupted signals $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{y}}$, respectively. The various existing cyber-attacks found in literature has been mapped in a three dimensional attack space which is illustrated in Figure 13.

There has been ongoing research in the area of attack modelling from Control systems engineers to design attack diagnosis, detection, recovery, tolerance and elimination tools for ICS. Every attack mathematically modelled is placed under one of the four attack categories which includes: i. Sensor attacks, ii. Actuator attacks, iii. Sensor and Actuator attacks, iv. Controller attacks.

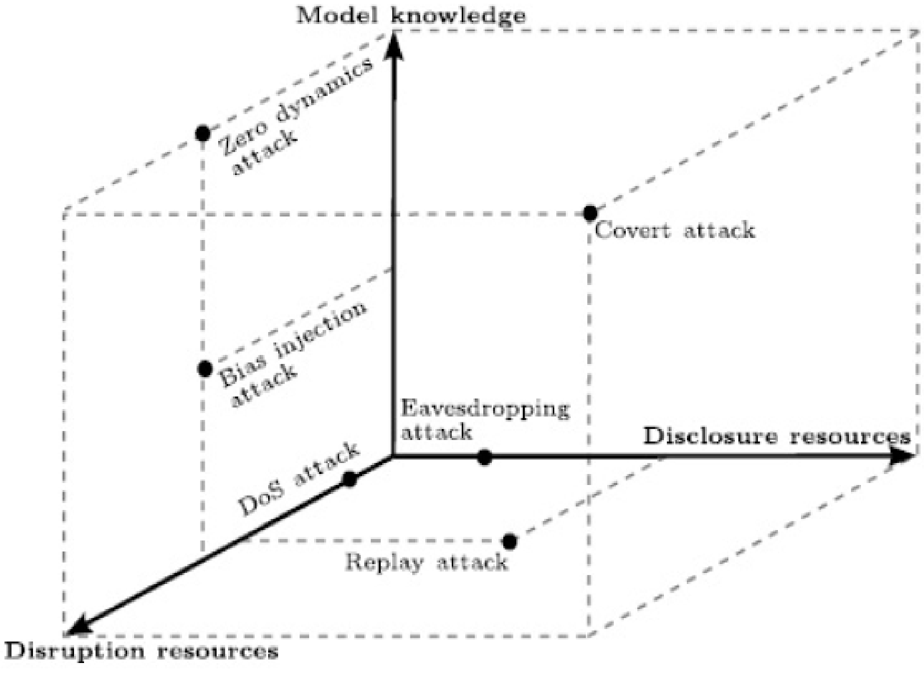


Figure 13. A three dimensional attack space containing mapping of various attacks which can be launched on ICS of CPS [53].

3.1.1 Sensor attacks

List of sensor attacks modelled are presented below. For all attacks T_a represents the attack period, $y_i(t)$ and $\tilde{y}_i(k)$ are the the *i-th* sensor measurement and sensor reading to the controller respectively.

- **Scaling Attack** [54]: During this attack the true sensor measurements are scaled to a different (higher or lower) value depending on the scaling parameter value λ_s as shown below.

$$\tilde{y}_i(t) = \begin{cases} y_i(t) & t \notin T_a \\ (1+\lambda_s)y_i(t) & t \in T_a \end{cases} \tag{17}$$

where λ_s is a constant.

- **Ramp Attack** [54]: As part of this attack the true sensor measurements are modified by adding a ramp function which gradually increases/decreases with time based on the gradient of ramp denoted by λ_r, as shown below.

$$\tilde{y}_i(t) = \begin{cases} y_i(t) & t \notin T_a \\ y_i(t) + \lambda_r . t & t \in T_a \end{cases} \tag{18}$$

- **Pulse Attack:** This type of attack involves modifying measurements through temporally-spaced short pulses with attack parameter λ_p [54].

$$\tilde{y}_i(t) = \begin{cases} y_i(t) & t \notin T_a \\ a(t)\, y_i(t) & t \in T_a \end{cases} \tag{19}$$

where *a(t)* is a binary signal which can either be 0 or 1.

- **Random Attack:** This attack involves the addition of randomly generated positive attack values generated by a uniform random variable rand(a,b) as [5], [54].

$$\tilde{y}_i(t) = \begin{cases} y_i(t) & t \notin T_a \\ y_i(t) + rand\ (a,b) & t \in T_a \end{cases} \tag{20}$$

- **Bounded time varying attack** [20]**:** The attack signal $\tilde{y}_i(t)$ is assumed to be bounded with unknown bound i.e.

$$\tilde{y}_i(t) = \begin{cases} y_i(t) & t \notin T_a \\ y_i(t) + a_i(t) & t \in T_a \end{cases} \tag{21}$$

where, $\|y_i(t)\|_2 \leq a$ for t $\geq 0,$ where a is unknown constant.

Moreover, it is assumed that,

$\|\dot{y}_i(t)\|_2 \leq b$ for t $\geq 0,$ b is small number, i.e. $y_i(t)$ is a slow time varying signal.

- **Denial-of-Service (DoS) Attack**: The DoS attack can be launched by jamming the communication channels, flooding packets in the network, and compromising devices to prevent data transfer, etc. [5].

As the lack of available sensor data, the DoS attack can be modeled as:

$$\tilde{y}_i(k) = \begin{cases} y_i(k) & k \notin T_a \\ (1 - D_s(k))\, y_i(k) + D_s(k)\, y_i(k - n) & k \in T_a \end{cases} \tag{22}$$

where D_s is a binary index and takes a value of 1 to resemble a scenario when a packet is denied and becomes 0 to resemble a scenario when a packet is not denied. To encompass energy limitations in [4], it is assumed that, within the attack time horizon *Ta*, the sensor can send at most *M* data packets, while the attacker can launch DoS attack at most *N* times where *N<M.* In (22) n is the number of consecutive packets which are jammed by the attacker and hence can take values from n = {1, 2, …….. N}. DoS attack is able to make the data channels unavailable by accessing the disruption resources. Note, a priori knowledge of the disruption resources are needed [6].

- **False Data Injection (Bias injection) attack:** In case of a False Data Injection attack, it is assumed that the attacker knows the system model and also has access to disruption resource [6].

A vector $\boldsymbol{y_a(k)}$ in (16) is called a (a,b)-data injection attack if there exists an index set $i \in$ A, where A is the set of manipulated measurements and A ⊂ P ≜ {1, . . . , m}, such that

(i) $\|\boldsymbol{y_a}(\boldsymbol{k})\|_{\ell 0} \leq a$

(ii) $y_{a,i}(k) = 0$, $\forall i \in$ P\ A;

(iii) $y_{a,i}(k) \neq 0, \forall i \in$ A.

To implement this class of attack, it requires the attacker to have the knowledge of either the measurements information ***y(t)*** or the topology configuration ***f*** and ***g***. Specifically, data injection attack can be written in the form of

$$\tilde{y}_i(t) = \begin{cases} y_i(k) & k \notin T_a \\ y_a(k) & k \in T_a \end{cases} \tag{23}$$

where $y_a(k)$ is the sensor measurement during attack.

3.1.2 Actuator attacks:

The attacks have been modelled for the plant and controller model given in (14) and (15) respectively. In all attacks T_a represents the attack period, $u_i(t)$ and $\widetilde{u_\iota}(t)$ are the the *i-th* controller output and plant actuator input respectively.

- **Data injection attack:** The data injection attack is modeled as follows

$$\tilde{u}_i(t) = \begin{cases} u_i(t) & t \notin T_a \\ a_r\, u_i(t) & t \in T_a \end{cases} \tag{24}$$

Where a_r is a constant. As per [9], the attacker has the knowledge of closed-loop dynamics, or of open-loop dynamics and the network operator's control law and hence can choose a_r accordingly to impart maximum damage.

- **Denial-of-Service (DoS) Attack:** The DoS attack can be modeled similar to that for the sensor as:

$$\tilde{u}_i(k) = \begin{cases} u_i(k) & k \notin T_a \\ (1 - D_a(k))\, u_i(k) + D_a(k)\, u_i(k - n) & k \in T_a \end{cases} \tag{25}$$

where D_a is a binary index that can be 1 (for packet jam) and 0 otherwise. Due to energy restriction, if *M* data packets have been transmitted during attack period T_a the attacker can launch DoS attack at most *N* times where *N<M.* In (22) n can take values from {1, 2, ……. N} and corresponds to the number of successive data packets jammed during an attack.

3.1.3 Sensor and actuator attacks:

Cyber-attacks corrupting states (sensors) and outputs (actuators) of the non-linear cyber-physical systems are addressed in this section. The plant and controller model in (14) and (15) under sensor and actuator attacks can be described in an alternative form given below.

$$\dot{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{t}), \boldsymbol{u}(\boldsymbol{t}) + \boldsymbol{a_u}(t), \boldsymbol{t}) + \boldsymbol{w}(t)$$

$$\boldsymbol{y}(\boldsymbol{t}) = \boldsymbol{g}(\boldsymbol{x}, t) + \boldsymbol{v}(t) \qquad (26)$$

And the dynamics of a discrete time digital nonlinear controller described as below.

$$\dot{\boldsymbol{z}}(k) = \boldsymbol{p}\big(\boldsymbol{z}(\boldsymbol{k}), \boldsymbol{y}(\boldsymbol{k}) + \boldsymbol{a_y}(\boldsymbol{k}), \boldsymbol{k}\big) + \boldsymbol{e}(\boldsymbol{y})$$

$$\boldsymbol{u}(k) = \boldsymbol{q}\big(\boldsymbol{z}(\boldsymbol{k}), \boldsymbol{a_y}, k\big) \qquad (27)$$

where $\boldsymbol{a_u} \in R^m$ and $\boldsymbol{a_y} \in R^o$ are vector functions representing attack signals.

- **Bounded time-varying attack** [12]**:** The following model for attack signals is considered both in the case of actuator and sensors attacks:

$$\boldsymbol{a_u}(t) = \boldsymbol{a_y}(t) = \boldsymbol{\gamma}(\boldsymbol{x}, t) \qquad (28)$$

where $\boldsymbol{\gamma}(\boldsymbol{x}, t)$ is an unknown but bounded function.

Available bounds for the function $\boldsymbol{\gamma}(\boldsymbol{x}, t)$ and for its time derivative are considered as

$$|\boldsymbol{\gamma}(\boldsymbol{x}, t)| \leq \rho_1(\boldsymbol{x}, t); \ |\dot{\boldsymbol{\gamma}}(\boldsymbol{x}, t)| \leq \rho_2(\boldsymbol{x}, t); \qquad (29)$$

- **Replay Attack** [10]**:** In case of replay attack the models given by (14) and (15) have been used. The replay attack has access to disclosure and disruption resources. At first, the adversary gathers sensor readings with a disclosure attack. Subsequently the attacker replays this collected data to the controller, which renders his primary interference invisible for any diagnosis system on the controller side.

Stage 1 (*$0 \leq t < t_o$*): gather sensor readings

$$\begin{bmatrix} \boldsymbol{a_u}(t) \\ \boldsymbol{a_y}(t) \end{bmatrix} = 0$$

$$I_K = \begin{bmatrix} \gamma_u & 0 \\ 0 & \gamma_y \end{bmatrix} \begin{bmatrix} \boldsymbol{u}(t) \\ \boldsymbol{y}(t) \end{bmatrix} \tag{30}$$

The collected data is stored in I_K.

Stage 2 (*$0 \leq t < 2t_o$*): relay and interfere:

$$\begin{bmatrix} \boldsymbol{a_u}(t) \\ \boldsymbol{a_y}(t) \end{bmatrix} = \begin{bmatrix} -\boldsymbol{u}(t) + \ \boldsymbol{u_a}(t - t_o) \\ -\boldsymbol{y}(t) + \ \boldsymbol{y_a}(t - t_o) \end{bmatrix}$$

$$I_K = I_{K-1} \tag{31}$$

3.1.4 Controller attacks

The controller attack has been modelled as the following presented in [7].

$$\dot{\boldsymbol{z}}(k) = \boldsymbol{p}(\boldsymbol{z}, \widetilde{\boldsymbol{y}}, k) + \triangle\, \boldsymbol{p}(\boldsymbol{z}, \widetilde{\boldsymbol{y}}, k)$$

$$\boldsymbol{u}(k) = \boldsymbol{q}(\boldsymbol{z}, \widetilde{\boldsymbol{y}}, k) \tag{32}$$

where, $\boldsymbol{z}, \widetilde{\boldsymbol{y}}$ and $\mathbf{u}$ represents the controller states, input from the plant to the controller and output of the controller to the plant respectively. $\triangle\, \boldsymbol{p}(\boldsymbol{z}, \widetilde{\boldsymbol{y}}, k)$ corresponds to a moderate change in the controller state dynamics. It is assumed that the attacker has the capability to intrude the logical part of the cyber physical system and change the dynamics of the controller.

3.2 Cyber attack and effect on distillation column:

3.2.1. Attack surface identification

Advancement in the area of computer science and IT has led to critical physical infrastructures being increasingly connected to the cyber world giving rise to CPS. This has led to these systems becoming more vulnerable to cyber-attacks. The attackers usually launch attacks to cause maximal damage to the physical infrastructure with the aim of either causing financial damage or damage to human lives. The thesis assumes that the attacker has breached the IT security configurations and firewall settings and has access to the hardware linked to the physical plant.

In order to protect these systems from external intruders the first step is to draw a footprint of the system highlighting the infrastructures that are critical to ensure normal operation of the system which can be potentially targeted by the attackers.

Identification of the attack surface is the first step towards designing attack detection, diagnostics, recovery and elimination tools for CPS. As mentioned before a continuous binary DC has five DoF as there are five manipulated variables, which includes distillate and bottoms flow rate, reflux and vapor flow rate and condenser duty cycle. The five variables are responsible for controlling five outputs of the system, which are distillate and bottoms product concentration, condenser liquid level, bottoms liquid level and column pressure.

In the steady state operation, the distillation column is made to operate around steady state operating values and the control objective is to maintain the output at their desired values. This is achieved by using mass and energy balance. Mass balance is achieved by balancing the feed flow rate against the distillate and bottoms flow rate. This is necessary as the amount of lighter component which can be extracted per hour from the column (as distillate and bottoms product) cannot exceed the amount of lighter component fed to the column per hour. Similarly rate at which the heavier component is extracted from the column can’t exceed the rate it is fed to the column. Energy balance is done by balancing the energy input to the column against the energy output from the column. A distillation column has primary and secondary energy ports. There are two primary energy input ports which are the feed preheater power and reboiler power and one energy output port which is the condenser duty. On the other hand, the secondary energy port is the reflux flow rate which can be used to extract energy from the DC. Energy imbalance alternates the temperature and pressure inside the column which affects the quality of the product streams. The aim of the attacker will be to upset the mass and energy balances to disrupt the normal operation of the distillation column.

In order to determine the critical points in a DC, motivation has been drawn from the research done in the area of fault diagnosis, fault recovery and fault tolerant control for DCs. These literatures predominantly propose online fault diagnosis and control tools thereby only considering faults on systems which are directly connected to the company network. The literatures aim to identify the most critical parts of the system which are susceptible to faults and their effect on the performance of the DC. It will be a fair assumption to believe the attacker is limited to targeting the controllers, sensors and actuators as they are the only hardware which are part of the control loop and are integrated to the cyber world. The proposed model of binary DC in section 2.2 and 2.3 have 2 and 5 control loops, respectively. It is assumed that each of these control loop will be networked. The attacker can upset the control loop by injecting one of the attacks modelled in section 3.1 by either targeting the sensor, actuator, sensor and actuator, controller or a combination of these hardware’s.

3.2.2: Modelling effect of attack on proposed model of a distillation column

This section demonstrates the effect of the attack models proposed in the former section on the performance of the DC. As mentioned previously the proposed control scheme contains two control loops one of each are used to maintain quality of one of the two products. Thus attacks are injected in these two control loops and their effect on the products quality are observed which have been included in this section. All the attacks are injected between t = 100 – 200 hours during simulation. During simulation process noise and measurement noise have been added as following:

$$\boldsymbol{x}(\boldsymbol{t}) = \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{t}), t) + \boldsymbol{w}(\boldsymbol{t})$$

$$\boldsymbol{y}(\boldsymbol{t}) = \boldsymbol{g}(\boldsymbol{x}(\boldsymbol{t}), t) + \boldsymbol{v}(\boldsymbol{t}) \tag{33}$$

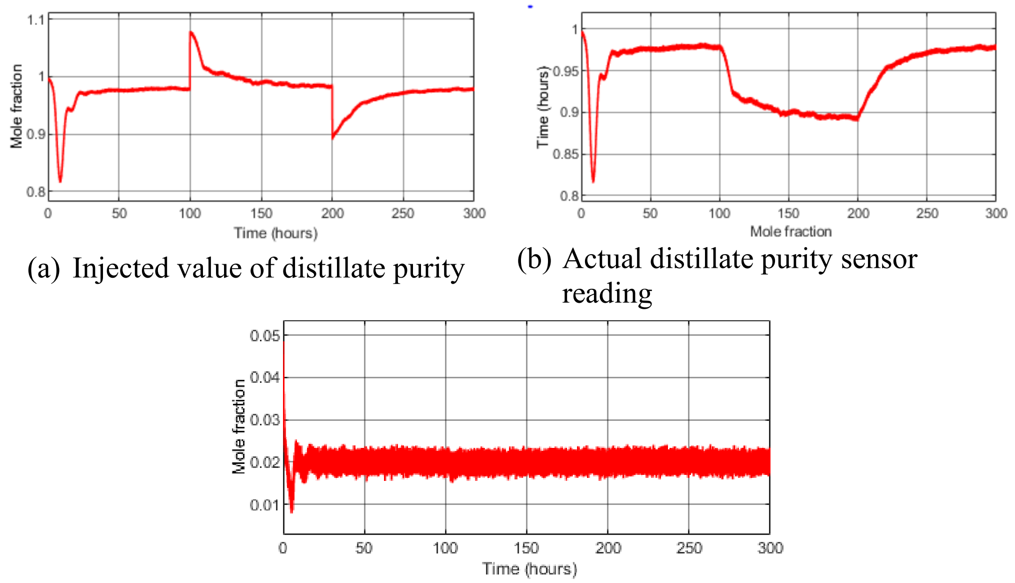
where $\boldsymbol{y}(\boldsymbol{t})$ and $\boldsymbol{v}(\boldsymbol{t})$ are output vector and noise vector, and $\boldsymbol{x}(\boldsymbol{t})$, and $\boldsymbol{w}(\boldsymbol{t})$ are state vector and process noise vector, respectively. Zero mean Gaussian White random noise are added to outputs and states. The system has 2 outputs and 58 states in total which includes the molar holdup, liquid and vapor flow rates and molar concentration of lighter product in each tray.

3.2.2.a Sensor attacks:

- **Scaling Attack:** This attack combines disclosure and disruption of resources. Once the sensor output is available the attacker can inject the modified measurement and disrupt the resource. This attack is injected as in equation (17) by setting the value of λ_s as 0.1 for distillate purity measurements and $\lambda_s = -0.95$ for bottoms impurity measurements. The results are presented in Figure 14 and Figure 15 respectively.

It can be observed from the results in Figure 14 that by affecting data integrity the distillate purity can be reduced below the required specification, thus making the product less valuable incurring financial loss; however the bottoms product quality is regulated within the required level by the controller with some transients.

Similarly illustrated in Figure 15 is effect of attacking the bottoms impurity sensor integrity which manifests on the column by increasing the impurity levels in the bottoms product thus causing financial loss; however the top product quality is maintained within required specification by the controller. In case of both the attacks had no impact on the remaining column outputs. To summarize the motive for this attack will be primarily to incur financial loss.

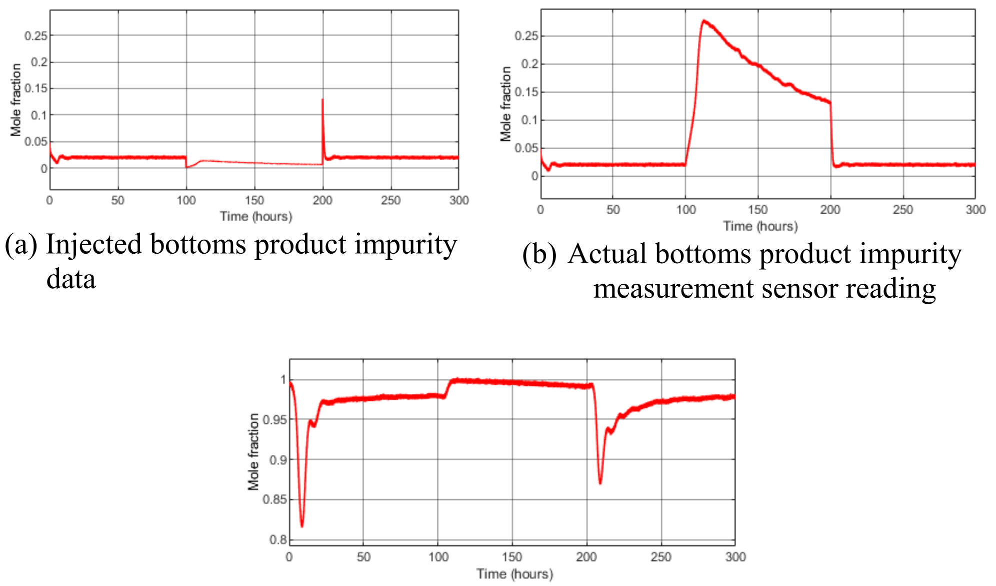


(a) Injected value of distillate purity

(b) Actual distillate purity sensor reading

(c) Impurity in bottoms product

Figure 14. Results illustrating effect of a attack on the distillate purity sensor.



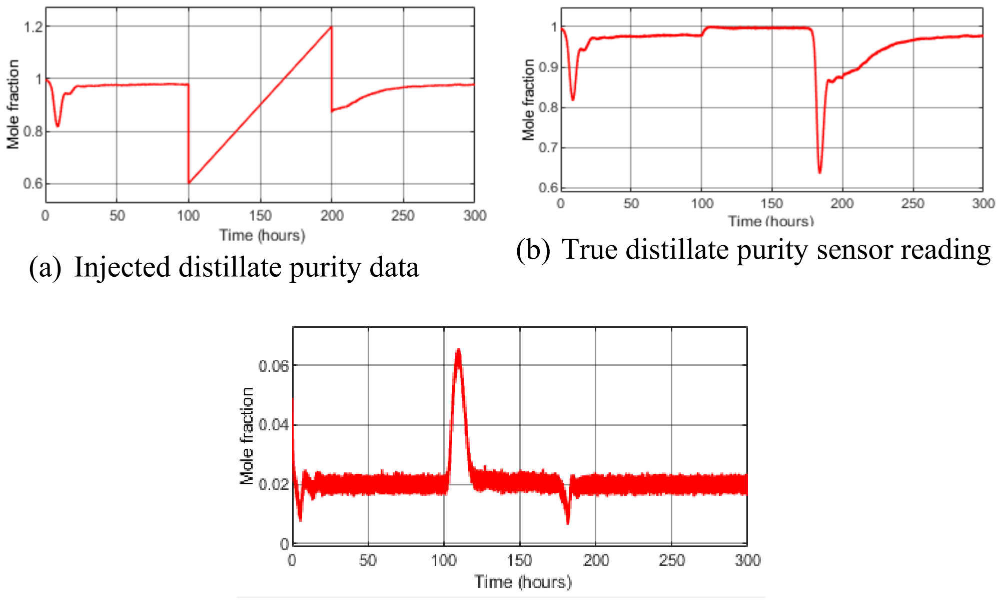
(a) Injected bottoms product impurity data

(b) Actual bottoms product impurity measurement sensor reading

(c) Distillate purity reading

Figure 15. Effect of a scaling attack on the bottoms impurity sensor.

- **Ramp attacks:** Results are presented in Figure 16 and Figure 17 respectively.

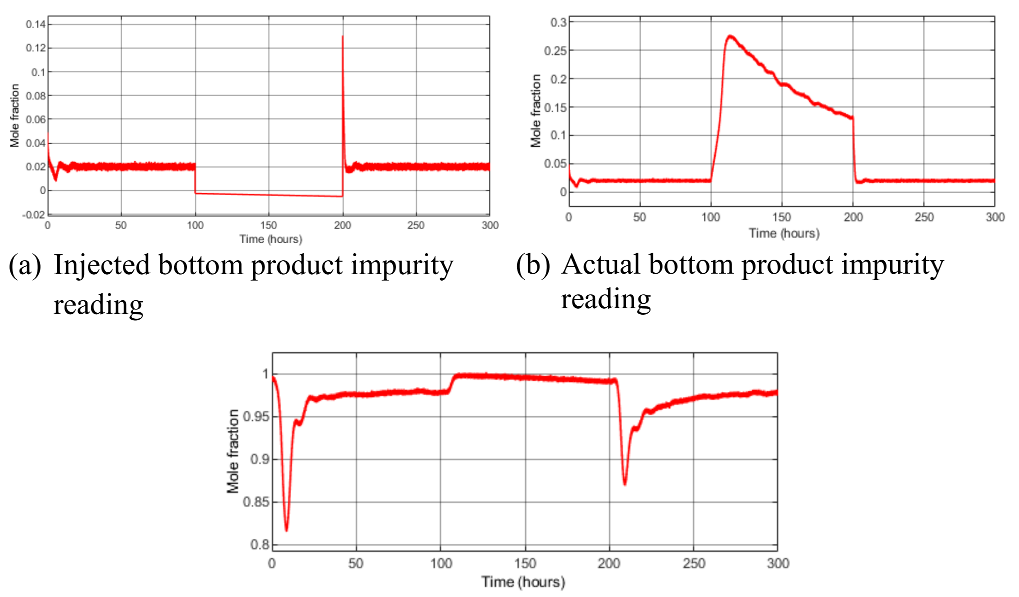


(a) Injected distillate purity data

(b) True distillate purity sensor reading

(c) Bottom product purity reading

Figure 16. Effect of a ramp attack on the distillate purity sensor.



(a) Injected bottom product impurity reading

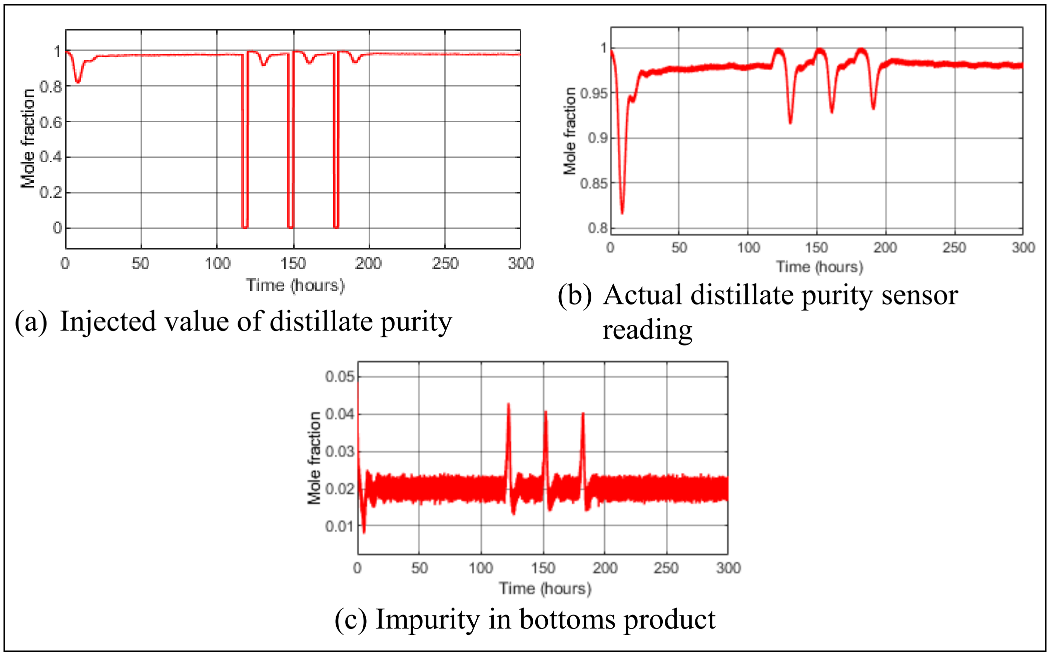
(b) Actual bottom product impurity reading

(c) Distillate purity plot

Figure 17. Effect of a ramp attack on the bottoms impurity sensor.

This type of attack involves disclosure and disruption of resources similar to scaling attack. Once sensor measurements are available they are modified and injected to the controller. The attack on distillate purity measurement sensor is simulated setting $\lambda_r = 0.006$ and the results are presented in Figure 16. The effect of ramp attack on bottoms impurity measurement sensor is carried setting value of λ_r = -.000025 and the simulation results are shown in Figure 17. Results shows that the attacks only degrade the quality of the product whose sensor is under attack; the remaining outputs remain unaffected by the attack. Presence of controller adds robustness to the control loop which is unaffected by attack by maintaining the product quality within specification during attack.

- **Pulse attack:** The attack on distillate purity is carried out by setting *a(t)* as a continuous time unit amplitude pulse with period of 30 hours and pulse width of 90%.

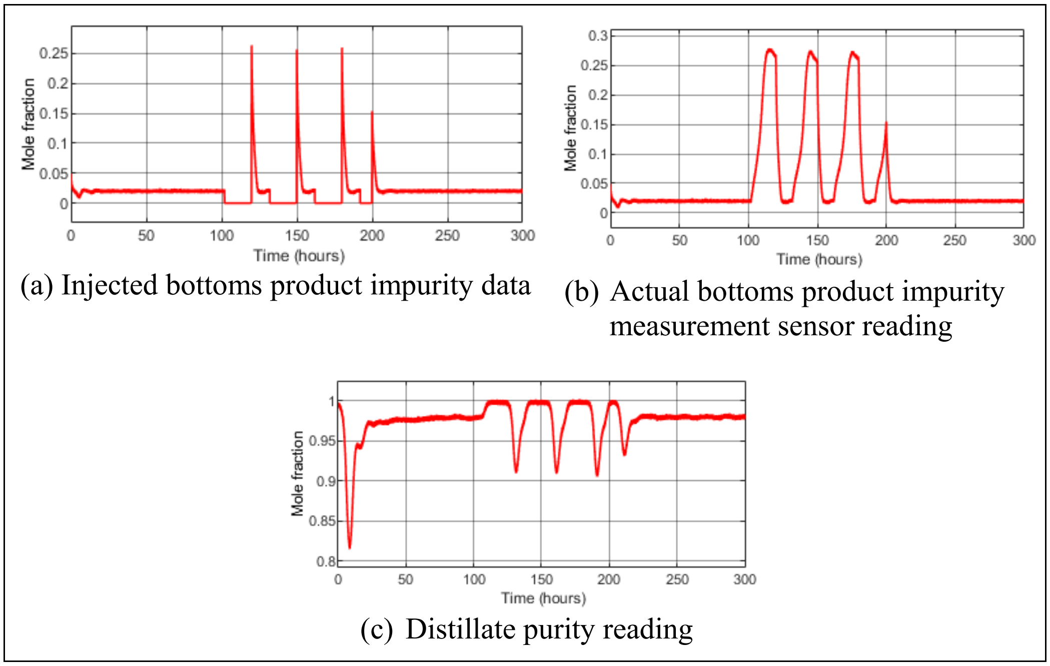


(a) Injected value of distillate purity

(b) Actual distillate purity sensor reading

(c) Impurity in bottoms product

Figure 18. Effect of a pulse attack on the distillate purity sensor.



(a) Injected bottoms product impurity data

(b) Actual bottoms product impurity measurement sensor reading

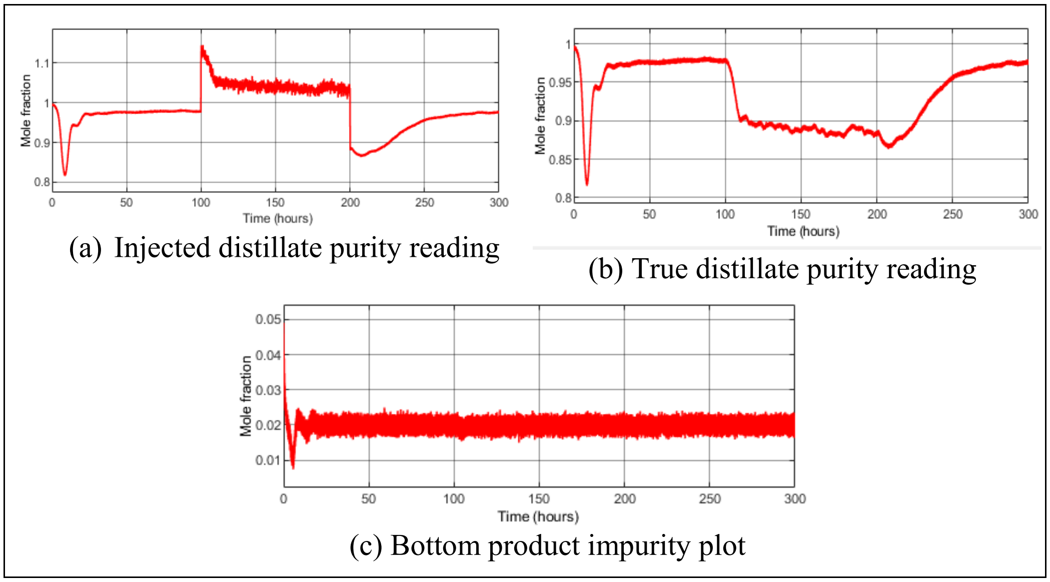
(c) Distillate purity reading

Figure 19. Effect of a pulse attack on the bottoms impurity sensor.

From the result presented below in Figure 18 it can be seen that the attack causes the distillate purity to fluctuate in a periodic manner dropping marginally below the 98%

requirement without any significant degradation in product quality. In case of bottoms impurity, the short pulses caused significant increase in bottoms impurity violating the product specification. Once the attack is removed the controller is able to return the purity back to the required level for both the products. It is observed that the greater the pulse width the greater the drop in the quality for both the distillate and bottoms purity.

- **Random Attack:** The attack vectors generated using random signals normally disturbed as N ~ (.15, .00005) and N ~ (-.08, 0.0000001) are fed separately as distillate purity readings and bottom impurity measurements respectively and there effects of on the column performance have been illustrated in Figure 20 and Figure 21 respectively. It is observed that the attacks are capable of upsetting the product qualities causing financial loss. However, except the control loop which is attacked, the feedforward and feedback controllers are able to regulate the remaining outputs around their desired values thus providing some degree of robustness against the attacks.

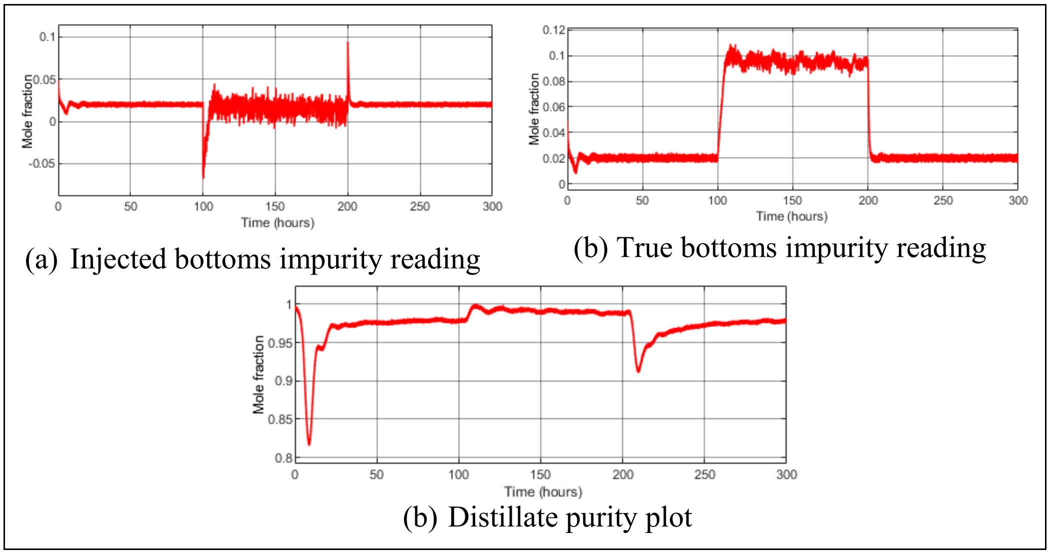


(a) Injected distillate purity reading

(b) True distillate purity reading

(c) Bottom product impurity plot

Figure 20. Illustration of the effect of a random attack on the distillate purity sensor.



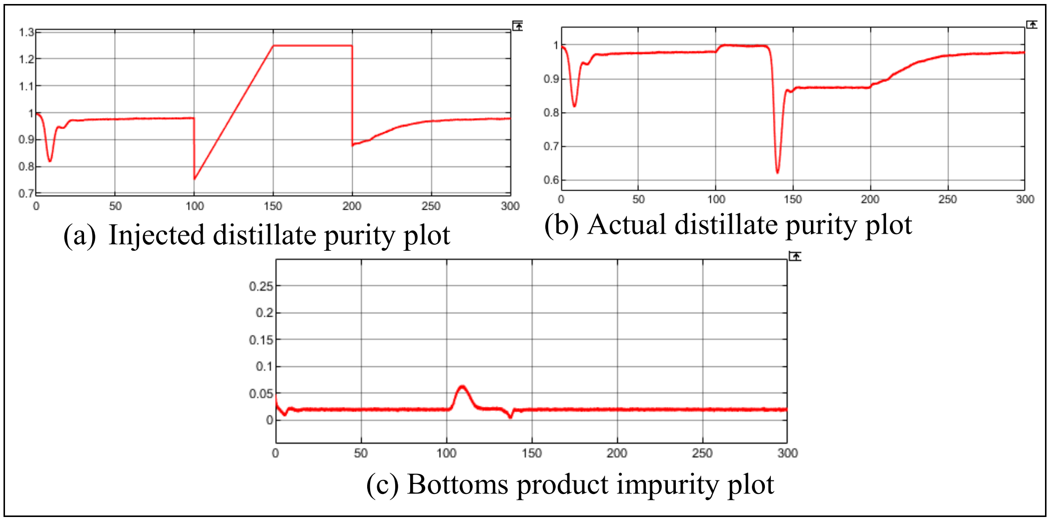
(a) Injected bottoms impurity reading

(b) True bottoms impurity reading

(b) Distillate purity plot

Figure 21. Effect of a random attack on the bottoms impurity sensor.

- **Bounded time varying attack:** The attack signal is chosen as $y_i\ (t)= .75-sat\ ((t-100)/50)$ for the distillate purity measurement sensor and the results obtained have been presented from Figure 22. For bottoms impurity, the attack signal is chosen as $y_i\ (t) = 1.58-tan^{-1}(t)$ and the effect has been demonstrated in Figure 23.

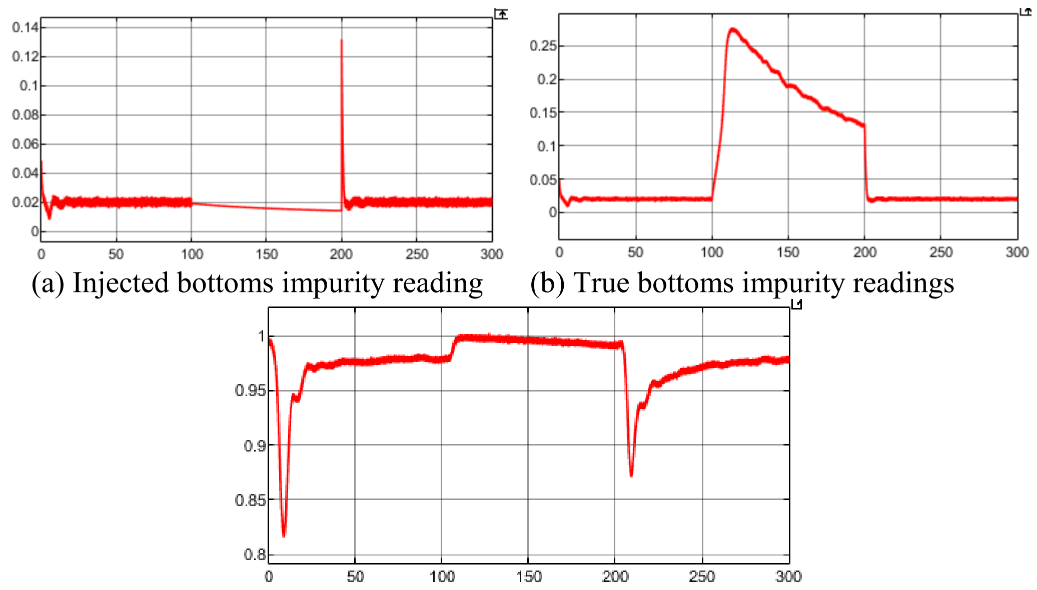


(a) Injected distillate purity plot

(b) Actual distillate purity plot

(c) Bottoms product impurity plot

Figure 22. Effect of a bounded attack on the distillate purity sensor.



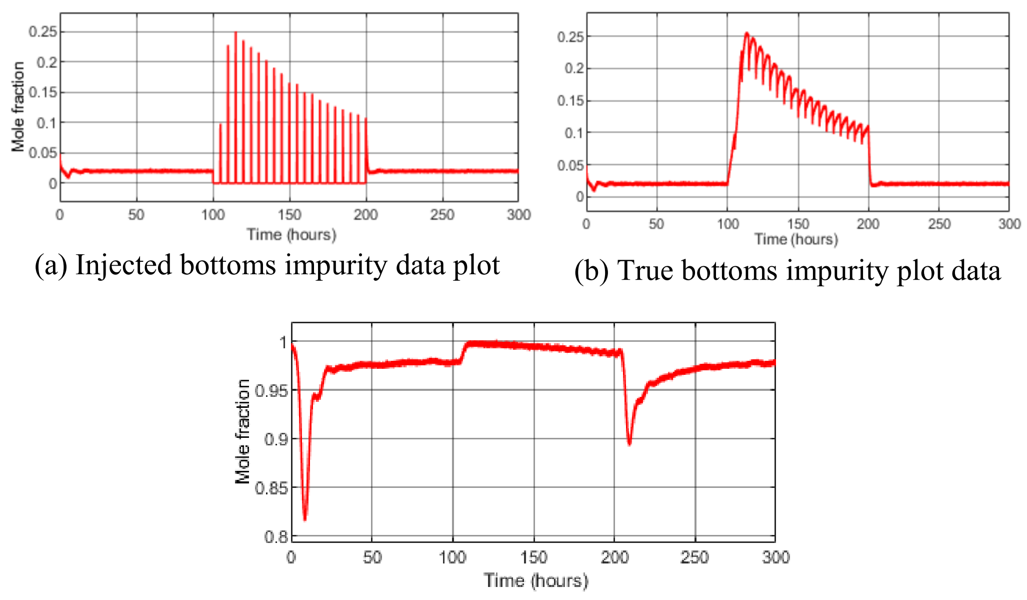
(a) Injected bottoms impurity reading (b) True bottoms impurity readings

(c) Bottoms product impurity plot

Figure 23. Effect of a bounded attack on the bottoms impurity sensor.

Both the attacks cause damage to the product quality of the loop being attacked violating the purity specifications which will introduce financial loss. The remaining outputs are unaffected by the attacks suggesting that the presence of control scheme provides some degree of robustness against the attack. Besides that, once the attack subsided the controller is able to return the product qualities back to the desired levels thus providing attack recovery.

- **Denial-of-Service (DoS) Attack:** The attack is simulated using square wave pulses with period of 5 hours and pulse width of 5 % with amplitude 1 and 0. When the value of the pulse is zero the data exchange between the controller and actuator is blocked. This is achieved by sending the last available output value from the controller before the blockage in every subsequent sample. As the pulse width is 95% hence for 4.75 hours the communication between the controller and actuator is interrupted during each pulse thus denying data availability. The attack results are presented in Figure 24.



(a) Injected bottoms impurity data plot

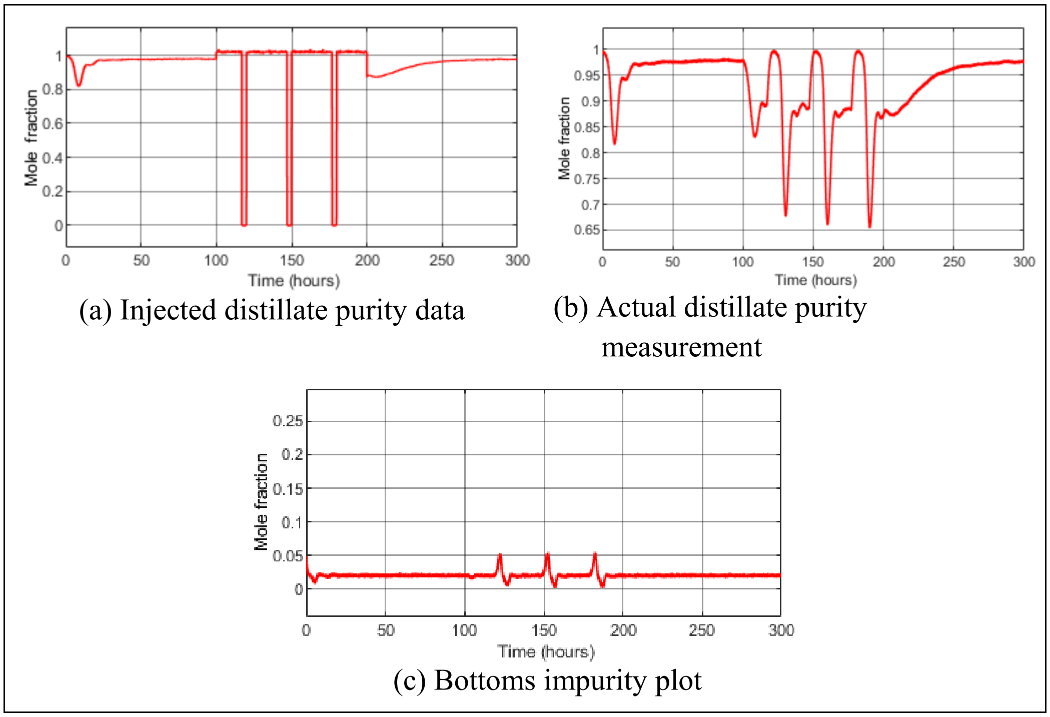
(b) True bottoms impurity plot data

(c) Distillate purity plot

Figure 24. Effect of a DoS attack on the bottoms impurity sensor.

Looking at Figure 24, the attack upsets the quality of bottoms product reducing its purity. The distillate purity increases during the attack which is due to increase in bottoms impurity. Once the attack subsides, the controller is able to track the output back to the desired value.

- **False Data Injection (Bias injection) attack:** The false data has been generated using normally distributed random signals multiplied by a continuous time pulse in order to bias the sensor output data. The random signals for distillate measurement and bottoms impurity measurements are chosen as N~ (1.02, .00002) and N ~ (.01, .000001) respectively. In case of distillate measurements, the pulse time period is selected as 30 hours with pulse width of 90% while for bottoms impurity the pulse time period is set to 5 hours with pulse width of 80%. The results of attacks on distillate purity and bottoms impurity have been presented in Figure 25 and Figure 26 respectively.

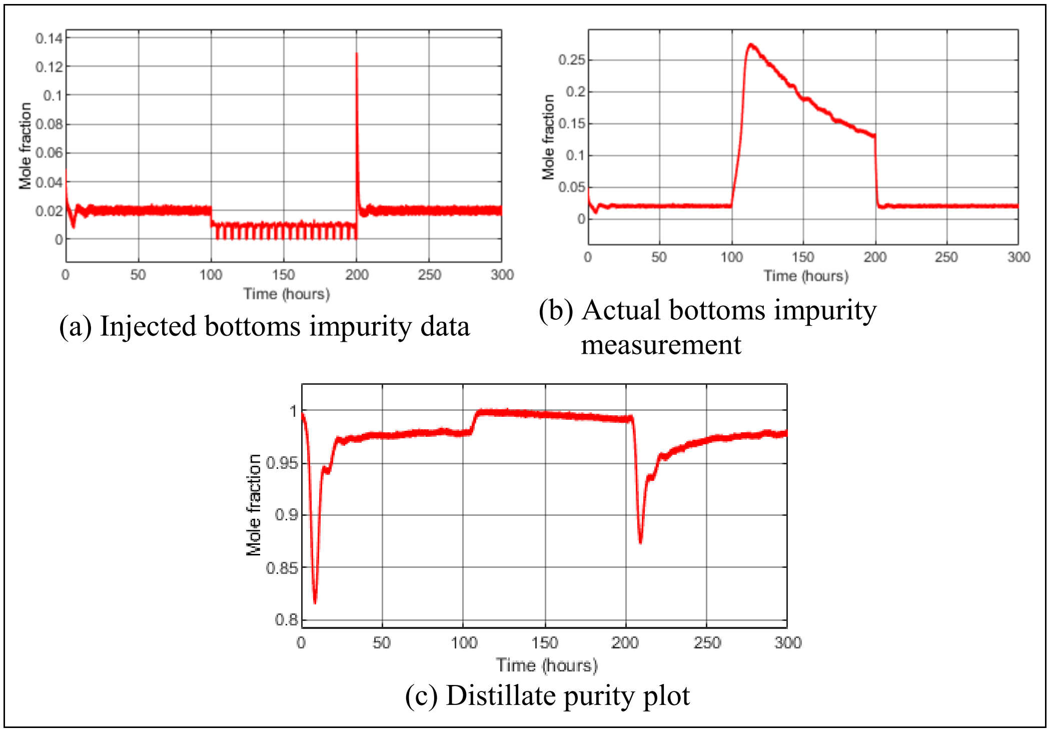


(a) Injected distillate purity data

(b) Actual distillate purity measurement

(c) Bottoms impurity plot

Figure 25. Column performance against a false data injection attack on distillate purity sensor.



(a) Injected bottoms impurity data

(b) Actual bottoms impurity measurement

(c) Distillate purity plot

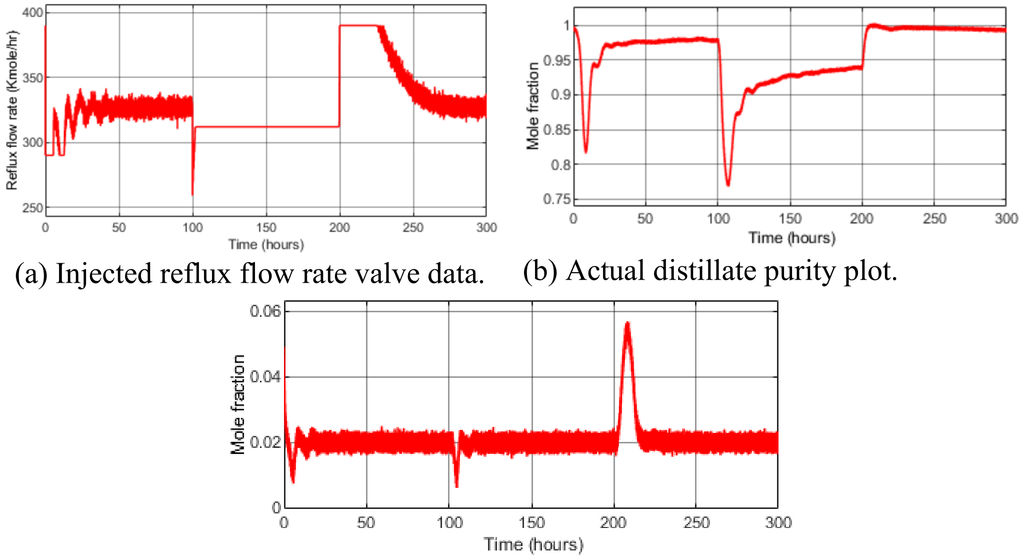
Figure 26. Column performance against a false data injection attack on bottoms impurity sensor.

The results presented illustrate that the quality of the distillate and bottoms products are degraded in presence of attack which will lessen their market price; thus incurring financial loss. However, for every attack, except the control loop attacked the remaining outputs are maintained around their desired values by the controllers thus adding robustness to a certain extent against the arrack. Once the attack subsides the controller returns the product quality back within the specification.

3.2.2.b Actuator attack:

The actuators associated with the two closed loop control network responsible for maintaining the distillate and bottoms product qualities (using reflux flow rate and reboiler duty cycle respectively) are primarily attacked and the effect of the attacks have been illustrated using the presented results.

- **Data injection attack:** This attack is simulated by independently attacking the controller outputs by setting a_r as 0.8 for reflux flow rate and a_r = 0.6 for reboiler duty cycle. Results for the attack have been presented in Figure 27 and Figure 28 respectively. The results only include the key column parameters which have been affected by the attack. As can be seen the attacked degraded product qualities thus incurring financial loss. Similar to what is observed for sensor attacks, only the output associated with the control loop under attack is affected by attack while the remaining outputs are regulated around their desired value by the proposed control scheme. The bottoms impurity is returned to desired set point once attack subsided as shown in Figure 28 providing inherent attack recovery however the distillate purity is higher than desired set point once attack subsided as seen in Figure 27 due to large controller integral error. There is high frequency chattering observed in reflux flow rate actuator data which is due to presence of process and measurement noise which the controller counteracts by changing the actuator input to maintain the outputs at their desired set points. The chattering has been reduced by applying a running average filter on the reflux flow rate output from controller since averaging is the optimal method for filtering random noise.

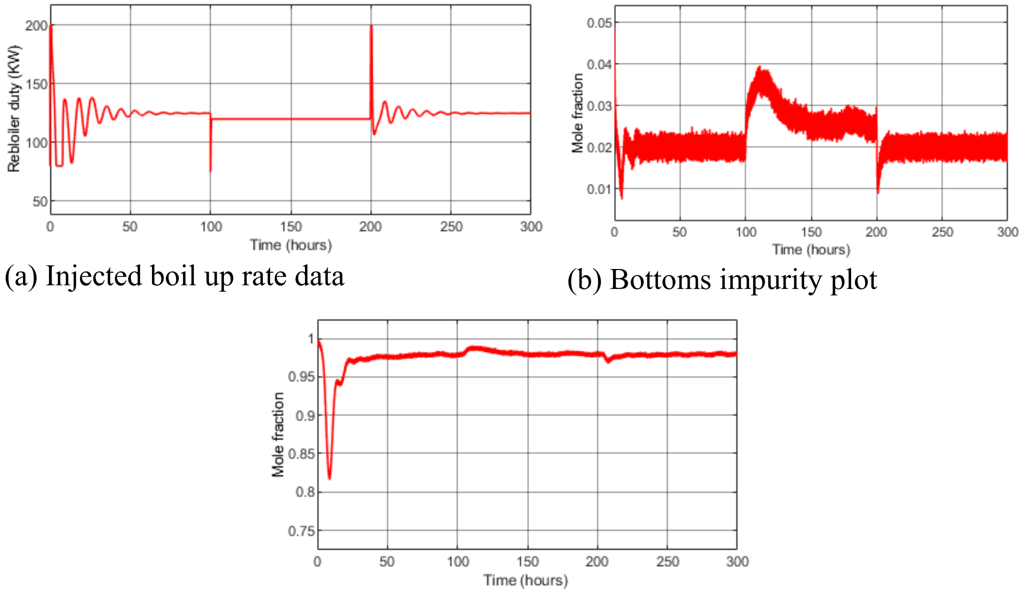


(a) Injected reflux flow rate valve data.

(b) Actual distillate purity plot.

(c) Bottoms impurity plot.

Figure 27. Effect of a data injection attack on the reflux flow rate.



(a) Injected boil up rate data

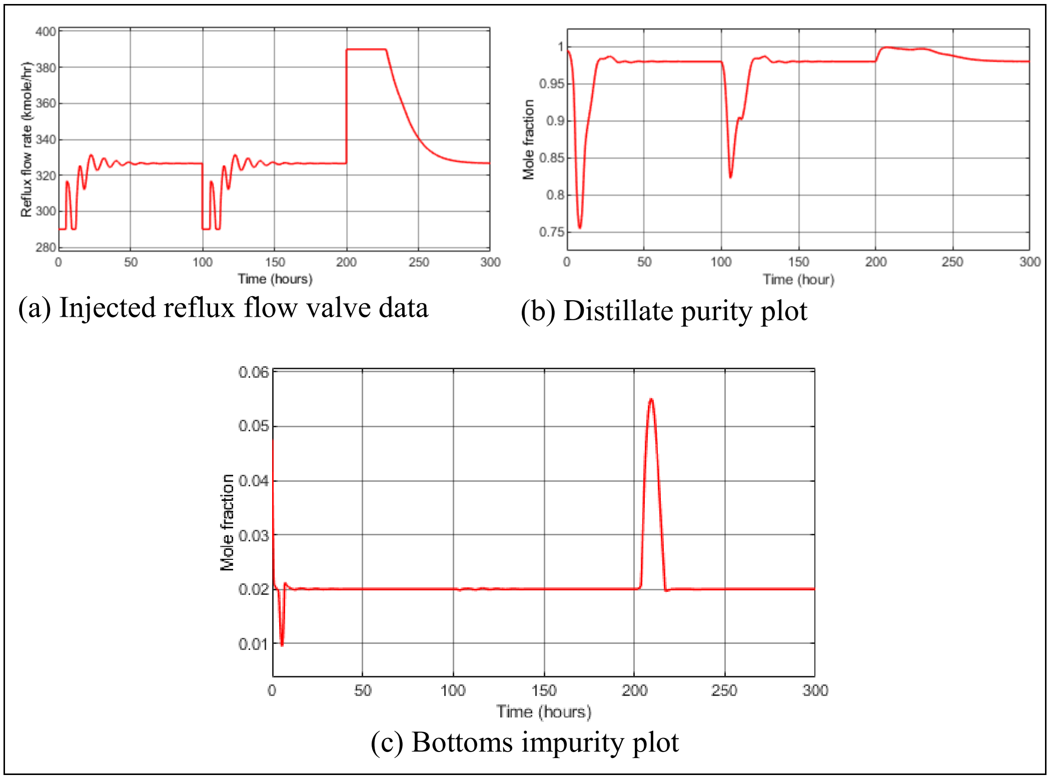
(b) Bottoms impurity plot

(c) Distillate purity plot

Figure 28. Effect of a data injection attack on the reboiler duty cycle.

- **Replay Attack:** This attack is simulated by combining eavesdropping and replay attack. At first the data between 0 – 100 hours is recorded which is then replayed between 100 – 200 hours during simulation to emulate replay attack. The attack is launched on reflux flow rate valve data.

The results from the attack has been provided in Figure 29. It can be observed that the attack results in the degradation of the product quality causing financial loss. Apart from the output associated with the attacked loop, the remaining outputs are unaffected by the attack by the presence of control scheme. Once the attack subsides the distillate becomes purer that 98% which is the setpoint for the controller. This is caused by the large integral error which causes the controller output to saturate thus setting the actuator at its maximum value.



(a) Injected reflux flow valve data

(b) Distillate purity plot

(c) Bottoms impurity plot

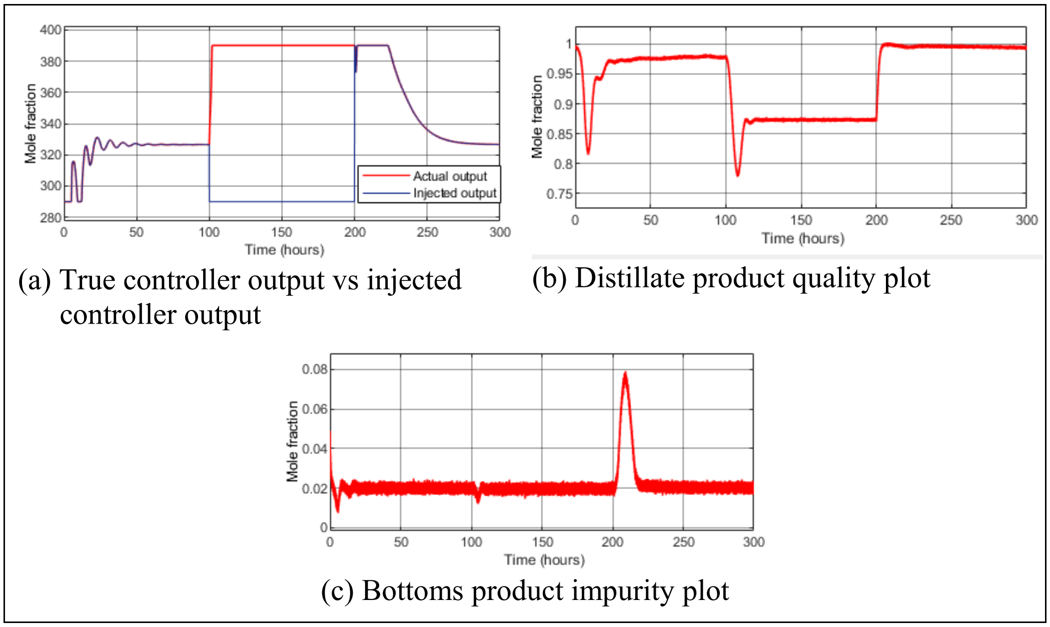
Figure 29. Effect of a replay attack on the reboiler duty cycle.

3.2.2.c Controller attacks:

The attack is injected by changing the proportional and integral gain of the PID controllers responsible for maintaining the distillate quality given in (10) as following.

$$u(t) = (K_P + \triangle K_P) + \ (K_i + \triangle K_i) \int e(t)dt + \ K_d \frac{de(t)}{dt} \tag{34}$$

The value of $\triangle K_P$ and $\triangle K_i$ is chosen to reduce the nominal proportional and integral gains by 40% and 70% respectively. From the results presented in Figure 30 it can be seen that the attack degrades the distillate quality however the quality of the bottom product along with the remaining outputs are unaffected (with minor transients) by the attack due to the presence of the control scheme which added some level of robustness against it. Once the attack subsided the controller is able to restore the product quality within the required specification however the purity is higher than 98% which is the controller set point. This is due to the large integral error which saturated the controller output.



(a) True controller output vs injected controller output

(b) Distillate product quality plot

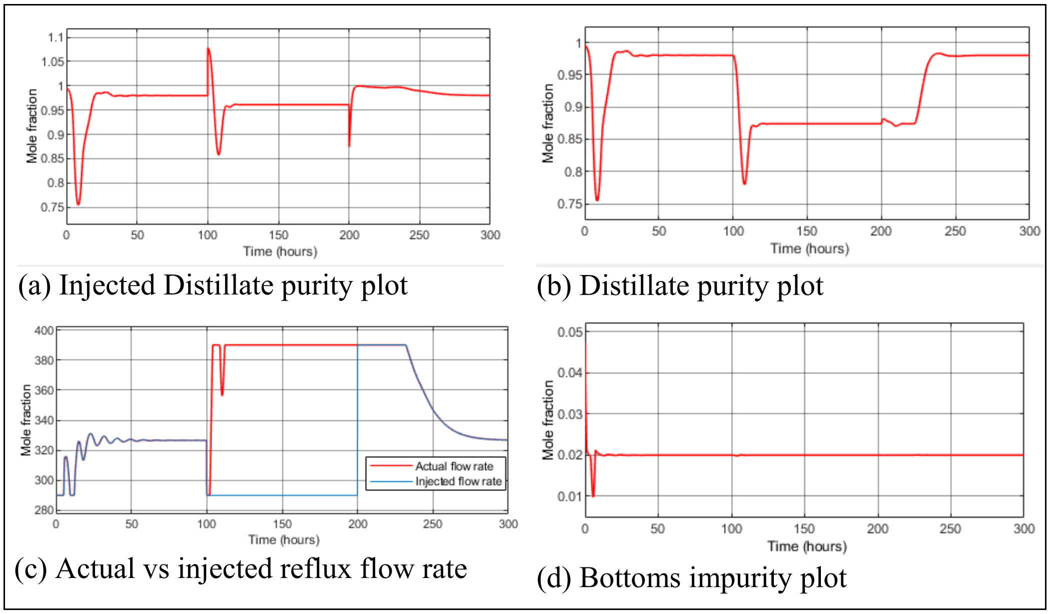
(c) Bottoms product impurity plot

Figure 30. Controller attack and its effect on the column performance.

3.2.2.d Chaining attack:

- **Controller and sensor attack:** The attack is simulated by combining the controller attack from section 3.2.2c and scaling attack on distillate measurements (λ_s = 0.1) from section 3.2.2.a. The results obtained from simulation is presented below. Sensor data for distillate purity measurement are replaced with corrupted data. And the controller parameters for the distillate purity controller are altered. The attack is injected from time 100 to 200 hours during simulation.

The result of the attack has been presented in Figure 31. Combining more than once attack caused the distillate quality to degrade while the remaining outputs including the bottoms impurity level are regulated around their set-point by the proposed control scheme thus adding robustness. Once the attack subsided the controller successfully brings the product qualities within their quality specification providing attack recovery. The motivation for chaining attack is to add stealth to the attack thus reducing the likelihood of detection.



(a) Injected Distillate purity plot

(b) Distillate purity plot

(c) Actual vs injected reflux flow rate

(d) Bottoms impurity plot

Figure 31. Illustration of effect of chaining attack on the column performance.

Provided below in Figure 32 and Figure 33 are summary of the effect of different attacks on the distillate and bottoms product quality.

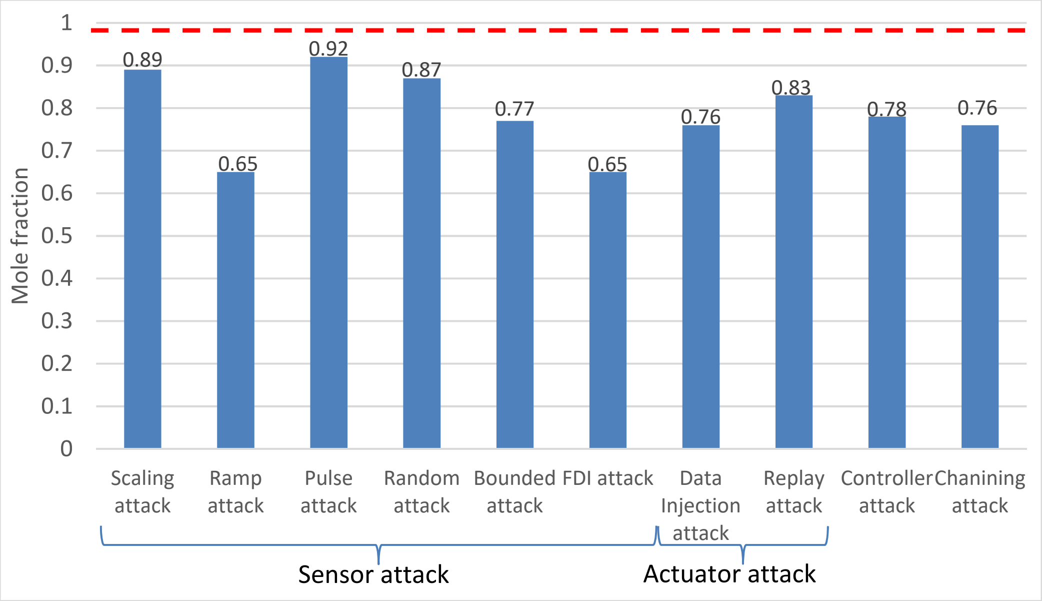


Figure 32: Illustration of effect of cyber attacks on quality on distillate.

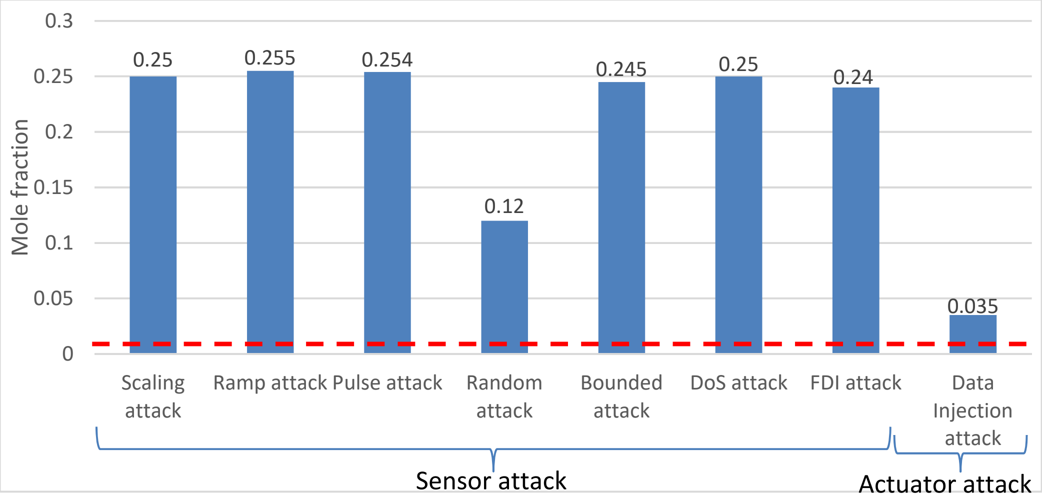


Figure 33: Illustration of effect on cyber attacks bottoms product quality.

From the results presented the maximum degradation on distillation purity is caused by the Pulse attack and the bottoms product suffers the worst quality decline by DoS attack and scaling attack respectively based on the selected attack parameters.

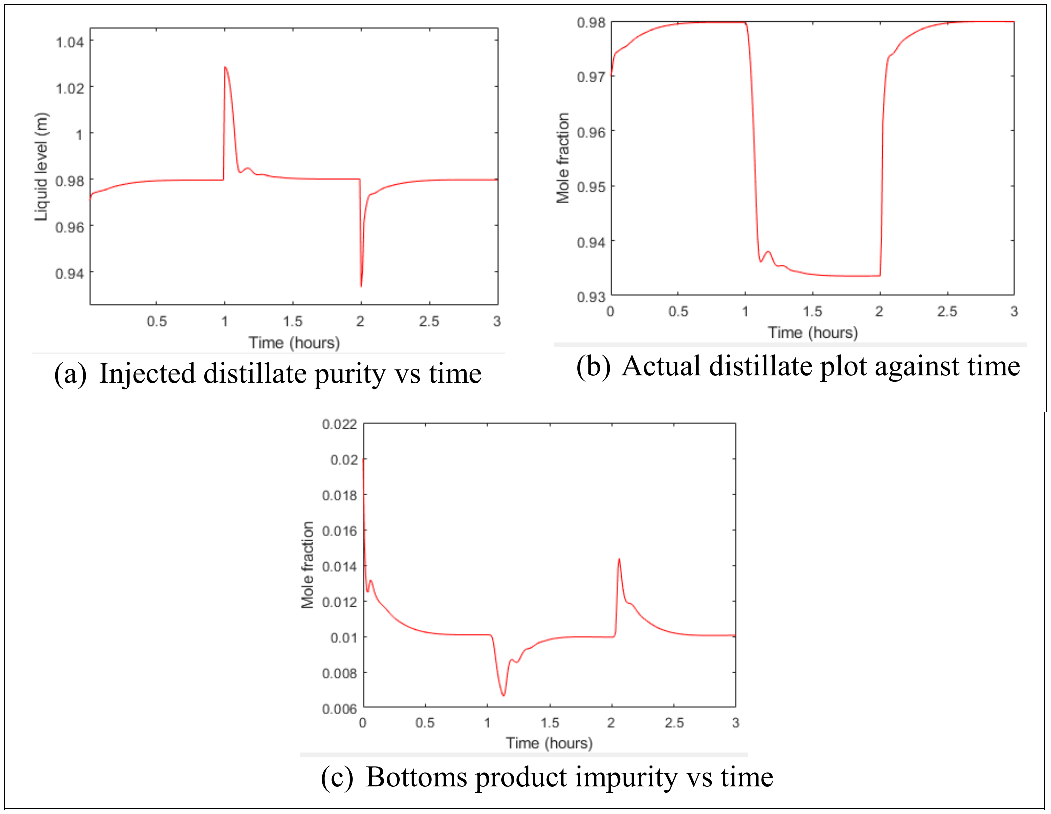
3.2.3. Modelling effect of cyber attack on Aspen Dynamics based model of the column

The attacks are simulated on a DC using their mathematical model on Simulink. All the attacks discussed in this report are injected in simulation from t = 1 – 2 hours. The attacks are intended to deviate the plant outputs from their specified requirement. As per the specification the distillate purity has to be equal or greater than 98% and the bottom product impurity has to be equal or less than 1%, and the liquid levels in the two drums has to be 0.75 m. The simulated attacks along with their effect on the column performance have been illustrated using the provided results in the rest of this section.

3.2.3.a Sensor attack:

- **Scaling attack*:***

Distillate purity sensor attack: The attack is simulated by setting λ_s as 0.05 and results have been presented in Figure 34.



(a) Injected distillate purity vs time

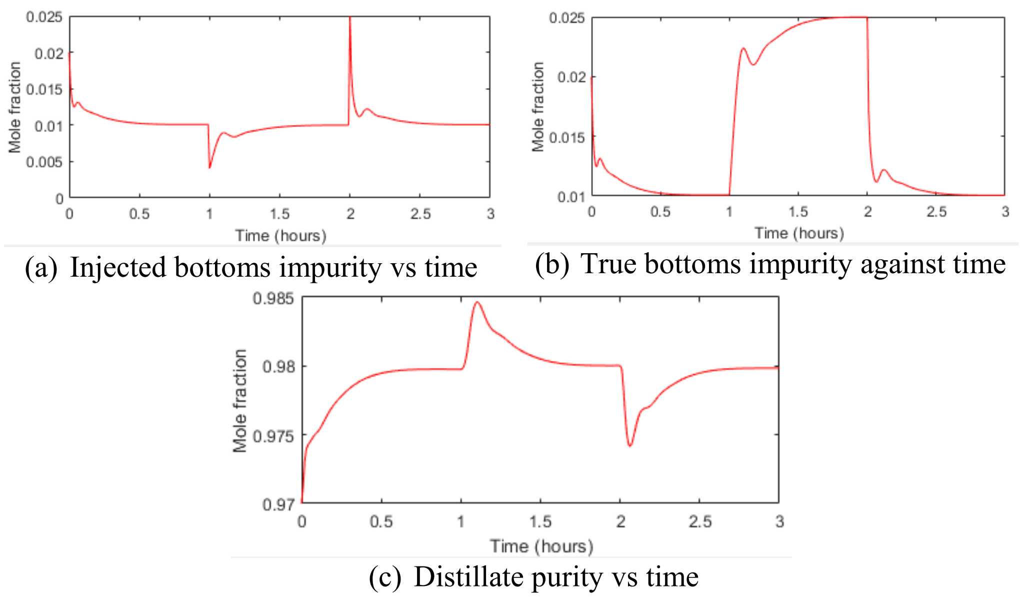
(b) Actual distillate plot against time

(c) Bottoms product impurity vs time

Figure 34. Results of a scaling attack on the distillate purity sensor.

It is observed that feeding increased distillate purity readings results in the controller taking countermeasure by decreasing the true purity level in the distillate. The attack caused some transients in bottoms impurity which is counteracted by the controller. There is no effect on the performance of the other control loops associated with the remaining outputs due to the presence of feedback controllers during this attack thus providing some degree of robustness against the attack. Once the attack subsided the controllers returned the output to its desired value thus providing attack recovery.

Bottoms impurity sensor attack: Using λ_s as -0.6 the attack is injected to emulate a case where the impurity level is lower than 0.01 and results are presented in Figure 35. The controller responds to a lower injected bottoms impurity level by increasing the true bottoms impurity level beyond 1%. The remaining outputs are regulated around their desired values by the feedback controllers except the distillate purity which increases momentarily before the controller bring it back to the set value. Once the attack subsides the outputs are returned back to their desired values by the feedback controllers.



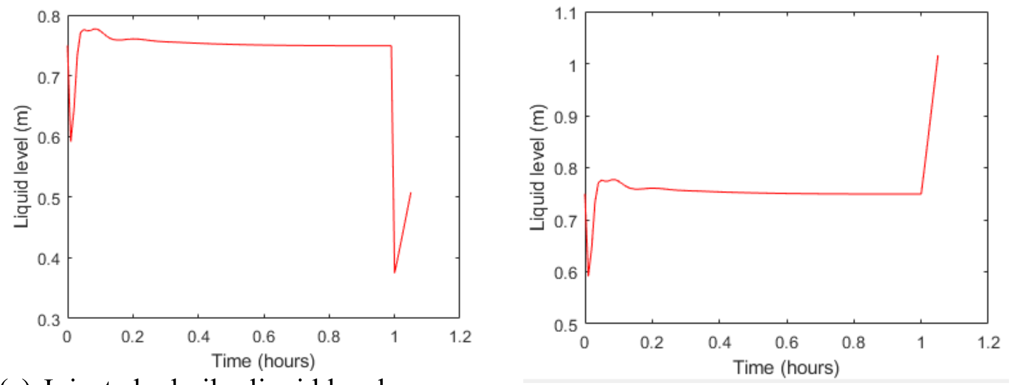
(a) Injected bottoms impurity vs time

(b) True bottoms impurity against time

(c) Distillate purity vs time

Figure 35. Results for a scaling attack on the bottoms impurity measurement sensor.

Reboiler drum level measurement sensor attack: The attack is simulated for λ_s = -0.9. This resulted in feeding liquid level data which is lower than the true liquid level in the reboiler drum. Hence the controller reduces the bottoms product flow rate which results increase in the true level of liquid in the reboiler causing it to overflow causing an emergency shutdown. Results presented in Figure 36 confirms this.

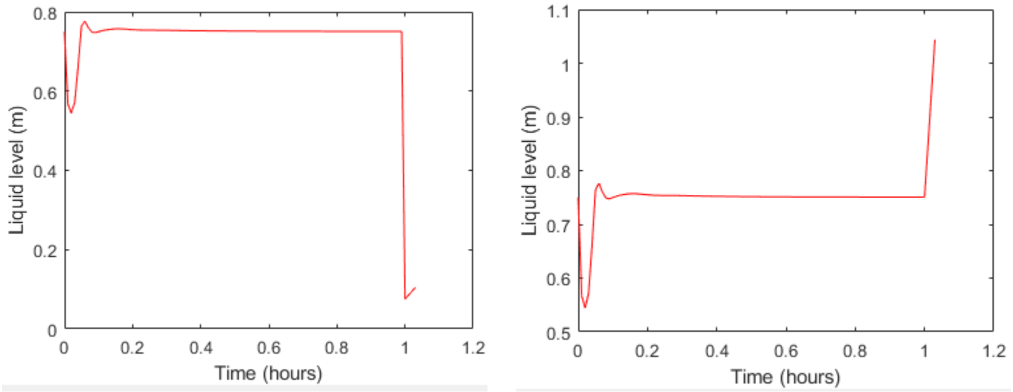


(a) Injected reboiler liquid level

(b) Actual liquid level in reboiler

Figure 36. Results for a scaling attack on the reboiler drum level measurement.

Reflux drum level measurement sensor attack: Attack results are shown in Figure 37.



(a) Injected reflux drum liquid level

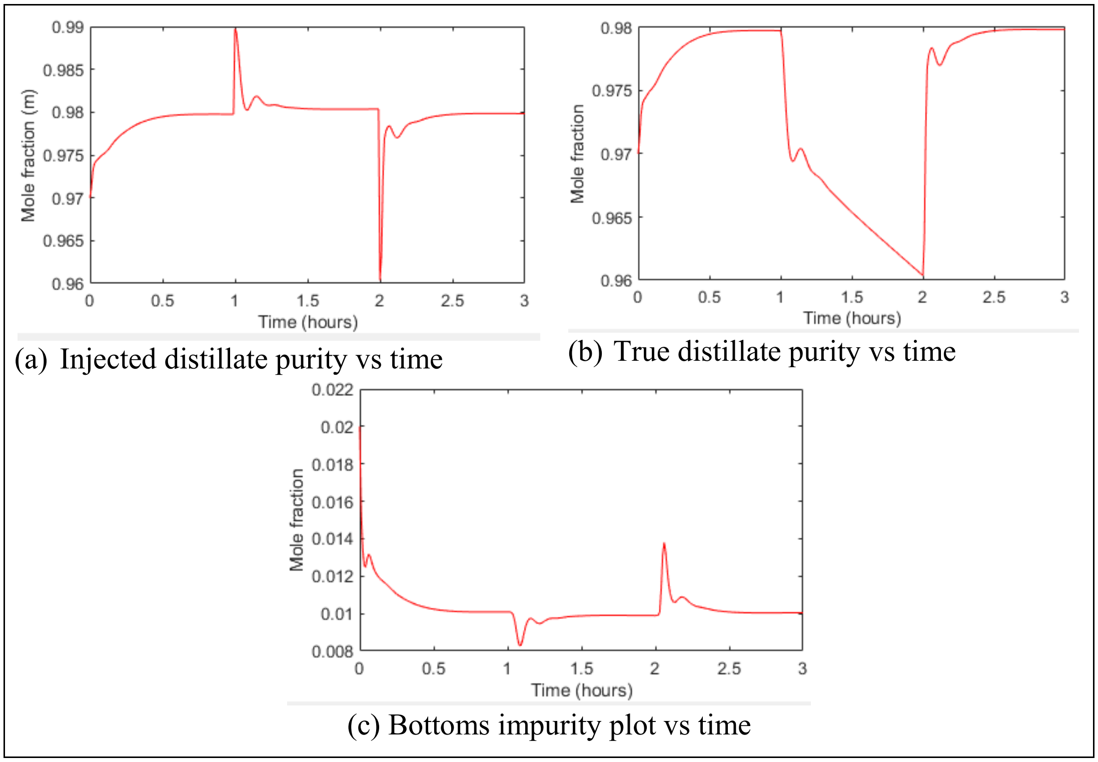
(b) Actual liquid level in reflux drum

Figure 37. Results for a scaling attack on the reflux drum level measurement.

The value of λ_s is set to -0.5 for generating attack vector. The attack vector fed controller liquid level values which are lower than the true liquid level in the drum which the controller acts against by increasing the liquid level in the drum until it overflows which results in an emergency shutdown which is confirmed from Figure 37.

Ramp attack:

Distillate purity sensor attack: The value of λ_r is set as .01 and result is given in Figure 38. It is observed that out of the five control loops only the top and bottom product qualities are affected. The fed distillate purity reading are higher than true purity value which resulted in reduced reflux flow thus reducing the true quality of the distillate below 98% and in the process reducing the bottoms impurity below 1% as confirmed from Figure 38 . It can be also observed that the controllers are able to bring the products back to their normal required specification upon removal of the attack.



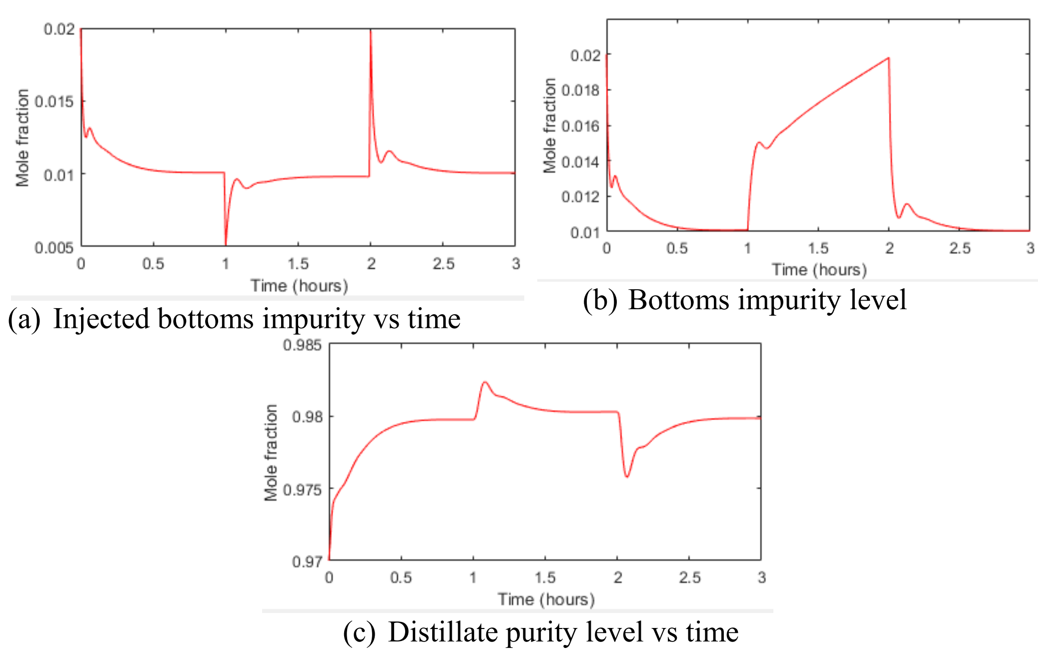
(a) Injected distillate purity vs time

(b) True distillate purity vs time

(c) Bottoms impurity plot vs time

Figure 38. Results of a ramp attack on the distillate purity sensor

Bottoms impurity sensor attack: The attack is simulated using λ_r as -0.005 and result is presented in Figure 39. This value is chosen to inject impurity readings which are less than the actual impurity sensor measurements to make the controllers act to increase the impurity level thus violating the specification of the bottoms product and thus its value. As can be seen from the results in Figure 39 the attack is able increase the impurity level in the bottoms product however due to this the purity levels of the top product is increased. All the remaining loops are controlled as per the specification without any hick-ups.



(a) Injected bottoms impurity vs time

(b) Bottoms impurity level

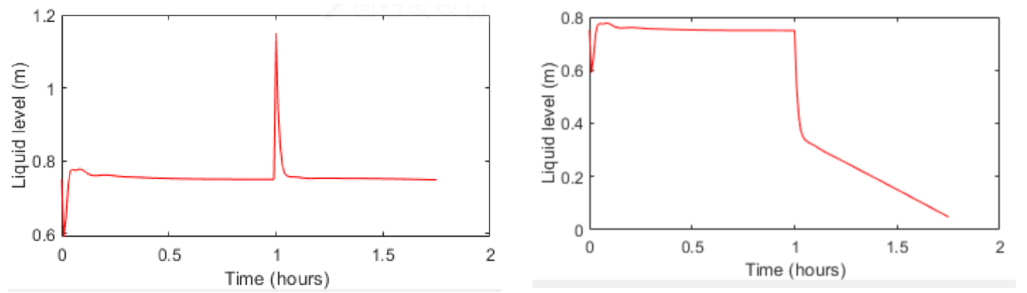
(c) Distillate purity level vs time

Figure 39. Results of a ramp attack on the bottoms impurity sensor.

Attack on reflux drum level measurement sensor:

<u>Case i: High liquid level in the drum:</u> The attack is injected by setting λ_r to 0.40 to simulate a situation when the liquid level is higher than desired liquid level in the drum. Results presented in Figure 40 illustrates the effect of the attack. The controller counteracts the increased measurements by reducing the true liquid level in the reflux

drum below 0.05 m which causes an emergency shutdown which is necessary because an empty reflux drum will cause the degradation of the distillate quality thus violating the purity specification. The remaining outputs are regulated around its desired value by their respective controllers.

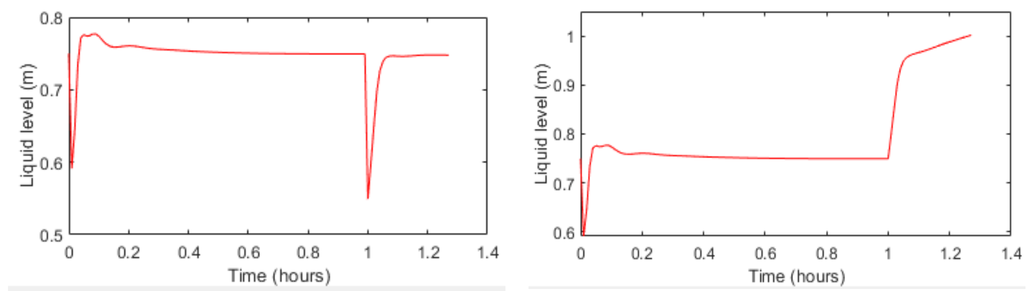


(a) Injected liquid level in the reflux drum (m)

(b) True liquid level in the reflux drum (m)

Figure 40. Effect of a ramp attack on the reflux drum level sensor.

Case ii. Low liquid level in the drum: The attack vector is generated setting λ_r to -0.20 and the results for the attack is presented in Figure 41.



(a) Injected liquid level (m) in reflux drum

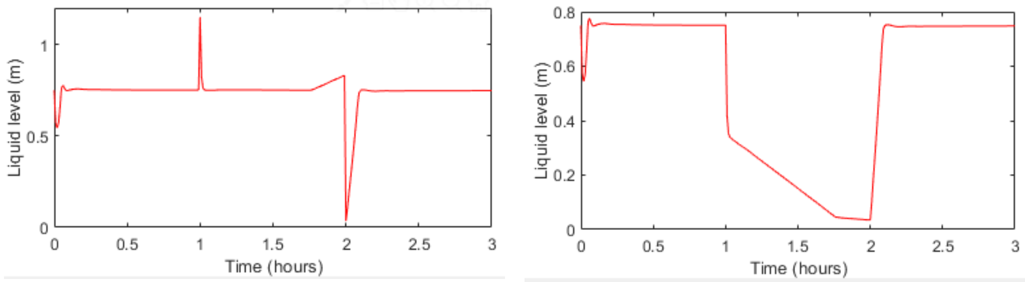
(b) Actual level of liquid (m) in reflux drum

Figure 41. Effect of a ramp attack on the reflux drum level sensor.

In this case the attack is simulated to inject liquid level readings in the reflux drum which are lower than the true liquid level in the drum. It is expected that the controller will counteract this drop in liquid level by reducing the distillate flow rate which in turn will increase the true level in the reflux drum thus potentially causing the liquid to overflow. This effect can be observed from results acquired using this attack illustrated in Figure 41. As mentioned previously the length of the drum is 1 meter. As can be seen from the results, the DC is shutdown when the liquid level exceeds 1 meter to avoid any spillage. Since hydrocarbons are highly flammable; hence a spillage poses fire hazard.

Attack on reboiler drum level measurement sensor:

Case I. High liquid level in reflux drum: The attack is simulated setting λ_r as 0.40 which simulated a case where the injected liquid level data is higher than the true liquid level in the reboiler drum to which the controller is expected to respond to by increasing the bottoms product flow resulting in a drop in true liquid level in the reboiler drum. This can be confirmed from the results presented below in Figure 42.



(a) Injected reboiler liquid level (m) vs time

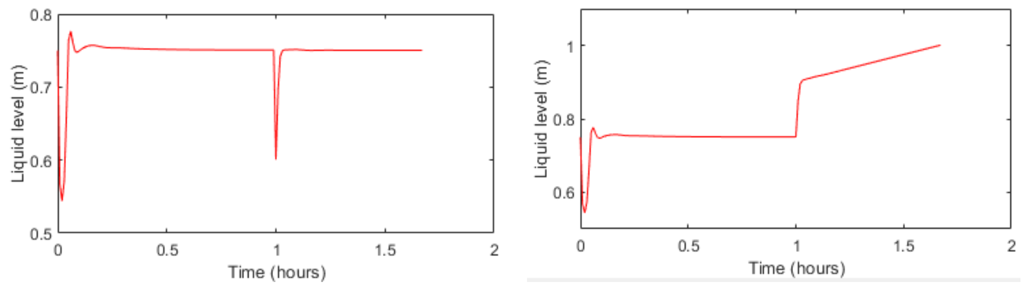
(b) True liquid level in reboiler drum (m)

Figure 42. Effect of a ramp attack on the reboiler drum level sensor.

Case ii. Low liquid level in reboiler drum:

This case simulates an attack scenario where the attacker fiddles with liquid level in reboiler drum data by feeding readings which are lower than true readings. The value of λ_r is set to -0.15 for the attack. The result for the attack is presented in Figure

43. As can be seen from the results, upon detecting low liquid level in the drum the controller ceases the distillate flow rate thus causing the actual level in the reboiler drum to rise and eventually overflow which causes the plant to shut down.

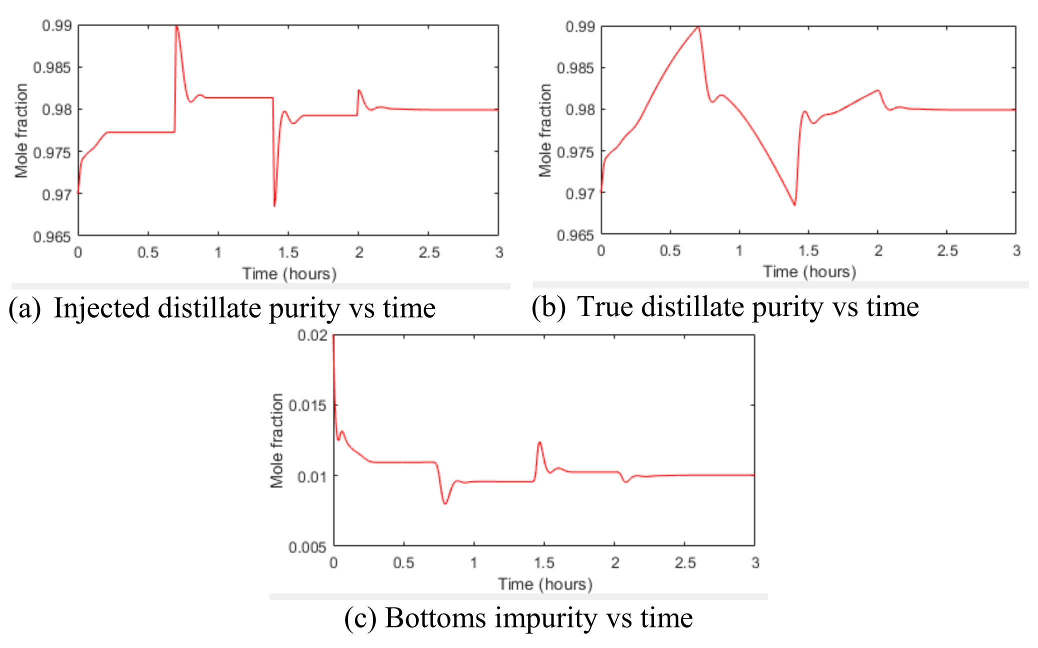


(a) Injected reboiler liquid level (m) vs time

(b) True liquid level (m) in the reboiler vs time

Figure 43. Effect of a ramp attack on the reboiler drum level sensor.

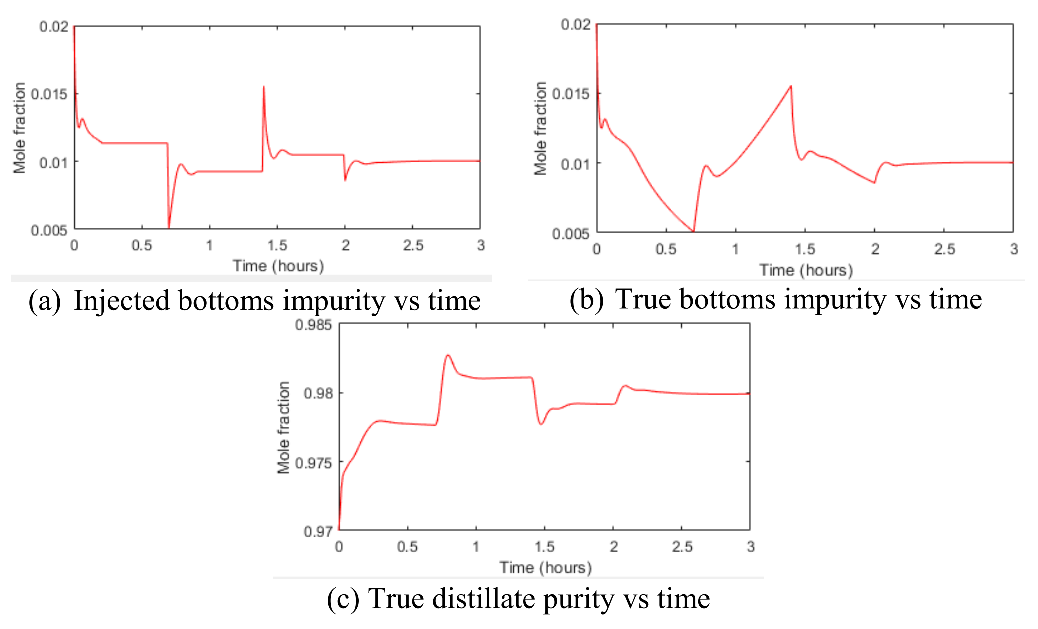
- **DoS attack:** The results for the attack on distillate purity sensor and bottoms impurity sensor are presented in Figure 44 and Figure 45 respectively. The DoS attack is simulated using continuous time pulses. When the amplitude of the pulse is 1 the sensor measurements are transmitted to controller. And when the pulse value is 0 the sensor readings are denied to the controller by sending the last available sensor measurement available when the pulse amplitude is 1. In case of both distillate purity sensor and bottoms impurity sensor, the pulse width is set to 30% with a period of 0.70 hour i.e. the communication between the sensor and controller is blocked for 30% of the duration of attack. The attack is injected during simulation from t = 0 – 2 hour at the start of simulation because once the plant reaches steady state the outputs remains fairly constant; hence are unaffected by DoS attack. The results show that during the attack the controllers are unable to track the distillate purity and bottoms impurity at their desired set-points.



(a) Injected distillate purity vs time (b) True distillate purity vs time

(c) Bottoms impurity vs time

Figure 44. Effect of a DoS attack on the distillate purity sensor.



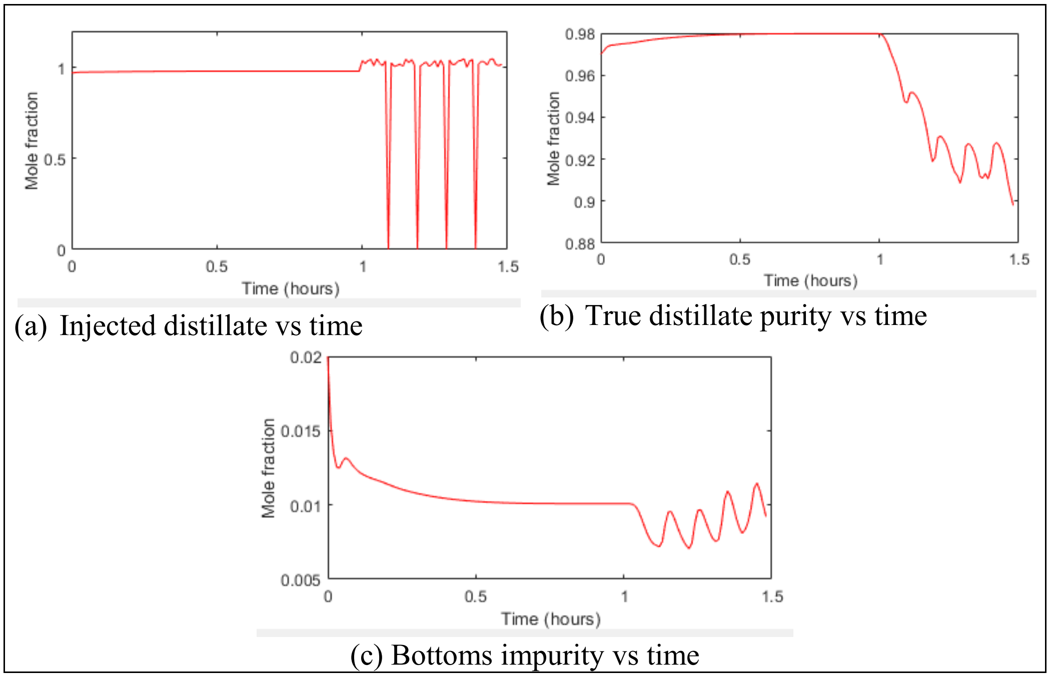
(a) Injected bottoms impurity vs time (b) True bottoms impurity vs time

(c) True distillate purity vs time

Figure 45. Effect of a DoS attack on the bottoms impurity sensor.

- **False data injection (FDI) attack:** The attack is generated using continuous time pulse biased by multiplying with a uniformly distributed random variable.

Attack on distillate purity sensor: The period of the pulse is set as 0.1 hour with the pulse width is set to 80% and the RV is selected with distribution U ~ (1, 1.05). It is expected that when the pulse amplitude is 1, the RV will bias the pulse amplitude and generate distillate purity readings for the controller which are higher than .98 (since the range of RV is between 1 and 1.05) which the controller will act to by reducing the true distillate purity. This is confirmed from results presented in Figure 46. In fact, the controller reduces the distillate purity below the minimum purity threshold of 90% causing an emergency shutdown of the DC.



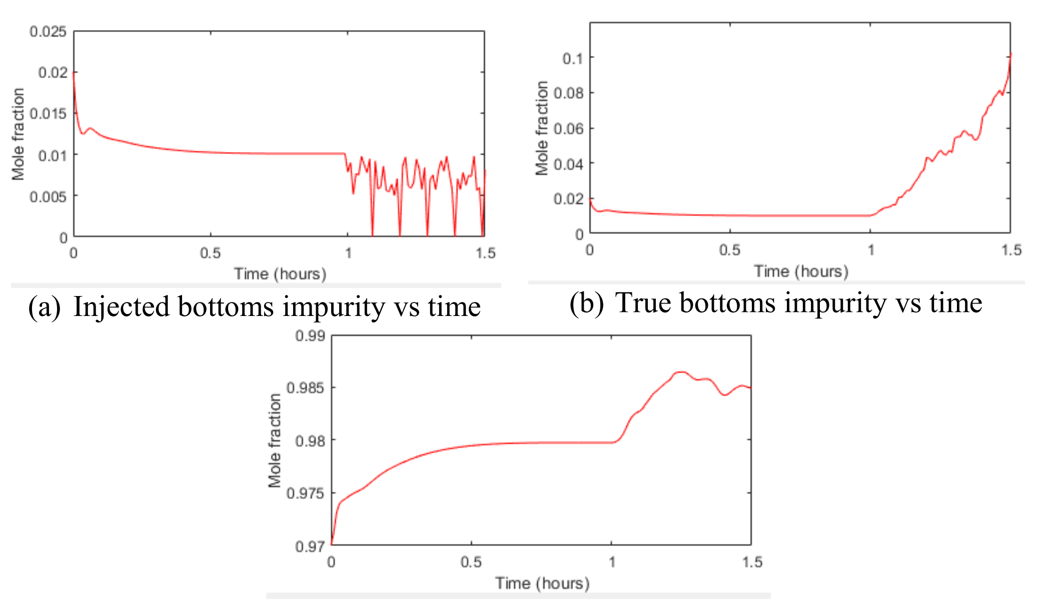
(a) Injected distillate vs time

(b) True distillate purity vs time

(c) Bottoms impurity vs time

Figure 46. Effect of a FDI attack on distillate purity sensor.

Attack on distillate purity sensor: Similarly, as in the case of distillate purity sensor, for this attack the pulse time period is set to 0.1 hour and width to 80%. The RV is chosen as U ~ (.005, .01). The result for the attack is presented in Figure 47. It can be seen that the attack causes the bottoms impurity to increase beyond its maximum threshold of 10% thus causing an emergency shutdown. This happens because during the attack when the pulse amplitude is 1, the RV causes the attack signal to be biased between 0.005 and 0.01 which is sent to the controller. As the bottoms impurity setpoint is 0.01, the controller reacts to the attack vector by increasing the bottoms impurity.



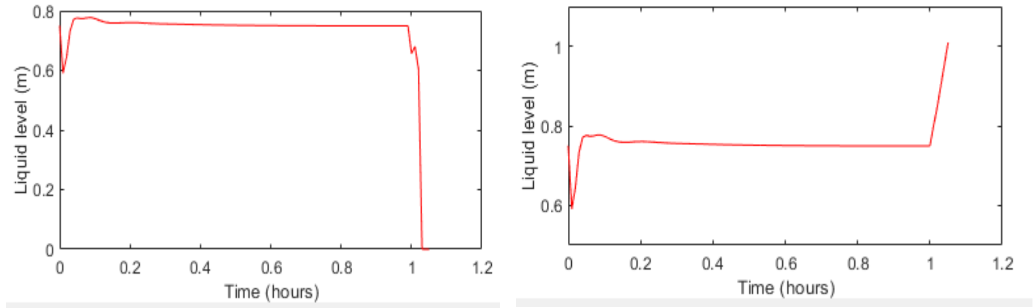
(a) Injected bottoms impurity vs time

(b) True bottoms impurity vs time

(c) Distillate purity vs time

Figure 47. Effect of a FDI attack on the bottoms impurity sensor.

Attack on reflux drum level sensor: The attack result is presented below in Figure 48.



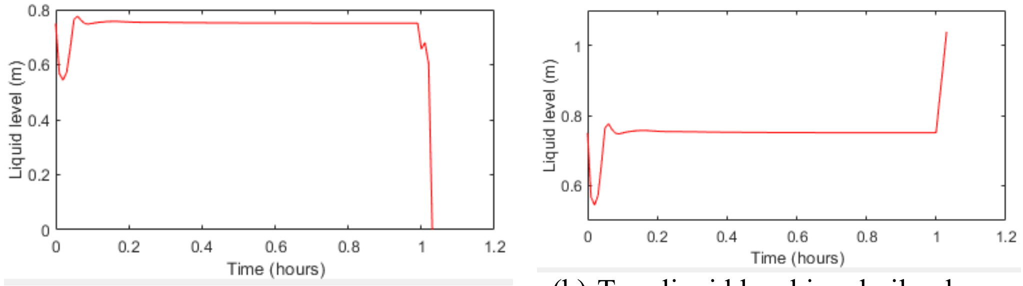
(a) Injected liquid level in reflux drum

(b) True liquid level in reflux drum

Figure 48. Effect of a FDI attack on the liquid level in reflux drum.

The value of the pulse width is set to 20% with a period of 0.1 hour and the RV is chosen as U ~ (0.6, 0.7). Hence during attack the injected value will be between 0.6 and 0.7 when the pulse amplitude is 1. Since the liquid level setpoint is 0.75 hence the controller will try to increase the liquid level in the drum which is verified from results in Figure 48. In fact, the controller increases the liquid level beyond 1 m. which causes an emergency shutdown to avoid spillage.

Attack on reflux drum level sensor: Similar to attack on reflux level, the value of the pulse width is set to 20% with a period of 0.1 hour and the RV is chosen as U ~ (0.6, 0.7) for this attack. The results for the attack is presented in Figure 49. It is seen that the attack causes the liquid level to exceed the 1 m. threshold thus causing the plant to shut down. This is because of the range of RV due to which the attack signal feeds reboiler liquid level values which are lower than the desired liquid level in reboiler drum. This causes the controller to increase the liquid level in reboiler drum causing it to overflow.



(a) Injected liquid level in reboiler drum

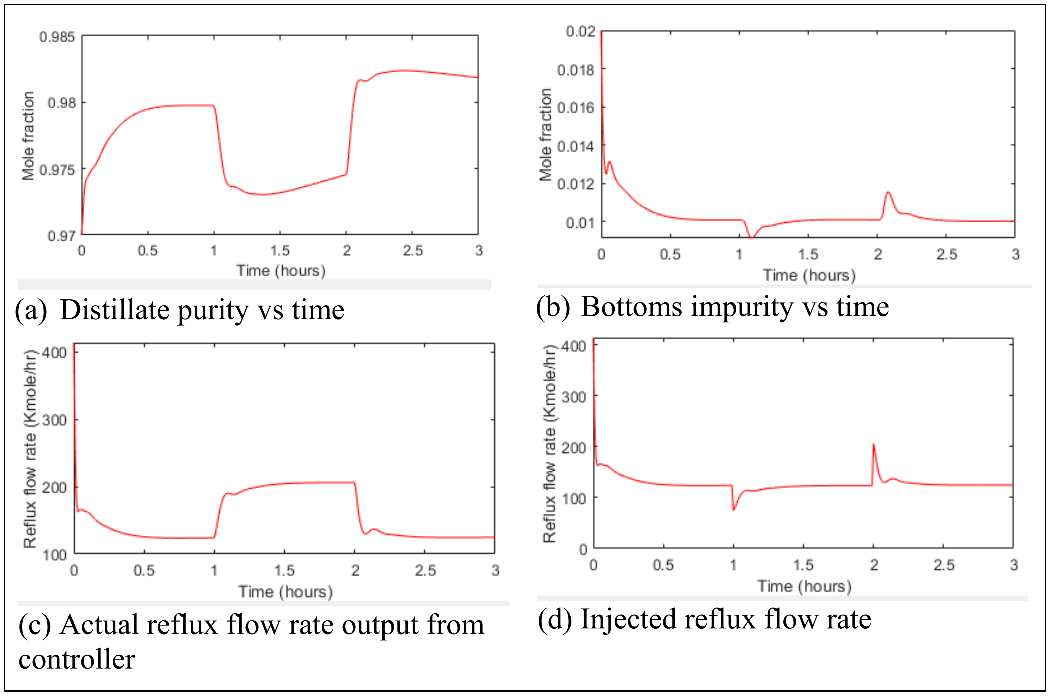
(b) True liquid level in reboiler drum

Figure 49. Effect of a FDI attack on the liquid level in reboiler drum.

3.2.3.b Actuator attack:

- **Data injection attack:**

Attack on reflux flow rate: The results for the attack is presented below in Figure 50.



(a) Distillate purity vs time

(b) Bottoms impurity vs time

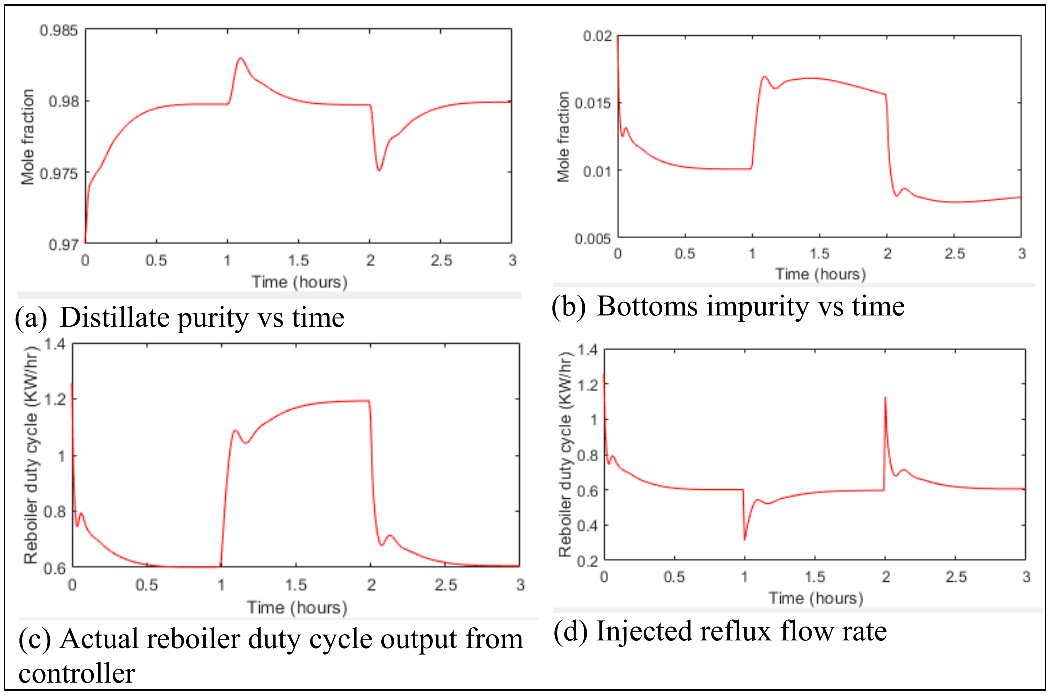
(c) Actual reflux flow rate output from controller

(d) Injected reflux flow rate

Figure 50. Effect of a data injection attack on the reflux flow rate.

This attack is simulated by independently attacking the reflux flow rate output from the controller by setting a_r as 0.6. It can be seen from the figure that the injected reflux flow rate is less than the flow rate output by the controller. As the reflux flow rate is used to control the distillate purity and the gain between reflux flow rate and distillate purity is positive hence the distillate purity is reduced below its desired value during the attack. The remaining outputs remain un-affected by the attack thus providing robustness against attack. Once the attack subsides the controller brings the output back to desired value thus providing attack recovery.

Attack on reboiler duty cycle: The value of a_r is set to 0.5 during this attack. From the results of the attack presented in Figure 51, it is observed that the attack causes an increase in the bottoms impurity level. This is because the injected reboiler duty cycle during attack period is less than the output from the controller. Since the reboiler duty cycle is used to control the bottoms impurity level hence a decrease in reboiler duty cycle causes increase in bottoms impurity. Once attack subsides the controller is able to bring the impurity level back at the desired setpoint thus providing attack recovery.



(a) Distillate purity vs time

(b) Bottoms impurity vs time

(c) Actual reboiler duty cycle output from controller

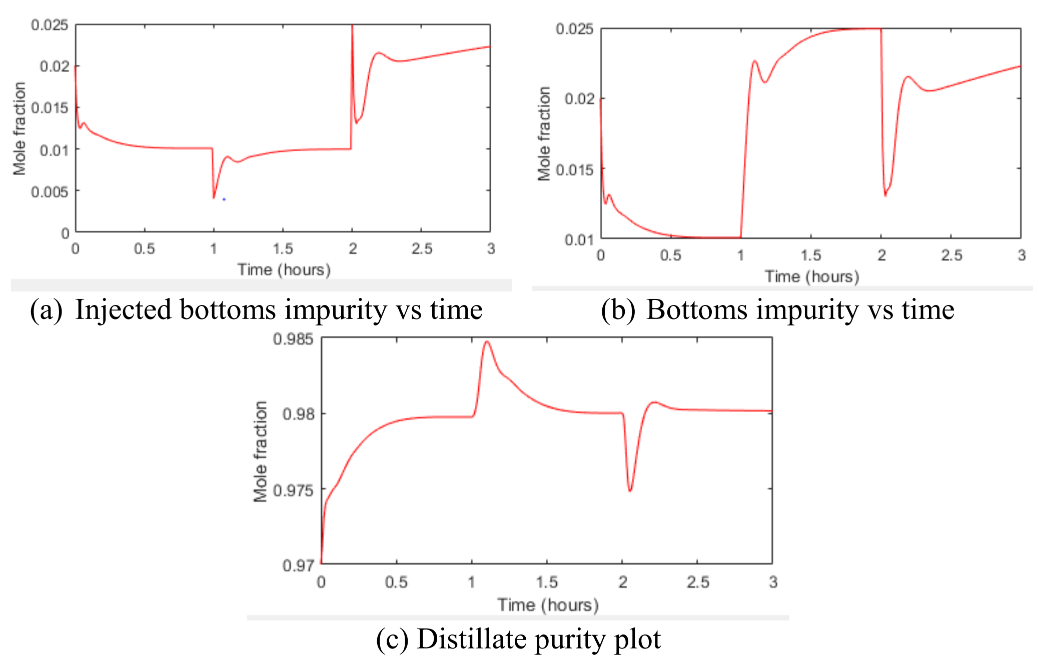
(d) Injected reflux flow rate

Figure 51. Effect of a data injection attack on reboiler duty cycle.

3.2.3.c Chaining attack:

The attack combines controller attack and sensor attack to generate the attack vector. As part of the attack the bottoms impurity controller proportional gain and integral gain is increased by 10% and 20% of their nominal value and the bottoms impurity sensor is attacked with a scaling attack by setting λ_s as -0.60.

The result for the attack is presented below in Figure 52. The results show that the attack causes an increase in the bottoms impurity level. There are two reasons behind it. Firstly, during scaling attack the controller is fed bottoms impurity readings which are lower than the bottoms impurity setpoint. The controller responds to this by increasing the impurity level in the bottoms product. Besides that, increasing the controller gain causes the impurity level to rise more quickly. Once the attack subsides the controller is unable to bring the impurity level back within the required specification as it does during scaling attack.



(a) Injected bottoms impurity vs time

(b) Bottoms impurity vs time

(c) Distillate purity plot

Figure 52: Effect of chaining attack on the column performance.

CHAPTER 4: DETECTION OF CYBER ATTACKS

This chapter will present the attack detection techniques developed based on the models of the two DC presented in Chapter 2. All the detection techniques have been designed based on the model of the columns and functions based on state estimation. There have been two techniques proposed for the grey box model of the DC based on EKF and UKF whereas a Luenberger observer based detection technique has been designed for the Aspen Plus Dynamics based model. Firstly, the mathematical implementation of the EKF and UKF has been presented along with the simulation results to validate and compare their ability in detecting attacks. Finally, the mathematical model and results for Luenberger observer have been presented to validate its ability in detecting attacks.

4.1 State estimator based attack detection for DC model

This section presents two attack detection techniques along with illustrating their performance and efficiency using results for the attacks presented in Section 3.2.2. The two techniques are based on Extended Kalman Filter (EKF) and Unscented Kalman filter (UKF). Fundamentally both techniques rely on state estimation for detection. The detector is integrated to the ICS as shown in Figure 53 for detection of attack on sensor, actuator or both.

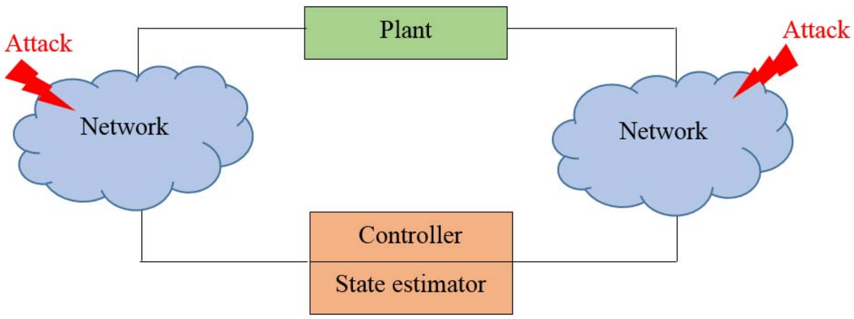


Figure 53. Illustration of state estimation based attack detection for ICS.

4.1.1 Implementation of Extended Kalman Filter

A Kalman filter estimates the system states and outputs from the knowledge of input and output measurements in the presence of measurement and process noise with limitation to linear systems. An EKF is able to predict nonlinear system states provided the state transition functions and output measurement functions are differentiable. Consider the following discrete time system

$$\boldsymbol{x_k} = f(\boldsymbol{x_k}, \boldsymbol{u_k}) + \boldsymbol{w_k}$$

$$\boldsymbol{y_k} = g(\boldsymbol{x_k}) + \boldsymbol{v_k} \tag{34}$$

where $\boldsymbol{x_k}$, $\boldsymbol{u_k}, \boldsymbol{y_k}$, $\boldsymbol{u_k}$, $\boldsymbol{w_k}$ and $\boldsymbol{v_k}$ are system states, inputs, outputs, process noise and measurement noise, respectively, and functions *f* and *g* correspond to state transition function and output measurement function respectively. Q_k and R_k, are process noise and measurement noise covariance matrices respectively. There are two steps in the estimation process namely Prediction and Update steps as given below

Prediction step:

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k) \tag{35a}$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k \tag{35b}$$

Update step:

$$K_k = P_{k-1|k-1} H_k^T (H_k P_{k-1|k-1} H_k^T + R_k) \tag{36a}$$

$$\hat{x}_{k|k} = \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} + K_k(\tilde{y}_k - g(\hat{x}_{k|k-1})) \tag{36b}$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \tag{36c}$$

where $F_k = \frac{\partial f}{\partial x_{|\hat{\boldsymbol{x}}_{\boldsymbol{k-1|k-1}}, \boldsymbol{u_k}}}$ and $H_k = \frac{\partial g}{\partial x_{|\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}}}$, $\hat{x}$ and $\tilde{y}_k$ represent the predicted mean state estimate and measurement input available at the controller side respectively, respectively. Jacobians F_k and H_k are state transition and observation matrices, matrices $P_{k|k}$ and K_k are error covariance estimate and optimal Kalman gain, respectively, and the subscript $k|k-1$ is parameter value at sampling instant *k* based on samples up to and including *k-1*.

The state distribution is approximated by a Gaussian Random Variable (GRV). Fundamentally EKF relies on linearizing the nonlinear system equations using Taylor series and then generating the linear model of the system in the state space form by computing the Jacobians using the approximated mean value of previous state estimates. The previous mean state estimates are used to approximate the Gaussian distribution (mean and covariances) of the posteriori state which are used to estimate the error covariance. The posterior state covariance, error covariance along with the Jacobians are used to compute the optimal Kalman gain. Finally, the filter gain along with previous mean state estimates and output measurements are used to predict the state of the system. The main limitation of the filter is that it uses an approximated Gaussian distribution (and approximated mean) to linearize the system. This approximated state distribution can induce large errors in the posteriori mean and covariances of the nonlinear transformed GRV. All these limitations can collectively introduce errors in the state prediction.

4.1.2 Unscented Kalman Filter

Unscented Kalman Filter (UKF) relies on unscented transformation to predict the states of a nonlinear discrete time system. Unscented transformation is a method for calculating the statistics for a random variable (RV) which undergoes a nonlinear transformation. Consider the nonlinear system model is described by equation (34). The state distribution similar to EKF is Gaussian. However, in this case the distribution is represented by a set of sample points called sigma points. These sample points accurately captures the state mean and covariance and when evaluated using the nonlinear system equations accurately gives the mean and covariance of posteriori state up to second order Taylor series expansion for any nonlinearity [55].

Similar to EKF there are two main steps in UKF implementation as follows:

Prediction step:

$$\boldsymbol{X}_{\boldsymbol{k-1}}^{\boldsymbol{a}} = [\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{\boldsymbol{a}} \pm \sqrt{(L+\lambda)P_{k-1}^{a}}] \tag{37a}$$

$$\boldsymbol{X}_{\boldsymbol{k|k-1}}^{\boldsymbol{x}} = f[\boldsymbol{X}_{\boldsymbol{k-1}}^{\boldsymbol{x}}, \boldsymbol{X}_{\boldsymbol{k-1}}^{\boldsymbol{w}}] \tag{37b}$$

$$\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} = \sum_{i=0}^{2L} W_i^{(m)} \boldsymbol{X}_{\boldsymbol{i,k|k-1}}^{\boldsymbol{x}} \tag{37c}$$

$$P_{k|k-1} = \sum_{i=0}^{2L} W_i^{(c)} \left[\boldsymbol{X}_{\boldsymbol{i,k|k-1}}^{\boldsymbol{x}} - \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}\right]\left[\boldsymbol{X}_{\boldsymbol{i,k|k-1}}^{\boldsymbol{x}} - \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}\right]^T \tag{37d}$$

Update step:

$$\boldsymbol{Y}_{\boldsymbol{k|k-1}}^{\boldsymbol{x}} = h\left[\boldsymbol{X}_{\boldsymbol{k|k-1}}^{\boldsymbol{x}}, \boldsymbol{X}_{\boldsymbol{k-1}}^{\boldsymbol{v}}\right] \tag{38a}$$

$$\hat{\boldsymbol{y}}_{\boldsymbol{k}} = \sum_{i=0}^{2L} W_i^{(m)} \boldsymbol{Y}_{\boldsymbol{i,k|k-1}} \tag{38b}$$

$$P_{\tilde{\boldsymbol{y}}_{\boldsymbol{k}},\tilde{\boldsymbol{y}}_{\boldsymbol{k-1}}} = = \sum_{i=0}^{2L} W_i^{(c)} \left[\boldsymbol{Y}_{\boldsymbol{i,k|k-1}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}\right]\left[\boldsymbol{Y}_{\boldsymbol{i,k|k-1}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}\right]^T$$

(38c)

$$P_{\boldsymbol{x}_{\boldsymbol{k}},\boldsymbol{y}_{\boldsymbol{k}}} = = \sum_{i=0}^{2L} W_i^{(c)} \left[\boldsymbol{X}_{\boldsymbol{i,k|k-1}}^{\boldsymbol{x}} - \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}\right]\left[\boldsymbol{Y}_{\boldsymbol{i,k|k-1}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}\right]^T \tag{38d}$$

$$K = P_{\boldsymbol{x}_{\boldsymbol{k}},\boldsymbol{y}_{\boldsymbol{k}}}, P_{\tilde{\boldsymbol{y}}_{\boldsymbol{k}},\tilde{\boldsymbol{y}}_{\boldsymbol{k}}}^{-1} \tag{38e}$$

$$\hat{\boldsymbol{x}}_{\boldsymbol{k|k}} = \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} + K(\boldsymbol{y}_{\boldsymbol{k}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}) \tag{38f}$$

$$P_{k|k} = P_{k|k-1} - KP_{\tilde{\boldsymbol{y}}_{\boldsymbol{k}},\tilde{\boldsymbol{y}}_{\boldsymbol{k}}}K^T \tag{38g}$$

where $\boldsymbol{x}^{\boldsymbol{a}} = [\boldsymbol{x}^T \quad w^T \quad v^T]^T$, $X^a = [(X^x)^T \quad (X^w)^T \quad (X^v)^T]$, $W_i^{(m)} = 1/\{2(L+\lambda)\}$, $W_i^{(m)} = W_i^{(c)}$, $\lambda = \alpha^2(L+k) - L$, and $i = \{1, \dots \dots \dots .., 2L\}$, λ is the composite scaling parameter, L is the dimension of augmented state, $\hat{\boldsymbol{x}}$ is the mean state estimate, $\hat{\boldsymbol{y}}$ is the mean output estimate, P is the covariance matrix, and $\boldsymbol{X}_{\boldsymbol{i}}$ are the sigma points. The parameter α determines the spread of the sigma points around $\hat{\boldsymbol{x}}$ and is usually set to a positive value (between 0 – 1) [35] and k is a secondary scaling which is usually set to 0.

The discrete time model of the system is generated using Euler method using the controller sampling time. The residual which will be used for detection is defined as following.

$$r_D\,(k) = |\hat{x}_D(k) - \tilde{x}_D(k)| \tag{39(a)}$$

$$r_B\,(k) = |\hat{x}_B(k) - \tilde{x}_B(k)| \tag{39(b)}$$

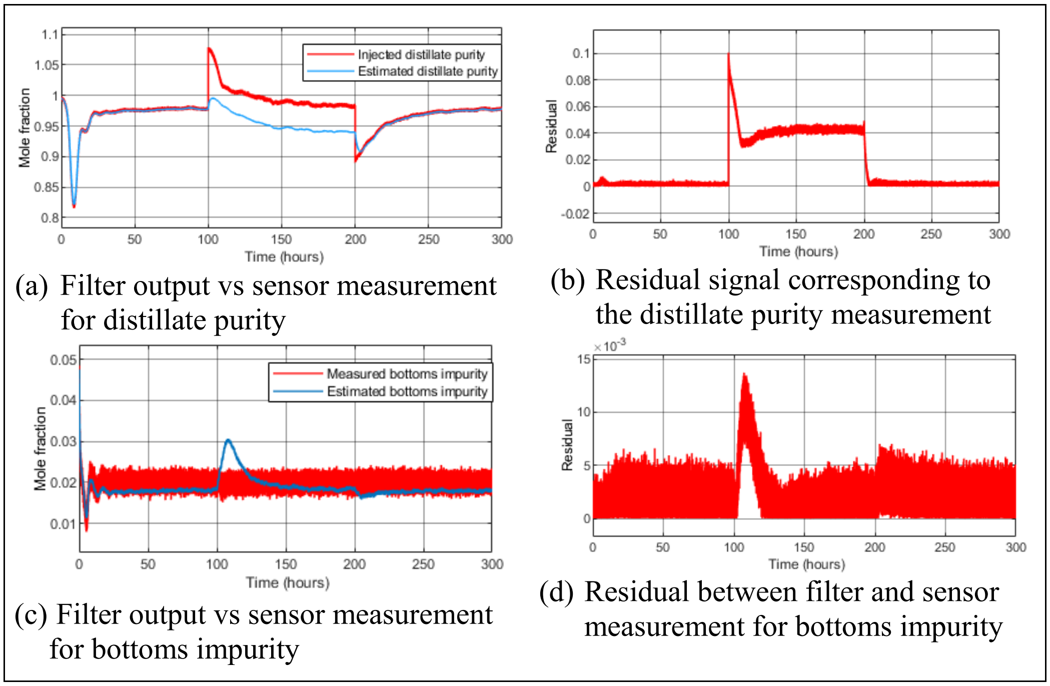
Where $\tilde{x}_D, \tilde{x}_B, \hat{x}_D$, and $\hat{x}_B$ corresponds to distillate purity measurement to the controller, bottoms impurity measurement to the controller, estimated distillate purity, and estimated bottoms impurity respectively. r_D and r_B denotes the residual in distillate purity estimation and bottoms impurity estimation respectively.

4.2 Validation of attack detection techniques

This section illustrates the application of EKF and UKF in detecting attacks presented in Section 3.2.2. Fundamentally, during the normal operation there will be no difference between the estimated filter output and sensor measurements to the controller. However, during attack there will be a difference which can be used for detection. The sampling time for the controller is set as 0.001 hour. In our case, any residual value larger than 0.02 for both the distillate and bottoms product will trigger an attack warning.

4.2.1 Sensor attack:

- **Scaling attack:** The value of λ_s is set to 0.1 for attack on distillate measurement while for attack on the bottom impurity measurement λ_s is selected as -0.95. The results using EKF and UKF for the scaling attack is presented in Figure 54 and Figure 55, and Figure 56 and Figure 57, respectively. EKF and UKF detected scaling attack on distillate purity measurements in 0.003 and 0.005 hours as shown in Figure 54 and Figure 56 respectively. Both estimators are able to successfully identify the loop being attacked as confirmed from the result below which can be used for attack diagnosis.



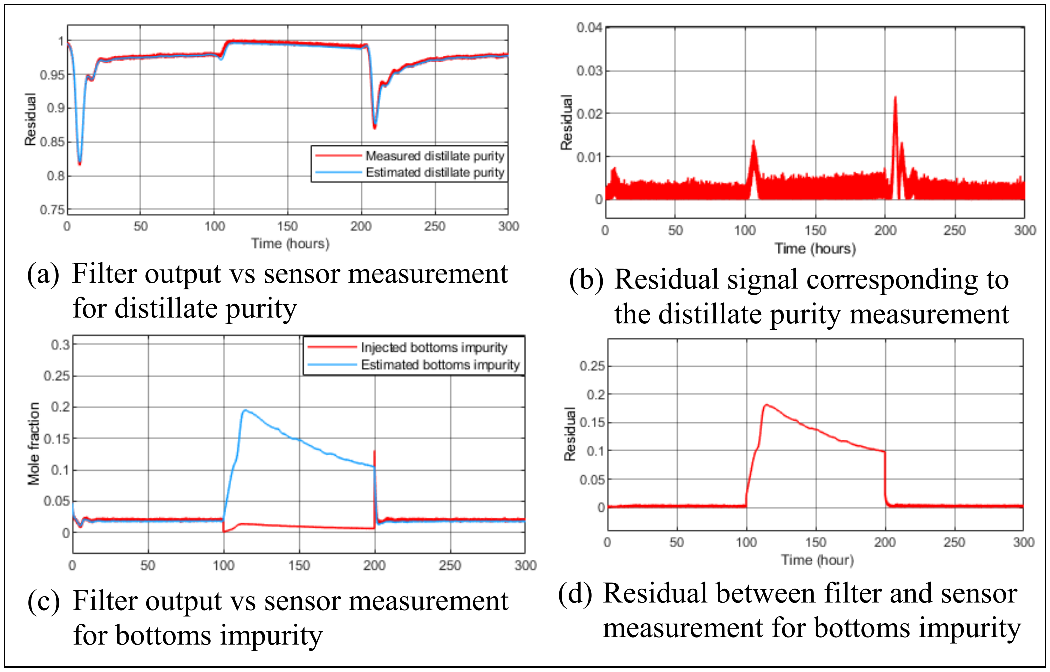
(a) Filter output vs sensor measurement for distillate purity

(b) Residual signal corresponding to the distillate purity measurement

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 54. Scaling attack detection on the distillate purity sensor using EKF.



(a) Filter output vs sensor measurement for distillate purity

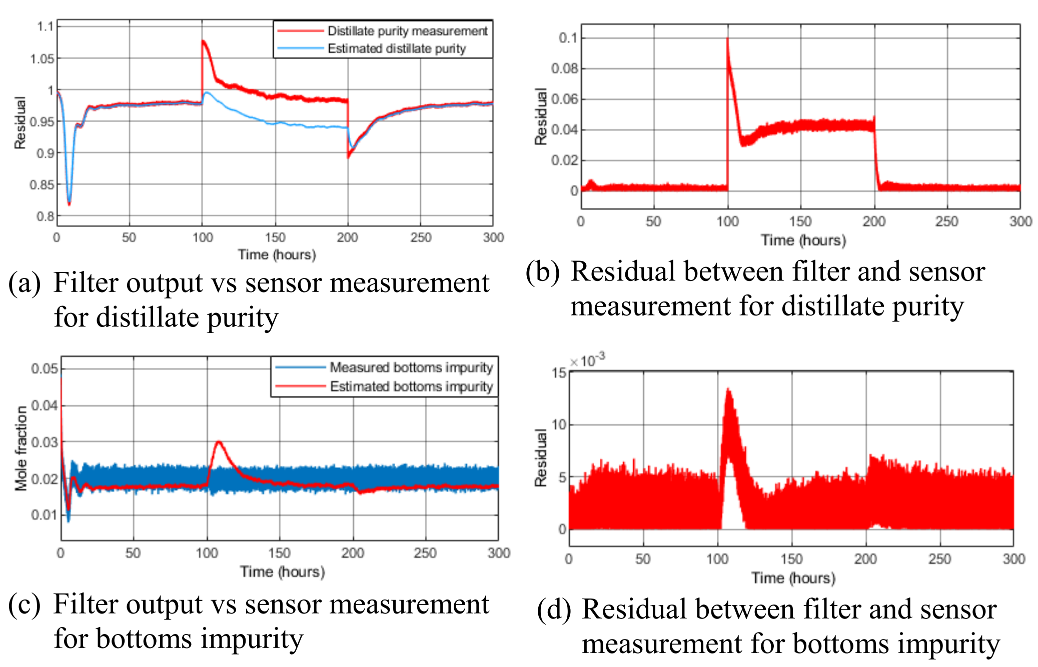
(b) Residual signal corresponding to the distillate purity measurement

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 55. Detection of scaling attack on the bottoms impurity sensor using EKF.

In the case of attack on bottoms impurity sensor as seen from Figure 55 and Figure 57 EKF and UKF detects the attack in .010 and 0.007 hours respectively. Both algorithms facilitate attack isolation by estimating un-attacked outputs correctly.



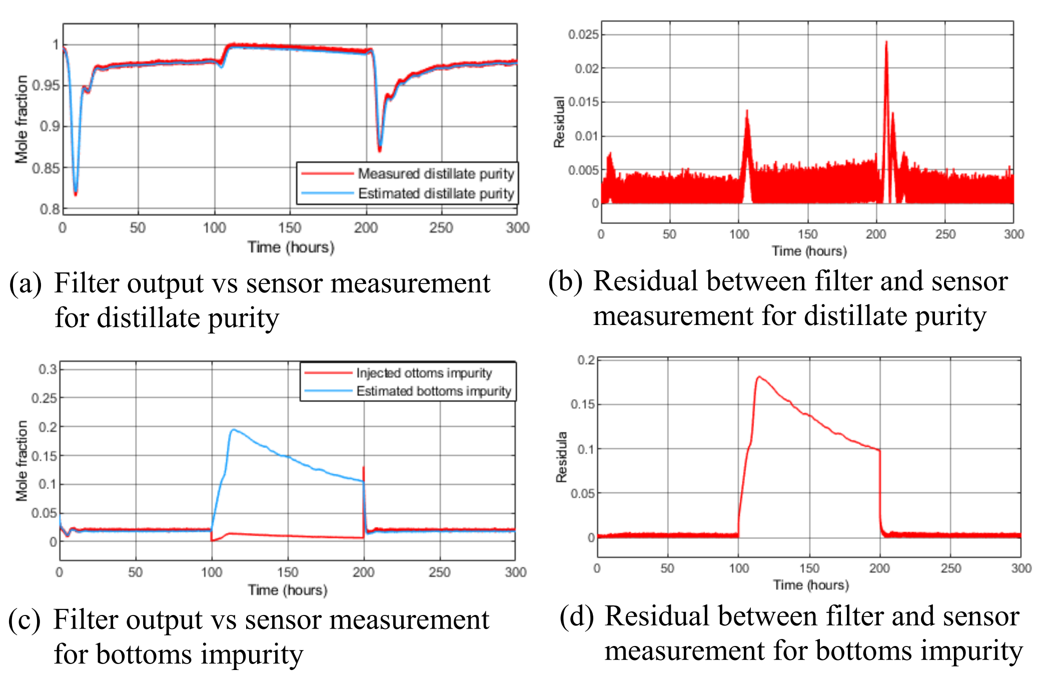
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 56. Scaling attack detection on the distillate measurement using UKF.



(a) Filter output vs sensor measurement for distillate purity

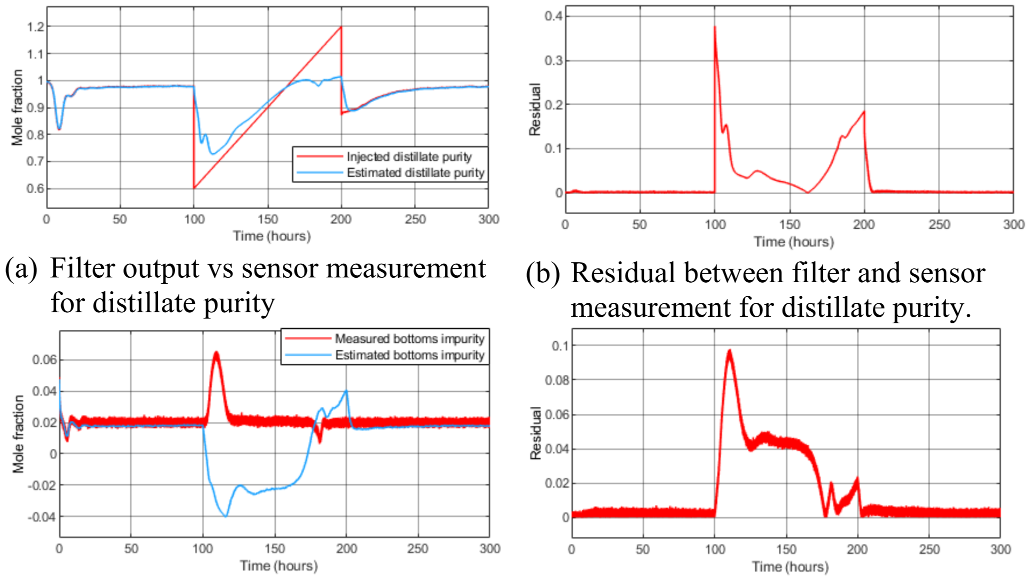
(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 57. Scaling attack detection on the bottoms impurity measurement using UKF.

- **Ramp attack:** The results for EKF and UKF have been presented in Figure 58, and Figure 59, and Figure 60, and Figure 61 respectively.



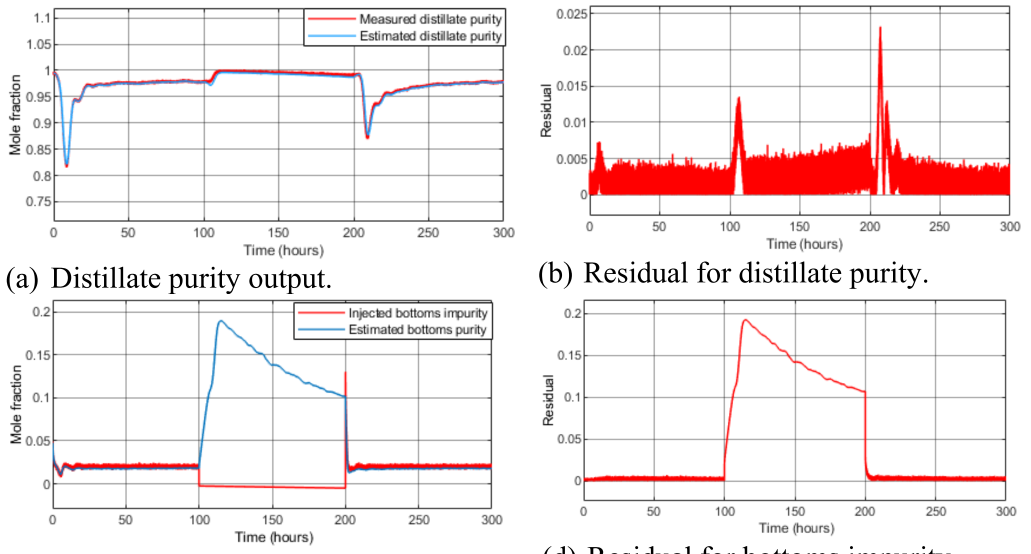
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity.

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 58. Ramp attack detection on the distillate measurement sensor using EKF.



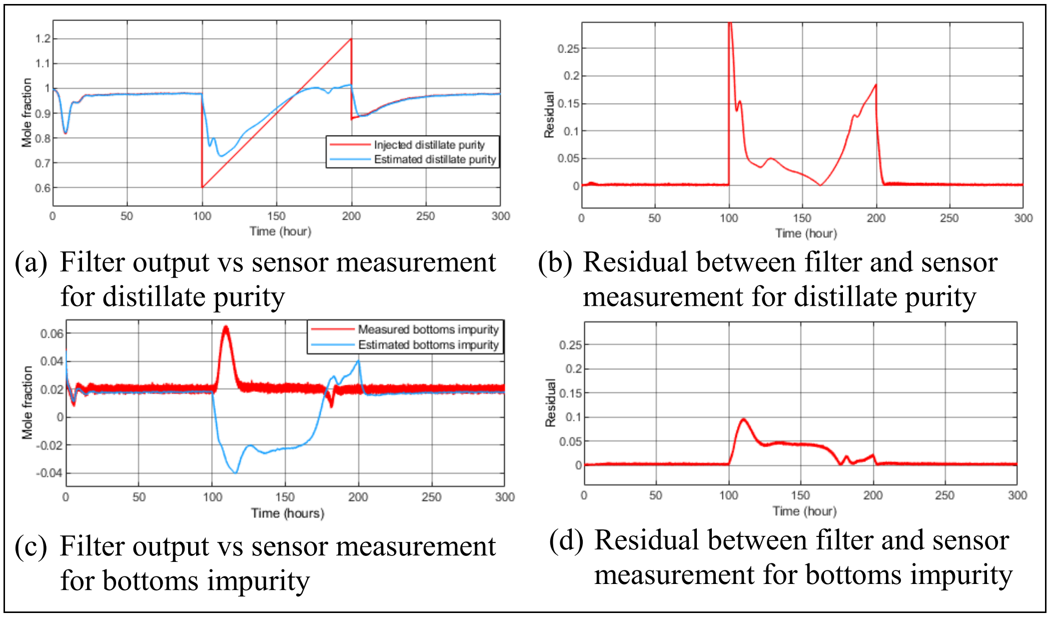
(a) Distillate purity output.

(b) Residual for distillate purity.

(c) Bottoms impurity output.

(d) Residual for bottoms impurity.

Figure 59. Ramp attack detection on the bottoms impurity sensor using EKF.



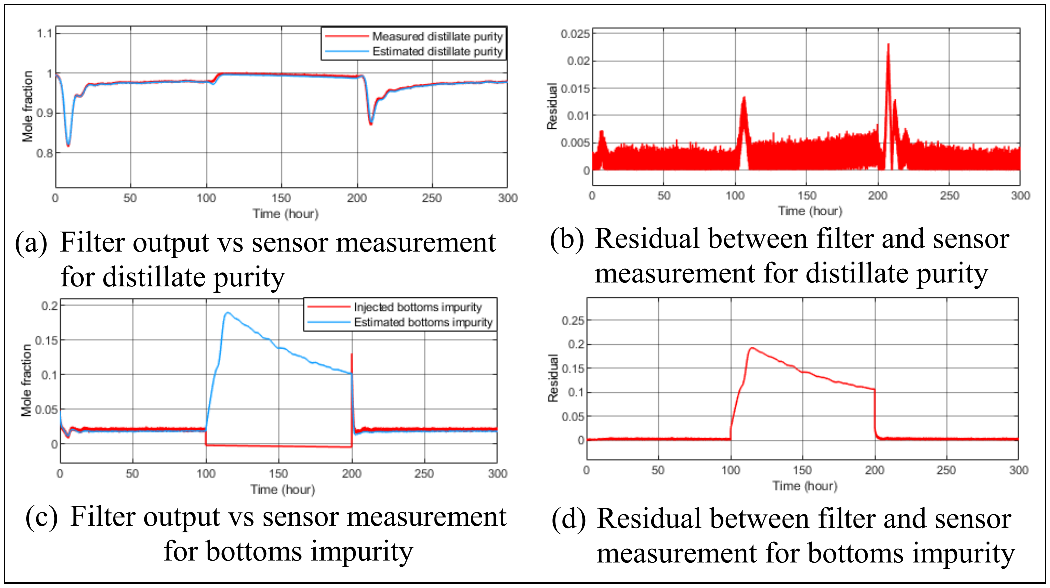
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 60. Detection of a ramp attack on the distillate purity sensor using UKF.



(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

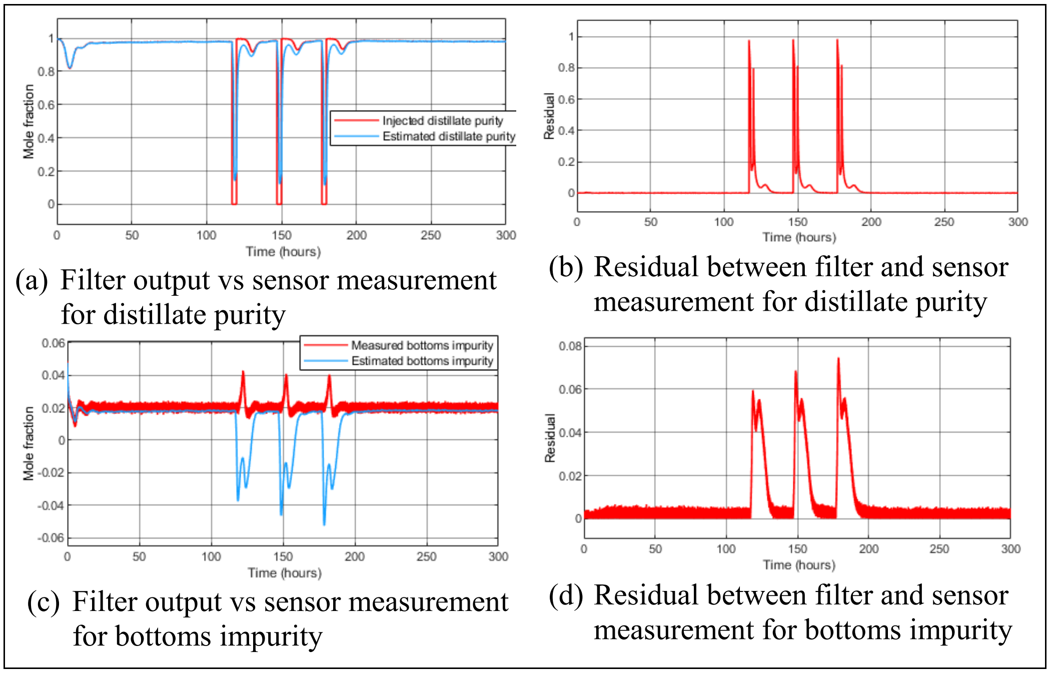
Figure 61. Ramp attack detection on the bottoms impurity measurement using UKF.

The ramp attack is simulated setting $\lambda_r = 0.006$ for distillate purity measurements and λ_r = -.000025 for bottoms impurity measurement. Both $r_D$and r_B can be used to detect the attack as can be seen from Figure 58 and Figure 60 respectively attack on distillate measurements. But for both EKF and UKF, r_D detected the attack faster in 0.007 and 0.003 hour respectively. Since both outputs get corrupted for both filters during attack it is not possible to trace the location of attack.

Comparing results for bottoms impurity attack in Figure 59 and Figure 61 the main difference lies in detection times. EKF detected the attack in 0.002 hours whereas UKF took 0.005 hours to detect the attack. In this case location of the attack can be traced since both filters only give corrupted output for sensors that are under attack.

- **Pulse attack**: In case of the distillate purity measurements the pulse signal is chosen with period of 30 hours and 90% pulse width whereas the period is set to 30 hours with 40% pulse width in case of attack on the bottoms impurity measurements. The model of pulse attack provided in Section 3.1.1 either latches to the true sensor measurements and outputs them or sends zeros to the controller. During the first few hours, for both the attacks, it sends the true sensor measurements to the controller. Although such case indicates disruption of resources but the attack detection time will be counted from the time the attacks starts to affect the outputs.

From the results in Figure 62 and Figure 64 it can be observed that, in case of both EKF and UKF, there appears discrepancy between the filter estimated output and controller input in both control loops which can be used for attack detection. The distillate purity attack is detected in 0.008 and 0.013 hour by EKF and UKF respectively. Since both the outputs gets corrupted hence neither EKF nor UKF provides any attack diagnosis feature for this attack. In the case of attack on the bottoms impurity measurements, the simulation results presented in Figure 63 and Figure 65 indicate that EKF and UKF detected the attack in 4.7 hours and 3.8 hours respectively. Besides that, it can be seen that both EKF and UKF based detection methods allows for isolation of attack since only the output measurements being tempered with by the attacker is affected by the attack.



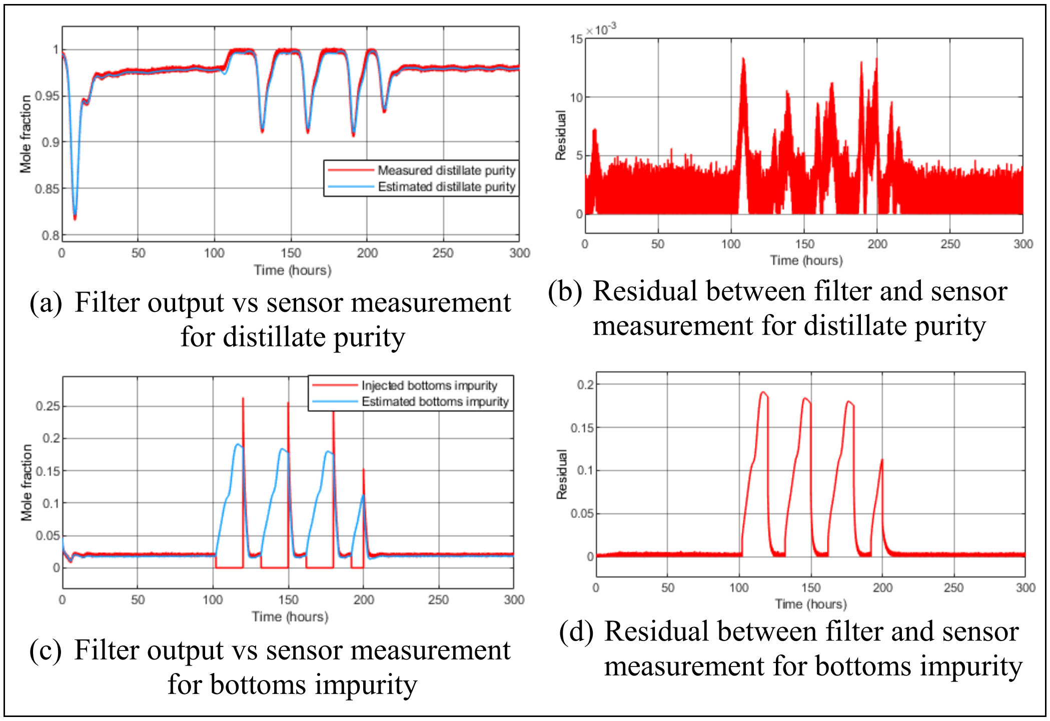
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 62. Detection of a pulse attack on the distillate purity sensor using EKF.



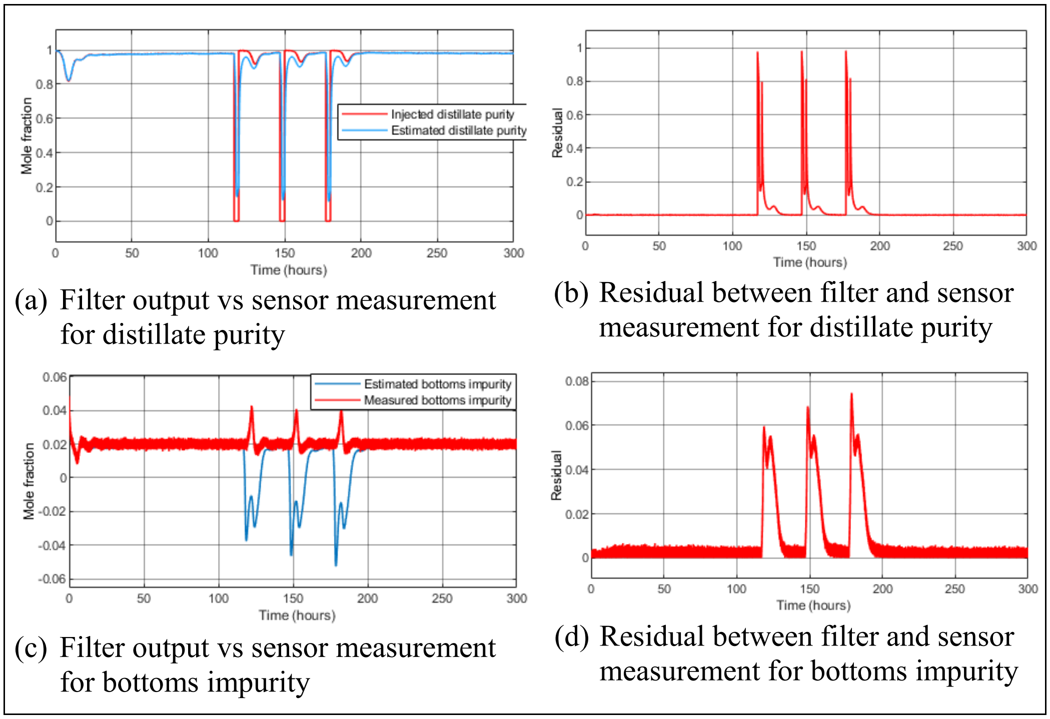
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 63. Detection of a pulse attack on the bottoms impurity sensor using EKF.



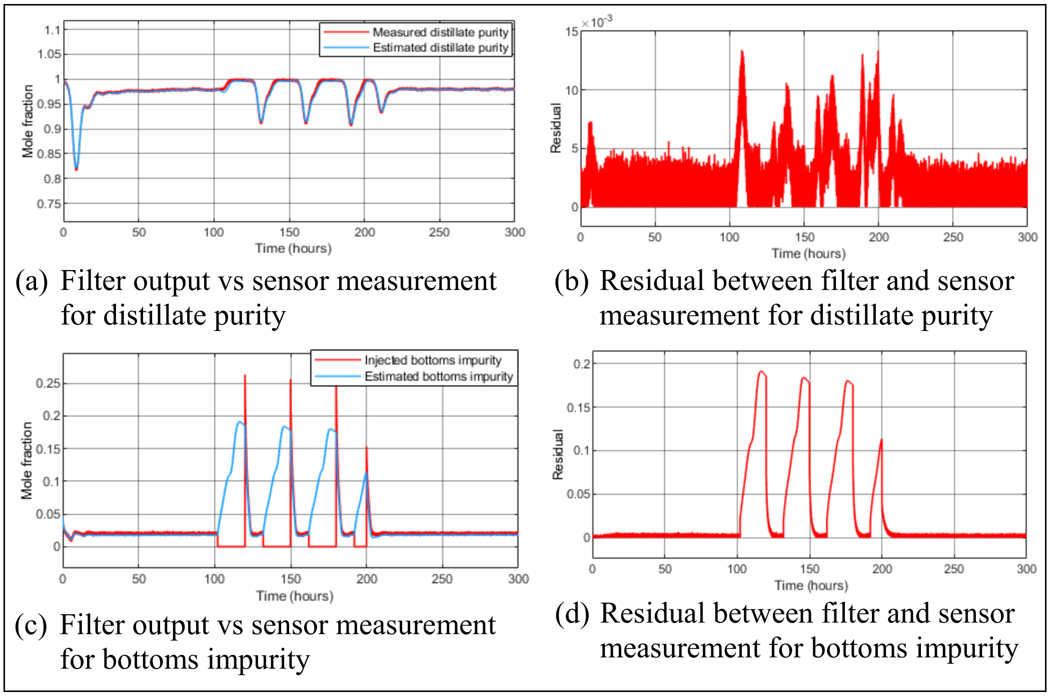
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 64. Detection of a pulse attack on the distillate purity sensor using UKF.



(a) Filter output vs sensor measurement for distillate purity

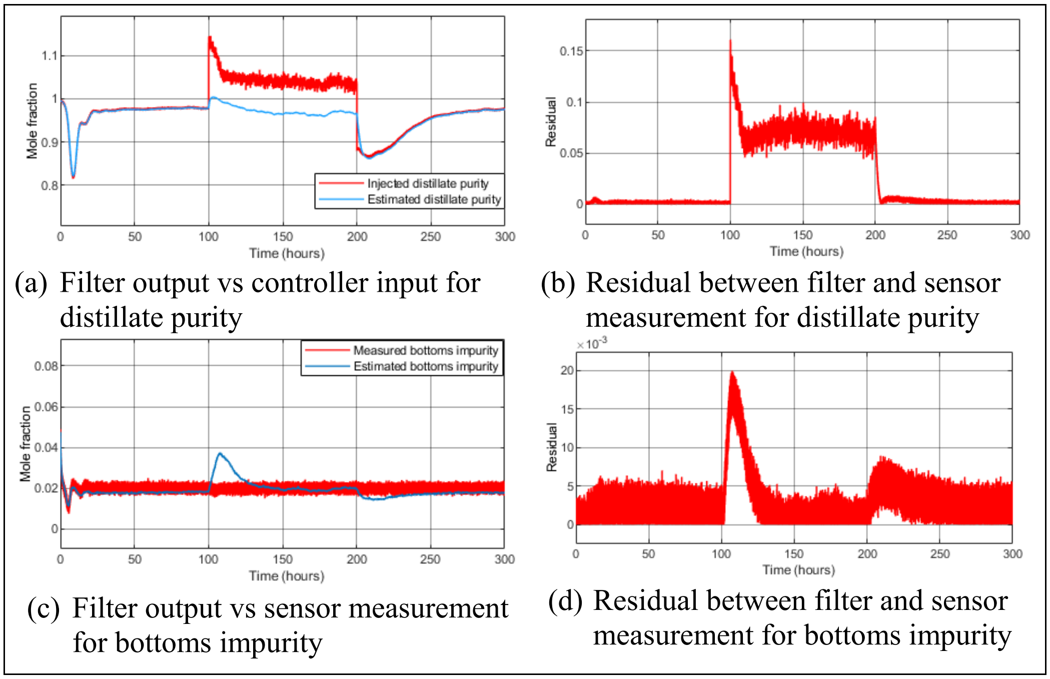
(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 65. Detection of a pulse attack on the bottoms impurity sensor using UKF.

- **Random attack:** The attack detection results for EKF are given in Figure 66 and Figure 67 and results for UKF are presented in Figure 68 and Figure 69.



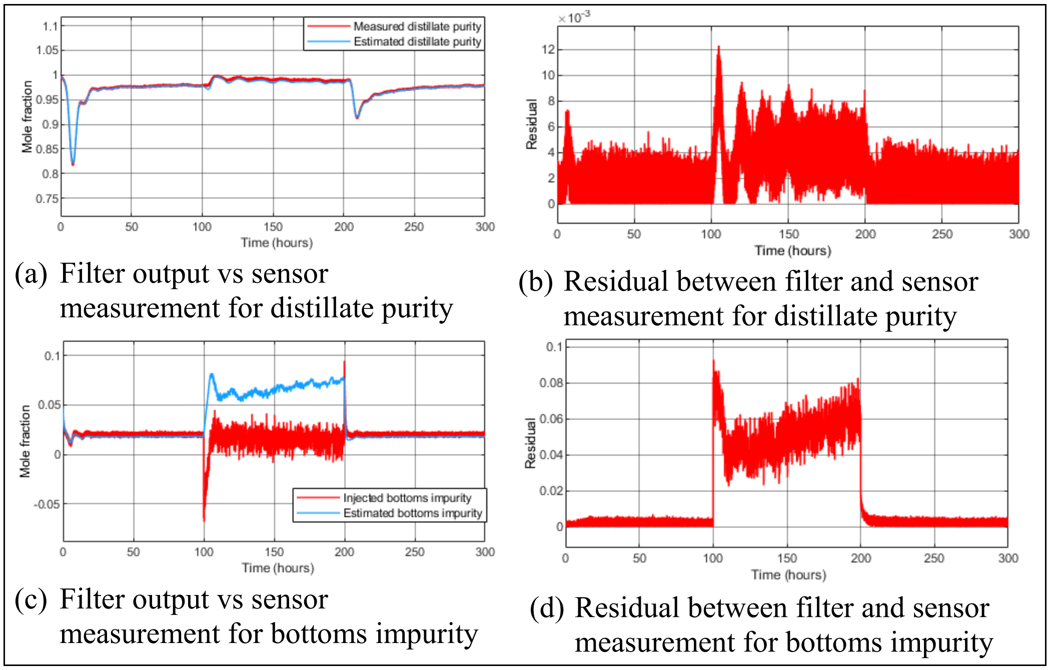
(a) Filter output vs controller input for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 66. Detection of a random attack on the distillate purity sensor using EKF.



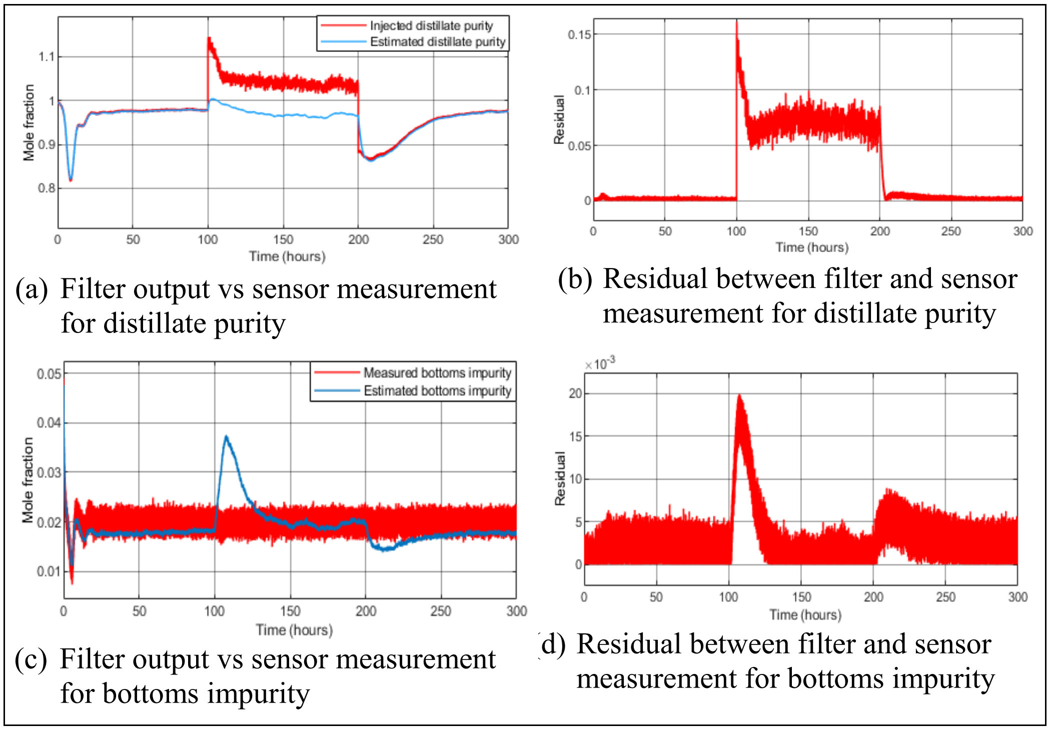
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 67. Detection of a random attack on the bottoms impurity sensor using EKF.



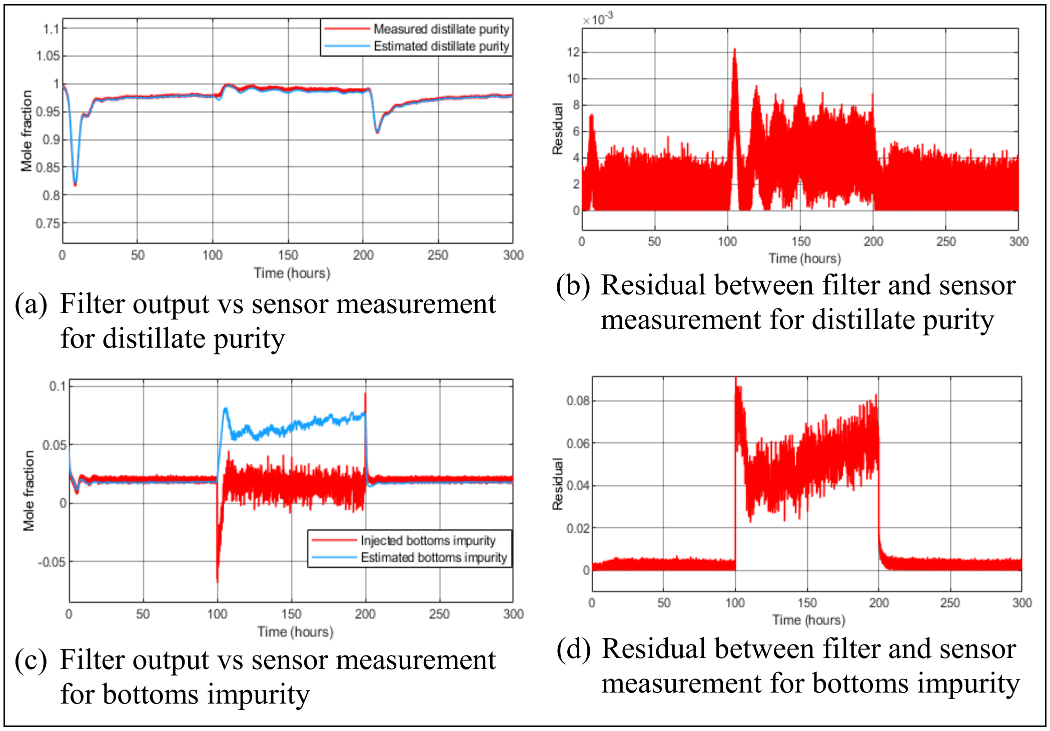
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 68. Detection of a random attack on the distillate purity sensor using UKF.



(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 69. Detection of a random attack on the bottoms impurity sensor using UKF.

The attack vectors are generated using random signals normally disturbed as N ~ (.15, .00005) and N ~ (-.08, 0.0000001) for the distillate purity and bottoms impurity measurements respectively.

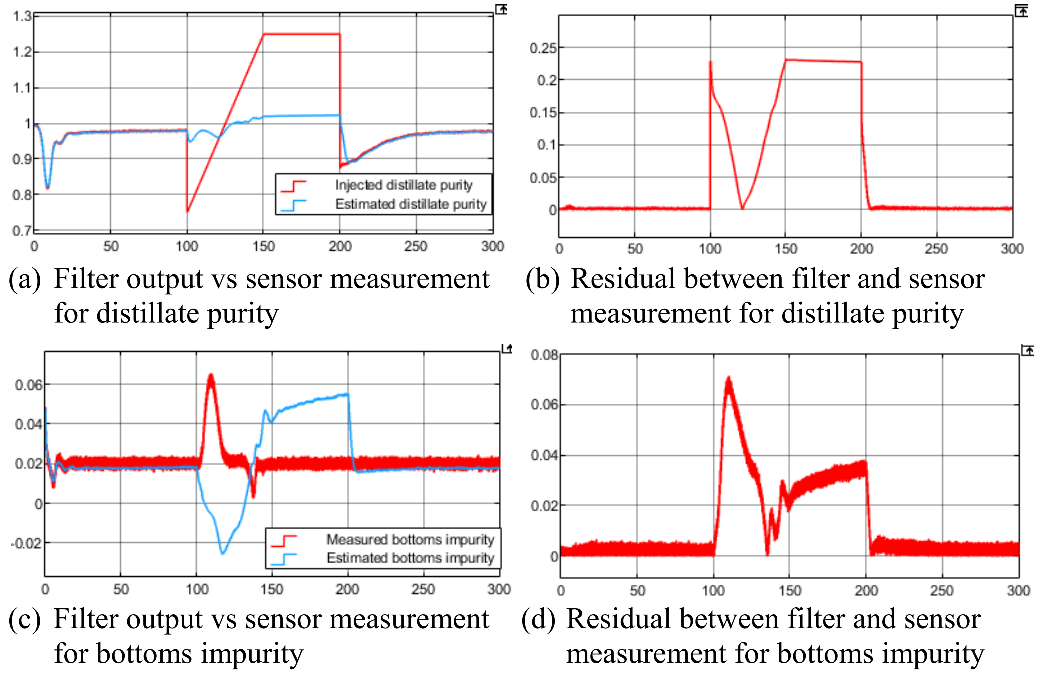
Looking at Figure 66 and Figure 68 it can be seen that the attack detection time for EKF and UKF are 0.011 and 0.005 hour respectively. Besides that, both EKF and UKF are able to track outputs which are unaffected by attacks accurately thus facilitating attack diagnosis by identifying the location of the attack.

From the results in Figure 67 and Figure 69, it can be seen that during attack on the bottoms impurity measurements, the residual r_B exceeds the threshold of 0.02 in 0.004 and 0.002 hours for EKF and UKF respectively. Similar to distillate purity, both filters are able to isolate the attack by correctly estimating the un-attacks outputs.

- **Bounded time varying attack:** Similar to section 3.2.2.a the attack signal is chosen as $y_i\ (t)= .75\text{-}sat\ ((t\text{-}100)/50)$ for the distillate purity measurement sensor and the attack signal for bottoms impurity measurement is chosen as $y_i\ (t) = 1.58\text{-}tan^{-1}(t)$.

Results for EKF are illustrated in Figure 70 and Figure 71 and results for UKF are presented in Figure 72 and Figure 73. EKF and UKF detects the attack on the distillate purity measurement sensor at 0.007 and 0.003 hour respectively which can be confirmed from Figure 70 and Figure 72. The other notable thing is both estimators incorrectly estimates the bottoms impurity hence can’t be used for attack isolation.

The detection time for attack on the bottoms impurity measurements is 2 hours and 3.710 hours for EKF and UKF respectively. Similar to the case during the distillate measurement attack both EKF and UKF provides attack diagnosis feature by correctly estimating the distillate purity during attack which can be used to trace the location of the attack in the control system.



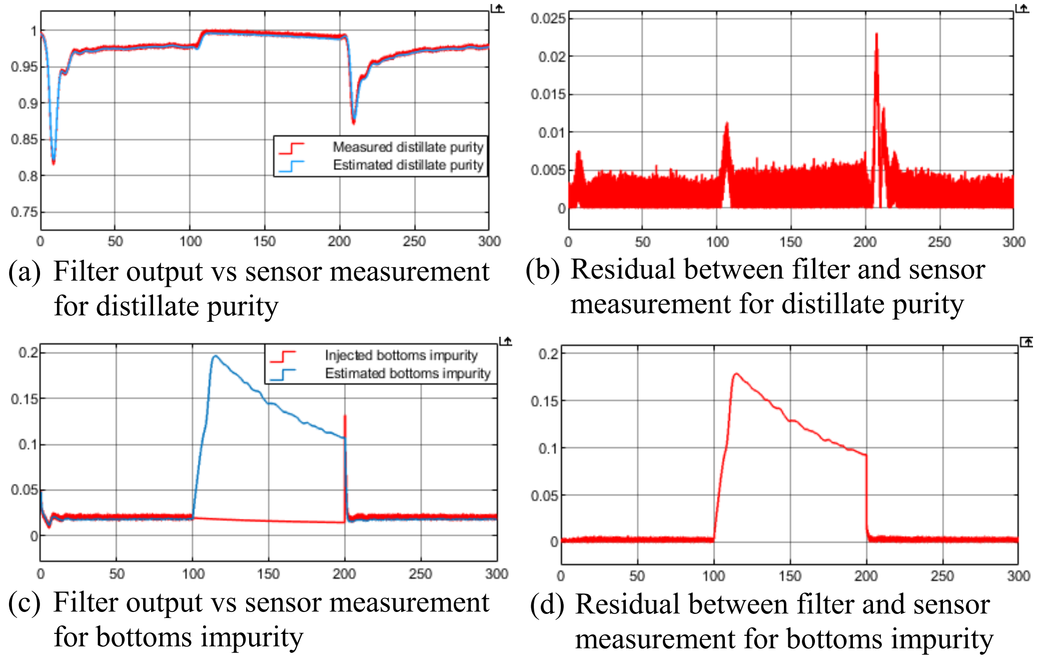
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 70. Detection of a bounded attack on distillate purity sensor data using EKF.



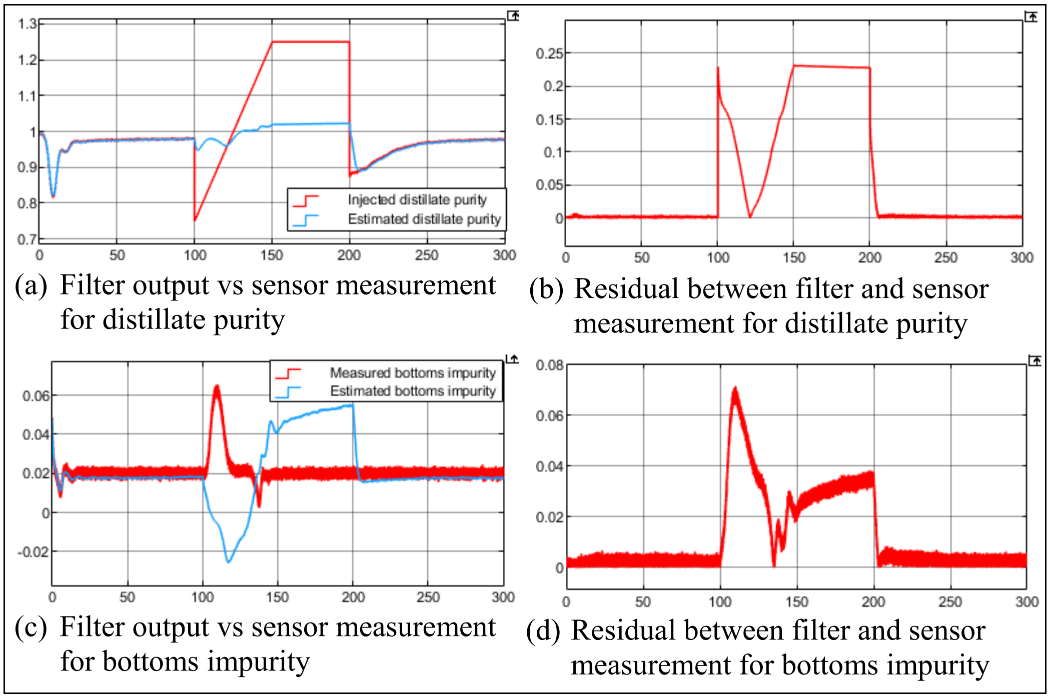
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 71. Detection of a bounded attack on the bottoms impurity sensor using EKF.



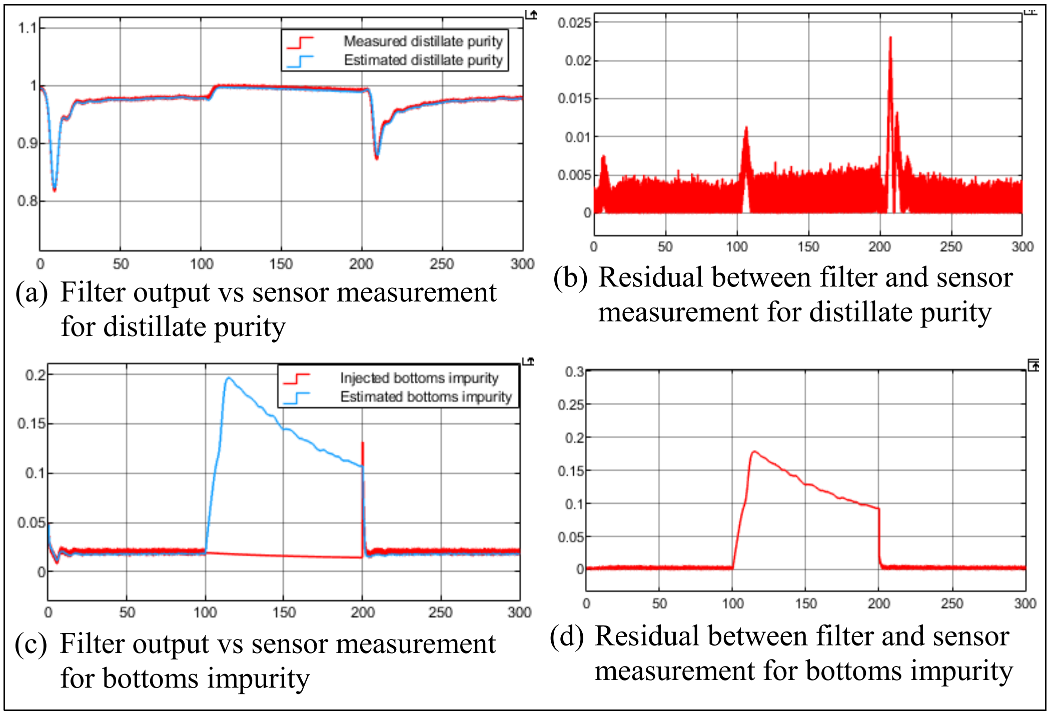
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 72. Detection of a bouned attack on distillate purity sensor using UKF.



(a) Filter output vs sensor measurement for distillate purity

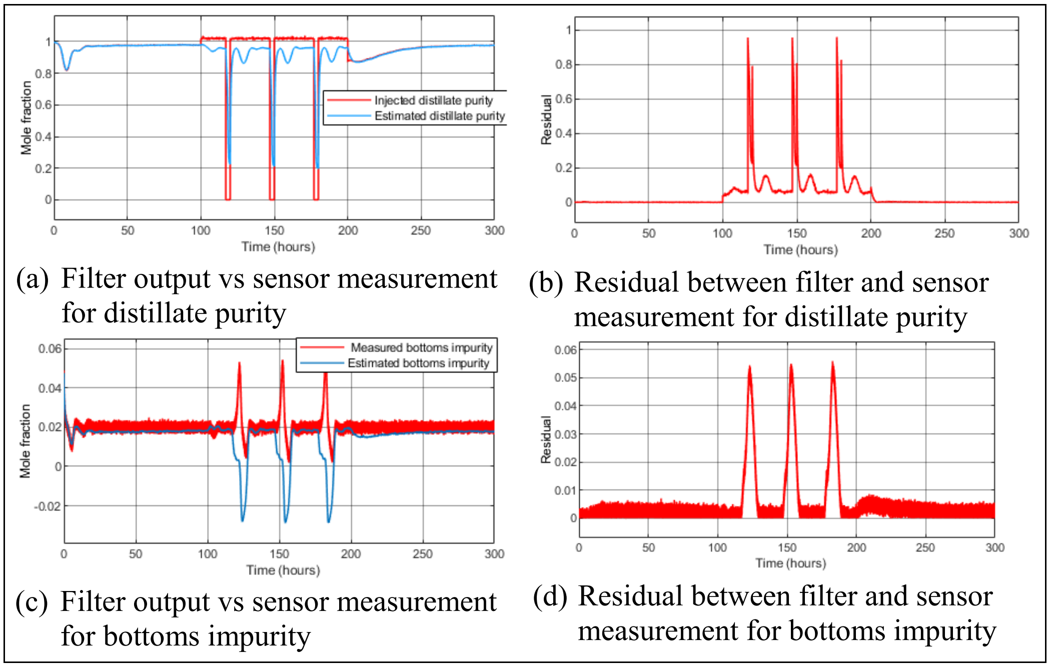
(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 73. Detection of a bounded attack on the bottoms impurity sensor using UKF.

- **False data injection attack (FDI):** Results for detection using EKF and UKF are presented in Figure 74 and Figure 75, and in Figure 76 and Figure 77 respectively.



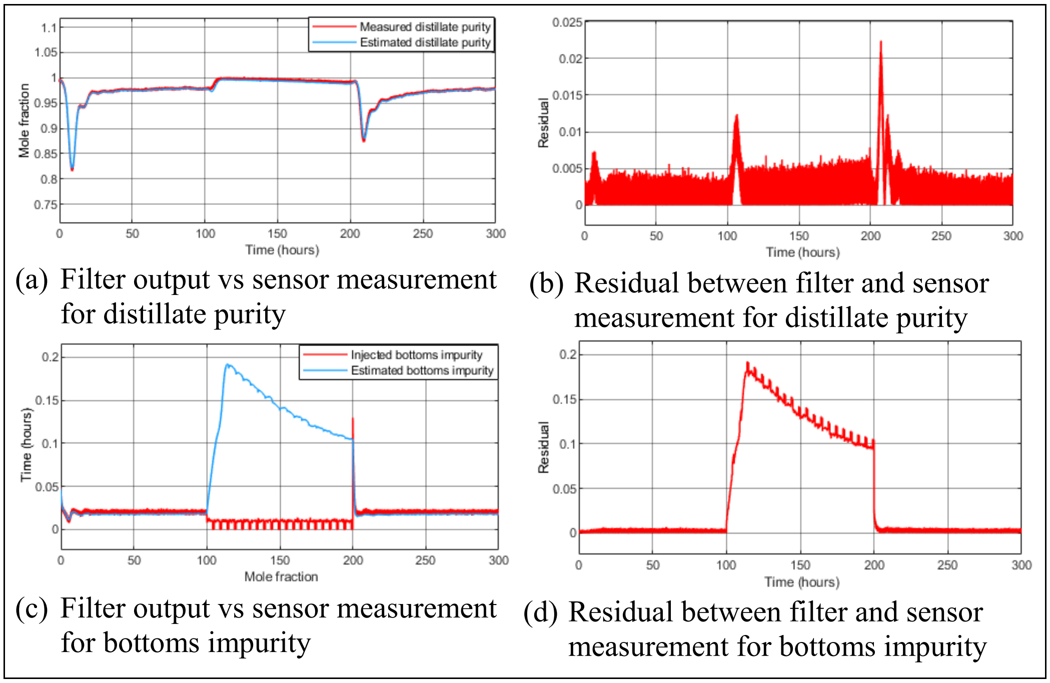
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 74. FDI attack detection on the distillate purity sensor using EKF.



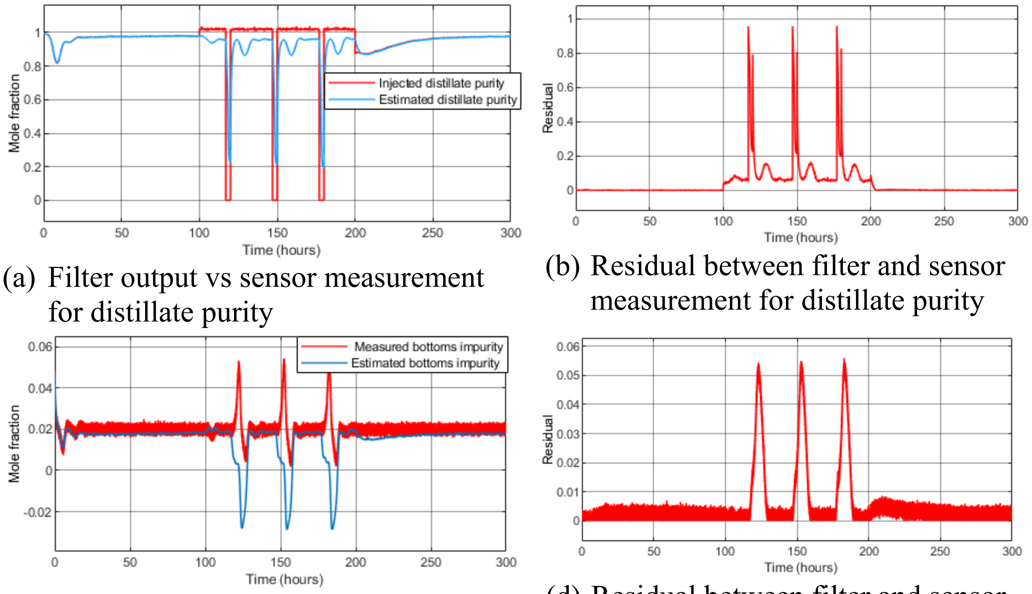
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 75. FDI attack detection on the bottoms impurity sensor using EKF.



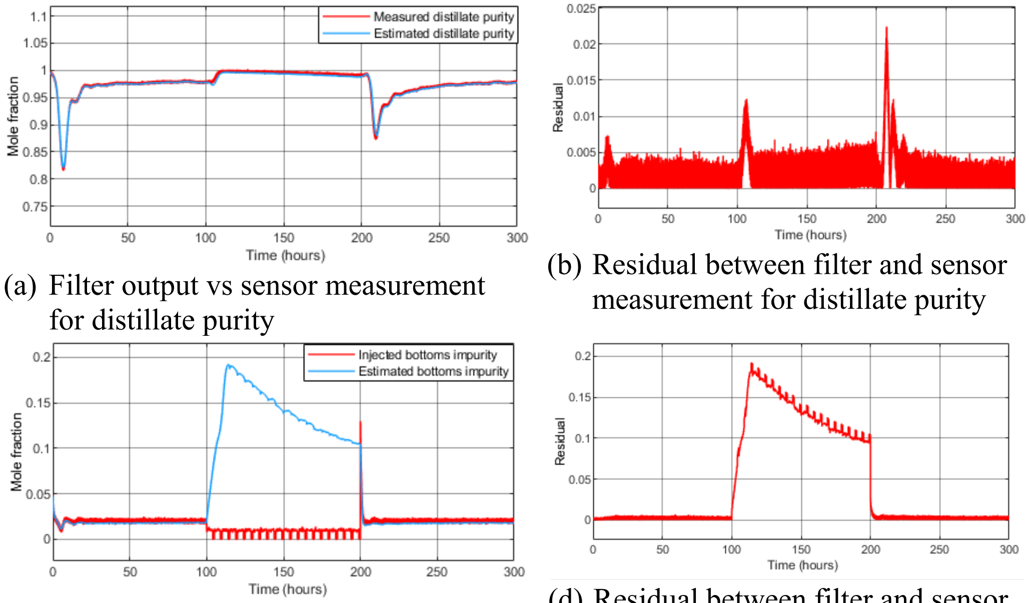
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 76. FDI attack detection on the distillate purity sensor using UKF.



(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

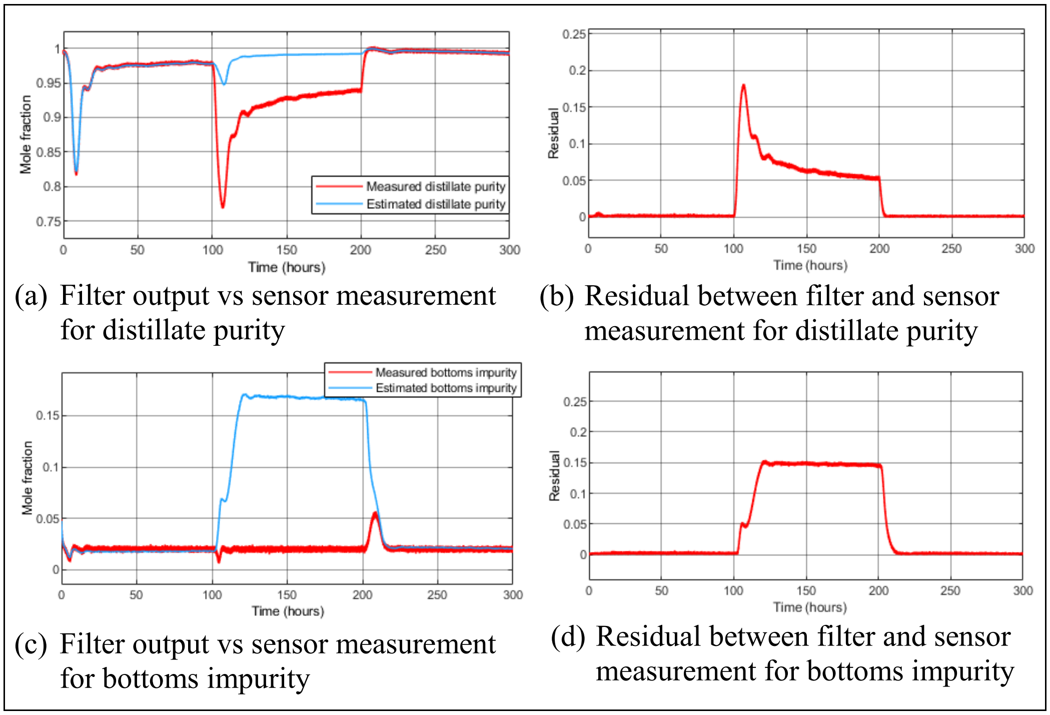
(d) Residual between filter and sensor measurement for bottoms impurity

Figure 77. FDI attack detection on the bottoms impurity sensor using UKF.

The attack vectors are generated using normally distributed random signals with distribution N~(1.02, .00002) and N ~ (.01, .000001) for distillate purity and bottoms impurity measurements respectively multiplied by continuous time pulses in order to bias the output data. The attack on the distillate purity measurement is detected by EKF and UKF in 0.003 and 0.001 hour respectively as can be seen from Figure 74 and Figure 76. The attacks affect both outputs of the filters for both cases of EKF and UKF which prevents from identifying the location of attack in the control system. From the results for detection of bottoms impurity sensor attack presented in Figure 75 and Figure 77 the detection times are 2.135 hours and 2.016 hours respectively. Both filtering algorithms are able to estimate the sensor measurements which are not under attack (i.e. distillate purity) correctly during the attack which can be used for attack isolation.

4.2.2 Actuator attack:

- **Data injection attack:** Attack detection results for EKF is presented in Figure 78 and Figure 79, and for UKF in Figure 80 and Figure 81 respectively.



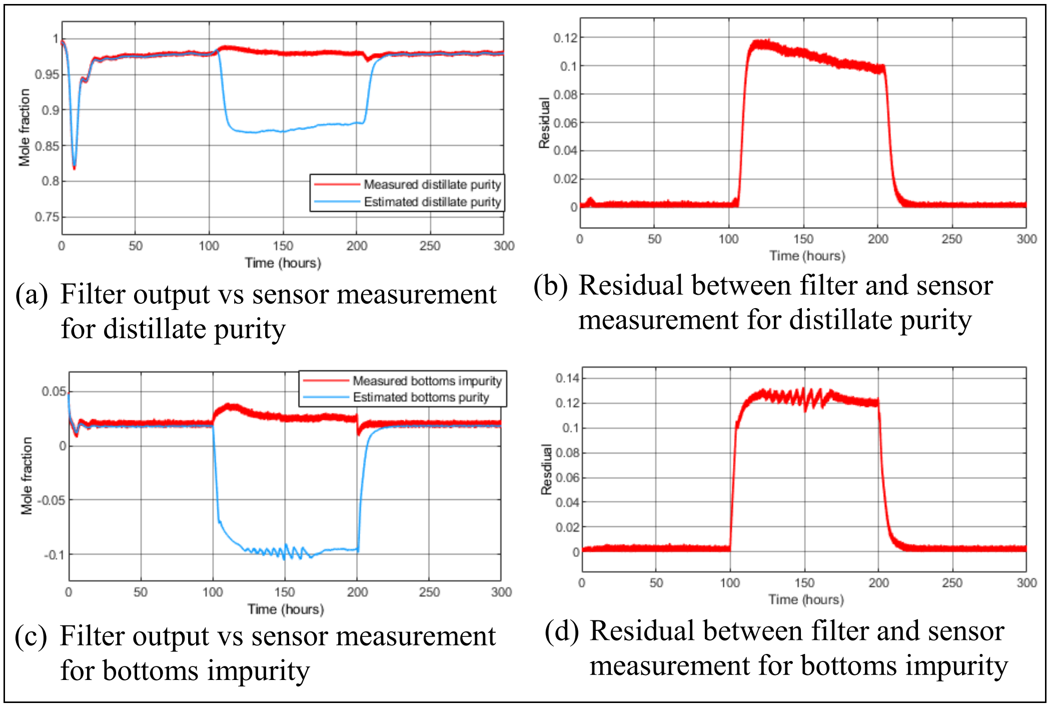
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 78. Detection of data injection attack on reflux flow rate data using EKF.



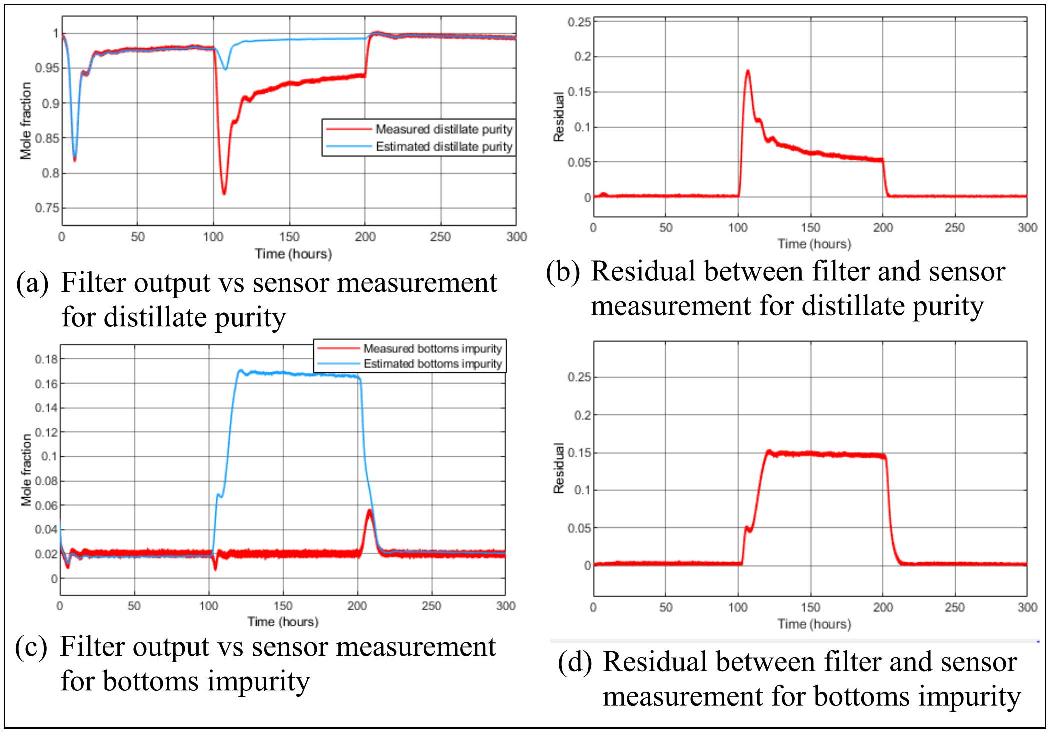
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 79. Detection of data injection attack on reboiler duty cycle data using EKF.



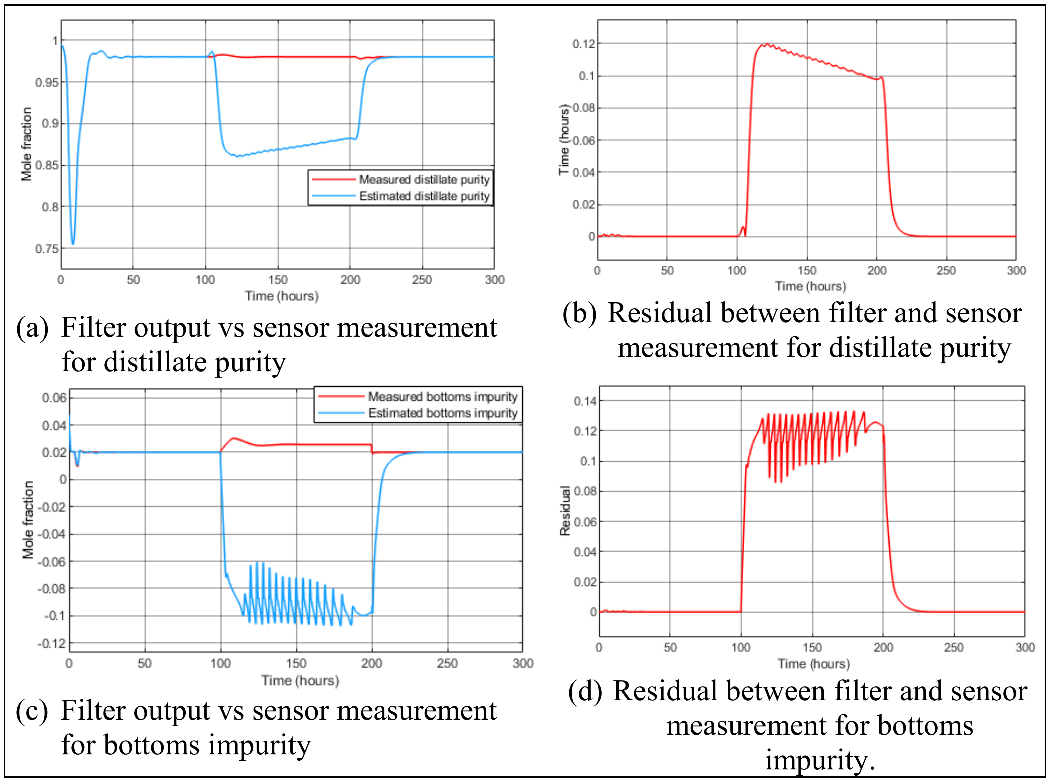
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 80. Detection of data injection attack on reflux flow rate data using UKF.



(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity.

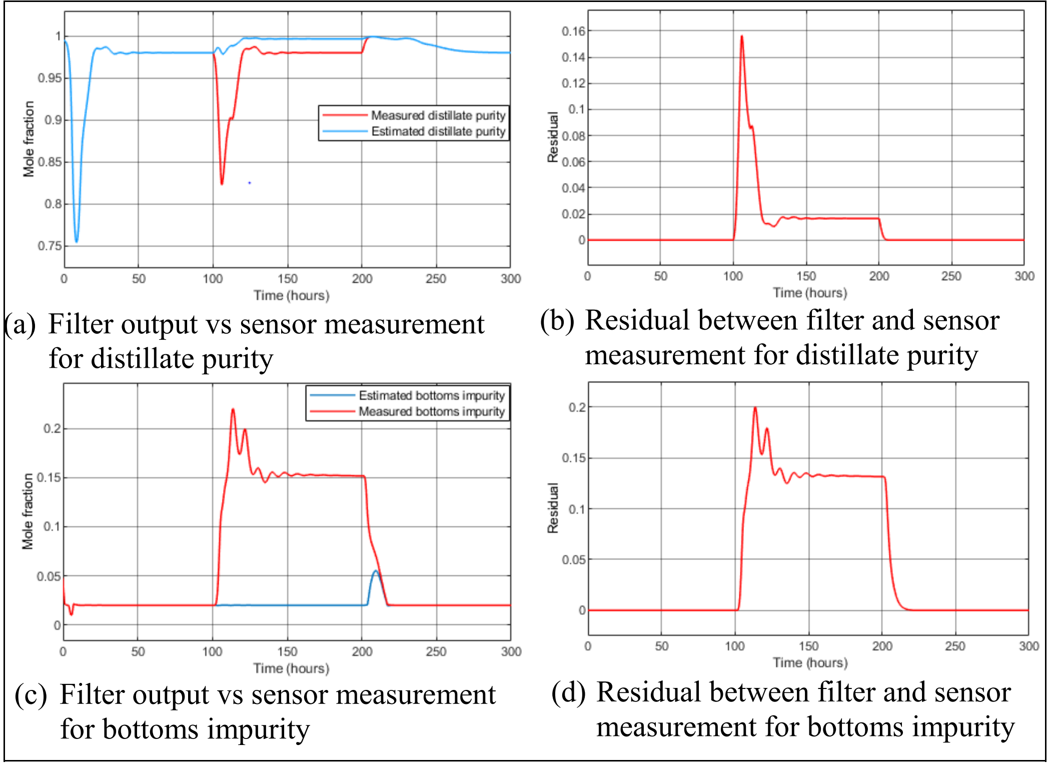
Figure 81. Detection of data injection attack on reboiler duty cycle data using UKF.

To generate the attack vector the value of a_r is set as 0.8 for reflux flow rate and a_r is set to 0.6 for reboiler duty cycle. The attack on reflux flow in Figure 78 and Figure 80 shows both r_D and r_B are large enough for attack detection. However, for both EKF and UKF, the attack is detected earlier by r_D in 1.760 and 1.783 hour respectively. This is expected since the reflux flow rate directly controls the distillate purity; thus it is more sensitive to any attack on the reflux flow rate. The comparatively large detection time compared to sensor attacks is because the distillate purity changes slowly to abnormality in reflux flow rate than distillate purity sensor.

Similar to reflux flow rate when the reboiler duty cycle data is attacked both r_D and r_B detects attack in case of both EKF and UKF as can been seen from Figure 79 and Figure 81. However since the bottoms impurity is more sensitive to changes in reboiler duty cycle hence r_B detects the attack faster. The detection times for EKF and

UKF are 0.210 and 0.133 hour respectively. As both control loops trigger attack warning hence the location of attack can't be traced by neither of EKF and UKF during this attack.

- **Replay attack:** The attack is injected by recording the reflux flow rate data up to t = 100 and then replaying this data to the plant between t = 100 – 200. The results for EKF and UKF is presented in Figure 82 and Figure 83 respectively. It can be seen that, for both filters, the attacks causes both r_D and r_B to exceed attack threshold. However r_D detects the attack faster in case of both EKF and UKF in 2.143 and 1.836 hours respectively. This is expected since the distillate purity is more sensitive to changes in reflux rate than bottoms impurity (or in other words the gain for distillate purity vs reflux flow rate is higher than bottoms impurity vs reflux flow rate).



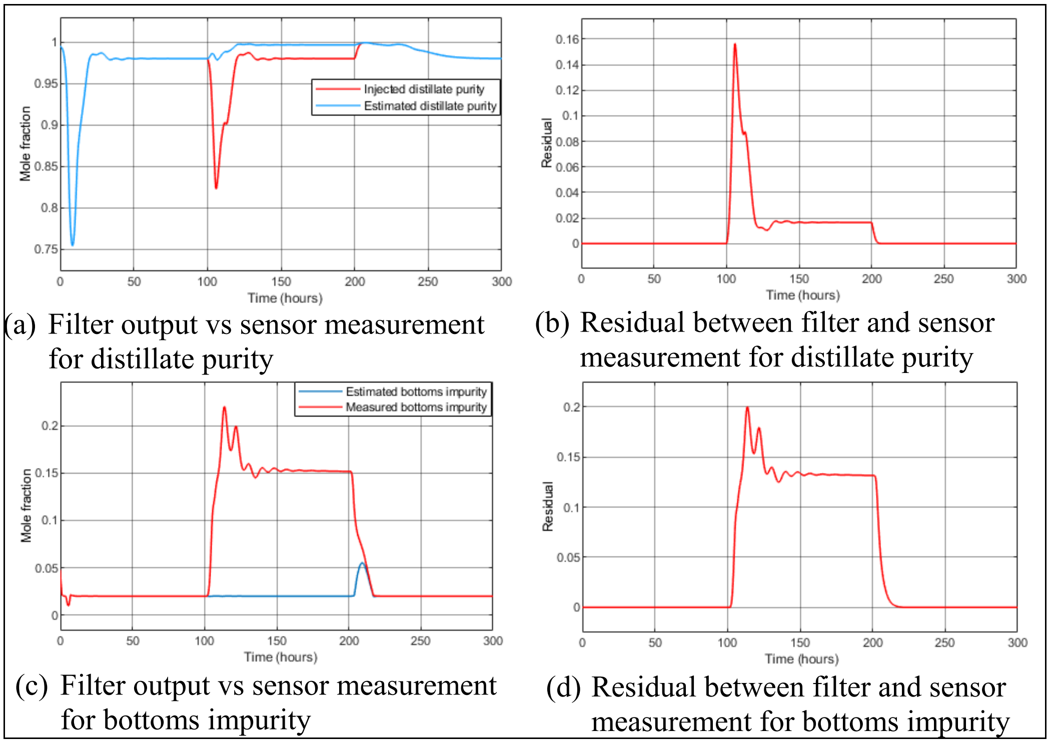
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 82. Results for replay attack detection on reflux flow rate using EKF.



(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

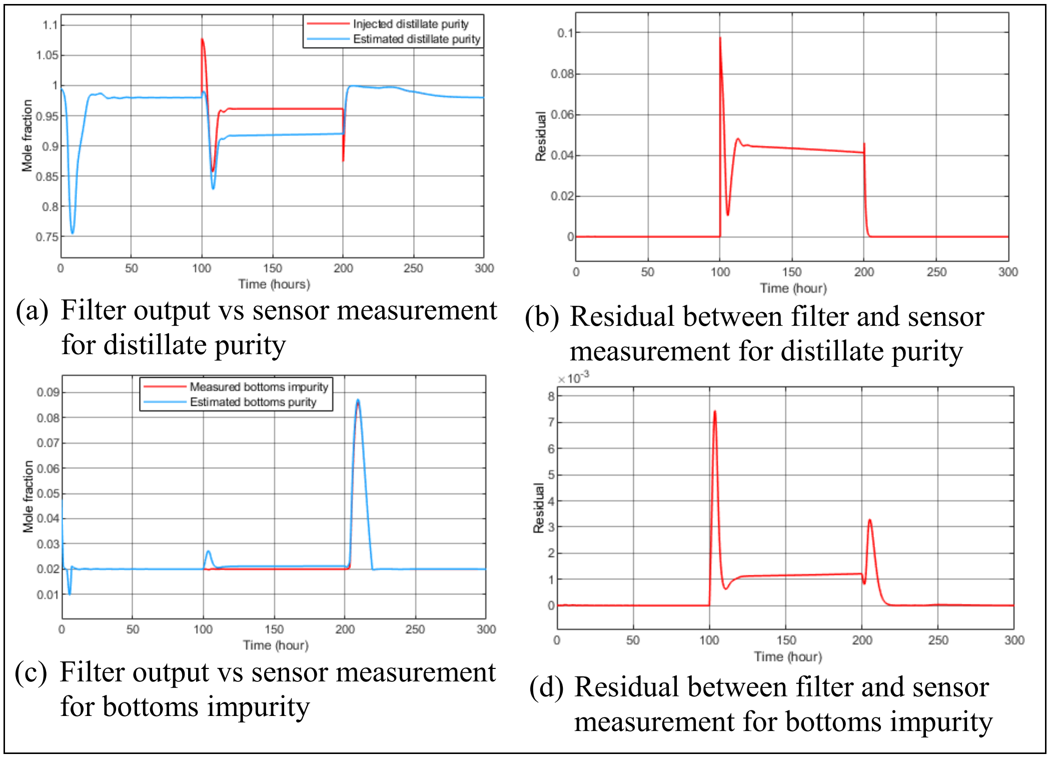
(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 83. Results for replay attack detection on reflux flow rate using UKF.

4.2.3 Chaining attack:

The ability of EKF and UKF in detecting the attack presented in Section 3.2.2.d whereby the controller is attacked by changing the controller gain and distillate purity sensor is attacked using a scaling attack is presented here. From the results in Figure 84 and Figure 85 both EKF and UKF detects the attack and the detection times are 0.003 and 0.001 hour respectively. The notable point here is both filters estimated the bottoms impurity accurately. It is observed previously in 4.1.1.a that during scaling attack on the distillate purity measurement the filters are able to locate the attack. However, in this case it can be seen that chaining scaling attack with controller attack makes the attack stealthy i.e. the only the sensor attack can be located. Thus the attacker uses these form of attacks to change controller parameters sneaking underneath the detection system.



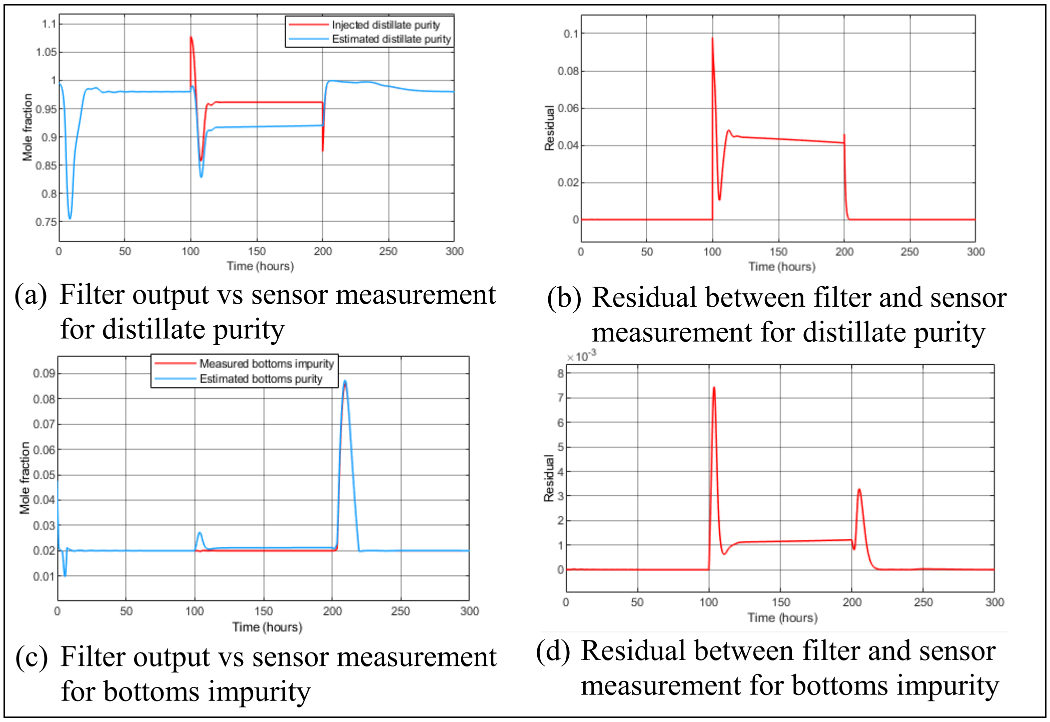
(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 84. Detection of chaining attack using EKF.



(a) Filter output vs sensor measurement for distillate purity

(b) Residual between filter and sensor measurement for distillate purity

(c) Filter output vs sensor measurement for bottoms impurity

(d) Residual between filter and sensor measurement for bottoms impurity

Figure 85. Detection of chaining attack using UKF.

Table 9 summarizes results for various attacks obtained using EKF and UKF.

Table 9. Summary of attack detection result.

Attack type	Attack name	Attacked parameter	Filter Type	Detection time (hour)	Attack detection	Attack isolation
Sensor attack	Scaling attack	Distillate purity	EKF	0.003	✓	✓
			UKF	0.005	✓	✓
		Bottoms impurity	EKF	0.010	✓	✓
			UKF	0.007	✓	✓
	Ramp attack	Distillate purity	EKF	0.007	✓	✗
			UKF	0.003	✓	✗
		Bottoms impurity	EKF	0.002	✓	✓
			UKF	0.005	✓	✓
	Random attack	Distillate purity	EKF	0.011	✓	✓
			UKF	0.005	✓	✓
		Bottoms impurity	EKF	0.004	✓	✓
			UKF	0.002	✓	✓
	Bounded time varying attack	Distillate purity	EKF	0.007	✓	✗
			UKF	0.003	✓	✗
		Bottoms impurity	EKF	2.000	✓	✓
			UKF	3.710	✓	✓
	False Data injection attack	Distillate purity	EKF	0.003	✓	✗
			UKF	0.001	✓	✗
		Bottoms impurity	EKF	2.135	✓	✓
			UKF	2.016	✓	✓
	Pulse attack	Distillate purity	EKF	0.008	✓	✗
			UKF	0.013	✓	✗
		Bottoms impurity	EKF	4.700	✓	✓
			UKF	3.800	✓	✓
Actuator attack	Data injection attack	Reflux rate	EKF	1.760	✓	✗
			UKF	1.783	✓	✗
		Reboiler duty cycle	EKF	0.210	✓	✗
			UKF	0.133	✓	✗
	Replay attack	Reflux rate	EKF	2.143	✓	✗
			UKF	1.836	✓	✗
Chaining attack	Controller and sensor attack	Distillate purity, Controller gain	EKF	0.003	✓	✗
			UKF	0.001	✓	✗

A summary of attack detection results using EKF and UKF have been illustrated with the aid of bar graphs in Figure 86 and Figure 87 respectively.

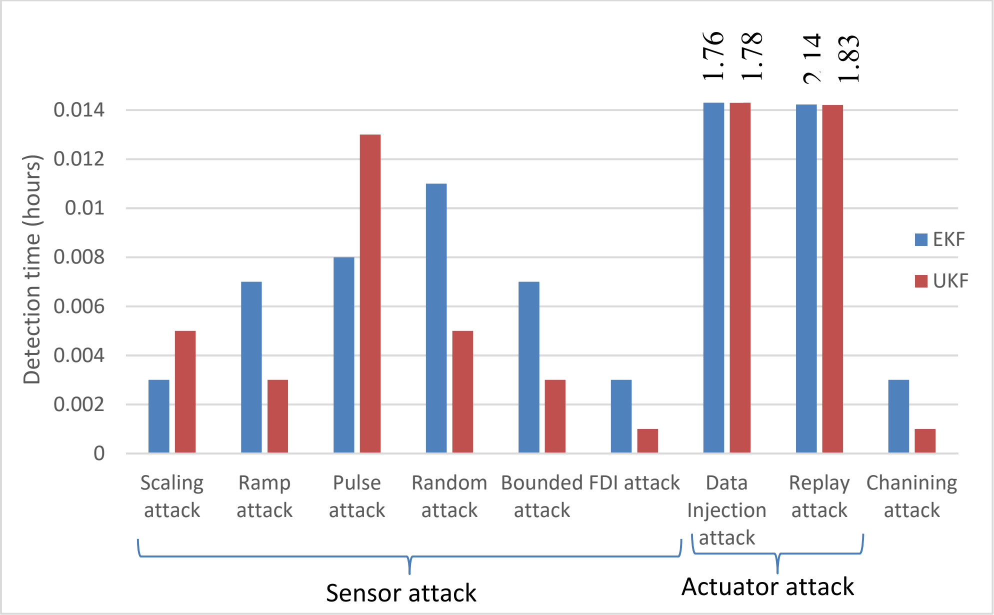


Figure 86: Summary of detection results for EKF vs UKF for distillate purity.

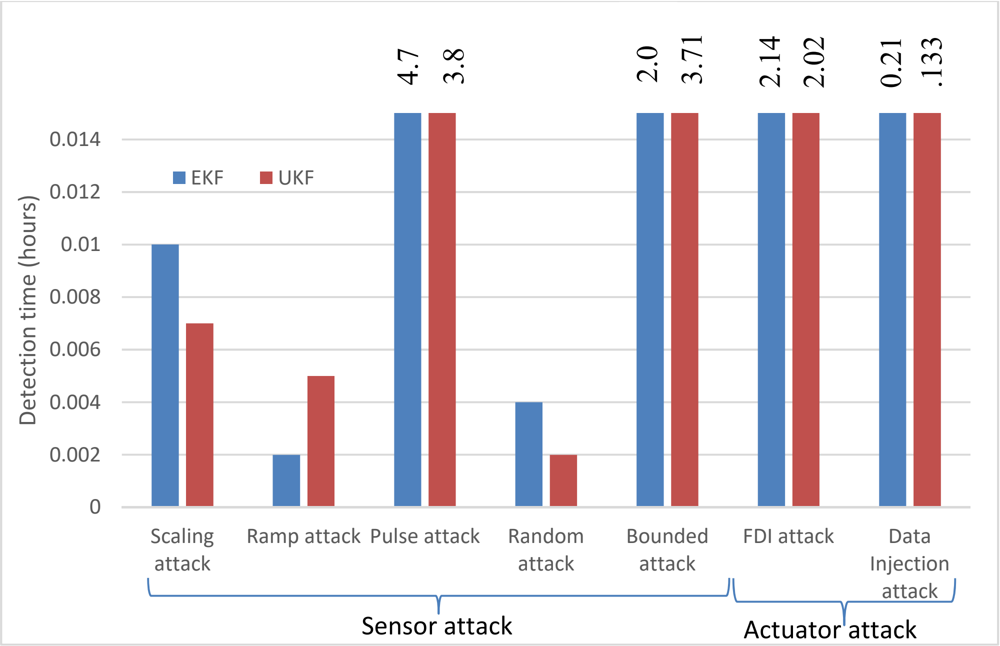


Figure 87: Summary of detection results using EKF and UKF for bottoms impurity.

As can be seen there are differences in detection times for EKF and UKF. One of the reason is the way fundamentally the algorithms are implemented; EKF uses mean state estimates for output estimation whereas UKF uses sigma points for estimation which makes it more accurate. Besides that, there is random process and measurement noise which also affects the detection times. The average detection times for EKF and UKF in case of distillate purity are .438 hour and .406 hour respectively, and for EKF and UKF are 1.29 hours and 1.22 hours respectively.

4.3 Attack detection for Aspen Based Column Model:

The proposed detection technique for the column model in Section 2.3 is based on the state estimation using an observer. An observer can predict the output and the internal states of a system from the knowledge of the system inputs and outputs (i.e. the sensor and actuator readings) which can be used to check the integrity of the communication channels between the sensor and controller, and controller and actuator. At first we place our focus on the observer design following which the method of attack detection will be elaborated on. During the steady state the column operation is deemed to be stable. A linear model of the column is generated around the stable operating point using Aspen Plus Dynamics which is used for the observer design. Thus it is valid to design detection techniques using the linear model of the plant.

4.3.1 Linearization of the distillation column model:

The linearized model of the plant is obtained by applying Taylor series around the steady state stable operating point of the DC and is expressed in the state space form as follows. Table 10 presents the values of inputs and outputs used during linearization. The linearized state space matrices of the DC can be written as:

$$\dot{x_\delta}(t) = A\, x_\delta(t) + B\, u_\delta\, (t)$$
$$y_\delta(t) = C\, x_\delta(t) + D\, u_\delta(t) \qquad (40)$$

Where, $x_{\delta}(t) = x(t) - x_e$; $u_{\delta}(t) = u(t) - u_e$; $y_{\delta}(t) = y(t) - y_{\delta}$, *x(t), u(t)* and *y(t)* represents the system states, input and output respectively. The delta quantities represent the deviation away from the stable equilibrium point by taking the difference between the current value and the equilibrium point. For the case of a distillation column, molar holdup of the lighter and heavier product and the enthalpy across all trays in the column along with the condenser and reboiler are the system states. In total the proposed column in section 2.3 containing 32 trays including condenser and reboiler as mentioned in Table 7 has 96 states, 6 inputs and 5 outputs. Therefore, $A \in R^{96x96}$, $B \in R^{96x6}$, $C \in R^{5x96}$ and $D \in R^{5x6}$. This model is used towards designing a Luenberger type observer which is described below.

Table 10. Values of inputs and outputs during stable operation.

Input parameter name	Value	Output parameter name	Value
Condenser duty (MMKCal/hr)	-.527	Column pressure (atm)	14.18
Reflux flow rate (kmole/hr)	124.28	Concentration of distillate (mole fraction)	.974
Distillate flow rate (kmole/hr)	40.179	Liquid level (m)	.749
Re-boiler duty (MMKCal/hr)	.602	Concentration of bottoms (mole fraction)	.009
Bottoms flow rate (kmole/hr)	59.763	Liquid level in column base (m)	.751
Feed flow rate (kmole/hr)	100		

4.3.2 Luenberger Observer design:

In full order observer, given the system parameters A, B, C, D and the values of the inputs and outputs over a time interval, it is possible to estimate the state when the system is observable if the pair (A,C) is observable. Unfortunately, the linearized model of the column acquired from Aspen Plus Dynamics is not observable. In order to resolve this issue the transfer function matrix of the linear model is derived using the step response which is used to derive a reduced order linear model of the system. Since the reduced model is derived using transfer function that only contains the observable and controllable states hence the model is fully observable. The state space matrices for the reduced system is given below.

A_r = [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 -1/1.45 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 -1/1.36 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 -1/1.44 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 -1/4.56 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 -1/1.35 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 -1/1.60 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 -1/6.30 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 -1/3.73 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 -1/2.57 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 -1/2.53 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 -1/2.59 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 -1/3.20 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/1.45 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/4.10 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/1.60 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/3.00 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/3.70 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/2.51]

B_r = [-6/46.1 -.258/50 -137/50 -3.07/50 847/48.9 .645/50
-0.028/50 -.1666 144/50 3.96/50 -1150/50 7.77/50
1 0 0 0 0 0
0 1 0 0 0 0
0 0 1 0 0 0
0 0 0 1 0 0
0 0 0 0 1 0
0 0 0 0 0 1
1 0 0 0 0 0
0 1 0 0 0 0
0 0 1 0 0 0
0 0 0 1 0 0
0 0 0 0 1 0
0 0 0 0 0 1
1 0 0 0 0 0
0 1 0 0 0 0
0 0 1 0 0 0
0 0 0 1 0 0
0 0 0 0 1 0
0 0 0 0 0 1]

C_r = [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 -.0134 -.0195 60.417 -.0029 66.148 -.0259 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 .000002 .00023 -.043 .00389 -1.143 .0014 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 .000036 -.000118/4.1 -.0000287 -.0000522 .1124 -.000693]

D_r = 0;

The aim is to design an observer to estimate small changes in states around the equilibrium point. Based on the reduced linear system model the state space representation of the full order observer or Luenberger observer is determined as follows:

$$\dot{\hat{x}}_\delta(t) = A\hat{x}_\delta(t) + Bu(t) + K(y_\delta(t) - \hat{y}_\delta(t))$$

$$\dot{\hat{y}}_\delta(t) = C\hat{x}_\delta(t) + Du_\delta(t) \qquad (41)$$

where $\hat{x}_\delta$, $\hat{y}_\delta$, y_δ, and K are estimated state, estimated output, output measurement and observer gain respectively. The actual output estimate is computed from the observer output estimate as following:

$$\hat{y}(t) = \hat{y}_\delta(t) + y_{eq} \qquad (42)$$

where $\hat{y}(t)$ is the estimated output and y_{eq} is the output value used during linearization.

The error dynamics in estimation is given by,

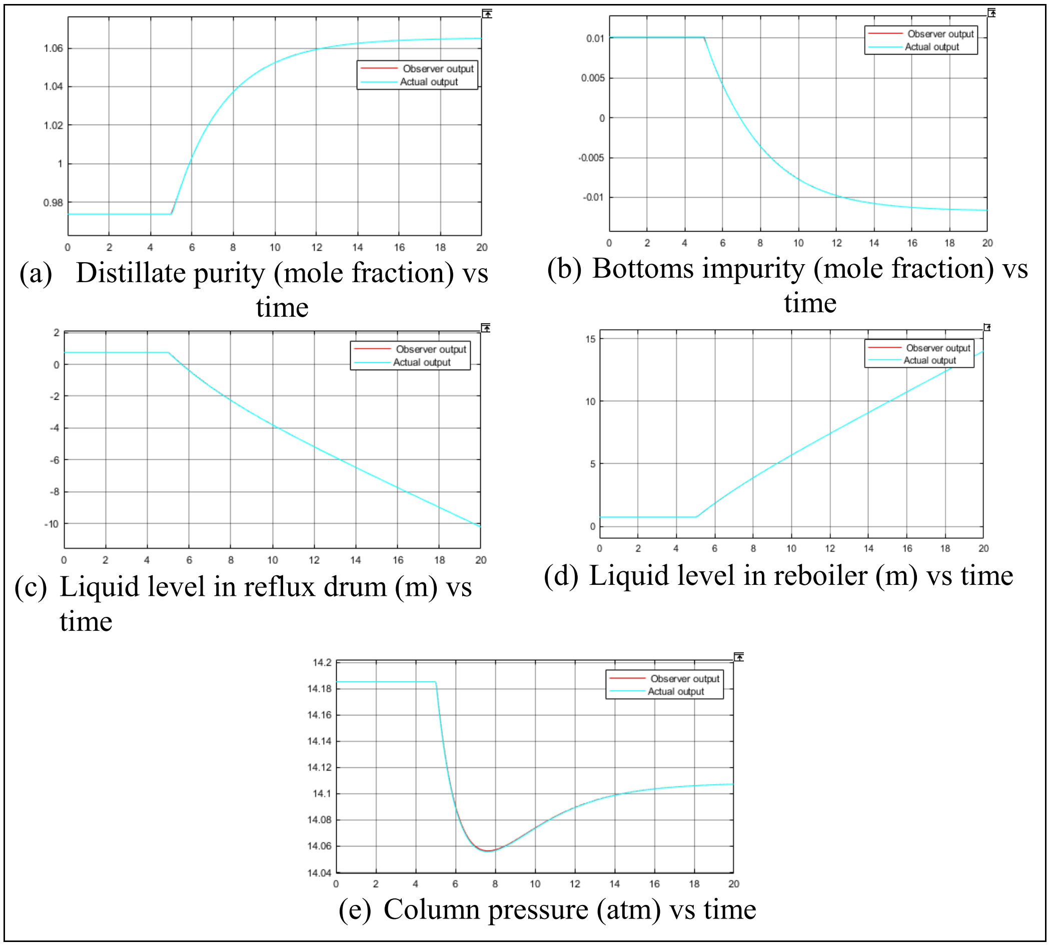
$$\dot{e}(t) = (A - KC)e(t) \qquad (43)$$

The poles of *(A-KC)* determine the error convergence rate. Thus they had to be selected large enough negatively to be able to track state changes of the system within an acceptable time period. The gain matrix is computed in Matlab using linear quadratic regulator (LQR) by arbitrarily choosing Q and R that rendered the largest negative poles of *(A-KC)* in order to guarantee fast convergence. LQR is a well-known technique used for computing the optimal gain for state feedback which for (40) is defined as follows.

$$u_\delta(t) = -Kx_\delta(t)$$

Hence, $\dot{x_\delta}(t) = (A - BK)\, x_\delta(t)$ (44)

The transpose of (A-BK) takes the same form as the error dynamics in (43) and hence can be used to calculate the optimal observer gain for convergence. The observer design is validated by comparing its output against the output from the full order linear model of the column with 96 states for a step change of 5 kmole/hr in the reflux flow rate applied at t= 5 hours during simulation; results have been presented in Figure 88. This scheme assumes that the attacker can only launch attack which has been mathematically modelled in Section 3.1; thus the observer is not targeted by the adversary.



(a) Distillate purity (mole fraction) vs time

(b) Bottoms impurity (mole fraction) vs time

(c) Liquid level in reflux drum (m) vs time

(d) Liquid level in reboiler (m) vs time

(e) Column pressure (atm) vs time

Figure 88. Validation of Luenberger Observer design.

4.4 Validation of Luenberger observer based attack detection technique:

In this section the ability of Luenberger observer in detecting attacks presented in Section 3.2.3 will be evaluated. The observer is connected to the ICS as in Figure 53. The fundamental concept is to use the residual between the sensor measurements fed to the controller and the observer output to detect attacks. The observer is used to estimate the output based on the measured sensor and actuator data. The detection scheme relies on discrepancy between the sensor measurements and predicted output during attack scenario which can be computed as a residual as in (45) for attack detection.

During normal operation the residual will remain small. During attack the residual value will increase. A threshold can be applied on the residual to detect attacks. The output residuals for the DC are defined as below.

$$r_{Lc}\ (t) = \left|\hat{L}_C(t) - \tilde{L}_C(t)\right| \tag{45(a)}$$

$$r_{Lr}\ (t) = \left|L_r(t) - \tilde{L}_r(t)\right| \tag{45(b)}$$

$$r_P\ (t) = \left|\hat{P}(t) - \tilde{P}(t)\right| \tag{45(c)}$$

$$r_D\ (t) = |\hat{x}_D(t) - \tilde{x}_D(t)| \tag{45(d)}$$

$$r_B\ (t) = |\hat{x}_B(t) - \tilde{x}_B(t)| \tag{45(e)}$$

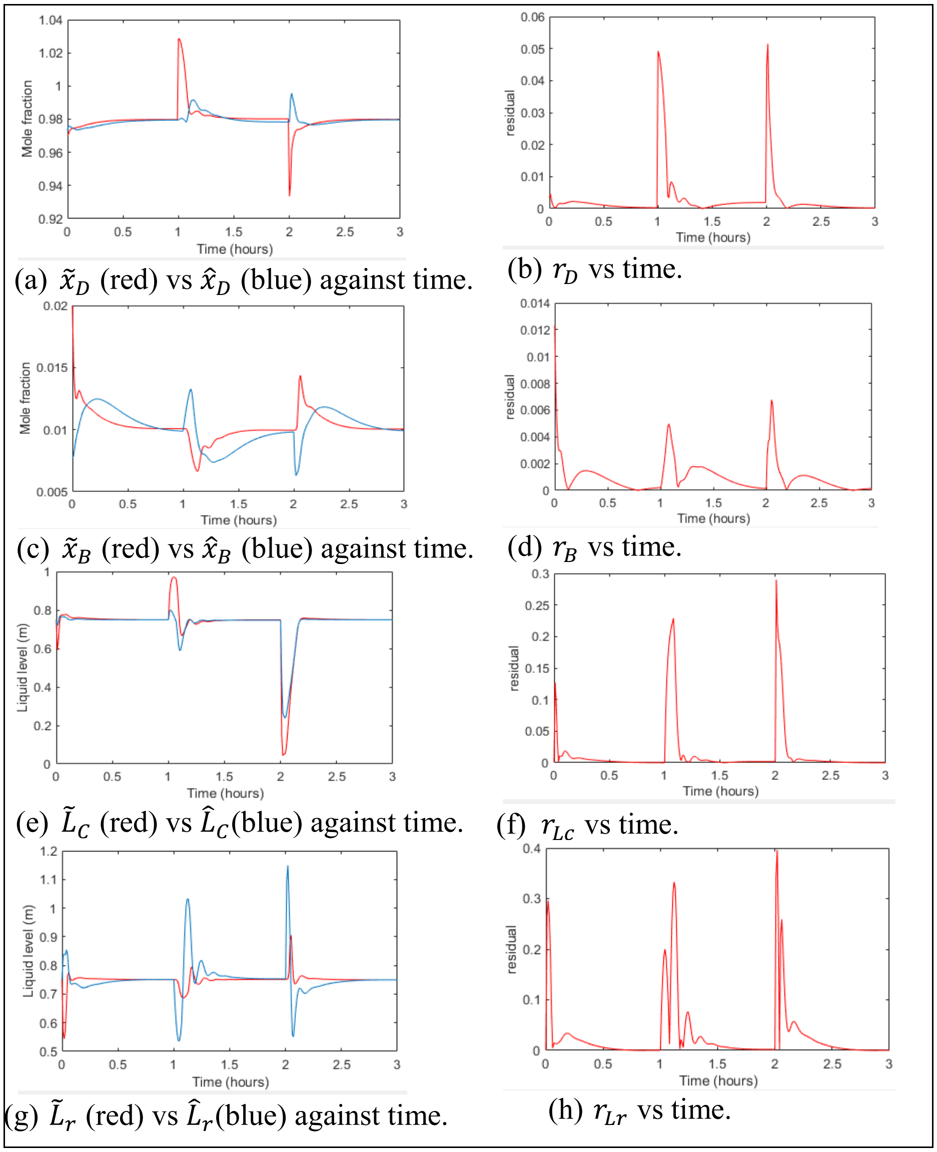
where $\tilde{L}_C$ and $\tilde{L}_r$, $\tilde{P}$, $\tilde{x}_D$, and $\tilde{x}_B$ are the liquid levels (m) in the reflux drum and the column sump, the column pressure at the top (atm), the distillate purity and bottoms impurity at the controller side, respectively. Any parameter $\hat{z}$ implies value of parameter z estimated by the observer. Table 11 presents the threshold set on the residuals for attack detection on different sensor measurements. Controller sampling time is set to 0.01 hour. The results are obtained for the same attacks mentioned in section 3.2.3. From the results it will be seen that there lies discrepancy between the observer and the sensor at start of simulation which is because the observer initial state is different to that of the model hence it takes time for observer to converge to true sensor measurements (i.e. error dynamics to converge). The convergence time observed is around 0.8 hours during simulation. Hence attack is injected between t = 1-2 hours.

Table 11. Threshold values for the residuals.

Residual	Threshold value	Residual	Threshold value
r_{Lc} (m)	0.2	r_B	.005
r_{Lr} (m)	0.2	r_P (atm)	5.0
r_D	0.01		

- **Scaling attack:**

Distillate purity sensor attack: The attack is injected setting λ_s as 0.05 and result is presented in Figure 89. Although there are multiple residuals which detected the attack but r_D detected in quickest time in .02 hour. Since the observer fails to estimate all un-affected outputs correctly during attack hence it is not possible to isolate the attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

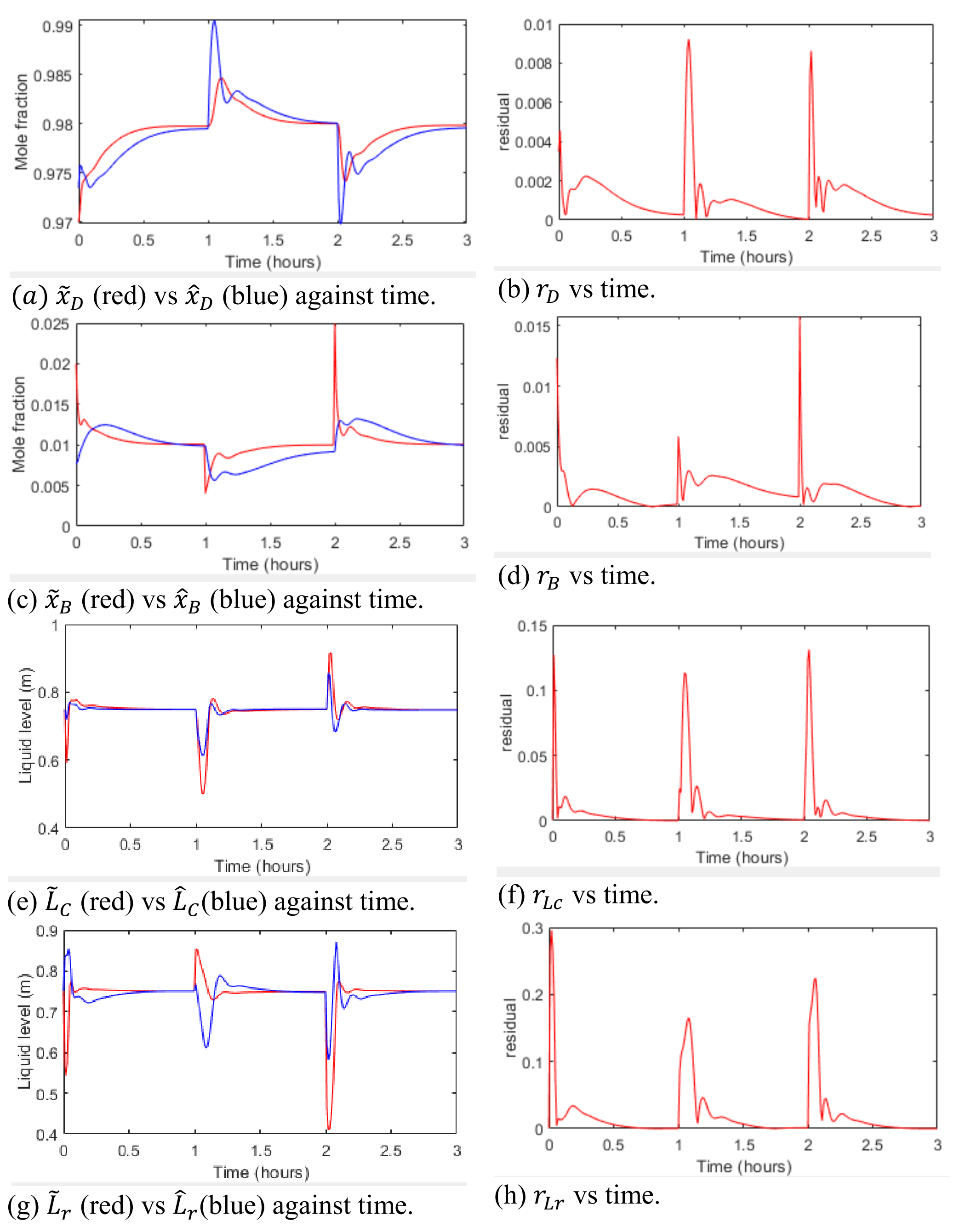
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time.

(h) r_{Lr} vs time.

Figure 89. Detection of a scaling attack on distillate purity sensor using observer.

Bottoms impurity sensor attack: The value of λ_s is set to -0.6 during this attack. The attack is detected using value of r_B in 0.03 hours. Since the remaining residuals do not cross the thresholds, hence this method allows for attack diagnosis for this attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

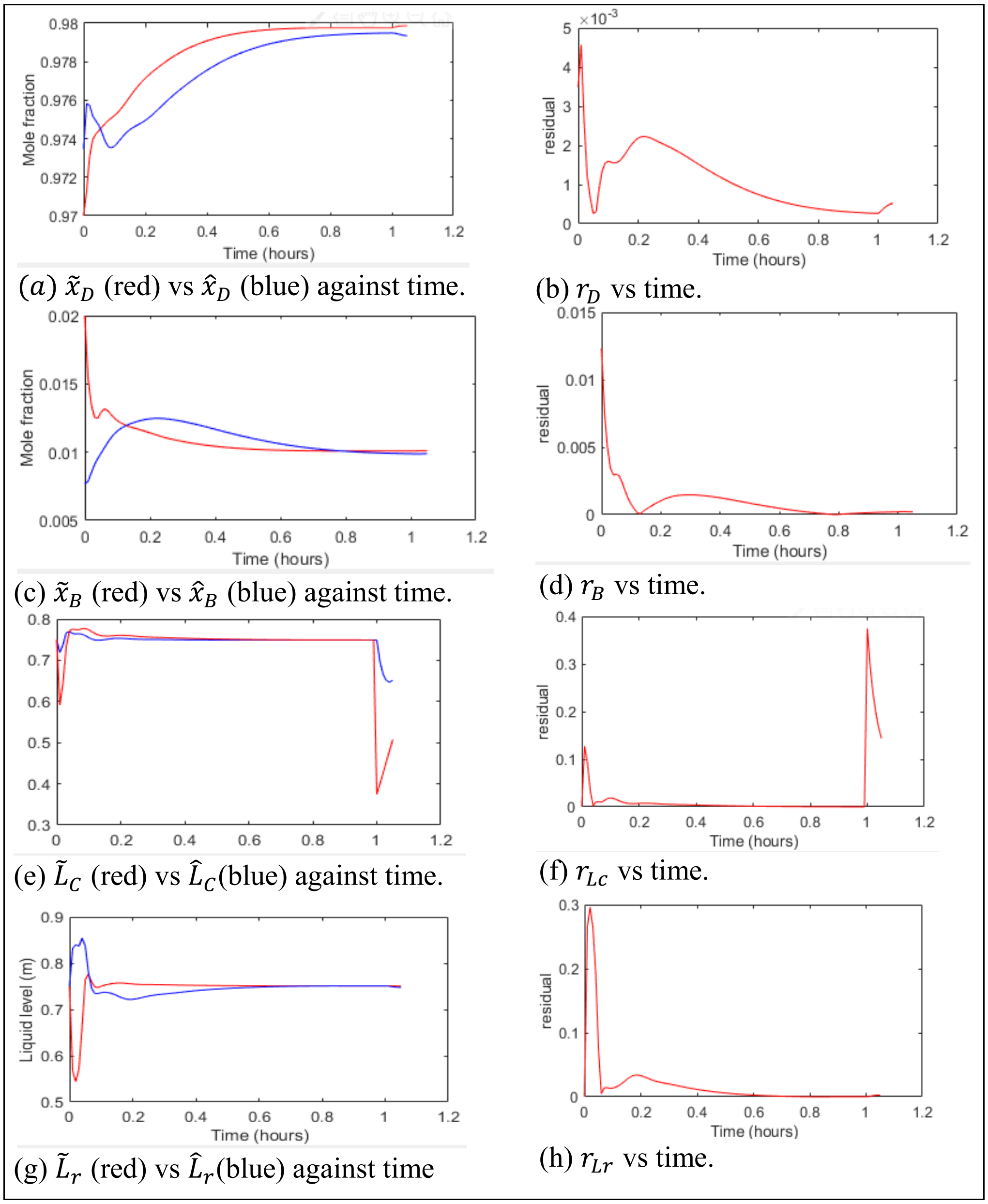
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time.

(h) r_{Lr} vs time.

Figure 90. Obserbed based attack detection for the bottoms impurity sensor attack.

Reflux drum liquid level sensor attack: The attack is injected setting λ_s as -0.5. From the results presented in Figure 91 the attack is detected by r_D in 0.01 hour. The detector is able to detect attack before the emergency shutdown due to the flooding in reflux drum caused by the attack. Since only r_D violates the threshold during attack hence this method can be used for tracing the location of attack in ICS.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

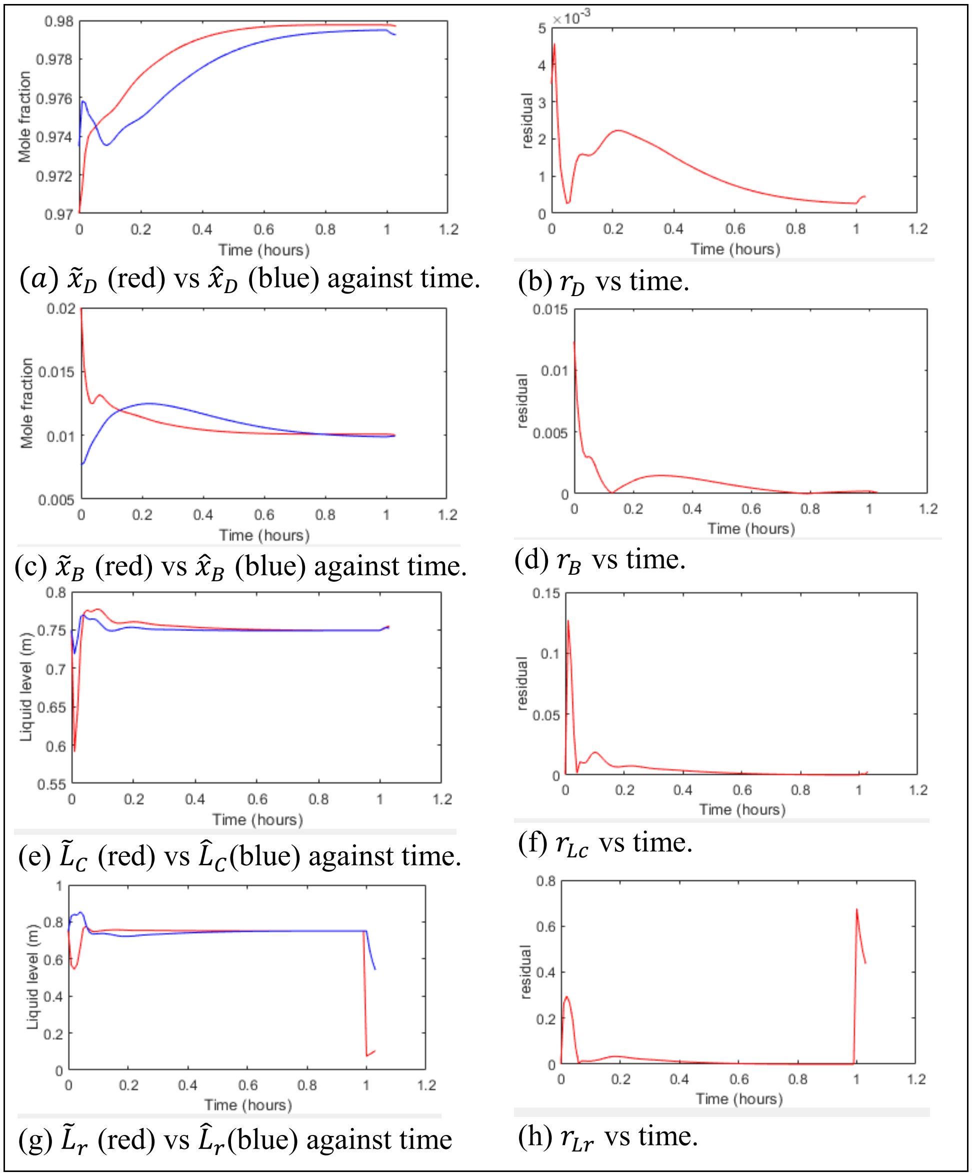
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 91. Observer based detection for attack on the reflux drum liquid level sensor.

Reboiler drum liquid level sensor attack: The value of λ_s is set to -0.9. This attack triggers the emergency shutdown due to the overflow of liquid in the reboiler drum as can be seen from Figure 92. However, the observer is able to detect the attack in 0.01 hour. Hence with the presence of detector an emergency shutdown can be prevented. The attack is detected by r_B only; hence this method provides attack isolation.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

(f) r_{Lc} vs time.

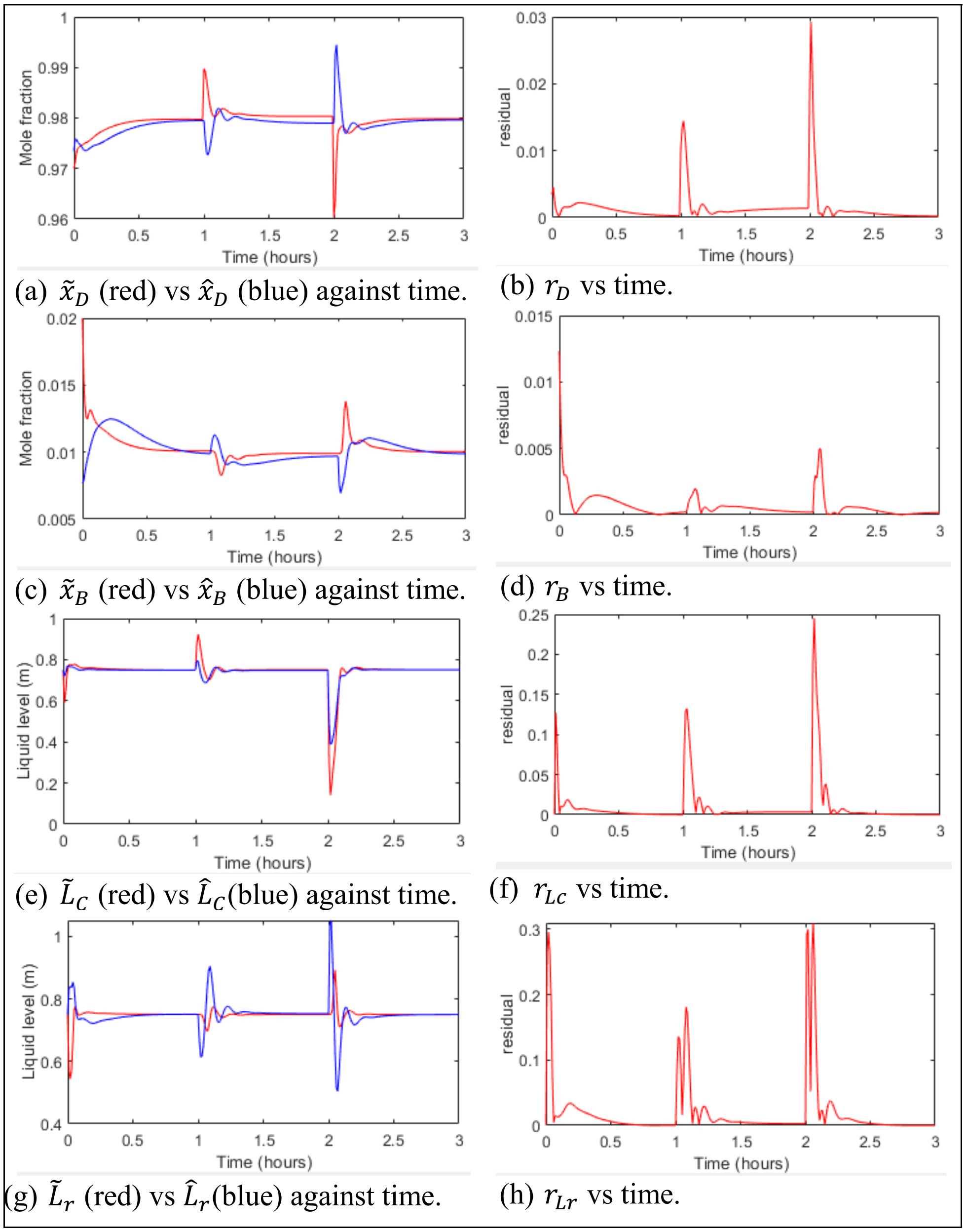
(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 92. Scaling attack detection on the liquid level in reboiler drum using observer.

- **Ramp attack:**

Distillate purity sensor attack: The attack is simulated by setting λ_r as 0.01 similar to section 3.2.3. Results for the attack is presented below in Figure 93. The attack is detected using $r_D$in 0.02 hour. The observer can be used for finding the location of attack in the ICS since only r_D crosses its threshold during attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

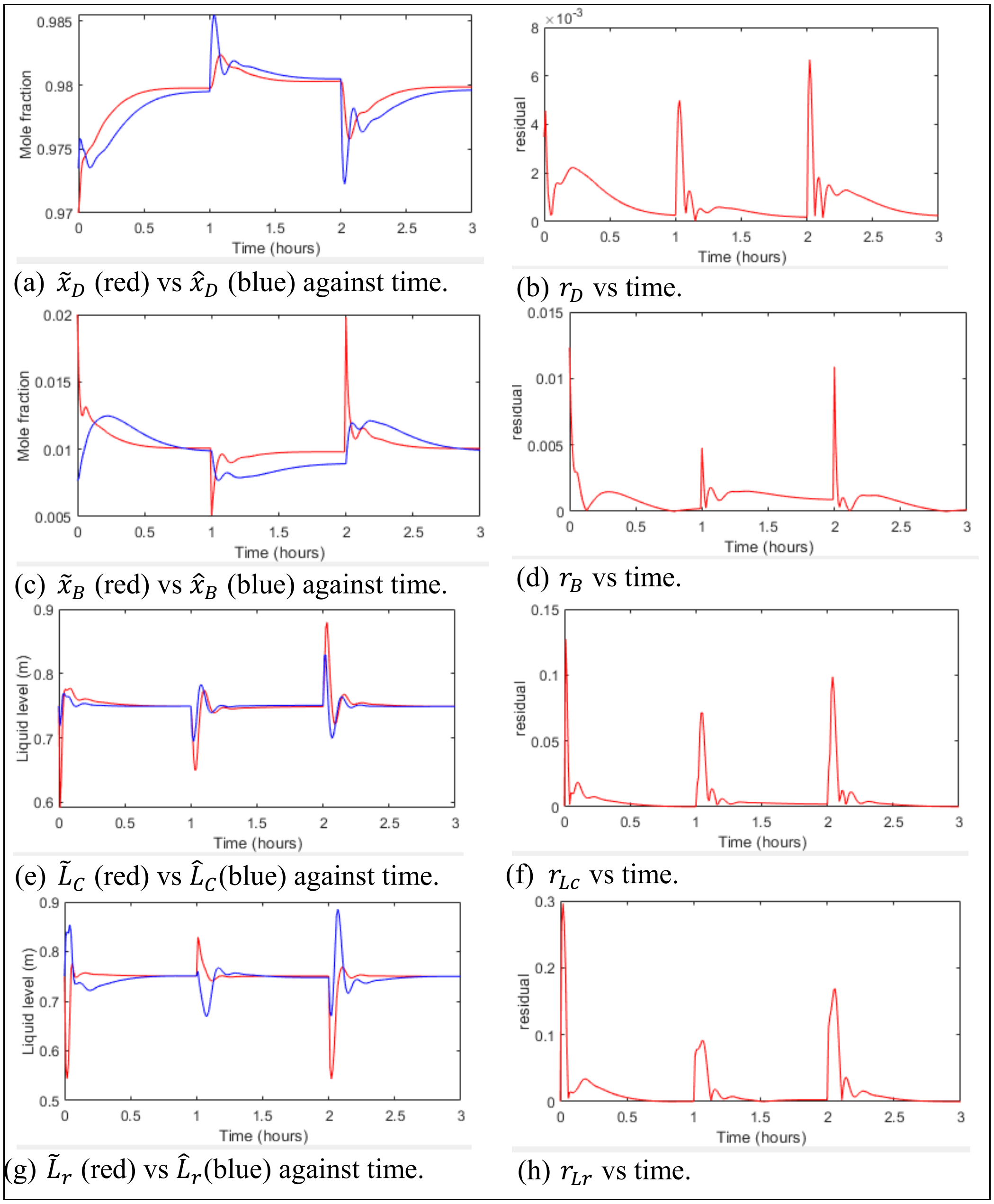
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time.

(h) r_{Lr} vs time.

Figure 93. Detection of a ramp attack on the distillate purity sensor using ramp attack.

Bottoms impurity sensor attack: The value of λ_r is selected as -0.005 for this attack and the result is presented in Figure 94. The attack is detected by r_B in 0.03 hour. Since the observer estimates the remaining outputs which are not attacked correctly hence this method allows for attack isolation.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

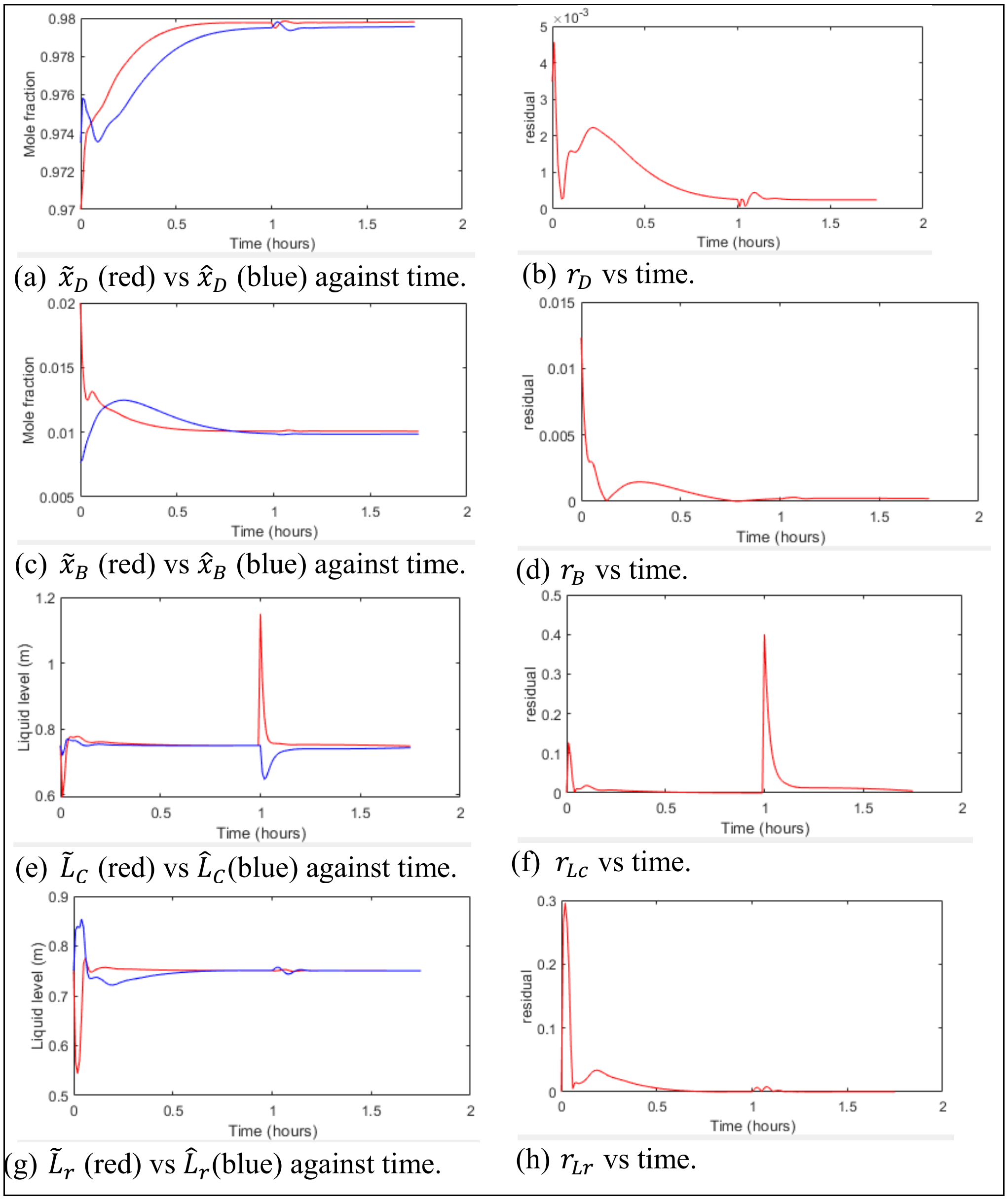
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time.

(h) r_{Lr} vs time.

Figure 94. Detection of a ramp attack on the bottoms impurity sensor using observer.

Reflux drum level measurement sensor attack: The first attack is simulated setting λ_s = 0.40. As said in section 3.2.3, this attack reduces the reflux level below the minimum threshold causing an emergency shutdown. The results in Figure 95 shows that r_{Lc} detects the attack before the shutdown in 0.02 hour. Since remaining residuals doesn't detect the attack hence this method can be used to trace the location of attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

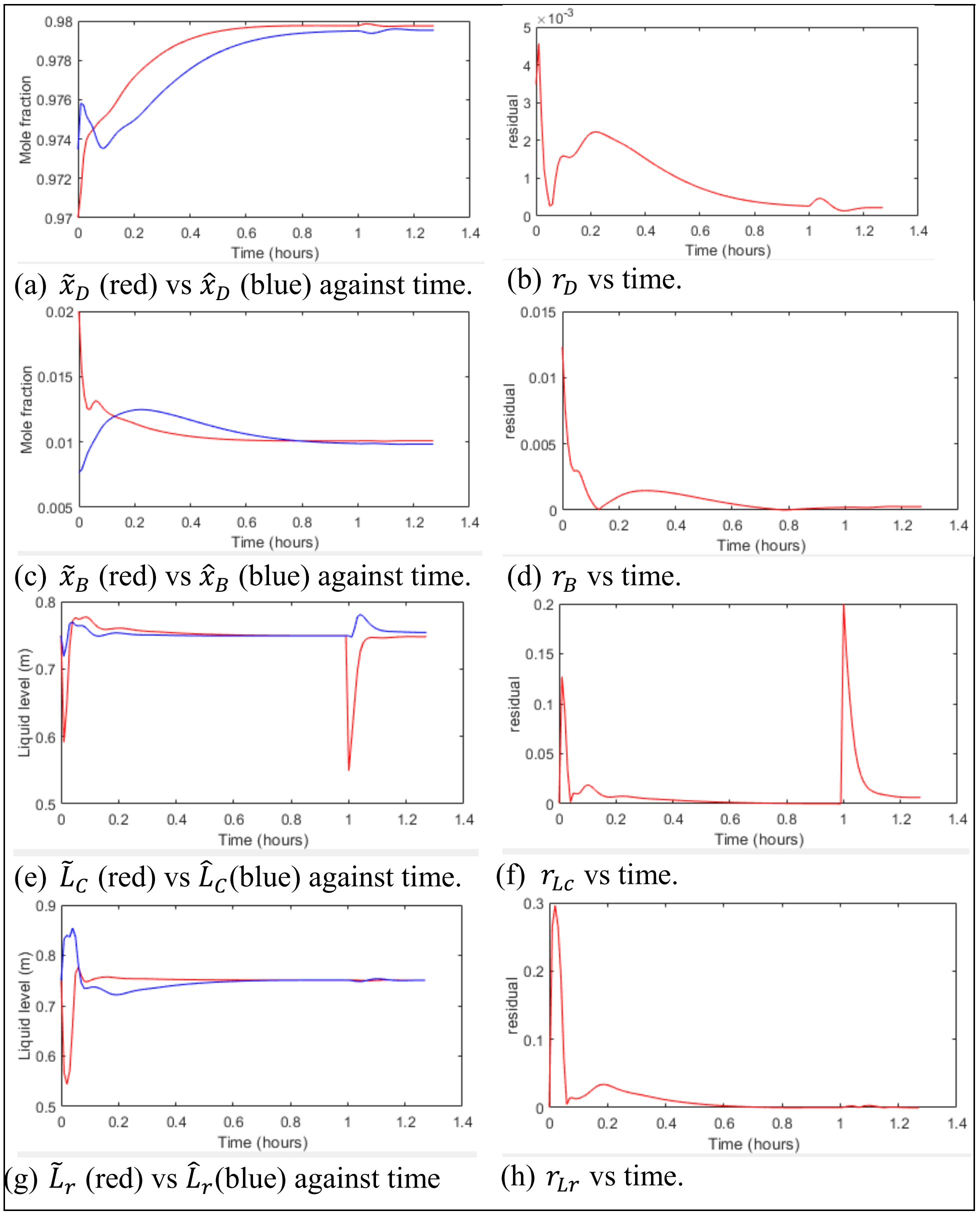
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time.

(h) r_{Lr} vs time.

Figure 95. Detection of a ramp attack on the reflux level for λ_s = 0.40.

During the second case the attack is simulated setting λ_s = -0.20. As mentioned in Section 3.2.3.a, this attack causes the liquid level in the reflux drum to overflow causing emergency shutdown. From the results in Figure 96 it is seen that r_{Lc} detects the attack in 0.04 hour long before the plant is shut down. This method can be used for attack isolation since the observer estimates the outputs which are not attacked correctly.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

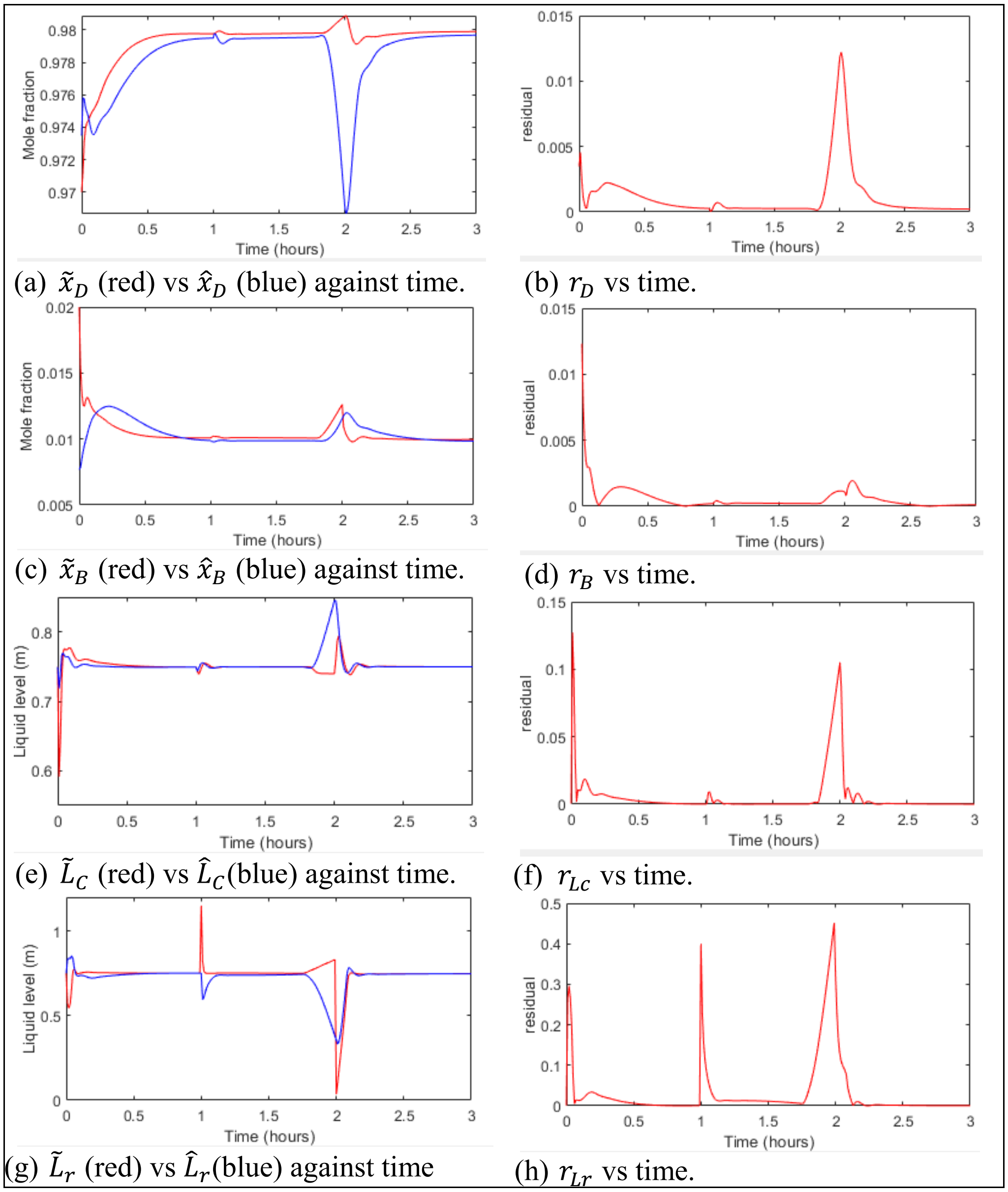
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 96. Detection of a ramp attack on the reflux level for λ_s= - 0.20.

Reboiler drum level sensor attack: The value λ_s is set to 0.40 for the first scenario. As said in section 3.2.3, this attacks drains the reboiler drum beyond minimum threshold causing an emergency shutdown. The results in Figure 97 shows that r_{Lr} detects the attack before the shutdown in 0.02 hour. The method allows for attack isolation as no other threshold except r_{Lr} detects the attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

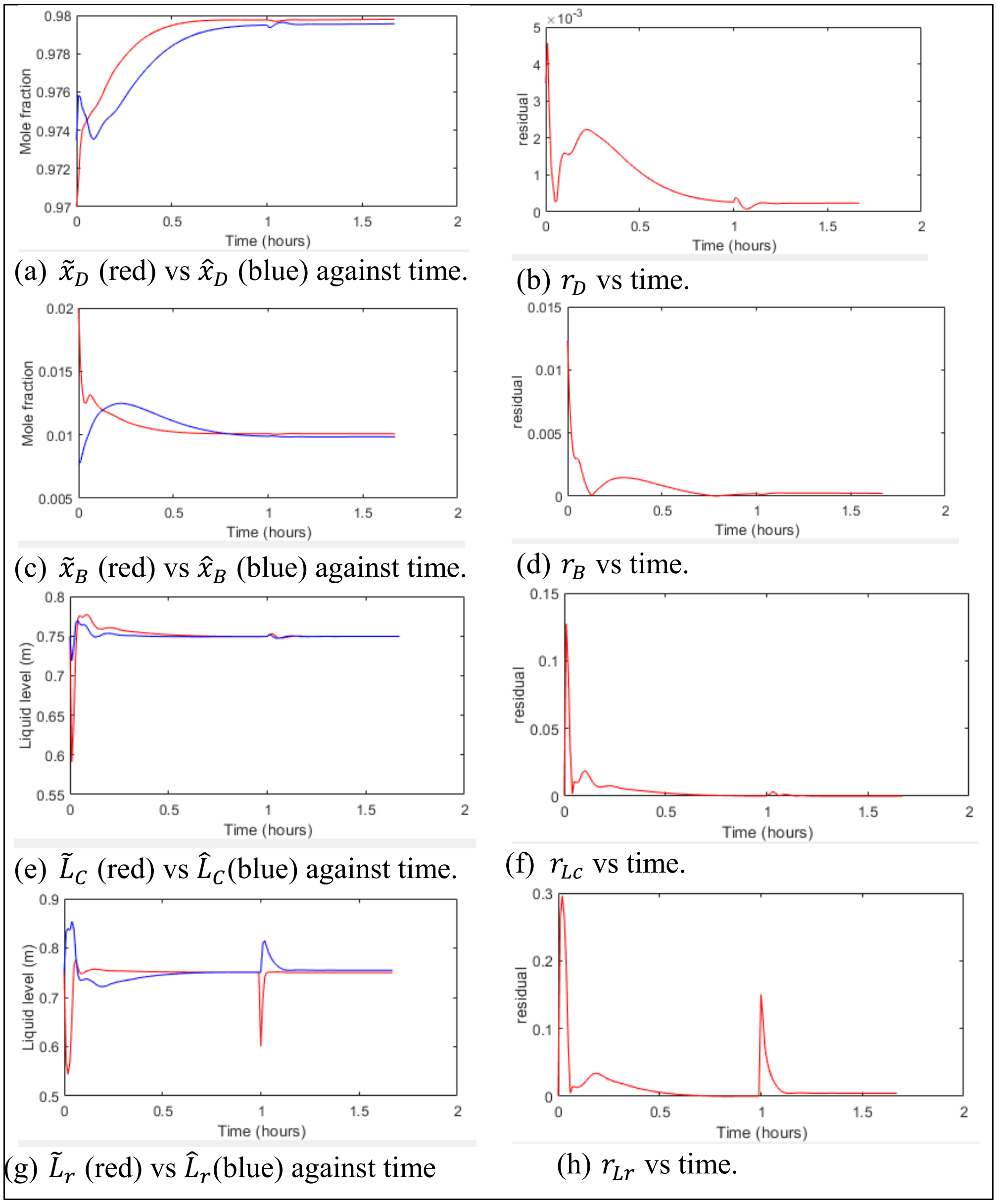
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 97. Detection of a ramp attack on the reboiler level for λ_s = 0.40

In case of this attack λ_s is set to -0.20. As mentioned in Section 3.2.3.a, this attack causes the liquid level in the reboiler drum to overflow causing emergency shutdown. The results in Figure 98 shows that r_{Lc} detects the attack in 0.01 hour long before the emergency shutdown. This method can be used for attack isolation since the observer estimates the outputs which are not attacked correctly.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

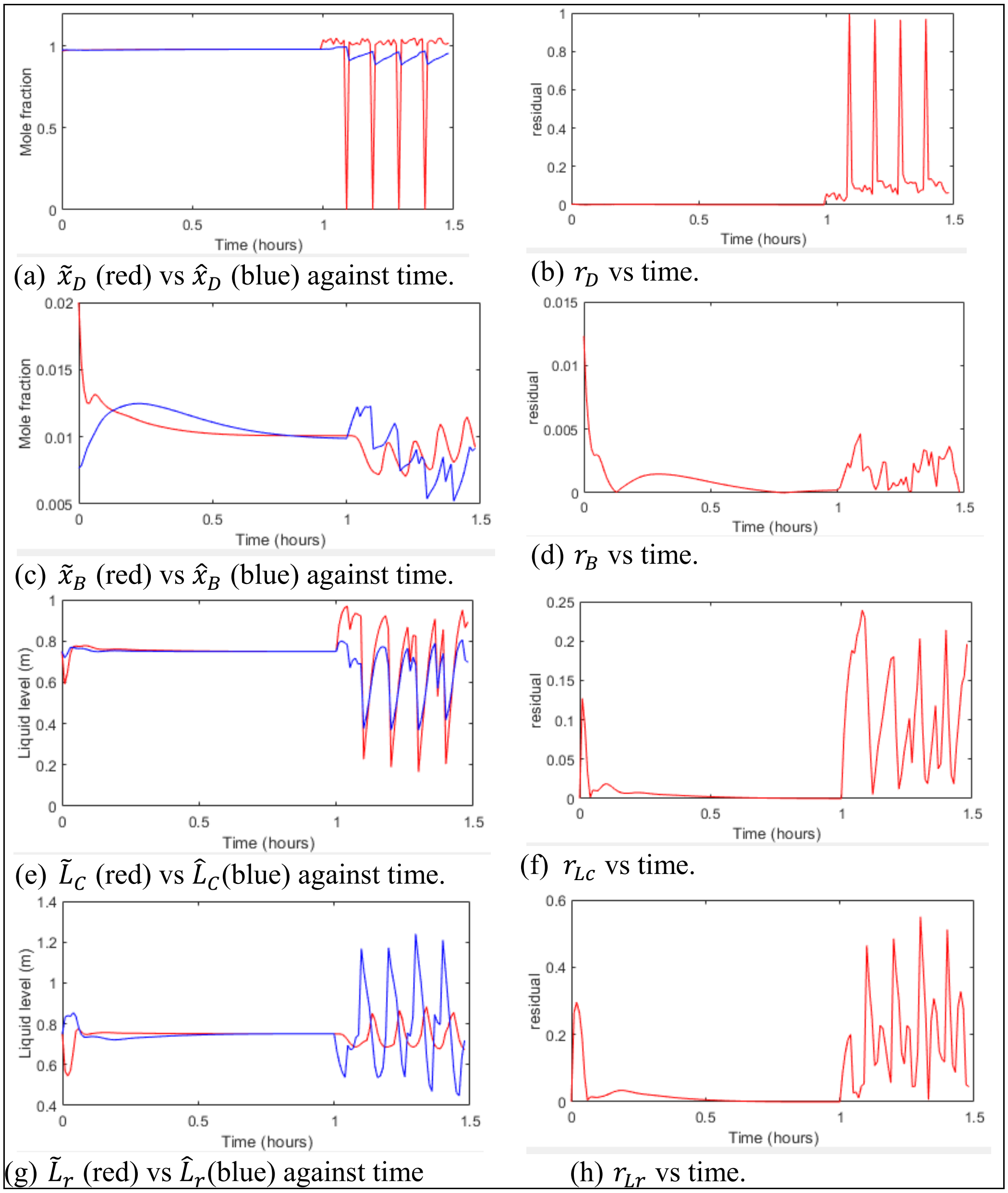
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 98. Detection of a ramp attack on the reflux level for λ_s= - 0.20.

- **FDI attack:** The attacks presented in Section 3.2.3.a are used for detection. *Distillate purity sensor attack:* As said previously the attack causes emergency shutdown since the distillate purity drops below 90%. The attack is detected by r_{Lc} in the fastest time in 0.64 hour way before the emergency shutdown. The method provides no attack diagnosis for this attack as multiple residuals trigger attack warning.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

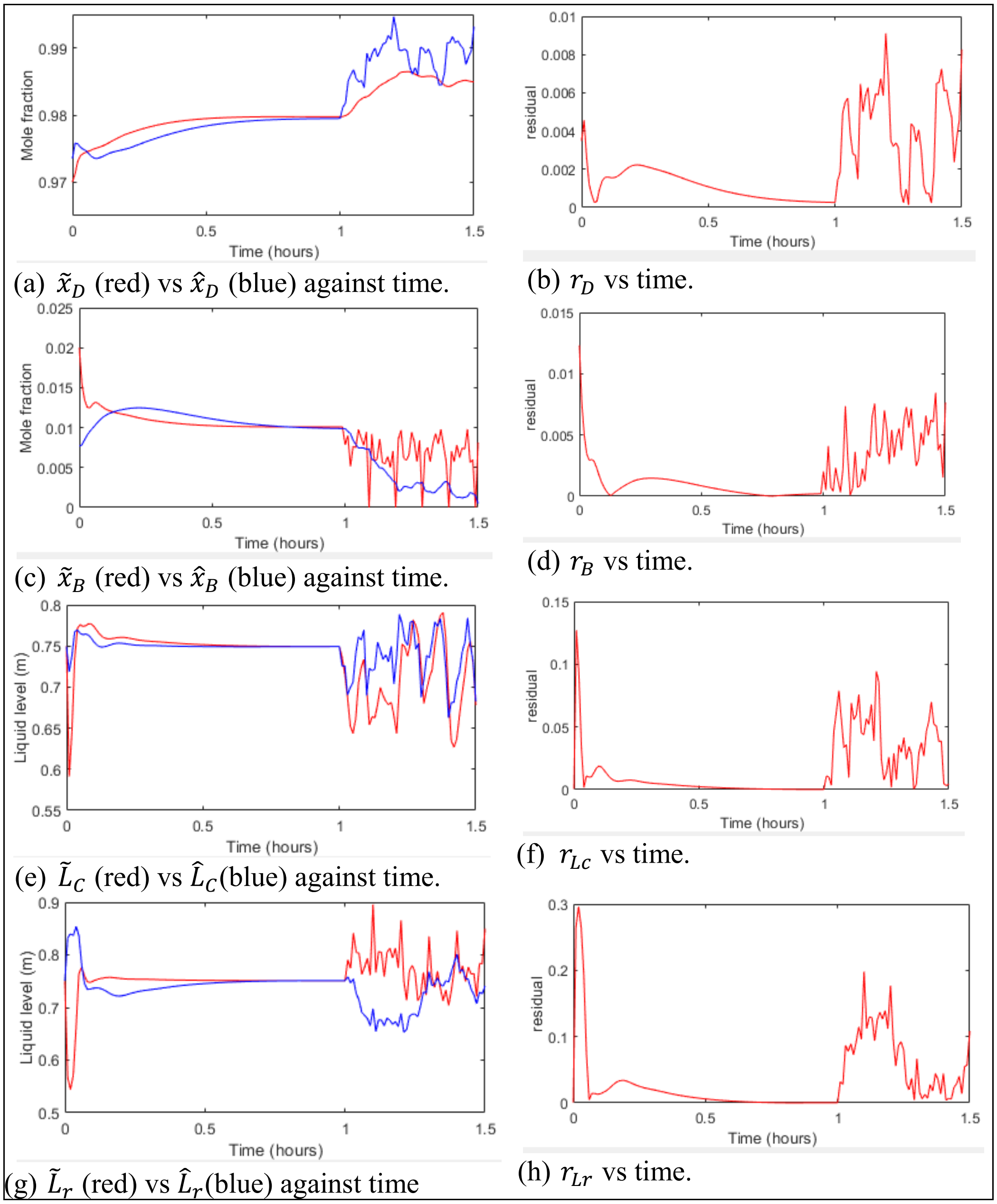
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 99. Detection of a FDI on the distillate purity sensor using observer.

Bottoms impurity sensor attack: The results of the attack presented in Figure 100 shows that r_B detects the attack in .53 hour. For this attack the observer is able to identify the location of the attack since all the remaining residuals other than r_B remains lower than their set threshold during attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

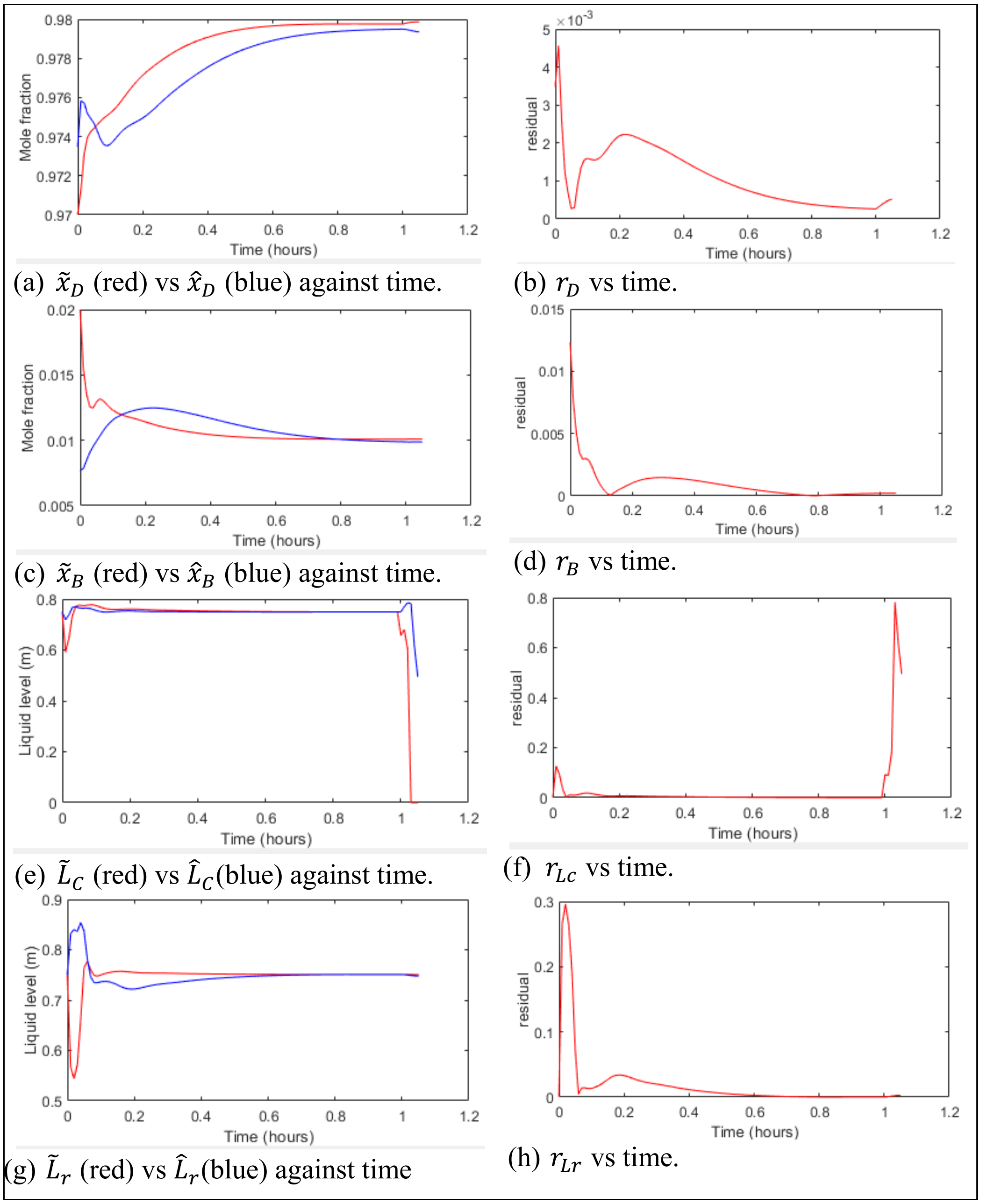
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 100. Detection of a FDI attack on the bottoms impurity sensor using observer.

Reflux drum level measurement sensor attack: The attack as seen from the results in Figure 101 is detected by r_{Lc}in 0.04 hour. As can be seen the observer estimates the outputs which are not being attacked correctly which can be used to trace the location of the attack in the ICS.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

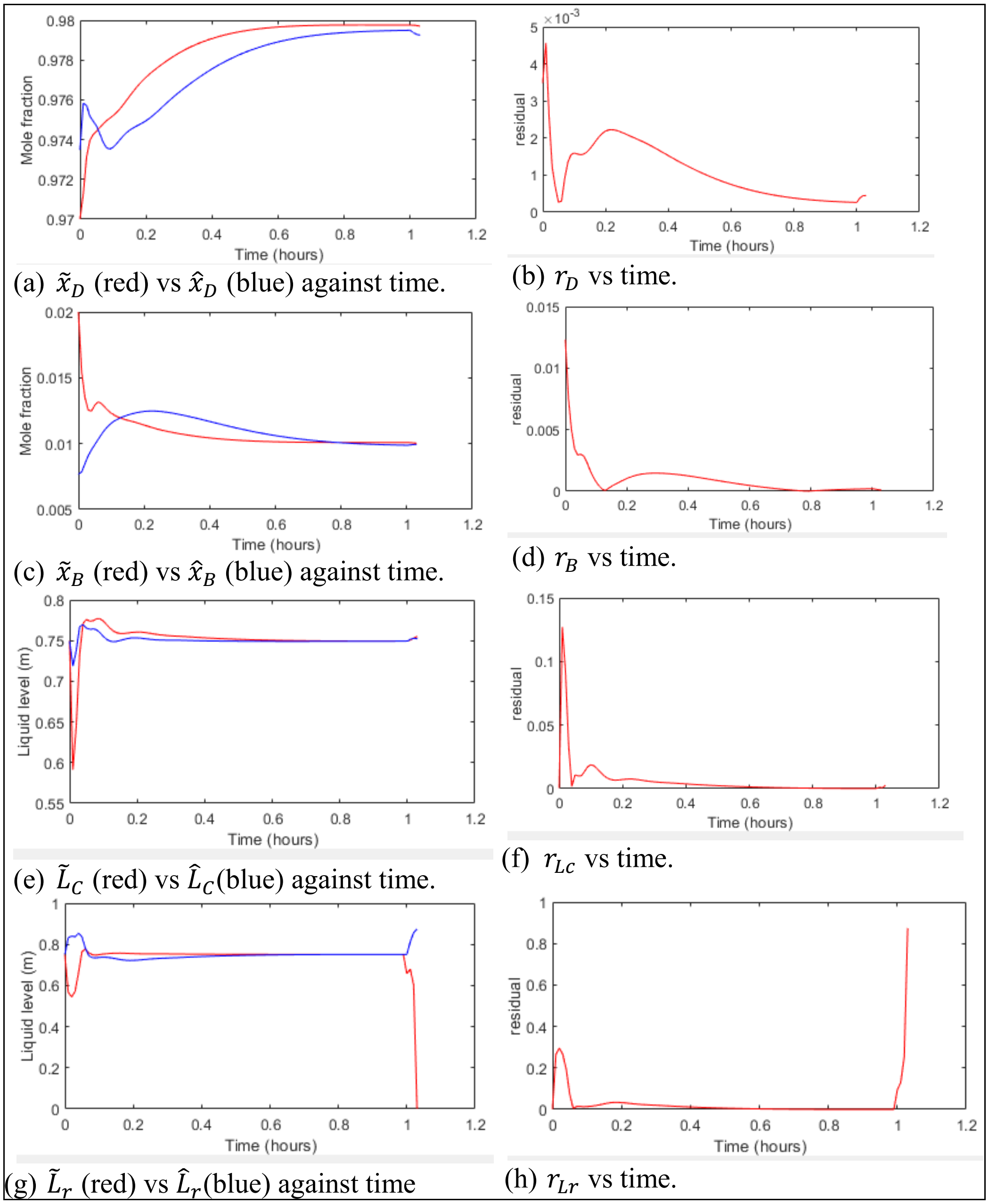
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 101. Detection of a FDI attack on the reflux level using observer.

Reboiler drum level measurement sensor attack: From the attack results presented in Figure 102 the attack is detected r_{Lr} in .02 hour. As the remaining residuals are unaffected by the attack hence the observer can be used to isolate the attack location in the ICS.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

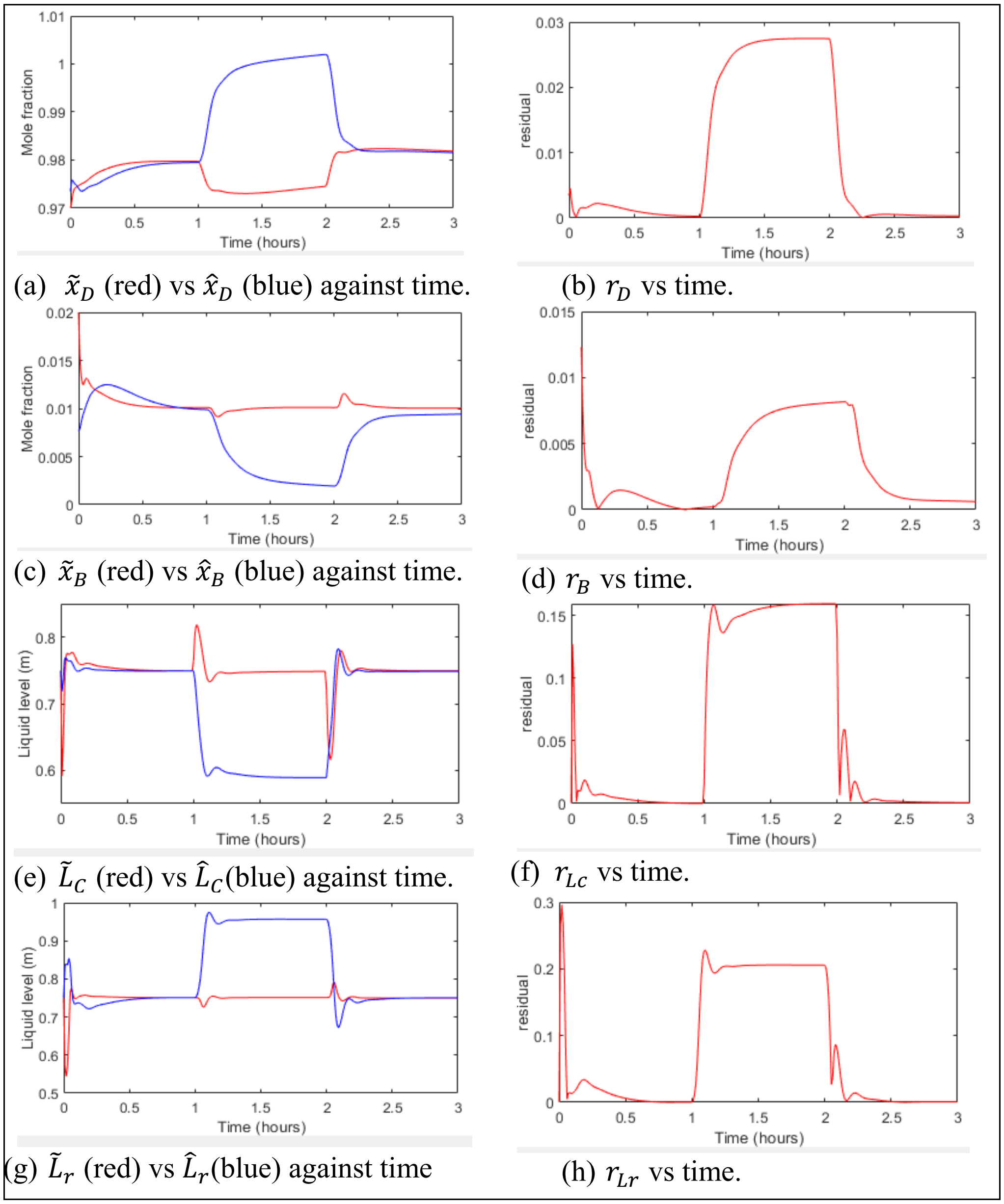
(h) r_{Lr} vs time.

Figure 102. Detection of a FDI attack on the reflux level using observer.

4.4.2 Actuator attack

- **Data injection attack:**

Attack on reflux flow rate: Using a_r as 0.6 the attack is detected by r_D in 0.1 hour as shown in Figure 103. This method can't detect the location of the attack as the observer fails to estimate multiple outputs correctly during this attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

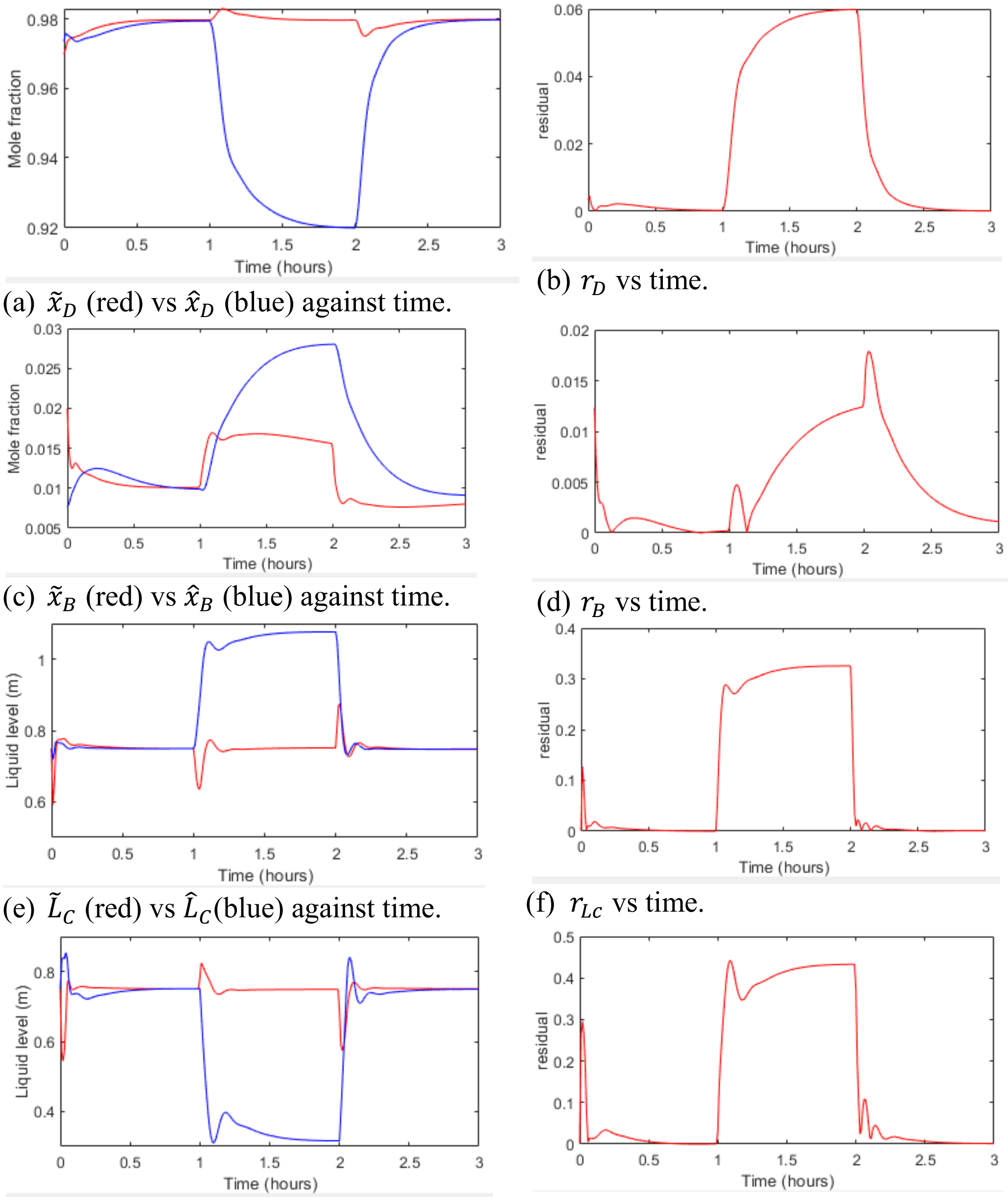
(f) r_{Lc} vs time.

(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 103. Detection of a data injection attack on the reflux flow rate using observer.

Attack on reboiler duty cycle: The value of a_r is set to 0.5 during this attack. Based on the results presented in Figure 104 the residual r_{Lc} detects the attack in the fastest time in 0.04 hour. As can be seen all the residuals cross their attack detection thresholds during this attack. Hence this method can't provide attack isolation for this attack.



(a) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(b) r_D vs time.

(c) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(d) r_B vs time.

(e) $\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

(f) r_{Lc} vs time.

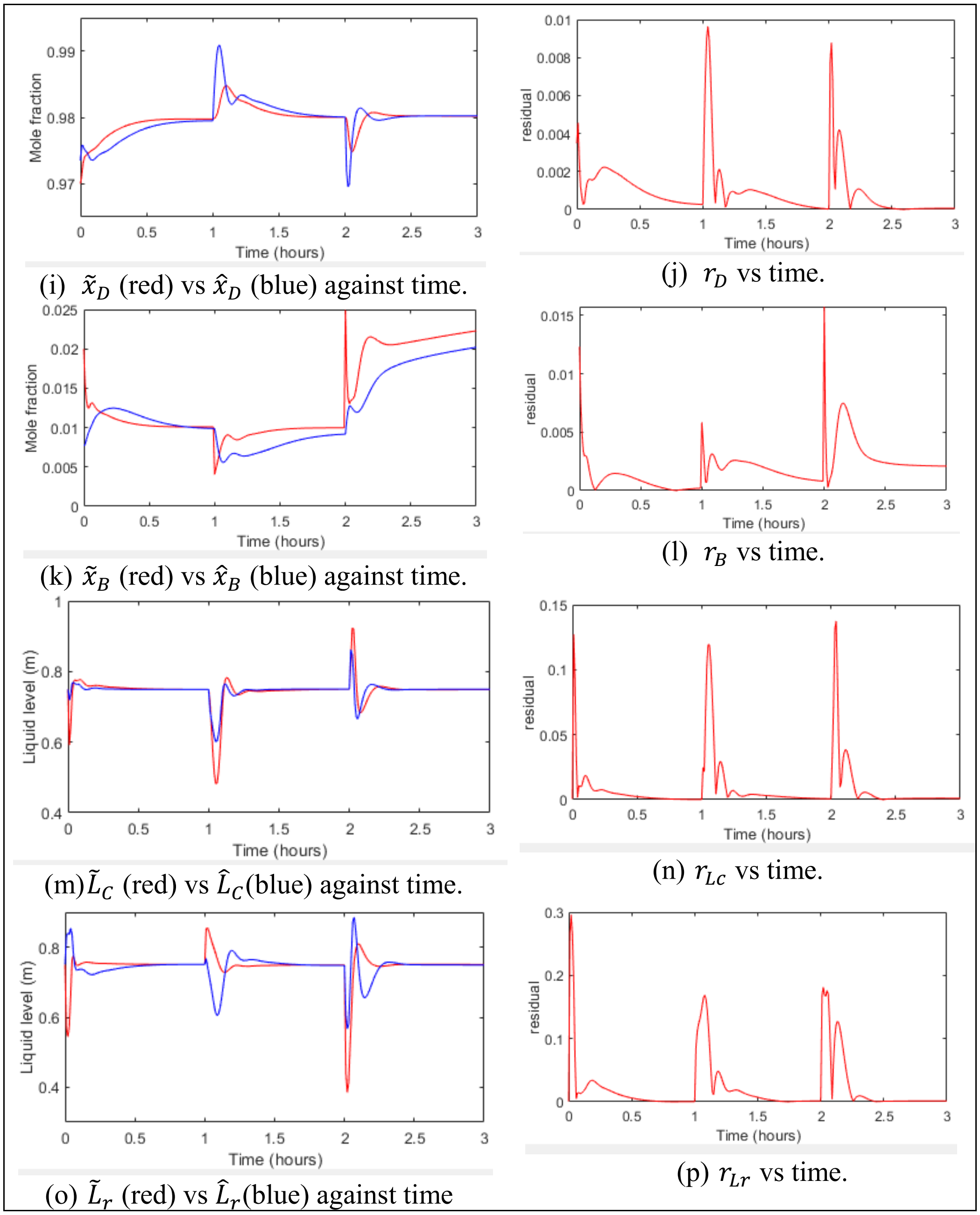
(g) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(h) r_{Lr} vs time.

Figure 104. Detection of a data injection attack on reboiler duty cycle using observer.

4.4.3. Chaining attack

The detection result for the chaining attack presented in section 3.2.3.c is given below in Figure 105.



(i) $\tilde{x}_D$ (red) vs $\hat{x}_D$ (blue) against time.

(j) r_D vs time.

(k) $\tilde{x}_B$ (red) vs $\hat{x}_B$ (blue) against time.

(l) r_B vs time.

(m)$\tilde{L}_C$ (red) vs $\hat{L}_C$(blue) against time.

(n) r_{Lc} vs time.

(o) $\tilde{L}_r$ (red) vs $\hat{L}_r$(blue) against time

(p) r_{Lr} vs time.

Figure 105. Detection of chaining attack using an observer.

From the results it can be seen that r_B detects the attack in 0.02 hour. However all the remaining residuals don’t exceed attack detection threshold thus implying the presence of attack in the bottom impurity control loop. However it fails to detect the controller attack. Hence injecting a controller attack in this manner makes it stealthy and non-detectable as has been illustrated using the results in Figure 105.

Table 12. Summary of attack detection results uisng Luenberger Observer

Attack type	Attack name	Attacked parameter	Detection time (hour)	Attack detection	Attack isolation
Sensor attacks	Scaling attack	Distillate purity	0.03	✓	✗
		Bottoms impurity	0.02	✓	✓
		Liquid level in reflux drum	0.01	✓	✓
		Liquid level in reboiler drum	0.01	✓	✓
	Ramp attack	Distillate purity	0.01	✓	✓
		Bottoms impurity	0.03	✓	✓
		Liquid level in reflux drum (case 1)	0.02	✓	✓
		Liquid level in reflux drum (case 2)	0.04	✓	✓
		Liquid level in reboiler drum (case 1)	0.02	✓	✓
		Liquid level in reboiler drum (case 2)	0.01	✓	✓
	FDI attack	Distillate purity	0.64	✓	✗
		Bottoms impurity	0.53	✓	✓
		Liquid level in reflux drum	0.04	✓	✓
		Liquid level in reboiler drum	0.02	✓	✓
Actuator	Data	Reflux flow rate	0.10	✓	✗

attack	injection attack	Reboiler duty cycle	0.50	✓	✗
Chaining attack	Controller and scaling attack	Bottoms impurity sensor and controller	0.02	✓	✗

5. CONCLUSION AND FUTURE WORK

To conclude, the objectives set for thesis and how they have been achieved will be discussed. Besides that, this section will outline the future works that can be done in the field of this thesis. As mentioned previously, there are four objectives of this thesis. The remainder of this section will outline the objectives and elaborate on how they have been accomplished.

- **Mathematical modelling of cyber-attacks:** As part of this thesis, the attacks commonly found in ICS have been modeled. It has been assumed that an ICS for a CPS can be characterized by its number of feedback control loops each containing sensors, actuators and controllers. Hence the adversary can target the communication channel between one or more of these devices to launch an attack. Thus the modeled attacks have been categorized under namely sensor attacks, actuator attacks, sensor and actuator attacks and controller attacks. The attack models in the existing literature have been modeled for linear plant models. However, considering that real world plants exhibit nonlinear dynamics, the attacks have been modeled for nonlinear CPS.
- **Mathematical modelling of a DC:** This task has been aimed to facilitate in the analysis of the effect of cyber-attacks on the DC. As part of the thesis, a binary continuous DC has been modeled. A binary continuous DC is designed to process one specific feed stream and separate it into two product streams which are i. Distillate, and ii. Bottoms product during its entire life cycle. The distillate contains the more volatile lighter fractions in the feed stream and is collected from the top of column. On the other hand, the heavier fractions in the feed stream are collected as bottoms product from the

bottom of the column. The main objective is to maintain the quality of the two products under their desired specification which necessitates the introduction of control system to the plant. There have been two different DC models proposed as part of this thesis.

The first column has been designed based on data from [45] and [46]. At first the data has been used to design the column in Aspen Plus. In order to simulate the dynamics of the DC, the model then has been transported to Aspen Plus Dynamics. The steady state calculations from Aspen Plus has been used to derive the dynamic model of the DC using MESH equations. However, the dynamics of some of plant parameters have been found to be dissimilar to the dynamics observed in Aspen Plus Dynamics. Hence those dynamics have been added to obtain a grey box model of the DC. After that the control system has been designed for the DC. There are two feedback control loops present in the control system of the DC one of which is used to control the distillate purity and the other one for controlling the bottoms impurity as per the quality specification.

The second model has been designed using data presented in Section 2.3. At first the data has been used to design the column in Aspen Plus which is then transported to Aspen Plus Dynamics. Then, the feedback control system for the DC has been designed in Simulink. Finally, the Aspen Plus Dynamics based plant model and the ICS implemented in Simulink have been integrated to create a hybrid model of the CPS. The control system of the plant contains five feedback control loops which are used to control the column pressure at the top (atm), the liquid level (m) in reflux drum and reboiler drum, and the product quality of the distillate and bottoms product.

- **Analyze the effect of cyber-attacks on control performance of the DC:** As a part of this task, the attacks have been injected to the two models of the DC and their effect on the column performance has been observed.

At first the attacks have been injected to sensors, actuators and controllers of the grey box model of the DC. Besides that, different types of attacks have been injected simultaneously in the form of chaining attack. The results of all different types of attacks on the DC has been presented in Section 3.2.2. It has been observed that the adversary can upset the column performance by targeting any aspect of communication (i.e. between sensor and controller, and controller and actuator) in control system. Since two different completely decoupled control loops have been used to control the product qualities independently hence every attack only upsets the performance of the loop it has been injected in. From the results, it has been observed that the adversary can launch attacks mainly to upset the product qualities of the DC to introduce financial loss (since the value of the product is dependent upon its quality). Besides that, it has been observed that the proposed control scheme provides a degree of robustness by providing attack recovery upon removal of attack.

Similarly, the different attacks modeled have been injected to the various control loops of the hybrid model of the DC as sensor attacks, actuator attacks, controller attacks, and chaining attacks and the obtained results have been presented in section 3.2.3. Since each output of the plant is controlled by an input independently, hence it has been observed that injecting an attack in any control loop only upsets that particular control loop; although the attack introduces unprecedented transients in some other control loops (depending on type of attack) however the controller is able to suppress the transients and bring the output back to the desired set-point. Other than that, the control scheme is able to bring the outputs back to their desired set-points upon removal of attack thus providing attack recovery. The motivation behind attacks can be to introduce financial loss by violating product qualities or threaten the safety of the column operation by either increasing the pressure in the column or by causing flooding in either of the two liquid drums.

- **Design and validation of attack detection techniques for ICS by using the model of DC:** Finally, based on the results of the attacks on the two DCs, detection techniques have been designed for the modeled attacks. As part of this thesis, only model based detection techniques have been designed. The designed techniques mainly rely on estimating the outputs of DC using the knowledge of the sensor outputs to the controller side and controller outputs to detect the attacks. During the normal operation, the estimated outputs are expected to closely match the sensor measurements. However, if the integrity of any control loop is violated by attack there is expected to be discrepancy between the estimated output and the sensor output to the controller which is used for attack detection. This proposed detection scheme can only detect sensor and actuator attacks by integrating the detector across the controller as illustrated in Figure 53. In total three different detection techniques for the two different models of the DC are proposed.

The first two detection techniques have been implemented using EKF and UKF for the grey box model of the DC. EKF and UKF can be used to provide state estimation for nonlinear plants. The implementation details of EKF and UKF along with results for detection have been provided in Section 4.2 and 4.3, respectively. From the results it has been validated that both EKF and UKF are able to detect all the sensor and actuator attacks modeled as part of this thesis. However, the detection times using the two estimators for the attacks are different as can be seen from Table 9 which can be attributed to the difference in the implementation techniques for EKF and UKF, and process and measurement noise. Finally, it has been also seen that for certain attacks the estimators are able to isolate the attack by tracing the location of the attack in the ICS.

A Luenberger observer based detection technique has been proposed for detecting attacks on the Aspen Plus Dynamics based hybrid. The linear model required for the observer design has been acquired by linearizing the model around its closed-loop steady state operating point. It has been observed that the observer is able to detect sensor and actuator attacks. Besides that for some attacks, it is also able to pinpoint the location of attack in the ICS thus facilitating attack isolation.

- **Future work:** There is many scope of work in the area of designing cyber secure ICS for crude oil DC as there are no literatures available in this area. The majority of the literatures on attack modelling focuses on linear plant model. However as stated, in real world most CPS exhibit nonlinear dynamics. The thesis consider various attack models from existing literatures on ICS of CPS and modelled them for nonlinear systems. But there is still scope for modelling other attacks found in literatures e.g. Covert attack, Zero dynamics attack and so on for the class of nonlinear systems. The thesis limits the study for the case of a binary continuous DC. However as said, DCs can be categorized based on mode of operation and number of output product streams. The existing study in this thesis can be extended for other categories of DCs. This thesis primarily addresses designing techniques for attack detection. So far, techniques based on availability of the plant model has been considered. Motivations can be drawn from data driven algorithms for designing further detection techniques as certain plant dynamics can be challenging to model. The next step following detection is attack isolation in order to trace the location of the attack in the ICS for attack mitigation and survivability. Although the presented detection techniques offers attack isolation for certain attacks however it is not been included as an inherent feature in the proposed techniques which leaves room for investment of research effort in this area. Finally, the topic of attack mitigation and survivability can be addressed in order to put in place a definitive combat action to prevent plant disruption by drawing motivation from the field of fault tolerate control.

REFERENCES

[1] F. Pasqualetti, F. Dorfler, and F. Bullo, "Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems," *IEEE Control Systems Magazine,* vol. 35, no. 1, pp. 110-127, 2015.

[2] F. Pasqualetti, F. Dorfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE transactions on automatic control,* vol. 58, no. 11, pp. 2715-2729, 2013.

[3] Y. Nakahira, and Y. Mo, "Dynamic state estimation in the presence of compromised sensory data," in *IEEE 54th Annual Conference on IEEE*, 2015.

[4] Y. Li, L. Shi, P. Cheng, J. Chen, and D. E. Quevedo, "Jamming attacks on remote state estimation in cyber-physical systems: A game-theoretic approach," *IEEE Transactions on Automatic Control,* vol. 60, no. 10, pp. 2831 - 2836, 2015.

[5] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using Kalman filter," *IEEE Transactions on Control of Networked Systems,* vol. 1, no. 4, pp. 370 - 379, Dec. 2014.

[6] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica,* vol. 51, pp. 135-148, 2015.

[7] Z.A. Biron, P. Pisu, and B. HomChaudhuri, "Observer design based cyber

security for cyber physical systems," in *Proceedings of the 10th Annual Cyber and Information Security Research Conference*, 2015.

[8] C. Z. Bai, F. Pasqualetti, and V. Gupta, "Data-injection attacks in stochastic control systems: Detectability and performance tradeoffs," *Automatica,* vol. 82, pp. 251 - 260, 2017.

[9] I. Shames, F. Farokhi and T. H. Summers, "Security analysis of cyber- physical systems using H2 norm," *IET Control Theory & Applications,* vol. 11, no. 11, pp. 1749 - 1755, 2017.

[10] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit," *IEEE Control Systems,* vol. 35, no. 1, pp. 93 - 105, 2015.

[11] A. Hoehn, and P. Zhang, "Detection of covert attacks and zero dynamics attacks in cyber-physical systems," in *American Control Conference*, 2016.

[12] M. L. Corradini, and A. Cristofaro, "Robust detection and reconstruc- tion of state and sensor attacks for cyber-physical systems using sliding modes," *IET Control Theory & Applications,* vol. 11, no. 11, pp. 1756-1766, 2017.

[13] L. Aggoune, Y. Chetouani, and T. Raissi, "Fault detection in the distillation column process using Kullback Leibler divergence," *ISA Transactions,* vol. 63, pp. 394-600, 2016.

[14] S. Lawal, and J. Zhang, "Fault monitoring and fault tolerant control in distillation columns," in *21st International Conference on Methods and Models in Automation and Robotics*, 2016.

[15] S. Lawal, and J. Zhang, "Actuator fault monitoring and fault tolerant control in distillation columns," *International Journal of Automation and Computing,* vol.

14, no. 1, pp. 80 - 92, 2017.

[16] P. Akhlagi, A. Kashanipour, and K. Salashoor, "Intelligent fault diagnosis of distillation column system based on PCA and multiple ANFIS," in *IEEE Conference on Cybernetics and Intelligent Systems*, 2008.

[17] J. Yang, D. Li, and Q. Gao, "Application of improved DPCA to distillation column process monitoring," in *IEEE International Conference on Mechatronics and Automation*, 2016.

[18] A. Daher, Y. Chetouani, G. Hoblos, and M. Khalil, "Modified fuzzy c-means combined with neural network based fault diagnosis approach for a distillation column," in *IEEE International Multidisciplinary Conference on Engineering Technology*, 2016.

[19] J. Yang, D. Lil, and Q. Gao, "Weir flow and liquid height on sieve and valve trays," *IEEE Chemical Engineering Journal,* vol. 73, pp. 191 - 204, 1999.

[20] N. Jahanshahi, N. Meskin, F. Abdollahi and W. M. Haddad, "An adaptive sliding mode observer for linear systems under malicious attack," in *2016 IEEE International Conference on Systems, Man, and Cybernetics*, 2016.

[21] X. Luo, Q. Yao, X. Wang, and X. Guan, "Observer-based cyber attack detection and isolation in smart grids," *Internation Journal of Electrical Power & Energy Systems,* vol. 101, pp. 127 - 138, 2018.

[22] X. Luo, X. Wang, X. Pan, and X. Guan, "Detection and isolation of false data injection attack for smart grids via unknown input observers," *IET Generation, Transmission & Distribution,* vol. 13, no. 8, pp. 1277 - 1286, 2019.

[23] A. F. Taha, J. Qi, I. Wang, and J. H. panchal, "Dynamic State Estimation under Cyber Attacks: A Comparative Study of Kalman Filters and Observers," 2015.

[24] M. Lv, W. Yu, Y. Kv, J. Cao, and W. Huang, "An integral sliding mode observer for CPS cyber security attack detection," *An Interdisciplinary Journal of Nonlinear Science,* vol. 29, no. 4, 2019.

[25] W. Yuqin, "Detection and Characterization of Actuator Attacks Using Kalman Filter Estimation," *M. S. Thesis, Marquette University,* 2009.

[26] A. Meng, H. Wang, S. Aziz, J. Peng, and H. Jiang, "Kalman filtering based interval state estimation for attack detection," *Energy Procedia,* vol. 157, pp. 6589 - 6594, 2019.

[27] N. Zivkovic, and A. Saric, "Detection of false data injection attacks using unscented kalman filter," *Journal of Mordern Power Systems and Clean Energy,* vol. 6, no. 5, pp. 847 - 859, 2018.

[28] H. Wanga, A. Menga, Y.Liua, X. Fu and G. Caoa, "Unscented kalman filter based interval state estimation of cyber physical energy systems for detection of dynamic attack," *Energy,* vol. 188, 2019.

[29] J. Qi, A. F. Taha, and J. Wang, "Comparing Kalman Filters and Observers for Power System Dynamic State Estimation With Model Uncertainty and Malicious Cyber Attacks," *IEEE Access,* vol. 6, pp. 77155 - 77168, 2018.

[30] D. Kundur, X. Feng, S. Mashayekh, S. Liu, T. Zourntos, K.L.B. Purry, "Towards modelling the impact of cyber attacKs on a smart grid," *International Journal of Security and Networks,* vol. 6, no. 1, pp. 2 - 13, 2011.

[31] Y. Wadhawan, A. AlMajali, and C. Neuman, "A Comprehensive Analysis of Smart Grid Systems against Cyber-Physical Attacks," *Electronics,* vol. 7, no. 249, 2018.

[32] T. Meraz, S. Sharmin, and A. mahmud, "Studying the impacts of cyber-attack on

smart grid," in *2nd International Conference on Electrical Information and Communication Technologies (EICT)*, 2015.

[33] T. Zhang, Y. Wang, X. Liang, Z. Zhuang, and W. Xu, "Cyber attacks in cyber-physical power systems: A case study with GPRS-based SCADA systems," in *29th Chinese Control And Decision Conference (CCDC)*, 2017.

[34] "Assessing the Impact of Cybersecurity Attacks on," *Energies,* vol. 12, no. 725, 2019.

[35] A.AlDairi, and L. Tawalbeh, "Cyber Security Attacks on Smart Cities and Associated Mobile Technologies," *Procedia Computer Science,* vol. 109, pp. 1086 - 1091, 2017.

[36] S. Adepu, and A. Mathur, "An Investigation into the Response of a Water Treatment System to Cyber Attacks," in *2016 IEEE 17th International Symposium on High Assurance Systems Engineering (HASE)*, 2016.

[37] D. T. Ramotsoela, G. P. Hancke, and A. M. Abu-Mahfouz, "Attack detection in water distribution systems using machine learning," *Human-centric Computing and Information Sciences,* vol. 9, no. 13, 2019.

[38] A. A. Abokifa, K. Haddad, C. Lo and P. Biswas, "Real-Time Identification of Cyber-Physical Attacks on Water Distribution Systems via Machine Learning–Based Anomaly Detection Techniques," *Journal of Water Resources Planning and Management,* vol. 145, no. 1, 2019.

[39] M. Housh, and Z. Ohar, "Model-based approach for cyber-physical attack detection in water distribution systems," *Water Research,* vol. 139, no. 1, pp. 132-143, 2018.

[40] M. N. Kurt, O. Ogundijo, C. Li, and X. Wang, "Online Cyber-Attack Detection

in Smart Grid: A Reinforcement Learning Approach," *IEEE Transactions On Smart Grid,* vol. 10, no. 5, pp. 5174 - 5185, 2019.

[41] M. Ozay ; I. Esnaola ; F. T. Y. Vural, and S. R. Kulkarni , "Machine Learning Methods for Attack Detection in the Smart Grid," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 27, no. 8, 2015.

[42] X. Chen, L. Zhang, Y. Liu, and C. Tang, "Ensemble learning methods for power system cyber-attack detection," in *IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, 2018.

[43] H. Nishino, and H. Ishii, "Distributed Detection of Cyber Attacks and Faults for Power Systems," *IFAC Proceedings Volume,* vol. 47, no. 3, pp. 11932 -11937, 2014.

[44] K. C. Sou, H. Sandberg, and K. H. Johansson, "Detection and identification of data attacks in power system," in *American Control Conference* , 2012.

[45] V. T. Minh, and A. M. A. Rani, "Modeling and control of distillation column in a petroleum process," *Mathematical Problems in Engineering,* pp. 1-14, 2009.

[46] "Condensate processing plant project - project description document no. 82036-02BM-01," PetroVietnam Gas Company, 1999.

[47] M.S.Lopes, M. S. Lopes, R. M. Filho, M. R. W. Maciel, and L.C.Medina, "Extension of the TBP Curve of Petroleum Using the Correlation DESTMOL," *Procedia Engineering,* vol. 42, pp. 726 - 732, 2012.

[48] E. J. Hoffman, "Relations between true boiling point and ASTM distillation curves," *Chemical Engineering Science,* vol. 24, no. 1, pp. 113-117, 1969.

[49] E. E. Tarifa,E. Erdmann,D. Humana, and J. Martínez, "A New Method for Estimating the EFV Distillation Curve," *Petroleum Science and Technology,* vol.

27, no. 3, pp. 331 - 344, 2009.

[50] "e-Education Institute," PennState College of Earth and Mineral Sciences, [Online]. Available: https://www.e-education.psu.edu/fsc432/content/true-boiling-point-distillation-tbp.

[51] B. Alkhalili, A. Yahya, N. Abrahim, and B. Ganapathy, "Biodesulfurization of Sour Crude Oil," *Mordern Applied Science,* vol. 11, no. 9, p. 104.

[52] A. Demirbas, H. Alidrisi, and M. Balubaid, "API Gravity, Sulfur Content, and Desulfurization of Crude Oil," *Petroleum Science and Technology,* vol. 33, no. 1, pp. 93 - 101.

[53] A. Teixeira, D. Perez, H. Sandberg, and K. H. Johansson, "Attack models and scenarios for networked control systems," in *1st International Conference on High Confidence Networked Systems*, 2012.

[54] S. Sridhar, and M. Govindarasu, "Model-based attack detection and mitigation for automatic generation control," *IEEE Transcations on Smart Grid,* vol. 5, no. 2, pp. 580 - 592, March 2014.

[55] E. A. Wan, and R. V. D. Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium* , 2000.