Pontifícia Universidade Católica
do Rio de Janeiro

Francisco José Burok Teixeira Leite Strunck

Application of Siamesis Neural Network for fault detection in industrial processes in the production of polystyrene

Dissertação de Mestrado

Dissertation presented to the Programa de Pós-graduação em Engenharia Química, de Materiais e Processos Ambientais, do Departamento de Engenharia Química e de Materiais of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Química, de Materiais e Processos Ambientais.

Advisor : Profa. Dra. Amanda Lemette Teixeira Brandão
Co-advisor: Profa. Dra. Karla Tereza Figueiredo Leite

Rio de Janeiro
November 2023

Pontifícia Universidade Católica
do Rio de Janeiro

Francisco José Burok Teixeira Leite Strunck

Application of Siamesis Neural Network for fault detection in industrial processes in the production of polystyrene

Dissertation presented to the Programa de Pós-graduação em Engenharia Química, de Materiais e Processos Ambientais of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Química, de Materiais e Processos Ambientais. Approved by the Examination Committee.

Profa. Dra. Amanda Lemette Teixeira Brandão
Advisor
Departamento de Engenharia Química e de Materiais – PUC-Rio

Profa. Dra. Karla Tereza Figueiredo Leite
Co-advisor
Instituto de Matemática e Estatística – IME - UERJ

Dra. Marília Caroline Cavalcante de Sá
Universidade Federal do Rio de Janeiro – UFRJ

Dra. Andréa Pereira Parente
Universidade Federal do Rio de Janeiro – UFRJ

Rio de Janeiro, November 21st, 2023

Francisco José Burok Teixeira Leite Strunck
Graduated in Chemical Engineering from the Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) in 2020, currently pursuing a Master's degree at the same institution, focused on modeling, simulation, control, and automation of styrene polymerization, using Python and advanced Artificial Intelligence techniques. Interned in Chemical Engineering at Drew Marine. In scientific initiation projects, was a CNPq scholarship holder at LabSint (PUC-Rio), responsible for the synthesis of organic molecules, conducting research on 'Click Reactions' for potential Alzheimer's disease treatment.

Bibliographic data

Strunck, Francisco

Application of Siamesis Neural Network for fault detection in industrial processes in the production of polystyrene / Francisco José Burok Teixeira Leite Strunck; advisor: Amanda Lemette Teixeira Brandão; co-advisor: Karla Tereza Figueiredo Leite. – Rio de janeiro: PUC-Rio, Departamento de Engenharia Química e de Materiais, 2023.

v., 89 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Química e de Materiais.

Inclui bibliografia

1. Engenharia Química – Teses. 2. Polimerização;. 3. Estireno;. 4. Detecção de falhas;. 5. Redes Neurais Siamesas (SNNs);. 6. Indústria 4.0.. I. Lemette, Amanda. II. Leite, Karla. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Química e de Materiais. IV. Título.

CDD: 620.11

I express my deep gratitude to everyone involved who accompanied me on this journey, enriching the pursuit of knowledge with moments of genuine joy and collaboration.

Acknowledgments

To my family, especially my mother, Ana Maria, for the moments by my side and for every word of support and encouragement. I am equally grateful to my father, Gustavo, who, wherever he may be, continues to support me and rejoice in my achievements. And to my uncles Gisah and Márcio, Gilberto and Martha, and my cousin Joaquim Strunck, for being a supporter and friend at all times.

To my partner, Ananda Penczek, for her patience, deep understanding, and invaluable assistance. Her constant support during countless hours of study and her emotional encouragement during the most challenging times.

A special thanks to two incredible women. To Amanda Lemette, who was much more than a mentor, a friend, and a role model. To Karla Leite, whose guidance was crucial and decisive. I express my sincere gratitude to both for all the support and learning.

To all the professors who contributed in different ways to my development as a master.

To everyone at the Grupo de Aplicações Avançadas em Processos (GAAP), I express my gratitude for the valuable knowledge exchanges, significant contributions, collaboration in work, and moments of relaxation.

To all members of the Pontifícia Universidade Católica do Rio de Janeiro, especially the Department of Chemical and Materials Engineering, and also to the then graduate coordinator Rodrigo Souza, for the knowledge imparted, partnership, understanding, and availability.

To the National Agency of Petroleum, Natural Gas and Biofuels (ANP) for the scholarship that made the development of this research project possible.

To all those who, even though not directly involved, have always supported me on this journey, my sincere thanks. Especially, I thank Luiz Felipe Aquino for his encouragement and understanding during the final stage of this journey.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Strunck, Francisco; Lemette, Amanda; Leite, Karla. **Application of Siamesis Neural Network for fault detection in industrial processes in the production of polystyrene** . Rio de Janeiro, 2023. 89p. Dissertação de Mestrado – Departamento de Engenharia Química e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro.

Industrial processes face new challenges with the advancement of Industry 4.0 and the increasing demand for improvements in fault detection. Fault detection is based on various techniques of statistical methods and machine learning. Although effective, they have some disadvantages, such as process simplification, low capacity to deal with noise, low capacity to deal with complex nonlinear systems, high computational demand, and risk of overfitting. In response to these limitations, this work introduces an innovative approach on the polymerization field that employs siamese neural networks (SNNs) and long short-term memory (LSTM) cells for early detection of faults in styrene polymerization. The modeling of styrene polymerization in a CSTR reactor was carried out using the method of moments for mass and energy balance, and in this system, proportional-integral-derivative (PID) control was added to simulate a real process control situation in the context of an industrial process. From the model, it was possible to obtain thirteen simulations, of which five are non-fault processes and eight are processes with faults. These data were processed and used to train the siamese networks. With the ability to classify whether these input data are similar or dissimilar, it was possible to perform fault detection. The results found demonstrate a fault detection rate with an accuracy of up to 100%, demonstrating the capability of this model in detecting faults in complex, dynamic, and nonlinear chemical processes. This study represents a substantial advance in the field of fault detection and also offers valuable findings for future investigations and improvements in intelligent fault detection systems in the chemical industry.

Keywords

Polymerization; Styrene; Fault Detection. Siamese Neural Networks (SNNs); Industry 4.0.

Resumo

Strunck, Francisco; Lemette, Amanda; Leite, Karla. **Aplicação da Rede Neural Siamesa para detecção de falhas em processos industriais na produção de poliestireno** . Rio de Janeiro, 2023. 89p. Dissertação de Mestrado – Departamento de Engenharia Química e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro.

Os processos industriais enfrentam novos desafios com o avanço da Indústria 4.0 e a crescente demanda por melhorias na detecção de falhas. A detecção de falha fundamenta-se em diversas técnicas de métodos estatísticos e aprendizado de máquina. Embora sejam eficazes, possuem algumas desvantagens, tais como simplificação do processo, baixa capacidade em lidar com ruído, baixa capacidade em lidar com sistemas complexos não lineares, alta demanda computacional e risco de de $overfitting$. Em resposta a essas limitações, este trabalho apresenta uma abordagem inovadora na área da polimerização empregando redes neurais siamesas (SNNs) e células long short-term memory (LSTM) para a detecção precoce de falhas na polimerização de estireno. Foi realizado a modelagem da polimerização do estireno em reator CSTR utilizando o método dos momentos para o balanço de massa e energia e, neste sistema, foi adicionado controle proporcional-integral-derivativo (PID) para simular uma situação real de controle de processo no contexto de um processo industrial. A partir do modelo foi possível obter treze simulações, das quais cinco são processos sem falha e oito são processos com falhas. Esses dados foram tratados e serviram para treinar as redes siamesas. Com a capacidade de classificar se esses dados de entrada são semelhantes ou diferentes, foi possível realizar a detecção de falha. Os resultados encontrados demonstram uma taxa de detecção de falhas com uma acurácia de até 100%, demonstrando a capacidade desse modelo em detectar falhas em processos químicos complexos, dinâmicos e não-lineares. Este estudo representa um avanço significativo no campo da detecção de falhas, oferecendo oportunidades valiosas para futuras investigações e aprimoramentos em sistemas inteligentes de detecção de falhas na indústria química.

Palavras-chave

Polimerização; Estireno; Detecção de falhas; Redes Neurais Siamesas (SNNs); Indústria 4.0.

Table of contents

List of figures

List of tables

List of symbols

PCA – Principal Component Analysis

SNN – Siamese Neural Networks

CSTR – Continuous Stirred-Tank Reactor

PID – Proportional-Integral-Derivative Control

PS – Polystyrene

MoM – Method of Moments

PFD – Process Flow Diagram

ML – Machine Learning

ANN – Artificial Neural Networks

TPR – True Positive Rate

FPR – False Positive Rate

AUC – Area Under the Curve

SVM – Support Vector Machines

KPI – Key Performance Indicators

RMSE – Root Mean Square Error

, .

1
Introduction

Monitoring chemical processes plays a crucial role in decision-making across various industries, such as pharmaceuticals, petrochemicals, and food processing. The right tools and techniques ensure increased operational safety and efficiency. Industry 4.0, characterized by integrating cyber-physical systems and the Internet of Things (IoT), has provided access to enormous volumes of data. This has enabled the application of artificial intelligence techniques, such as machine learning, for failure detection in chemical processes [3].

Polystyrene is a synthetic polymer made from the monomer styrene, a liquid hydrocarbon commercially manufactured from petroleum. The production of styrene involves a series of complex processes that commence with the distillation of crude oil to produce naphtha. This naphtha is subjected to steam cracking to yield ethylene, propylene, and benzene. The benzene is alkylated with ethylene in the presence of a catalyst to produce ethylbenzene, which is then dehydrogenated to produce styrene. Over 90% of global styrene production employs iron oxide-catalyzed dehydrogenation of ethylbenzene [17].

Polystyrene is a significant contributor to the plastic industry. Its significance in various industrial sectors underscores its prominence in the global landscape, reflecting its essential role in modern manufacturing and commercial practices. More than 450 million tonnes of plastics were produced globally in 2022 and are set to double by 2045. Inert plastics such as polyolefins (polyethylene (PE) and polypropylene (PP)) and polystyrene (PS) comprise more than half [16] [55].

Polystyrene is extensively used in the food industry for containers and disposable cutlery. In the packaging industry, it is sought after for its cushioning properties, coupled with its durability and light weight, which propels the demand for polystyrene. The construction industry also values polystyrene for its excellent insulation properties. The variety of polystyrene forms - general-purpose polystyrene (GPPS), high-impact polystyrene (HIPS), and expandable polystyrene (EPS) - caters to a broad spectrum of industry needs [10] [17]. The growing variety of applications highlights the rising demand for polystyrene, which underscores the need for efficient and reliable production processes.

The synthesis of polystyrene is a dynamic non-linear system governed

by numerous operational conditions, including temperature, pressure, and reaction time. Traditional fault detection methods may be ineffectively handling this complexity, particularly due to the quasi-stationary state of the involved species and the potential large-scale impacts of small variable deviations, which could lead to substandard or even hazardous outcomes [9]. Hence, the intricate nature of this procedure poses challenges in synthesis control and fault detection, consequently emphasizing the demand for advanced monitoring and control techniques.

Utilizing Hotelling's T2 and Squared Prediction Error (SPE) in conjunction with PCA to identifies patterns for fault detection in industrial processes by transforming the original variables into principal components. Despite its advantages, such as reducing high operational costs and facilitating the modeling process, PCA has limitations, such as superficial modeling and an inability to diagnose specific types of faults. Other commonly used multivariate statistical analysis methods for fault detection include Partial Least Squares (PLS), Independent Component Analysis (ICA), Gaussian Mixture Model (GMM), Fisher Discriminant Analysis (FDA), and Qualitative Trend Analysis (QTA). These methods offer different advantages and disadvantages, characteristics, and restrictions in their application [2] [4] [61].

Due to industrial chemical processes' complex and non-linear nature, more advanced machine learning models are now being explored for fault detection. Among these are Neural Networks (NN) [62], Deep Neural Networks (DNN) [61], K-Nearest Neighbors (KNN) [62], Echo State Networks (ESN) [63], Convolutional Neural Networks (CNN) [64], and Support Vector Machine (SVM) models [65]. Deep neural networks are gaining attention due to their ability to handle complex, dynamic, and non-linear processes, which allow them to identify intricate faults [1]. However, these models also come with their challenges, including high computational demands, the risk of overfitting, and a lack of consistently precise results.

As a response to these challenges, this work uses Siamese Neural Networks (SNNs), often referred to as 'twin' neural networks, as a solution. SNNs consist of two identical networks; each one accepts an individual input and is connected to a top layer that computes the similarity between the inputs. SNNs offer a unique advantage by being designed to distinguish between two inputs effectively. This renders them exceptionally proficient in tasks aimed at discerning similarities or disparities. Their capability to discriminate patterns, even within complex datasets, sets SNNs ahead of

other models. Explicitly designed for tasks involving pattern recognition, this quality is invaluable in various applications, such as image recognition, signature verification, and, as shown in our research, identifying faults in polymerization processes [5].

This research introduces a novel approach to early fault detection in industrial processes, particularly in the polymer synthesis process. We have employed SNNs, using their unique ability to recognize patterns, with the aim of distinguishing between standard and abnormal operational conditions. We have modeled the process of styrene polymerization utilizing the method of moments for mass and energy balances. Normal and fault data were identified and used to train the 'twin' neural networks. This work is notably the first to use SNNs for fault detection in a styrene polymerization process. Integrating these new datasets will contribute to the study of polymerization and pave the way for more advanced and accurate models in chemical engineering and machine learning.

2
Objectives

This work aims to develop an advanced method for early fault detection in styrene polymerization processes using SNNs. The work aims to overcome the limitations of traditional statistical methods, such Hotelling's T2 and Squared Prediction Error (SPE) in conjunction with PCA and other machine learning models, by implementing more advanced and adaptable machine learning techniques.

Based on what was presented in Chapter 1, for a better understanding of the work, Chapter 3 presents the theoretical background on production process modeling of the polymerization of styrene, siamese neural networks, as well as depicting related works. Chapter 5 introduces the proposal for classifying faults using deep learning techniques. Chapter 6 presents the experiments and results obtained from the model and the classification. Finally, in Chapter 7, the main conclusions of the work are presented. The objectives are presented below:

- Review existing methods for fault detection in industrial chemical processes in the literature;
- Develop, implement and validate a polystyrene model using the method of moments for mass and energy balances in a CSTR reactor;
- Incorporate a PID control system to simulate realistic industrial conditions;
- Build a comprehensive database by introducing different types of faults in the styrene polymerization process;
- Implement and train Siamese Neural Networks with LSTM units using the constructed database;
- Evaluate the performance of the SNNs-LSTM in fault detection and compare its performance with SNNs-MLP;
- Interpret the experimental results to validate the efficacy of the SNNs in fault detection;
- Assess performance indicators such as accuracy and detection rate of the implemented SNNs;
- Examine the applicability and scalability of SNNs to other industrial chemical processes.

3
Literature Review

3.1
Objectives

This chapter aims to present a brief review of the polymerization mechanisms for the production of polystyrene, focusing on free-radical polymerization. Subsequently, the chapter presents kinetic mechanisms and models found in the literature to describe polymerization initiated by thermal means and monofunctional initiators, as well as the gel effect. Finally, polymerization processes are discussed, with a focus continuous reactors used for the commercial production of polystyrene.

3.2
Polymerization

At 1933, Dostal and Marklo proposed a mechanistic model, which comprises two fundamental steps: chain initiation and successive monomer addition. The chain initiation is a relatively slow process involving the 'activation' of a monomer, which then rapidly facilitates the addition of further monomers. Initially, it was assumed that these activated molecules could continuously grow by adding monomers indefinitely [35].

Separately, the concept of thermal initiation in vinyl polymerization was initially articulated by Flory in 1937. Flory proposed that the process commences with the combination of two styrene molecules, forming a diradical. This laid the groundwork for understanding how polymerization reactions can be triggered thermally [36].

The isolation of styrene monomer can be traced back to Newman in 1786, who conducted a distillation of liquidambar, a solid resin extracted from a family of trees native to the Far East and California. Liquidambar, historically utilized in medicine and perfumery, is characterized by a vanilla odor and contains cinnamic acid, which can be readily decarboxylated to produce styrene. In the intervening years between its discovery and commercialization, styrene emerged as a by-product in various chemical processes and was synthesized through multiple routes, albeit in limited quantities [13].

It wasn't until the advent of catalytic cracking of ethylbenzene and effective polymerization inhibitors that styrene, and consequently polystyrene,

became commercially viable. Polystyrene has been commercially available in the United States for 66 years, stemming from the straightforward polymerization of the styrene monomer [13].

One of the most important characteristics of crystal polystyrene is its appearance as a vitreous solid at temperatures below 100°C. Another notable feature is its high tensile strength, which makes it suitable for mechanical applications. It also exhibits good electrical insulating properties, which makes it useful in electrical components. This polymer is highly resistant to acids, alkalis, and most oils, broadening its applicability in chemical-resistant coatings. This means that the polymer is sufficiently fluid to be easily molded, making it applicable for thousands of uses. Above 100°C temperature, occurs glass transition, the polymeric chains (at the molecular level) gain rotational freedom, allowing segmental mobility of the chains.

Polymers can be classified based by polymerization mechanisms into addition and step polymerization. Addition polymerization involves the repetitive incorporation of monomers containing unsaturated double or triple bonds into the growing polymer chain. This process is generally governed by three fundamental stages initiation, propagation, and termination (Equations 4-1 - 4-8). Condensation polymerization, the chain growth occurs through reactions between the functional groups of monomers, resulting in non-uniform chain formation. This is because chains of any size can react with one another via their functional groups [24].

Hui and Hamielec (1972) developed a thermal initiation kinetic model that has become the foundational in the study of styrene polymerization, applicable over a temperature range of 100-200°C and a conversion spectrum of 0-100 %. Their research utilized both second and third-order reaction models to scrutinize experimental data concerning average molecular weights, molecular weight distribution and conversion rates. The third-order model was particularly effective in yielding accurate predictions for conversion and average molar weights. The kinetic constants we determined were based on the assumption that they remain constant regardless of the length of the chain, but they exhibit variations depending on the levels of conversion. Specifically, the constants related to the gel effect have proven to have a huge influence on the kinetics of styrene polymerization. These concepts are vital for understanding the rate equations used in the kinetic models of polymerization [8].

Free-radical polymerization is of substantial commercial interest because of its capacity to manipulate molecular weight distribution through

reaction. Styrene polymerization is an anionic polymerization that allows to produce of polystyrene with predefined molecular weight distributions, making it ideal for theoretical explorations of property changes as molecular weight varies [14]. In commercial applications, the resins used generally had average molecular weights ranging from 100,000 to 400,000 g/mol. These materials frequently exhibited a dispersity (Đ), defined as the ratio of the weight-average molecular weight to the number-average molecular weight, which varied between 2 and 4 [13].

Russo and Bequette (1998) applied bifurcation theory to investigate steady-state multiplicity in a jacketed Continuous Stirred Tank Reactor (CSTR) for styrene polymerization. They employed the Damkohler number to quantitatively express process, focusing on variables like the overall heat transfer coefficient and the cooling feed temperature. The kinetic model incorporated the following key reaction steps: initiator decomposition, chain initiation, propagation, and termination both via combination and disproportionation. Thermal initiation and gel effects were intentionally excluded, assuming a high benzene solvent concentration. The study was based on the assumption of constant physical properties for the reaction mixture, with temperature control managed via the cooling fluid in the reactor's jacket [37].

In 2000, Chen developed a mathematical model to predict the steady-state behavior in the continuous bulk polymerization of styrene, initiated by peroxide. The model accounted for factors such as polymerization kinetics, reactor dynamics, and pre-heating in the devolatilization stage. The model evaluated the effects of feed composition, thermal conditions, initiator type, and concentration on variables like monomer conversion rate, molecular weight, and key physicochemical properties. Nonlinear algebraic equations were solved using the Newton-Raphson method to determine the mass fraction of styrene in the reactor [38].

Giuping et al. (1999), investigated the molecular weight distribution in styrene polymerization within a batch reactor. Their empirical model revealed that the termination rate constant is dependent on the polymer molecular weight. The research also established that below 100°C, radical termination solely occurs through combination, and thermal initiation is inconsequential even at high reaction temperatures. However, the model was inadequate in accurately forecasting high molecular weight polymer chains [40].

Ultra-high molecular weight polystyrene, characterized by a molecular

weight exceeding 400,000 g/mol, is exclusively produced through polymerization processes that utilize rare earth metal-based catalysts. The kinetics of these high-viscosity polymerization systems have been modeled through Monte Carlo computational simulations, as evidenced by the research conducted by Ling et al. in 2001. Despite its innovative methodology, this approach to polystyrene polymerization has not seen widespread commercial adoption [39].

In 2022, Korkut conducted a study that delineated various methods for attaining specific molecular weights in polymer synthesis. These methods encompassed alterations to the initial monomer concentration, ultrasound power, and initiator concentration. The study concentrated on ultrasound-assisted emulsion polymerization of styrene, aiming to produce polystyrene of different target molecular weights. The research underscored that the most economically viable conditions for polymer synthesis are contingent upon the desired molecular weight, thus requiring adaptability in process design for manufacturers. According to Korkut, the dispersity (Đ) for the synthesized polymers varied between 1.2 and 1.5, while the viscosity-average molecular weights ranged from 100,000 to 1,500,000 g/mol [29].

The modeling of polystyrene is crucial for predicting polymer properties and ensuring industrial safety and quality. Its thermoplastic nature allows it to be easily molded and recycled, making it a versatile material in various applications. It is widely used in the production of disposable cutlery, food packaging, and insulation materials. Its high insulating properties make it ideal for use in foam board and expanded polystyrene (EPS) products, which are critical in construction and packaging industries. Moreover, polystyrene is employed in the electronics sector as an insulating material for components [10] [17].

Modeling techniques for polymerization typically fall into two categories: statistical methods such as Monte Carlo and deterministic methods like the method of Moments. Monte Carlo methods offer the ability to capture these complexities but come with disadvantages such as high computational cost, slow convergence, scalability and memory requirements. On the other hand, deterministic approaches, like the method of Moments, provide good solutions and lower computational costs. Deterministic models like the method of moments are based on kinetic theory and account for a "memory" of the reaction through mass balance equations [6] [7] [11].

The method of moments (MoM) has become a popular way to study how polystyrene is synthesized. A lot of research papers have looked at

different parts of this method, showing that it's a flexible and reliable way to predict what happens in the process of making polymers [12]. Mastan and Zhu (2015) offers a comprehensive tutorial that provides the step-by-step on how to apply the method of moments to different polymerization and polymer modification systems. Serving as an invaluable resource, this paper systematically demystifies the application of MoM, effectively lowering the barrier to entry for practitioners in the field [11].

Other study done by Riazi et al. (2019) focus on moment rate equations for high-temperature free-radical polymerization reactions. His research sets a significant foundation for understanding the kinetics involved and demonstrates the adaptability of MoM in accommodating diverse reaction conditions [26]. Bachmann (2017) extended the application of MoM to nonlinear free-radical polymerization. Unlike traditional applications that often assume steady-state conditions or statistical distributions of connections, Bachmann derives moment equations that allow for more complex modeling scenarios [27].

In other study, Tikhomirov (2016) gives a comprehensive examination of how method of moments can be utilized for simulating the synthesis of polymers, with a specific focus on modeling variations in molecular weight. Based on experimental studies of butadiene-styrene copolymer degradation, the research develops a mathematical framework that accounts for both technological properties like viscosity and structural characteristics such as molecular weight distribution parameters. The work goes beyond mere simulation to suggest how method of moments develop polymers with specific controlled properties and in the process control system. This research demonstrates method of moments ability to model not just idealized scenarios but also the intricate variables encountered in real-world polymer synthesis [].

The concept of monofunctional polymerization, involving monomers with a single functional group, is significant in polystyrene synthesis due to its potential for producing high purity and simplified reaction kinetics.Studies have looked into how the monofunctional nature of styrene monomers influences the final properties of polystyrene [8] [9].

Hayden and Melville (1960) studied the kinetics of methyl methacrylate polymerization, and calculated the gel effect, or Trommsdorf effect, can be described as a transition from a liquid state to a gelled solid. This transition is intimately linked with a phenomenon of auto-acceleration in the conversion rate [18]. During polymerization, especially at high conversions, this effect becomes significant. It is primarily caused by diffusion limita-

tions from large molecules that slow down termination reactions between the polymer radicals chains and relative increase of propagation due to the facility of monomer to move. As a result, the rate of polymerization accelerates, leading to rapid polymer network formation [30]. After that, Choi et al. (1988) acknowledges the impact of the gel effect in styrene polymerization. In other words, translational diffusion declines faster than segmental diffusion increases, triggering rapid self-acceleration. Higher conversion leads to more interconnected polymer chains due to increased viscosity and chain size [20].

3.3 Proportional-Integral-Derivative (PID)

Proportional-Integral-Derivative (PID) control is a closed-loop feedback system commonly applied in industrial control settings. In general, PID controller operates with the objective of minimizing the error term by modulating a control variable, such as inhibiting setting on a motor, and temperature or pressure of a reactor [31].

PID control system has been a pillar in industrial processes, including polymerization. PID control operates by adjusting process variables in real-time to maintain optimal conditions for polymerization. Previous research has demonstrated the efficacy of PID control in stabilizing the polymerization of polystyrene by continuously monitoring variables such as monomer concentration and temperature, PID control systems provide a dynamic adjustment mechanism [32].

PID controllers are predominant in control systems due to their straightforward design, ease of implementation, and the longstanding focus on enhancing PID tuning methodologies. In the context of control systems and biomedical applications, this document offers a literature review that chronicles the evolution from classic PID controllers to those augmented with intelligent control. Recent advancements in PID control, as identified in the literature, include the development of fractional-order PID controllers, fuzzy logic integration, IMC-PID controller design, and the synergistic use of PID with observer structures [58].

Hapoglu et al. (2003) applied of adaptive PID control combined with a genetic algorithm was implemented for maintaining optimal temperatures in a jacketed batch polymerization reactor. The optimal temperature path for the polymerization reactor was established using the Hamiltonian maximum principle method, with the reactor's model equations being solved using

Runge-Kutta-Felthberg methods. Genetic algorithms (GAs) provide a robust solution for optimizing PID parameters, presenting fewer constraints and simplicity compared to other techniques [56].

Altınten et al. (2008) worked on a self-tuning PID controller that was applied to manage the temperature of a jacketed batch polymerization reactor. The controller's effectiveness in tracking optimal temperature profiles. In the context of styrene production, the effectiveness of the self-tuning PID controller combined with GA was demonstrated. This controller displayed high proficiency in adhering to various optimal temperature trajectories, as established by experimental results [57].

3.4 Artificial Neural Networks

Artificial neural networks (ANNs), originally originally presented by McCulloch and Pitts (1943), are interconnected artificial neurons used for various regression and classification tasks. Their effectiveness in these domains is attributed to their universal function approximation properties, allowing for multidimensional non-linear mappings between dependent and independent variables [47].

ANNs are computational models designed to emulate aspects of human brain function. They are composed of interconnected units, analogous to neurons, and are capable of learning from data through adjustments to these interconnections, also known as weights. ANNs have shown efficacy in a wide range of applications, from forecasting, complex classification problems and distinguishing similarities [34].

Rosenblatt and Frank (1962) and Haykin (1999) elucidate that artificial neural networks are predominantly constructed using a multilayer architecture, known specifically as multilayer perceptrons (MLP). The MLP architecture comprises three essential components: an input layer, one or more hidden layers, and one output layer. This configuration defines the depth of the artificial neural network as can been see n the Figure 4.1 [50] [51].

In the field of fault detection, principal component analysis (PCA) has conventionally served as a standard approach in data analytics. PCA uses a non-parametric approach, and affords dimensionality reduction with computational efficiency, thereby distilling complex data into actionable insights. However, PCA has key limitations including its assumption of linearity, sensitivity to outliers, and focus on variance, which may not align with

fault characteristics. It also lacks temporal analysis and requires periodic re-training, with impacts real-time applicability [42]. Ku et al. (1995) has utilized PCA for monitoring chemical operations and anomaly detection. They have further contributed by isolating the sources of anomalies using static PCA models, T* and Q charts, and a catalog of probable disturbances. By incorporating a 'time-lag shift' approach they could introduce a temporal dimension to the traditionally static PCA models [43].

Recent research has used artificial neural networks for fault detection. The complexity of most chemical industry always tends to create a problem in monitoring and supervision system. Fast fault detection and diagnosis is a best way to handle and tackle this problem. There are different methods tackling different angle already proposed in literature. Artificial neural networks have demonstrated successful applications in fault detection within chemical systems. Models has been employed for pattern recognition in complex systems and for fault detection. These artificial neural networks offer a non-linear mapping between input and output variables, which can be particularly useful for capturing the complexities in polymerization kinetics and thermodynamics [1] [2] [4].

The paper of Patton (1994) presented a novel methodology for fault detection and isolation in nonlinear dynamic systems through the use of artificial neural networks. His approach consists of two stages: state prediction and fault classification. A multi-layer perceptron network is employed in the first stage to generate residual signals. Artificial neural networks offer advantages in addressing nonlinearities, primarily through their capability to approximate any nonlinear function given appropriate weights and architecture. This feature facilitates on-line learning, making the approach suitable for real-time fault diagnosis [33].

Nashalji et al. (2009) proposes a hybrid method combining PCA improved by fenetic algorithm and a ANN classifier for fault detection. This study introduces a hybrid multivariate technique that combines PCA to enhance the detection of faults in industrial processes. The developed model was validated using simulated data from the Tennessee Eastman chemical plant simulator [49].

Chadha et al. (2017) compared two deep artificial neural network architectures and found that the sparse stacked autoencoders model has superior fault detection capability. This article investigates the utility of deep artificial neural networks for early fault detection in industrial processes. Two architectures, deep stacking networks and sparse stacked autoencoders, are

compared using the Tennessee Eastman benchmark [48].

Zheng and Zhao (2020) implemented an unsupervised fault diagnosis method in chemical engineering, countering the issue of limited labeled data. The method involves three phases: feature extraction via a convolutional stacked autoencoder with long short-term memory (LSTM) and convolutional layers; feature visualization using t-SNE; and clustering. Their model was validated on the Tennessee Eastman Process and an industrial hydrocracking instance, showing high labeling accuracy and Q metric value but with limitations in identifying new fault types. Future work aims to refine feature extraction and explore clustering algorithms as a loss function. The authors evaluated the efficacy of this approach by conducting tests on the Tennessee Eastman Process (TEP) and an industrial hydrocracking instance, achieving a Q metric value of 0.986, a labeling accuracy of 97.8 %, and an average testing False Discovery Rate (FDR) of 93.0 % [44].

Xiaoyang Lua et al. (2019) utilized convolutional neural network for classifying fault types based on extracted features. In essence, the convolutional layers serve as automatic feature extractor. The diagnostic approach consists of two primary stages: feature extraction and classification. The raw sequential data of current and voltage in the photovoltaic cell are transformed into a two-dimensional electrical time series graph for visualization. The proposed CNN-based diagnostic model demonstrates a high efficacy, achieving over 99 % average accuracy in case studies [41].

Gravanis et al. (2022) addressed a critical concern in Industry 4.0 optimizing production costs by enhancing the fault detection and diagnosis (FDD) systems specifically for non-linear processes. The author propose a novel FDD framework that employs recurrent networks, including LSTM and time delay neural networks (TDNN). These networks are utilized for their capacity to manage sequential data, which is crucial for understanding, time-dependent industrial processes [45].

3.5 Siamese Neural Networks

The siamese neural network algorithm was first introduced by Bromley et al. (1994) for detection of falsified signatures comparing two handwritten signatures. During the training phase, these sub-networks extract feature vectors from two different signatures. Verification is achieved by comparing a newly extracted feature vector with a pre-stored vector representing the genuine signature of a signer. Signatures with feature vectors closer to the

stored representation—within a specified threshold are classified as authentic, while others are classified as forgeries. The algorithm's robustness is demonstrated by its resistance to forgeries among individuals who maintain consistent signing patterns and have a accuracy of 97.0 % [42].

SNNs offers unique functionalities by specializing in the identification of similarities or anomalies in paired datasets. This proves to be advantageous in identifying system deviations, which could signify potential failures or inefficiencies in the polymerization process. SNNs compute the similarity between two input vectors, which can be employed for tasks such as anomaly detection and fault identification in polymerization systems. Unlike traditional artificial neural networks, a SNN takes in two separate inputs that go through identical subnetworks, meaning the weights and parameters are the same for both paths. Given their capability to compare datasets, SNNs can be employed to benchmark real-time data against historical data, thus effectively identifying deviations that may warrant immediate solution [34].

Despite the limited research on fault detection in polymers using SNN, numerous references aim to develop models for distinguishing similarities and dissimilarities. In most cases, siamese neural network performs a nonlinear encoding of the input data with the aim of reaching a semantically meaningful space where related patterns are close to each other (such as faces of people, signatures, and others) and unrelated patterns are distant from each other [5].

Zhou et al. (2021) introduced a few-shot learning model with a SNN for intelligent anomaly detection in industrial cyber-physical systems, improving accuracy and false alarm rate [52]. Nagy et al. (2021) examines deep artificial neural network models for the recognition of traffic sign defects, focusing on both convolutional neural networks (CNNs) and siamese convolutional neural networks for defect detection. The accuracy results achieved were 99.50 % and 97.90 %, respectively [54]. Fernandez-Llaneza et al. (2021) introduced a siamese recurrent artificial neural network model for bioactivity prediction in drug discovery, highlighting its ability to learn task-specific chemistry features [53].

In this work, SNNs are used as an extension of traditional artificial neural networks designed to address problems that involve finding similarities between pairs of data. The aim of this work is to train the SNN to distinguish the normal data and the faulty data.

4
General Concepts

In this chapter, concepts related to the model, PID, artificial neural networks and the Siamese topology are presented, as well as how this can be applied to fault detection. Finally, a brief explanation is provided regarding the metrics that were used for the evaluation of the proposed model.

4.1
Model and Kinetics Polymerization

Styrene, an aromatic hydrocarbon, possesses a vinyl group that facilitates its polymerization through a free radical mechanism, a process pivotal to the formation of polystyrene. In conventional free-radical polymerization driven by a monofunctional initiator, the kinetic mechanism encompasses initiation, propagation, chain transfer and termination, as outlined in Equations 4-1 - 4-8 [8] [9]. Below, the reactions pertinent to each stage of free-radical polymerization are meticulously outlined:

– **Initiation:**

The initiation process commences with the formation of free radicals, typically arising from the decomposition of an initiator. This can be represented in the context of styrene as follows:

$$I \xrightarrow{k_d} 2R \tag{4-1}$$

$$R + M \xrightarrow{k_i} P_1 \tag{4-2}$$

I is the initiator, R is the free radicals, M is the monomer (styrene), and P_1 is the initial polymeric radical. The kinetic constant for initiator dissociation is represented by k_d, and k_i denotes the kinetic constant for the initiation phase.

– **Thermal Initiation:**

The thermal initiation, represented by equation 4-3, involves the direct transformation of monomers into polymeric radicals without the presence of a traditional initiator.

$$3M \xrightarrow{k_{dm}} 2P_1 \tag{4-3}$$

– **Propagation:**

The free radical $\cdot R$ then interacts with a styrene monomer, initiating a chain reaction with the live polymer $\cdot P$. This propagation phase is crucial in determining the molecular weight and structure of the resulting polymer.

$$P_i + M \xrightarrow{k_p} P_{i+1} \tag{4-4}$$

In this stage, P_i represents an existing polymeric chain, and P_{i+1} is the extended chain after the addition of one monomer unit. The rate constant for this reaction is k_p.

– **Chain Transfer:**

Chain transfer reactions play a significant role in controlling the molecular weight and distribution of the polymer.

$$P_i + M \xrightarrow{k_{tm}} L_i + P_1 \tag{4-5}$$

$$P_i + X \xrightarrow{k_{tx}} L_i + P_1 \tag{4-6}$$

$$P_i + S \xrightarrow{k_{ts}} L_i + P_1 \tag{4-7}$$

Reactions 4-5, 4-6, and 4-7 describe the transfer of the polymeric chain to another monomer M, impurity X or solvent S. The constants k_{tm}, k_{tx}, and k_{ts} represent the respective rate constants for these transfers, respectively.

– **Termination by Combination:**

The polymerization reaction culminates when two free radicals combine or react with an inhibitor, thus terminating the growth of the polymer chain.

$$P_i + P_j \xrightarrow{k_{tc}} L_{i+j} \tag{4-8}$$

In termination reaction 4-8, two polymeric radicals P_i and P_j combine to form a longer polymer chain L_{i+j}. The rate constant for this reaction is represented by k_{tc}.

4.2 Proportional-Integral-Derivative

The PID control system comprises three integral components: Proportional, Integral, and Derivative. The proportional component scales the error by a constant factor, referred to as K_p, the proportional gain constant [31].

$$P_{\text{out}} = K_p \cdot e(t)$$

The integral component is concerned with the cumulative sum of past errors. This sum is then scaled by the integral gain constant, K_i. The integral output can be mathematically described as

$$I_{\text{out}} = K_i \int_0^t e(\tau) d\tau$$

The derivative component predicts the future error by focusing on its rate of change. This prediction is then multiplied by the derivative gain constant, K_d. The derivative output is expressed as

$$D_{\text{out}} = K_d \frac{d}{dt} e(t)$$

Combining these three components, the PID control function $u(t)$ is articulated as

$$PID(t) = K_p \cdot e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \tag{4-9}$$

4.3 Method of Moments

The method of moments offers a significant advantage in reducing the computational complexity while maintaining the essential characteristics of the polymerization process. This method employs a set of ordinary differential equations (ODEs) derived from fundamental rate equations for initiation, propagation, and termination steps. In the case of polystyrene synthesis, these equations can be solved numerically to obtain moment values, which are subsequently used to calculate key polymer properties.

In radical polymerization, the i^{th} moments for living polymers and dead polymers are defined by Equations 4-10 and 4-11, as presented by μ and λ. Typically, due to the high reactivity of the living polymer chains, their moments are significantly smaller than those of dead chains and

other polymer populations, aligning with the quasi-steady-state assumption (QSSA). Consequently, in systems comprising only dead and radical chains, it was reasonable to approximate the overall polymer moments as being essentially equal to the moments of the dead chains.

$$\mu_k = \sum_{i=1}^{\infty} i^k \times P_i \tag{4-10}$$

$$\lambda_k = \sum_{i=1}^{\infty} i^k \times L_i \tag{4-11}$$

Where, μ_k is the k^{th} moment of living polymers, λ_k is the k^{th} moment of dead polymers. The summation runs from $i = 1$ to ∞, assuming that the distribution could, in principle, extend to infinitely long chains.

The zeroth-order moment ($\sum N_i$, where N_i is the number of polymer chains with i monomer units) is a physical representation of polymer chain concentration within the system, while the first-order moment ($\sum iN_i$) symbolizes the weighted count of monomeric units in the polymer chains. The second-order moment ($\sum i^2 N_i$) quantifies variance in the polymer's molecular weight distribution (MWD). These statistical moments facilitate the computation of the number-average molecular weight (Mn), the weight-average molecular weight (Mw), and the dispersity (Đ), properties that are crucial for characterizing the polymerization process and its end-product [11]. The equations are governed by the following set of equations:

$$M_n = \frac{\sum n_i M_i}{\sum n_i} = \frac{(\lambda_1 + \mu_1)}{(\lambda_0 + \mu_0)} \tag{4-12}$$

$$M_w = \frac{\sum n_i M_i^2}{\sum n_i M_i} = \frac{(\lambda_2 + \mu_2)}{(\lambda_1 + \mu_1)} \tag{4-13}$$

$$PDI = \frac{M_w}{M_n} \tag{4-14}$$

4.4 Artificial Neural Networks

Artificial Neural Networks (ANNs) represent a cornerstone in the field of machine learning and have been instrumental in advancing numerous applications in science and engineering. Mimicking the structure and function of biological neural networks, ANNs have the remarkable capability to learn from data and make intelligent decisions. This section delves into the architecture and functioning of ANNs, highlighting their key components and the roles they play in data processing and pattern recognition.

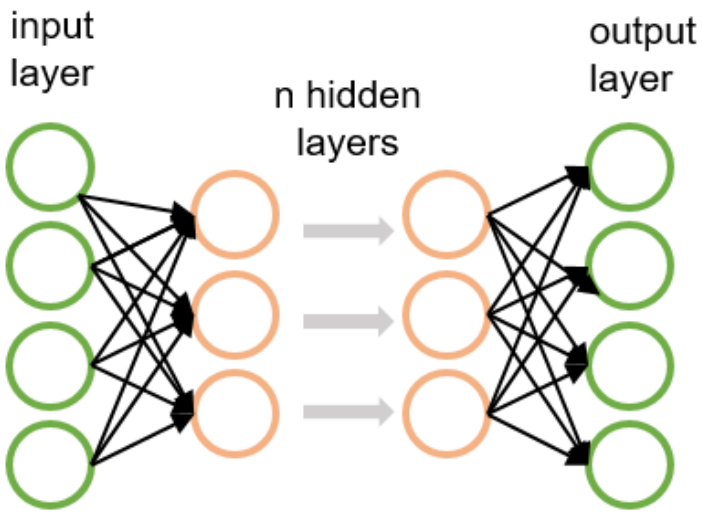

Figure 4.1: Architecture of NN.

The architecture of an ANN consists of neurons organized into layers: input, hidden, and output—along with associated weights and biases. These elements facilitate computations and are adjustable during the training phase to optimize performance. Each neuron computes its output y by applying a linear transformation followed by an activation function f to its inputs:

$$y = f\left(\sum_{i=1}^{n} w_i \cdot x_i + b\right)$$

Where:

- y is the output signal.
- f is the activation function.
- w_i are the weights associated with the input signals.
- x_i are the input signals.
- n is the number of input signals.
- b is the bias term.

The activation function in a artificial neural network governs the last decision from the neuron and introduces non-linearity, thereby affecting the output produced by the summation function. The activation functions were essential for assembling multiple perceptrons into a unified network while preserving its non-linear characteristics. Activation functions provide the crucial non-linearity needed for artificial neural networks to compute complex functions and perform tasks such as classification and regression.

They do this by determining the output of a neuron ($f(z)$) and the combined input signals (z).

- Linear:
$$f(z) = z \tag{4-15}$$
Used primarily for simplicity, but often insufficient for complex data patterns.
- Rectified Linear Units (ReLU):
$$f(z) = \max(0, z) \tag{4-16}$$
ReLU and its variants are pivotal in deep learning models due to their efficiency in activating multiple layers of the network without falling prey to the vanishing gradient problem.
- Hyperbolic Tangent (Tanh):
$$f(z) = \tanh(z) \tag{4-17}$$
Tanh and SELU functions provide scaled outputs, which are crucial in models where normalization of activation is required.
- Sigmoid:
$$f(z) = \frac{1}{1 + e^{-z}} \tag{4-18}$$
Sigmoid and its extended forms like Softmax and Swish are particularly useful in binary classification and probabilistic interpretations.

The efficacy of an ANN is highly dependent on the interplay of its architecture, activation functions, and training algorithms. Deep learning, a subset of machine learning involving ANNs with multiple hidden layers, exploits this interplay to model high-level abstractions in data, providing remarkable results in complex tasks like autonomous driving, precision medicine, and real-time language translation.

ANNs also integrate concepts like dropout and batch normalization to combat overfitting and ensure generalized performance across unseen data. Dropout randomly disables neurons during training, preventing co-adaptation and promoting robust feature learning. Batch normalization standardizes the inputs to each layer, accelerating training and stabilizing the learning process.

In summary, ANNs are a dynamic and potent tool in machine learning, capable of modeling complex, non-linear relationships in large and diverse datasets. Their versatility and adaptability make them suitable for a wide range of applications, from intricate pattern recognition in vast data

sets to real-time decision-making in dynamic environments. The continuous evolution in ANN architectures and training methodologies promises further groundbreaking advancements in artificial intelligence and computational modeling.

4.5 Siamese Neural Networks

The architecture of a SNN is characterized by its symmetry, with both input channels being structurally analogous and sharing the same parameters. The primary goal of the model is to either minimize or maximize a specified distance metric between dual outputs, typically the Euclidean distance. This process is key in determining the level of similarity or dissimilarity between the pair of inputs. Unlike conventional models, no activation function is applied to the output layer in SNNs when using Euclidean distance as the similarity measurement, as the raw output is used to calculate the distance between feature vectors.

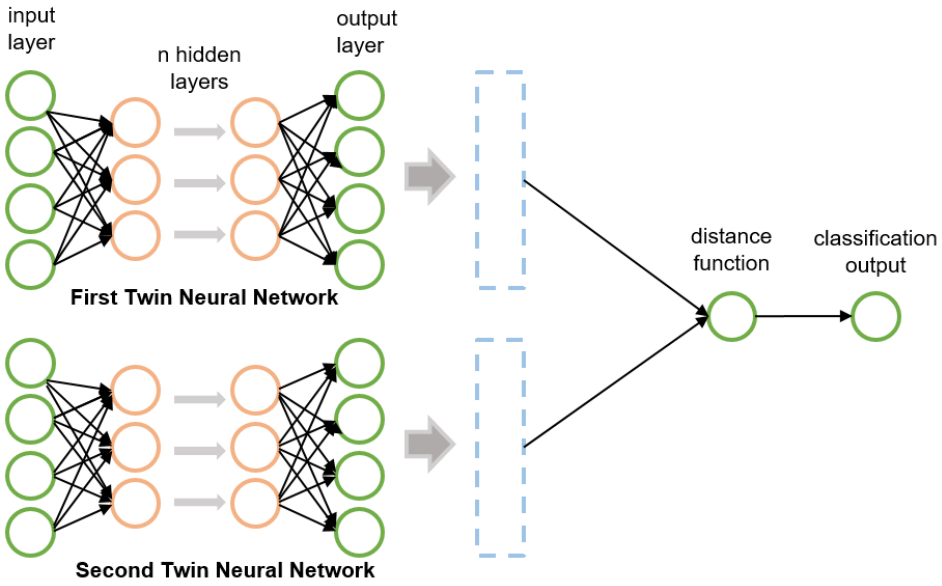

Figure 4.2: Architecture of SNN.

A siamese neural network processes pairs of input readings, aiming to increase the similarity between pairs from the same class, while distancing those from different classes. Input data undergoes sequential processing through the network's layers to produce the final output. The process of learning is centered around the concept of distance metrics. While the Euclidean distance (Equation 4-19) is commonly used, alternative distance

metrics such as the Manhattan distance, Cosine similarity, or more complex functions like learned distance metrics using a separate sub-network can also be employed.

$$\text{Euclidean distance}(y_{\text{pred}}) = \sqrt{\sum_{i=1}^{n}(x_{1i} - x_{2i})^2} \tag{4-19}$$

Based on the computed Euclidean distance and a label indicating whether the pair is similar or dissimilar, the contrastive loss is calculated according to Equation 4-20. This loss function aims to minimize the distance between similar pairs while maximizing the distance between dissimilar pairs [46]. The contrastive loss formula is expressed as:

$$\text{Contrastive Loss} = \frac{1}{N}\sum_{i=1}^{N}\left((1 - y_{\text{true,i}}) \cdot y_{\text{pred,i}}^2 + y_{\text{true,i}} \cdot \max(0, \text{margin} - y_{\text{pred,i}})^2\right) \tag{4-20}$$

In this equation:

- N is the total number of pairs in the dataset, with the loss averaged over all these pairs.
- $y_{\text{true},i}$ is the binary label for the i-th pair, where 0 signifies similar pairs, and 1 denotes dissimilar pairs.
- $y_{\text{pred},i}$ is the Euclidean distance between the feature vectors of the two inputs in the i-th pair.
- The term $(1 - y_{\text{true},i}) \cdot y_{\text{pred},i}$ calculates the loss for similar pairs, where the model is penalized if similar items are far apart.
- The term $y_{\text{true},i} \cdot \max(0, \text{margin} - y_{\text{pred},i})^2$ accounts for dissimilar pairs, increasing the loss when the distance $y_{\text{pred},i}$ is less than a specified margin. The margin is a hyperparameter defining the desired separation in the feature space for dissimilar pairs.

The objective of the contrastive loss function is to learn a feature space where the distances between similar pairs are minimized, while the distances between dissimilar pairs are large. This function effectively shapes the feature space during the training of the SNN, enhancing the network's ability to differentiate between similarities and differences in the input data.

Among its advantages is the efficacy in sculpting a feature space conducive to clustering similar entities closely, while ensuring a clear demarcation from dissimilar ones, a trait indispensable in applications such as facial

recognition. This loss function further enhances the model's capacity to generalize effectively to novel data, attributing to its focus on relative, rather than absolute, feature distances, thereby honing its ability to discern the crux of similarities and dissimilarities. Additionally, its robustness against noisy labels emerges from its reliance on relative pair positioning, presenting an edge over other loss functions grounded in absolute classifications.

Contrastive loss is effective and scalable, especially for large datasets, because it only considers distances between pairs of data points, rather than the whole dataset. This loss function allows customization through the 'margin parameter', which helps to set the appropriate level of difference between similar and dissimilar pairs. This customization is important as it suits the varying requirements of different applications. However, the effectiveness of contrastive loss depends on the assumption that the Euclidean distance between data points is a relevant measure, which might not always be true. The model's success is highly sensitive to the chosen value of the margin hyperparameter, requiring careful adjustment and possibly many trials to find the best value. This loss function, focusing only on pairwise distances, might miss more complex relationships that other loss functions, considering wider interactions, could catch.

The stage of backpropagation involves updating the network's weights and biases by propagating the loss from the output layer back through the hidden layers to the input layer. These values are then used to adjust the parameters in a way that minimizes the contrastive loss. The iterative process of forward and backward propagation continues over multiple epochs until a satisfactory performance metric is achieved or a pre-defined stopping criterion ceases the training.

Weights and biases within the SNN are iteratively updated to minimize the loss function, denoted as $\mathcal{L}$. Gradients of the loss function with respect to weights ($\mathbf{W}$) and biases ($\mathbf{b}$) are computed. Both equations incorporate a negative sign as the objective is to descend along the gradient towards a minimum in the loss function. The SNN adjusts its weights and biases by repeatedly applying these update rules, thereby improving its performance.

Here, η is the learning rate, determining the step size in the optimization progress. The term $\frac{\partial \mathcal{L}}{\partial \mathbf{W}}$ represents the gradient of the loss function concerning the weights. This equation modifies the existing weights in a direction that reduces the loss. Similar to the weight update, η is the learning rate, and $\frac{\partial \mathcal{L}}{\partial \mathbf{b}}$ is the gradient of the loss function concerning the biases. This

equation adjusts the biases to minimize the loss function further, as seen in Equations 4-33 and 4-34.

– Weight Update:
$$\Delta \mathbf{W} = -\eta \frac{\partial \mathcal{L}}{\partial \mathbf{W}} \tag{4-21}$$

– Bias Update:
$$\Delta \mathbf{b} = -\eta \frac{\partial \mathcal{L}}{\partial \mathbf{b}} \tag{4-22}$$

This process iteratively adjusts the weights and biases to minimize the contrastive loss, enhancing the network's ability to learn discriminative features from the input data. The cycle of forward propagation (to compute the loss) and backpropagation (to update the parameters) is repeated over multiple epochs until the network achieves the desired level of accuracy, or a pre-defined stopping criterion is met.

Diverse sub-network integrations augment the versatility of SNNs. For instance, the incorporation of Convolutional Neural Networks (CNNs) is instrumental in image-based applications like facial recognition and medical imaging, due to their proficiency in extracting hierarchical spatial features. Recurrent Neural Networks (RNNs), with their inherent 'memory' of past inputs, are integrated for sequential data analysis, enhancing performance in text processing and time-series analysis. Long Short-Term Memory (LSTM) units, a specialized RNN variant, further refine this capability, adeptly handling long-term dependencies in sequences, crucial in language translation or speech recognition.

Beyond conventional frameworks, SNNs also encompass Feedforward Neural Networks for simpler, non-temporal tasks and advanced RNN variants like Gated Recurrent Units (GRUs) for managing computational complexity in sequence data processing. Moreover, hybrid models in SNNs synergize the spatial pattern recognition of CNNs with the temporal data processing prowess of RNNs/LSTMs, exemplifying their adaptability across a spectrum of computational needs. This multifunctional integration not only showcases the adaptability of SNNs but also marks them as a cornerstone in the advancement of neural network architectures.

4.6 Long Short-Term Memory

LSTM networks are a specific type of recurrent neural network (Figure 4.3) [60]. Designed to learn long-term dependencies and address the vanishing gradient problem in traditional recurrent networks [59]. Individual blocks within the LSTM contain memory cells connected to multiplying

units, with forget gates that remove irrelevant information. This type of neuron excels in capturing long-term dependencies, making them useful in time series analysis and modeling time-dependent phenomena such as reaction kinetics and delayed effects common in polymerization reactors.

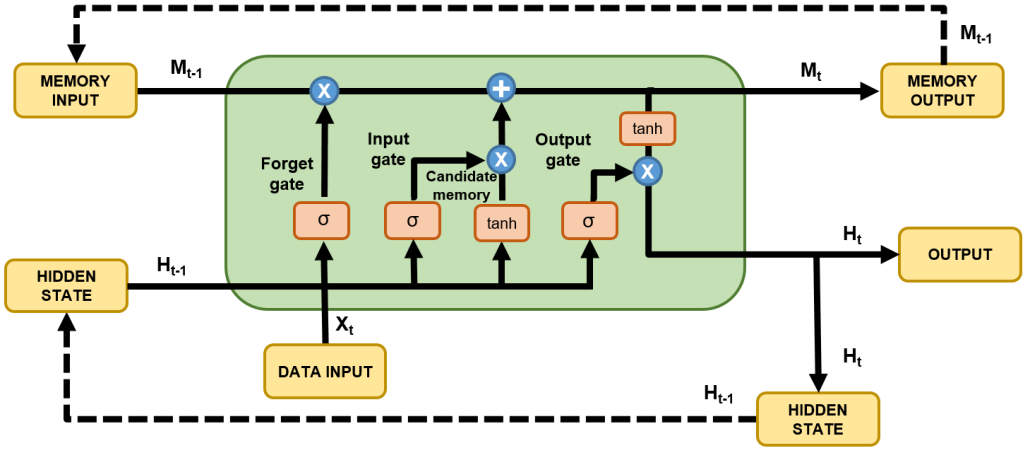

Figure 4.3: LSTM cell.

The fundamental operations within an LSTM cell are described by the following equations:

$$
\begin{aligned}
\text{Forget Gate:} \quad & f_t = \sigma(W_f \cdot [H_{t-1}, X_t] + b_f) \\
\text{Input Gate:} \quad & i_t = \sigma(W_i \cdot [H_{t-1}, X_t] + b_i) \\
\text{Cell State Update:} \quad & \Delta C_t = \tanh(W_c \cdot [H_{t-1}, X_t] + b_c) \\
\text{Cell State:} \quad & C_t = f_t \odot M_{t-1} + i_t \odot \Delta M_t \\
\text{Output Gate:} \quad & o_t = \sigma(W_o \cdot [H_{t-1}, X_t] + b_o) \\
\text{Hidden State:} \quad & H_t = o_t \odot \tanh(M_t)
\end{aligned}
$$

The LSTM cell's architecture is intricately designed with several gates and states to efficiently manage the flow of information through time. The forget gate (f_t), represented by $\sigma(W_f \cdot [H_{t-1}, X_t] + b_f)$ where σ denotes the sigmoid activation function, determines the portion of the previous memory state know as cell state (M_{t-1}) to retain or discard.

The input gate (i_t) evaluates the amount of new information to be introduced in the memory, while the cell state update ($\Delta M_t = \tanh(W_c \cdot [H_{t-1}, X_t] + b_c)$) computes the fresh information to be added to the cell state (M_t). This updated state is a combination of the old state and new inputs, modulated by both the forget and input gates.

The output gate (o_t) decides what portion of the cell state will influence the output. Here, σ is the sigmoid function, which scales the input between 0 and 1. This scaling is crucial as it determines the degree to which information in the cell state affects the output.

The hidden state (H_t), which is the output of the LSTM cell at the current time step. The cell state M_t is first normalized by the tanh function, which scales its values between -1 and 1. The output gate's activation o_t then performs an element-wise multiplication with this normalized cell state. The resulting H_t is the output of the LSTM cell for that time step and is also passed to the cell in the next time step as part of its input. This mechanism ensures that the network selectively propagates relevant information, thereby effectively modeling long-term dependencies.

4.7 Evaluation

In the realm of machine learning and statistical modeling, the evaluation of binary classification models holds a position of paramount importance. Binary classification involves categorizing data into one of two distinct classes. The efficacy of such a model is not just a matter of algorithmic robustness but also hinges on the appropriateness and precision of its evaluation metrics. Among the most widely used metrics are Accuracy, Precision, Recall, and the F1 Score. Each of these metrics offers a unique lens through which the performance of a binary classification model can be scrutinized and understood. Below the terms used to calculate the metrics are explained.

- **True Positive (TP)**: The number of examples that belong to a positive class and are predicted as positive.
- **True Negative (TN)**: The number of examples that belong to a negative class and are predicted as negative.
- **False Positive (FP)**: The number of examples that belong to a negative class but are predicted as positive by the model.
- **False Negative (FN)**: The number of examples that belong to a positive class but are predicted as negative by the model.

- **Accuracy:**

Accuracy is the most intuitive and initial measure often employed to evaluate a binary classification model. It is defined as the proportion of

true results (both true positives and true negatives) in the total dataset. Mathematically, it is represented as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4-23}$$

This metric offers a general overview of the model's performance. However, its reliance on a balanced dataset is its Achilles' heel. In scenarios where there is a significant class imbalance (one class is more prevalent than the other), accuracy can become a misleading indicator.

– **Precision:**

Precision steps into the spotlight where the cost of a false positive is high. It measures the quality of the positive predictions made by the model. Precision is defined as the ratio of correctly predicted positive observations to the total predicted positive observations:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4-24}$$

It tells us how many of the items identified as positive are actually positive. However, used alone, precision does not give a full picture, it does not account for the positive cases the model failed to identify.

– **Recall:**

Recall steps into the spotlight where the cost of a false negative is high. Also known as sensitivity or the true positive rate, addresses this limitation of precision. It measures how many of the actual positive cases the model successfully captures through its predictions:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4-25}$$

In fault detection, for instance, recall becomes a crucial metric. A high recall rate means that the model successfully identifies most faults with the condition. However, like precision, recall too, if used in isolation, does not provide a complete picture of the model's performance. It does not penalize the model for the number of incorrect positive predictions made.

– **F1-Score:**

The F1 Score harmonizes precision and recall, providing a single metric that balances both. It is particularly useful in scenarios where an equilibrium between precision and recall is necessary. The F1 Score is the harmonic mean of precision and recall, giving:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4-26}$$

This metric is especially useful when dealing with imbalanced datasets. Unlike accuracy, the F1 Score takes into account both false positives and false negatives. It is a more robust measure of a model's performance, especially in scenarios where either precision or recall alone could be misleading.

– **Matthews Correlation Coefficient (MCC):**

The Matthews Correlation Coefficient (MCC) is a robust statistical rate that yields a high score only if the predictor successfully predicts the positive class as positive and the negative class as negative. Unlike other metrics, MCC takes into account true and false positives, true and false negatives. It is defined as:

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4-27}$$

MCC returns a value between -1 and +1. A coefficient of +1 represents a perfect prediction, 0 no better than random prediction, and -1 indicates total disagreement between prediction and observation. This metric is particularly useful in binary classification problems, especially with imbalanced datasets, as it provides a more informative and truthful evaluation of the model's performance than accuracy alone.

MCC is considered one of the best measures of the quality of binary classifications, balancing the dataset's imbalance and reflecting all four confusion matrix categories. It is a reliable statistical rate even when the two classes are of very different sizes.

In sum, the evaluation of binary classification models requires a nuanced understanding of various metrics. Understanding and applying these metrics effectively can significantly enhance the evaluation and subsequent application of binary classification models in practical scenarios, leading to more reliable and contextually relevant outcomes.

5
Methodology

5.1
Objectives

The objectves of this metodology is we discuss the initial stages of the process. It starts with data generation through a phenomenological model, which captures the principles of thermodynamics of styrene polymerization reactions. The next step involves formulating a specific model for polystyrene polymerization with MoM. With this model, a Proportional-Integral-Derivative (PID) control system is integrated to maintain the desired process conditions by adjusting the control variables based on the feedback received. Subsequently, a database is constructed encompassing both normal operation and fault simulations to an SNN model. This database is essential for training machine learning models, as it provides a variety of scenarios for the algorithm to learn from.

Then to delve into the application of machine learning techniques. The gathered data, characterized by both normal and anomalous process behavior, is utilized to train SNN. Once trained, these networks enable continuous fault detection by continuously analyzing process data and comparing it to the learned profiles of normal and faulty conditions. The decision-making process involves a threshold-based system where, if the detected anomaly exceeds a set threshold, the process is evaluated for potential faults. If a fault is identified, corrective feedback is provided for continuous process monitoring, ensuring that the polymerization process remains within the desired operating parameters. If no fault is detected, the system continues its normal operation without intervention.

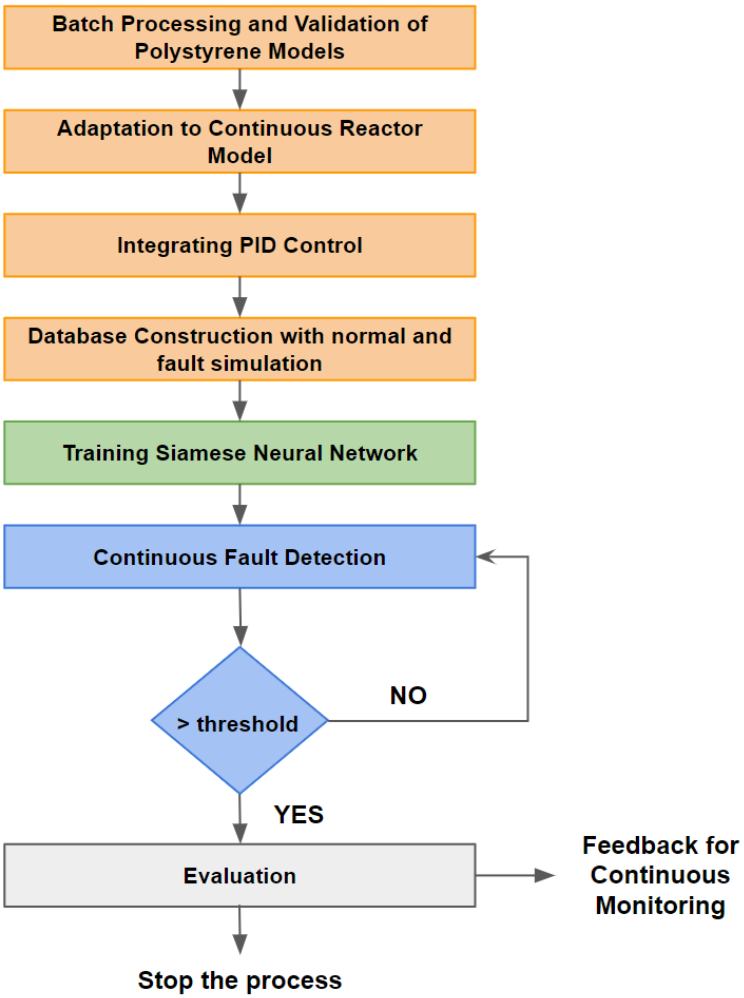

Figure 5.1: Process flowchart.

The flowchart (Figure 5.1) presents the structured methodology from this work for modeling and fault detection in styrene polymerization processes using a combination of phenomenological modeling, PID control integration, Siamese Neural Networks (SNNs) training, and continuous fault detection evaluation.

5.2 Polystyrene Model Simulation

The industrial process modeled involves the polymerization of styrene in a continuous stirred-tank reactor (CSTR) presented in Figure 5.2. The solvent was ethylbenzene and the initiator was tert-Butylperoxy 2-ethylhexyl carbonate (Luperox® TBEC). Within the CSTR, perfect mixing is maintained, allowing for uniform distribution of the substances. A jacket surrounding the reactor enables the temperature control by heating or cooling the system. The output consists of an unreacted monomer, solvent, initiator

and the polymer. Temperature and concentrations are controlled both in the input and output of the reactor, and the temperature is controlled in the jacket.

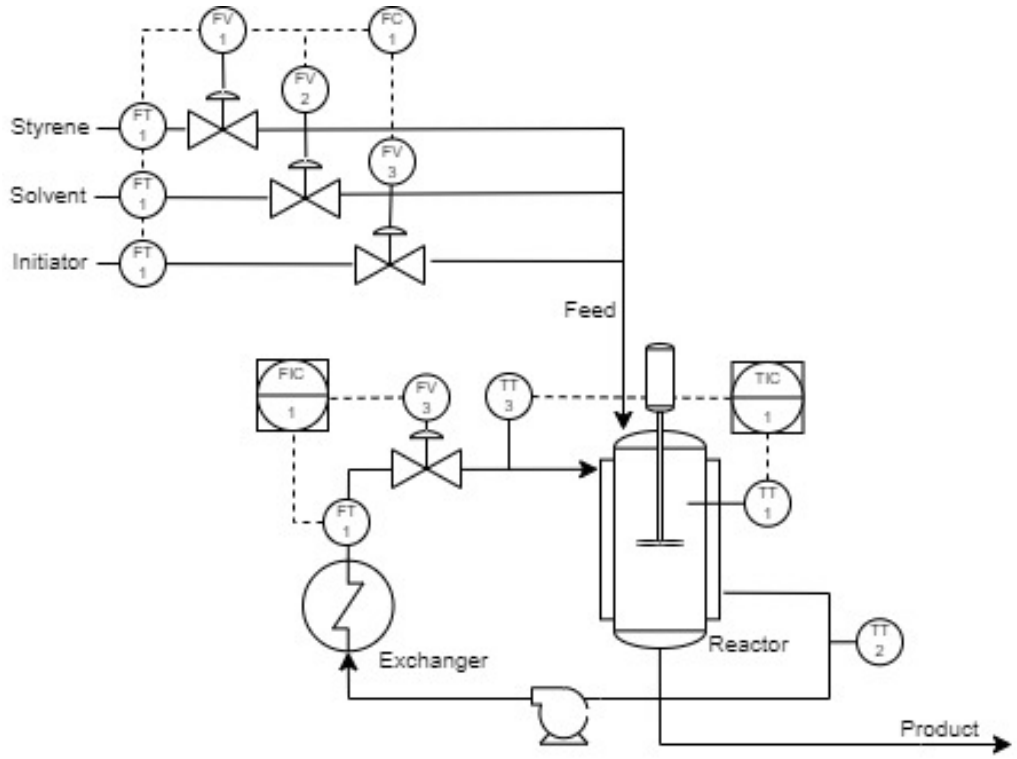

Figure 5.2: Illustration of the polymerization unit.

First, the initial moments are determined based on the known initial conditions and the properties of the monofunctional initiator where the particularly equations are present below:

$$\begin{aligned}\frac{dM}{dt} &= +Q_m\left(\frac{M_0}{V}\right) - Q_{t0}\left(\frac{M}{V}\right) - k_p\left(\frac{M}{V}\right)\left(\frac{\mu_0}{V}\right)V - \\ &\quad - 2k_{dm}\left(\frac{M}{V}\right)^3 V - (k_p + k_{tm})\left(\frac{M}{V}\right)\left(\frac{\mu_0}{V}\right)V\end{aligned} \tag{5-1}$$

$$\frac{dS}{dt} = +Q_s\left(\frac{S_0}{V}\right) - Q_{t0}\left(\frac{S}{V}\right) - k_{ts}\left(\frac{S}{V}\right)\left(\frac{\mu_0}{V}\right)V \tag{5-2}$$

$$\frac{dI}{dt} = +Q_i\left(\frac{I_0}{V}\right) - Q_{t0}\left(\frac{I}{V}\right) - k_d\left(\frac{I}{V}\right)V \tag{5-3}$$

$$\frac{dX}{dt} = +Q_x\left(\frac{X_0}{V}\right) - Q_{t0}\left(\frac{X}{V}\right) \tag{5-4}$$

$$\begin{aligned}\frac{d\lambda_0}{dt} &= -Q_{t0}\left(\frac{\lambda_0}{V}\right) + \frac{1}{2}\left(\frac{k_{tc}}{V}\right)\left(\mu_0^2\right) + \\ &\quad + \left(k_{tm}\left(\frac{M}{V}\right) + k_{tx}\left(\frac{X}{V}\right) + k_{ts}\left(\frac{S}{V}\right)\right)\left(\frac{\mu_0}{V}\right)V\end{aligned} \tag{5-5}$$

$$\begin{aligned}\frac{d\lambda_1}{dt} &= -Q_{t0}\left(\frac{\lambda_1}{V}\right) + \left(\frac{k_{tc}}{V}\right)(\mu_0\mu_1) + \\ &\quad + \left(k_{tm}\left(\frac{M}{V}\right) + k_{tx}\left(\frac{X}{V}\right) + k_{ts}\left(\frac{S}{V}\right)\right)\left(\frac{\mu_1}{V}\right)V\end{aligned} \tag{5-6}$$

$$\begin{aligned}\frac{d\lambda_2}{dt} &= -Q_{t0}\left(\frac{\lambda_2}{V}\right) + \left(\frac{k_{tc}}{V}\right)\left(\mu_0\mu_2 + \mu_1^2\right) + \\ &\quad + \left(k_{tm}\left(\frac{M}{V}\right) + k_{tx}\left(\frac{X}{V}\right) + k_{ts}\left(\frac{S}{V}\right)\right)\left(\frac{\mu_2}{V}\right)V\end{aligned} \tag{5-7}$$

$$\begin{aligned}\frac{dT}{dt} &= +Q_{t0}\left(\frac{T_{c_{\text{input}}} - T}{V}\right) + \left(\frac{\delta H}{Cp}\right)(k_p M\mu_0) - \\ &\quad - \left(\frac{h\text{Area}}{CpV}\right)\left(T - T_{c_{\text{input}}}\right)\end{aligned} \tag{5-8}$$

$$\frac{dTc}{dt} = +Q_c\left(\frac{T_{c_{\text{input}}} - T_{c_{\text{output}}}}{V_c}\right) + \left(\frac{h\text{Area}}{Cp_{\text{Cool}}V_c}\right)\left(T - T_{c_{\text{output}}}\right) \tag{5-9}$$

M_0 : the initial concentration of component M.

S_0 : the initial concentration of component S.

I_0 : the initial concentration of component I.

X_0 : the initial concentration of component X.

Q_s : are the flow rate of components or the cooling jacket

k_s : are the rate constant for a specific reaction

V : the volume of the reactor.

$\lambda_0, \lambda_1, \lambda_2$: are the moments.

T_s : are the temperatures.

δH : the enthalpy of reaction.

Cp : the heat capacity at constant pressure.

Cp_{Cool} : the heat capacity of the coolant at constant pressure.

$h\text{Area}$: the heat transfer coefficient multiplied by the surface area.

V_c : the volume of the coolant.

Then, the numerical solution derived from the moment equations are solved numerically to obtain the moments.

$$\mu_0 = \left(\frac{2fkdI + 2kdm\left(\frac{M}{V}\right)^3 V}{ktc} V \right)^{0.5} \tag{5-10}$$

$$\mu_1 = \frac{2fkdI + 2kdm\left(\frac{M}{V}\right)^3 V + kp\left(\frac{M}{V}\right)\mu_0 + ktm\left(\frac{M}{V}\right)\mu_0 + kts\left(\frac{S}{V}\right)\mu_0 + ktx\left(\frac{X}{V}\right)\mu_0}{ktm\left(\frac{M}{V}\right) + kts\left(\frac{S}{V}\right) + ktx\left(\frac{X}{V}\right) + ktc\left(\frac{\mu_0}{V}\right)} \tag{5-11}$$

$$\mu_2 = \frac{2fkdI + 2kdm\left(\frac{M}{V}\right)^3 V + kp\left(\frac{M}{V}\right)(2\mu_1 + \mu_0) + ktm\left(\frac{M}{V}\right)\mu_0 + kts\left(\frac{S}{V}\right)\mu_0 + ktx\left(\frac{X}{V}\right)\mu_0}{ktm\left(\frac{M}{V}\right) + kts\left(\frac{S}{V}\right) + ktx\left(\frac{X}{V}\right) + ktc\left(\frac{\mu_0}{V}\right)} \tag{5-12}$$

The results are then used to reconstruct the Mw, Mn and Đ, providing insights into the polymer microstructure. The calculated values are validated against experimental data to ensure the accuracy and reliability of the model.

In the study done by Choi et al. (1987), the polymerization model was simplified by assuming equal rate constants for the initiation k_i and propagation k_p process. The modeling as an approximation is based on

the notion that both initiation and propagation steps share similar barriers and, consequently, similar rate constants. This simplification is particularly applicable under conditions where radical initiation exhibits high efficiency and the propagation phase proceeds with considerable rapidity. [20]. The fraction (f) represents the percentage of the initiator that successfully decomposes to generate initiator radicals during the polymerization process. The parameter f is critical for understanding the efficiency of the initiator in the polymerization mechanism and in this case the value f is 0.7 [23]. These and other parameters are shown in Table 5.1

Parameter	Equation	Reference
k_{dm}	$7.326 \times 10^{8} \exp\left(-\frac{27440}{R \times T}\right)$ L^2/mol^2.h	Hui et al. (1972)
k_{tm}	$9.563 \times 10^{9} \exp\left(-\frac{12670}{R \times T}\right)$ L/mol.h	Yoon et al. (1992)
k_{tx}	$5.635 \times 10^{11} \exp\left(-\frac{7067}{R \times T}\right)$ L/mol.h	Moore et al. (1989)
k_{ts}	$7.085 \times 10^{7} \exp\left(-\frac{12523}{R \times T}\right)$ L/mol.h	Moore et al. (1989)
k_{p}	$3.816 \times 10^{10} \exp\left(-\frac{7067}{R \times T}\right)$ L/mol.h	Villalobos et al. (1991)
k_{d}	$2.033 \times 10^{17} \exp\left(-\frac{31500}{R \times T}\right)$ 1/h	Luperox TBEC (ATOFINA)
k_{i}	$k_{\mathrm{i}} = k_{\mathrm{p}}$	Choi et al. (1988)
f	0.7	Scorah et al. (2007)

Table 5.1: Kinetics parameters [8] [19] [20] [21] [22] [23].

The gel effect, also known as auto-acceleration or the Trommsdorf effect, becomes crucial in bulk polymerization at high conversions. This effect arises from diffusion limitations that slow down termination reactions between polymer radical chains. Due to increased viscosity, large polymer chains move less, inhibiting termination and favoring propagation. This leads to longer polymer chains and, eventually, the vitrification effect, where propagation slows down as monomers become scarce.

In the case of styrene polymerization, the gel effect can be described mathematically by considering the rate of chain transfer to polymer and the rate of propagation. The corresponding Equation 4-28 takes into account the concentration of active chains, monomer concentration, and specific rate constants [8].

$$ktc_0 = 60 \cdot 7.53 \times 10^{10} \cdot \exp\left(-\frac{1680}{R \cdot T}\right) \text{ L/mol} \cdot \text{h} \tag{5-13}$$

$$ktc = ktc_0 \cdot \exp\left(-2 \cdot \left(A1 \cdot x + A2 \cdot x^2 + A3 \cdot x^3\right)\right) \text{ L/mol} \cdot \text{h} \tag{5-14}$$

$$A1 = 2.57 - 5.05 \times 10^{-3} \cdot T(K) \tag{5-15}$$

$$A2 = 9.56 - 1.76 \times 10^{-2} \cdot T(K) \tag{5-16}$$

$$A3 = -3.03 + 7.85 \times 10^{-3} \cdot T(K) \tag{5-17}$$

For this system, we admit perfect level control. The outlet volumetric flow rates were calculated by the global mass balance, in order to maintain a constant volume in the reactor.

In the proposed simulations, we do the normal process #1 with a predefined set of initial conditions. Specifically, we establish the reactants' concentrations and the reactor's volume. Styrene, the monomer, along with the solvent and initiator, are introduced into the reactor. Additionally, the reactor and its cooling jacket are set to an initial temperature equal for the reactants.

Data collection is a crucial component of our simulation. We record the parameters at one-second intervals throughout the complete operation cycle. This dataset encompasses a range of variables such as species concentrations and temperatures. By systematically varying the operational conditions, we simulate different normal polymerization scenarios. Moreover, to enhance the robustness of our model, we introduce fault conditions as delineated in an accompanying table.

5.3 Validation

Mathematical models were developed and implemented to describe the trajectories of monomer conversions, average molar masses, and molar mass distributions in continuous systems. The simulation results were validated against experiments conducted by Juliana (2012) on a bench scale in a batch reaction. This validation involved a bench-scale experimental setup and the polystyrene computational simulations to confirm the accuracy of chemical reaction models in polymerization processes under different reactor conditions (R1, R2, and R3), as shown in Table 5.2 [67].

Table 5.2: Summary of Reaction Conditions

Reaction	**R1**	**R2**	**R3**
Conc. iniciator (mmol/L)	-	1.3 ± 0.1	2.5 ± 0.1
Temperature (°C)	393,15	393,15	393,15

Validation was conducted through a quantitative comparison between the experimental data and the simulation results. For conversion rates and molecular weight distributions (Mn and Mw), the model's predictions were evaluated against the empirical data, focusing specifically on the model's ability to replicate trends across all operational scenarios (R1, R2, and R3). The model's adaptability was further assessed by introducing variations, such as the addition of an initiator.

5.4 Failures

During the operation of polystyrene reactors, we frequently encountered different issues that impacted the process. Like reactor leakage, agitator and heat exchangers malfunctions, reactant impurities, data transmition etc. We can simply to several types of faulty: mechanical, software, measurement, and operational. A comprehensive understanding of these faults is imperative for enhancing process optimization and minimizing risks.

Analysis of past accident cases is critical for improving process safety, offering insights into practical occurrences of accidents. Statistical analysis of the accidents reveals that technical (mechanical, software, measurement) failures are the predominant cause, followed by organizational/operational cause. Technical Failures are the most significant contributors to accidents, that includes piping system failures (poor layout, wrong specification, dead ends, poor installation, inadequate hot bolting, and blockages), contamination of process stream (impurities, by-products, and external contamination), inappropriate selection of construction material (errors in design leading to physical and mechanical problems), mass transfer issues and corrosion/erosion (poor mixing, excessive charging, and operational scenarios), heat transfer problems (loss of cooling, incorrect heating methods, and problems with thermal expansion). Organizational Causes are subdivided into: management/procedural faults, knowledge-based errors and storage/handling of chemicals. Managerial errors such as inadequate hazard recognition, wrong policies, personal factors like incompetence, ignorance of technological advancements and lack of knowledge sharing [66].

Mechanical failures in reactor processes, manifest in various forms, each contributing to significant disruptions in operations. Problems with valves, such as leaks or blockages, interfere with the flow of chemicals and can be safety hazards. If agitators are not properly working, the mixing becomes uneven, affecting the quality of the end product. Heat exchangers,

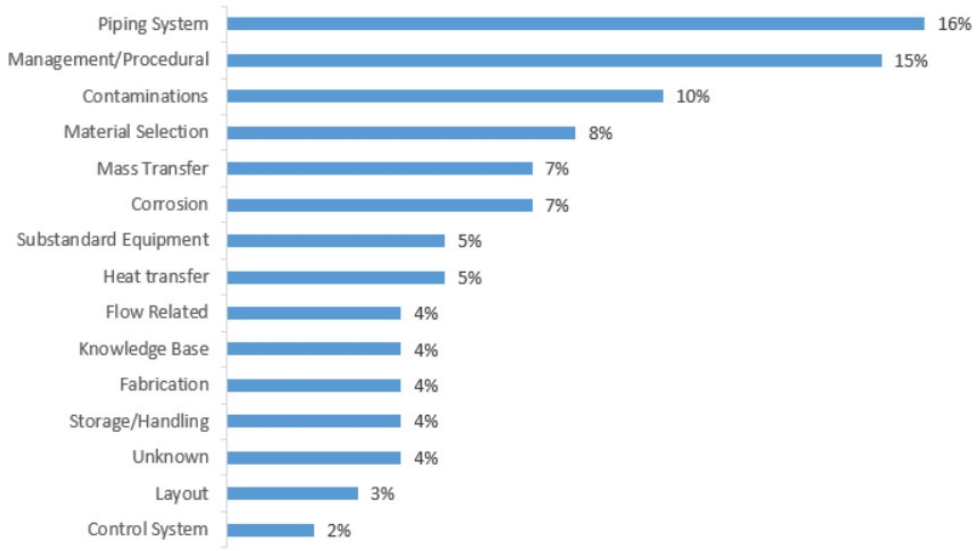

Figure 5.3: Causes of accident [66].

responsible for maintaining the right reaction temperature, can fail due to clogging or leaks, changing reaction speeds and altering the product. Sealing problems in the reactor structure result in material loss and contamination. Also, issues with pumps, which control the flow of reactants and products, can upset the balance of the process. Fixing these mechanical problems is crucial for maintaining reactor efficiency and ensuring the quality of the product.[66]

Measurement inaccuracies are a common issue affecting the polymerization process. Inaccurate monomer feed measurements led to variations in the polymer's properties. Similarly, temperature and pressure mismeasurements, due to faulty sensors or calibration errors, often resulted in uncontrolled reaction rates, posing potential safety risks. These inaccuracies not only affect product quality but can also lead to inefficient use of resources and increased production costs [66].

Software and system failures posed significant challenges in maintaining optimal reactor conditions. Sensor failures, particularly those related to temperature and pressure, can make the control systems lead to an incorrect value, resulting in inappropriate reaction environments. Regular system checks, timely maintenance, and redundancy in critical sensors could be crucial in mitigating these risks, ensuring both safety and consistency in the process [66].

Operational challenges in polystyrene reactors impacted the polymerization process. These included human errors due to training gaps, deviations in process control like changing reactant ratios and monitoring, and

variability from raw material inconsistencies like choosing the wrong initiator. Equipment maintenance lapses and poor emergency responses led to material waste and restart complexities. Additionally, non-compliance with safety regulations and communication breakdowns among teams further exacerbated operational faults [66].

This work aims to identify and analyze faults in chemical processes, focusing on mechanical failures like heat exchanger issues and operational challenges from changing reactant ratios.

Was done a series of normal and fault conditions to examine the dynamics of the polymerization process. The normal n°1 condition serve as a baseline from the others normal conditions from which we alter only the initial states: temperature and Q_m/Q_s ratio. For the fault conditions the initial condition that was the normal n°1 process. In a Continuous Stirred-Tank Reactor (CSTR), process variables change initially but stabilize at a steady state due to the reactor's continuous mixing and flow. The fault introduced at t = 17000, after the system's response these initial disturbances.

We simulate a range of faults, each designed to test the resilience and adaptability of PID control and then the model fault classification. These include step change or slow drift changes in monomer, solvent and initiator flow rate, reactor's cooling temperature and kinetic rate constants changing progressively over time.

Fault n°1 is a step change of increase in the Q_m/Q_s ratio, removing all solvent inflow and adding the same flow to the monomer flow, while fault n°2 involved a step decrease in the in the Q_m/Q_s ratio from 18.4 to 5.5. Faults n°3 and n°4 were characterized by an step change in the initiator flow rate (Q_i) from 0,00012 L/h to 0,010 L/h and 0,020 L/h, respectively. Fault n°5 is a step change in the coolant temperature (T_c) to 405k. Fault n°6 increased the impurity concentration by 140.0 % also in a step change. Faults n°7 and n°8 were distinguished by slow drifts in the reaction rate constants, with k_{tm} and k_{tc} increasing 0.10% per second for both. The specifics and implications of these faults, including their impact on the polymerization process, are further elucidated in Table 5.3, which provides detailed variations and corresponding data for reference.

The controlled variables Tc0, Qm, Qs, Qi, and Qx where the model changed the values and applied the fault, obviously, the values changes but this changes are not recorded so the model do not know that the fault occurred. The uncontrolled variables (time, x, Mn, Mw, T, Tc, D,

Table 5.3: Simulation operational conditions

#	Q_m/Q_s ratio	**T (K)**	**Description**	**Variation type**
Normal 1	18.07	402.15	Standard	Step
Normal 2	18.07	415.05	Increase of temperature	Step
Normal 3	18.07	389.25	Decrease of temperature	Step
Normal 4	950.2	415.05	Increase of Q_m/Q_s ratio	Step
Normal 5	0.730	415.05	Decrease of Q_m/Q_s ratio	Step
Fault 1	18.07	402.15	Increase of Q_m/Q_s ratio	Step
Fault 2	18.07	402.15	Decrease of Q_m/Q_s ratio	Step
Fault 3	18.07	402.15	Small increase of Q_i rate	Step
Fault 4	18.07	402.15	Big increase of Q_i rate	Step
Fault 5	18.07	402.15	Reactor Cooling Fluid	Step
Fault 6	18.07	402.15	Increase of impurity concentration	Step
Fault 7	18.07	402.15	Reaction k_tm Slow Drift	Gradual shift
Fault 8	18.07	402.15	Reaction k_tc Slow Drift	Gradual shift

eta, pr, M, S, I, X, λ_0, λ_1, λ_2),are sensitive to these changes will show alterations in response to the faults. The SNN model is designed to detect these indirect effects in these uncontrolled fault, signaling the occurrence of a fault through deviations in these uncontrolled variables. This approach allows us to evaluate the model's ability to recognize and classify faults, rather than direct feedback from the controlled variables. The controlled variables (Tc0, Qm, Qs, Qi, Qx) where the model get the fault, obviosly, changes values but this changes is not recored so the model dont know that the fault occured.

5.5 Data preprocessing

The high-dimensional time-series data, which comprises 2500 data records per second across all variables (faultNumber, simulationRun, sample, time, x, Mn, Mw, T, Tc, Tc0, D, eta, pr, M, S, I, X, λ_0, λ_1, λ_2, Qm, Qs, Qi, Qx). To train time series data is necessary to split the data in parts, each part is referred as window, and then each window is use to train the model. Windowing is a critical initial step in the data preprocessing for SNNs. In this project, the selected window size was 10 seconds with 1 second step to each other.

Twin neural networks (SNNs) need two data inputs from the training dataset (Figure 5.4). Each pair of data segments with a 10-second window is fed into the SNN for processing. One epoch is one complete cycle of the entire training dataset. During an epoch, the network's weights are updated to minimize loss by learning to distinguish between similar and dissimilar

data pairs. To train a computer program to recognize patterns, we feed it data in batches instead of all at once. This makes the computer's job easier and faster, just like doing a big task step by step. This also helps the network to avoid local minimal and potentially find better generalizing solutions [51].

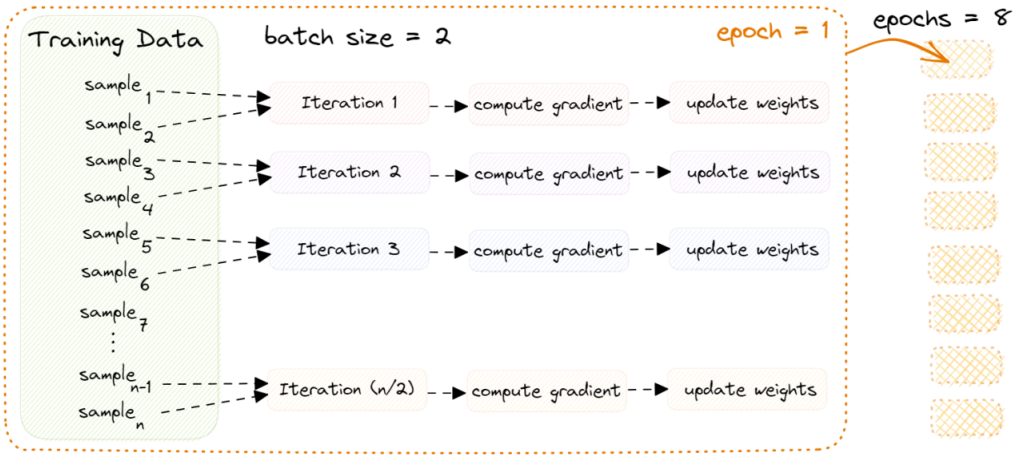

Figure 5.4: Traning processing.

Each entering is paired with two windows of 10 seconds as aforementioned. Similar pairs (anomalous-anomalous or normal-normal) were labeled as 0, or dissimilar pairs (anomalous-normal or normal-anomalous) were labeled as 1 as Figure 5.5. In a chemical process context, distinguishing normal from potentially faulty data is vital. SNNs are adept at this task due to their differential learning approach. By training SNNs to compare normal operational data with anomalous data, they learn to detect deviations indicative of faults. This comparison is essential for early anomaly detection, critical to ensuring safety and efficiency.

To streamline the training set and enhance the model's robustness, we exclude pairs where both series are faulty. Excluding anomalous-anomalous data pairs from training focuses the model's learning on identifying deviations from normality. This targeted training approach ensures the SNN remains sensitive to normal-anomalous faults. In the context of the chemical industry, it is necessary to compare normal data with potential faulty data. The aim is that the SNN becomes a precise tool that monitor using the real-time data and normal data.

The division of the database is the next phase in the preprocessing. One significant advantage of this project is the ability to create a large-sized dataset. This enables us to allocate a substantial portion of the total data for validation and testing. Sixty percent of the data were allocated for training/validation, and the remaining 40% was reserved for testing.

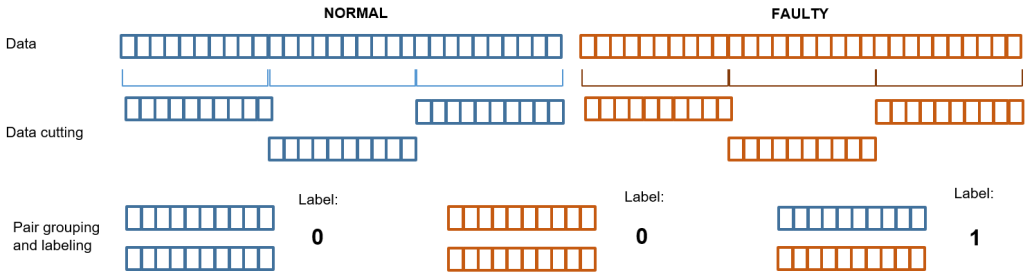

Figure 5.5: Data preprocessing.

From the training/validation data, 30 % was saved for validation data and the remaining 70 % composed the training dataset. This division allows for a robust training and validation process, which is essential for the ANN learning process.

Normalization in neural networks is a technique that is used to scale the inputs. This work used the scaled median absolute deviation (SMAD) method, which is effective for scale features that may contain outliers or huge noise, such as a reactor can have. SMAD normalizes the data using the median and the median absolute deviation, making the normalized data less sensitive to extreme values. First, it is necessary to determine the median of the original dataset to normalize. Then, subtract the median from each data record, centering your data around zero. After that, calculate the median of these absolute values and multiply the new median by the scale factor, which is a fixed number that depends on the distribution (approximately 1.4826). The output represents the SMAD value. Finally, divide each data record by the SMAD value. The factor 1.4826 is derived from the inverse cumulative distribution function of a standard normal distribution, representing the ratio between the standard deviation and the MAD for normally distributed data.

$$x_i'' = \frac{x_i - \text{Med}(x)}{\phi \times \text{MAD}(x)}, \quad \text{where MAD}(x) = \text{median}(|x_i - \text{Med}(x)|) \qquad (5\text{-}18)$$

where x_i'' is the normalized value, $\text{Med}(x)$ is the median of the variable, and $\text{MAD}(x)$ is the median absolute deviation. The constant scale factor (ϕ) is approximately 1.4826.

5.6 Siamese neural network architecture and training

Siamese Neural Network is employed to distinguish between normal and potentially faulty operational data in chemical processes. This distinc-

tion is crucial for early detection of anomalies, ensuring both safety and efficiency. The SNN learns by comparing what's normal to what's not. We train it with normal data and data with faults, smaking it proficient in identifying deviations that are indicative of faults. To enhance the robustness and effectiveness of the model, we don't use pairs of data where both are faulty. This exclusion focuses the SNN's learning on recognizing faults from normal operational patterns. This way, the SNN focuses on learning what's normal and what's not.

To mitigate the risk of overfitting, a dropout layer is strategically introduced between intermediate dense layers. This addition serves to randomize neuron activation, thereby fostering multiple prediction pathways and enhancing the model's generalization capabilities. The model utilizes contrastive loss as its loss function, which is designed to optimize the feature space in a manner that ensures that similar instances are brought closer together, while dissimilar ones are distanced.

Backpropagation in SNN is similar to that in conventional feedforward from NNs but with a focus on optimizing the specialized loss function. The Adam optimizer is applied for its ability to adjust learning rates adaptively, enhancing the efficiency of model optimization. The SNN training aims to change the weights and biases so that the network effectively learns to differentiate between similar and dissimilar pairs of inputs. The process begins with a forward pass, where a pair is passed through the twin networks of the SNN to obtain the vector. These are then used to calculate the loss using the contrastive loss.

The network was built using the Keras Python library since, through this tool, it is possible to create two identical subnetworks that receive separate inputs. These subnetworks comprise neurons with long short-term memory (LSTM) units and utilize the ReLU activation function in hidden layers.

The model was evaluated using accuracy. Different training phases were performed for each combination of faulty and hyperparameters, as shown in Table 6.7. Early stopping was utilized as a form of regularization to select the optimal neural network configuration without overfitting. All models were tested with different neural network configurations, batch sizes (10, 50, 200) and learning rates (0.02, 0.01).

In the context of fault detection, the SNN was tasked with classifying system states as either faulty or non-faulty, represented as binary outputs: 0 for non-faulty and 1 for faulty conditions. This binary classification

approach simplifies the decision-making process, focusing the network on distinguishing between two distinct classes of system states. The output of the SNN, following the calculation of Euclidean distance between feature vectors of a pair of inputs, is processed through a sigmoid activation function. This function maps the distance to a probability value between 0 and 1. A threshold, typically set at 0.5, is employed to categorize the output as either 0 or 1, based on the calculated probability.

This binary classification mechanism in SNNs for fault detection facilitates a more direct and interpretable decision-making process. It reduces the output to a simple binary decision, easing the integration into automated control systems for real-time fault monitoring and diagnosis. Such an approach, combined with the robustness and efficiency of SNNs, positions this methodology as an effective tool in the realm of fault detection and diagnosis, particularly in complex systems where early and accurate fault detection is imperative.

6
Results and Discussion

This chapter aims to show how the proposed model for the process was implemented and solved, as well as how the model simulates the operational conditions of the reactors and the properties of the produced polymer.

6.0.1
Polystyrene Model Simulation

6.0.1.1
Validation

This validation was done through experimental data conducted on a bench scale, as from Juliana (2012), offering foundation for our simulation results [67]. Acknowledging that experimental work, especially on a bench scale, inherently involves the potential for both systematic and random errors is crucial. These errors can arise from a variety of factors, such as changes in ambient conditions, inaccuracies in measurement instruments, and the unpredictable nature of chemical reactions.

Figure 6.3, illustrating the validation of conversion rates under different reactor conditions (R1, R2, and R3), the dashed line representing ±3 % of the model conversion, demonstrates the model's ability to replicate experimental trends across all operational scenarios, including those with (R1) and without the initiator (R2 and R3).

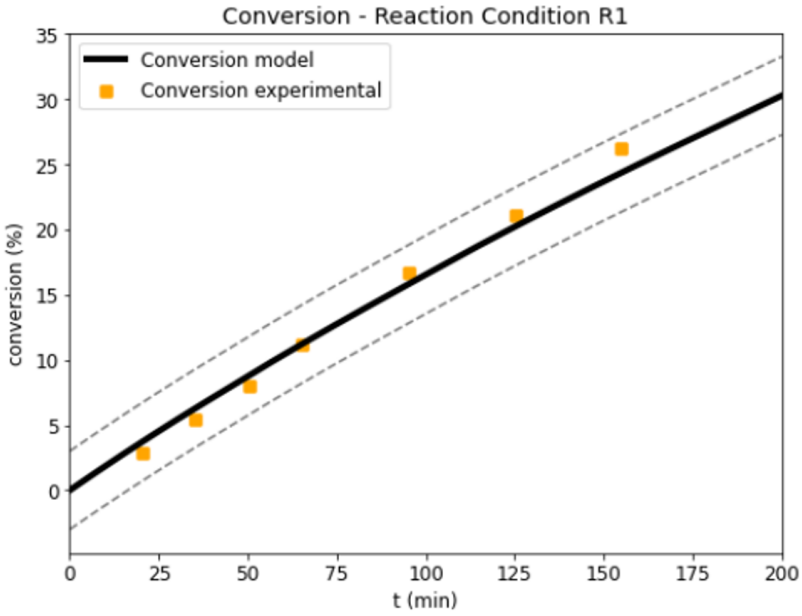

Figure 6.1: Validation of conversion rates for R1 conditions.

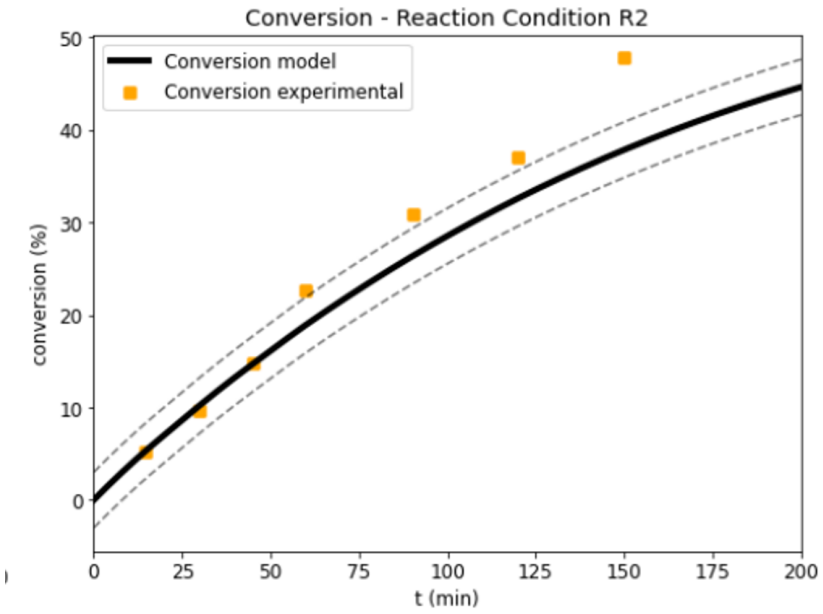

Figure 6.2: Validation of conversion rates for R2 conditions.

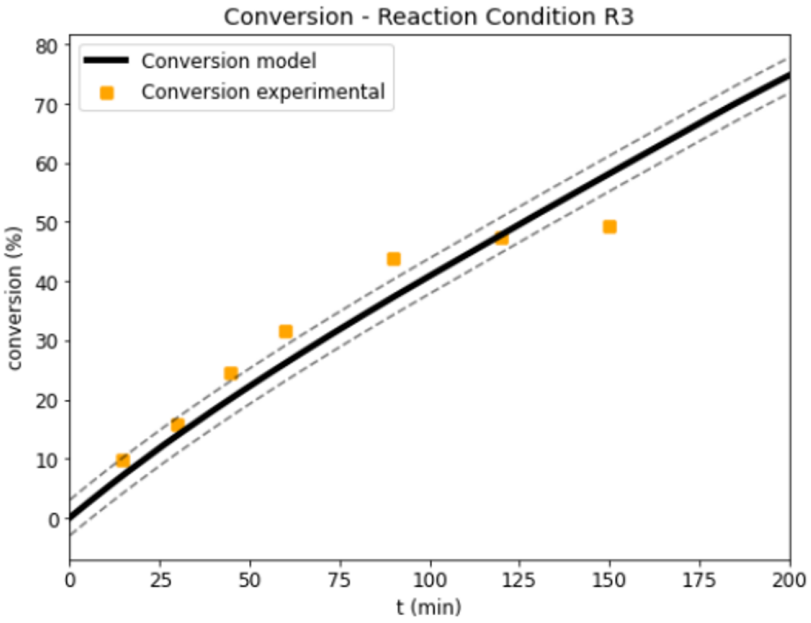

Figure 6.3: Validation of conversion rates for R3 conditions.

Similarly, Figure 6.6 validates Mn and Mw for the same reactor conditions, with the dashed line representing ±20000 g/mol of the model conversion. Although the model exhibited a deviation for condition R1 in terms of Mw, the addition of an initiator in conditions R2 and R3 significantly improved accuracy. This deviation is less concerning in the continuous model, where initiation is a consistent process, underscoring the model's adaptability in reflecting experimental realities.

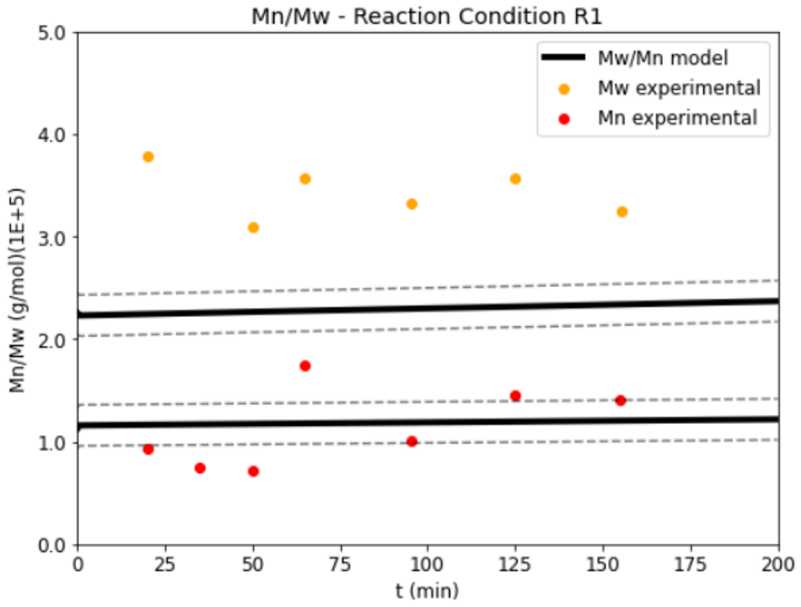

Figure 6.4: Validation of Mn and Mw for R1 conditions.

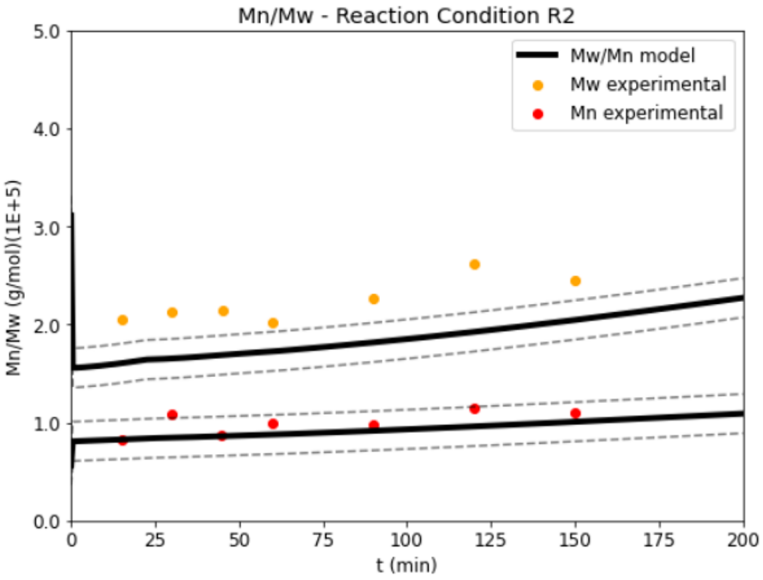

Figure 6.5: Validation of Mn and Mw for R2 conditions.

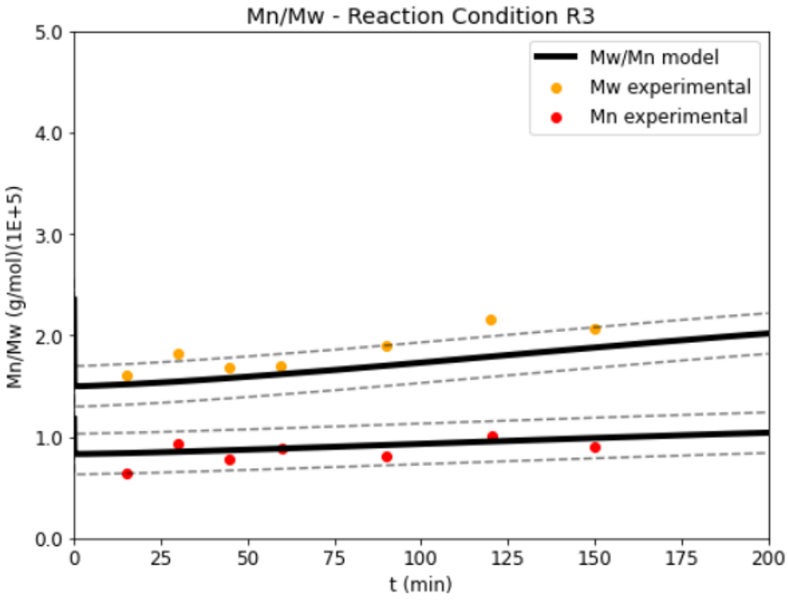

Figure 6.6: Validation of Mn and Mw for R3 conditions.

This alignment is crucial for understanding how changes in reactor conditions affect conversion rates, offering insights into optimizing the polymerization process. The validation of these models against experimental data confirms their accuracy, paving the way for application in the continuous stirred tank reactor operation, which differs only by adding the flow rate of reactants and the cooling jacket for temperature control.

6.0.1.2 CSTR Simulation

Simulations conducted within the CSTR provided significant insights into mass and energy balance. The granular-level analysis allowed for the detailed capture of moment transformations and polymerization kinetics. In our simulations, for the normal #1 process, at initial time (t = 0 s) the styrene concentration was set to 7.352 mol/L, the solvent concentration to 0.380 mol/L, the initiator concentration to 0.00087 mol/L and the volume to 0.480 L. The initial temperature for the reactor and the jacket was 402.15 K. We collect 2500 data records per run that is and each record is equal to 1 second, covering the entire operational cycle. The same interval of time was used for all simulations. Various normal polymerization scenarios were simulated by changing these operational conditions and other fault process were applied as described in the Table 5.3. For each time interval of 1 second, the species concentrations (M, S, I and X), flows (Qm, Qs, Qi and Qx), reactor temperature (T), jacket temperature (Tc and Tc0), MoM indicators (λ_0, λ_1, λ_2) Mn , Mw, faultNumber, simulationRun, sample were recorded and the resultant dataset was used to train the SNN model.

In each normal condition, the variations are detailed in the table, only the initial conditions are altered. For all fault simulations, the process begins as like Normal #1, and at $t = 1700$ s, the fault is introduced. In fault #1, there is a step increase in the Q_m/Q_s ratio, shifting from a ratio of 20 in relation to a scenario where Q_s concentration approaches zero. In fault #2, the reverse occurs, and the Q_m/Q_s ratio adjusts to 1. For fault #3, the initiator flow rate (Q_i) increases from 0.000115 L/h to 0.050 L/h, and in fault #4, Q_i increase to 0.20 L/h. In fault #5, the reactor cooling fluid temperature abruptly changes to 405K. In fault #6, an impurity concentration increases to a constant 0.0001 mol/L in the reactor. Lastly, faults #7 and #8 involve gradual changes in the kinetic rate constants k_tm and k_tc, respectively, which continue until the end of the simulation.

In Figure 6.7, the conversion trajectory exhibits two distinct regimes: an initial rapid ascent, followed by a stable plateau. This behavior is consistent with the theoretical expectations for a CSTR, where the rate of reaction converges towards a steady state after an initial transient phase.

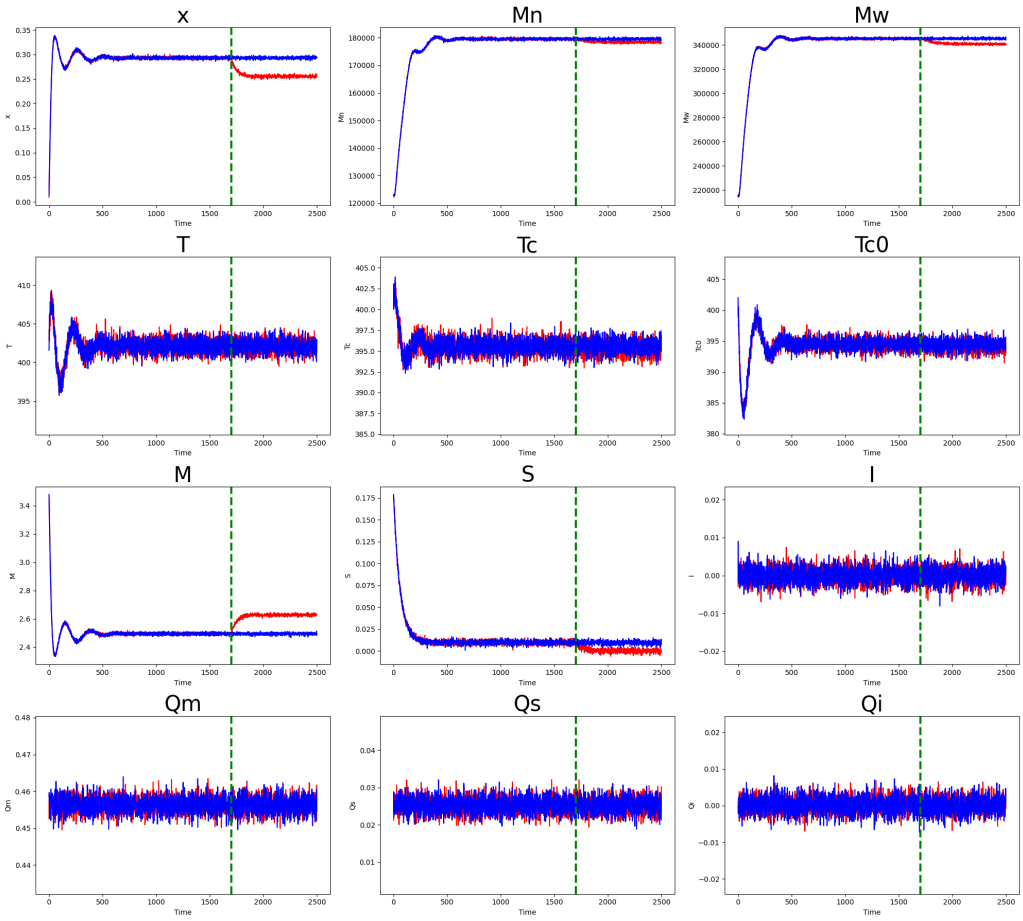

Figure 6.7: First normal process (blue), first faulty process (red) and start of the faulty (green).

The PID control algorithm effectively controls the thermal regulation of the reactor temperature (T) and the outlet temperatures (Tc) and thermal jacket's inlet (Tc0), ensuring the temperature remains in a predetermined operational value. It is important to show that the computational model employed is an approximation rather than an authentic reactor system. The simulation predicts the instantaneous introduction of reactants at t = 0, starting the polymerization. As an exothermic reaction, the polymerization makes the reactor temperature spike in the initial phase. In the first moments, the inlet temperature started to drop as the control began to try to control the reactor temperature. The temperature oscillates until the normalization.

When a system previously in a steady state experiences a sudden increase in the flow rate of monomer Qm, the immediate effect is to increase the total amount of monomer entering the reactor. However, since conversion is calculated based on the ratio of reacted monomer to the total monomer fed (or present) at any time, the immediate effect could indeed be a decrease in

conversion percentage. This decrease occurs because the increase in monomer feed momentarily surpasses the system's capacity to convert this additional monomer at the same rate, thus lowering the conversion percentage. This situation, however, impacts the conversion rate, which is defined by the ratio of the reacted monomer to the total monomer input at any given moment. Consequently, an instantaneous effect might be a reduction in the conversion percentage. This decline is attributed to the system's temporary inability to process the additional monomer influx at the same rate, thereby diminishing the conversion percentage. If the monomer feed rate adjustment is not complemented by a corresponding increase in temperature and initiation concentration, the conversion remains lower at the new steady state. This lower conversion rate slows the formation of new polymer chains, reducing Mn and leading to a decrease in the Mw, due to less efficient polymerization and the formation of fewer long chains.

In the instance of faulty two, as depicted in Figure 6.8, it's observed that the ratio of Qm/Qs reduces. This reduction is attributed to the decreased monomer entry rate (Qm reduction), which in turn slightly lowers the temperature due to less exothermic reactions. After an initial fast increase in Mn and Mw, a significant decline in these values is noted. The decrease in monomer feed rate enhances the conversion rate, which might lead one to expect an increase in Mn and Mw as the monomer to polymer reaction progresses more fully. However, at higher conversion rates, the chances of chain termination reactions, like combination or disproportionation, may rise due to increased concentrations of polymer radicals. An escalated rate of termination compared to chain initiation and growth could result in a predominance of shorter chains, thereby lowering Mn and Mw. Additionally, the increase in solvent-to-monomer ratio can influence the solubility of both the monomer and growing polymer chains. This change has the potential to affect polymerization kinetics and molecular weight distribution, where a higher solvent ratio might reduce polymer concentration and may promote termination over propagation.

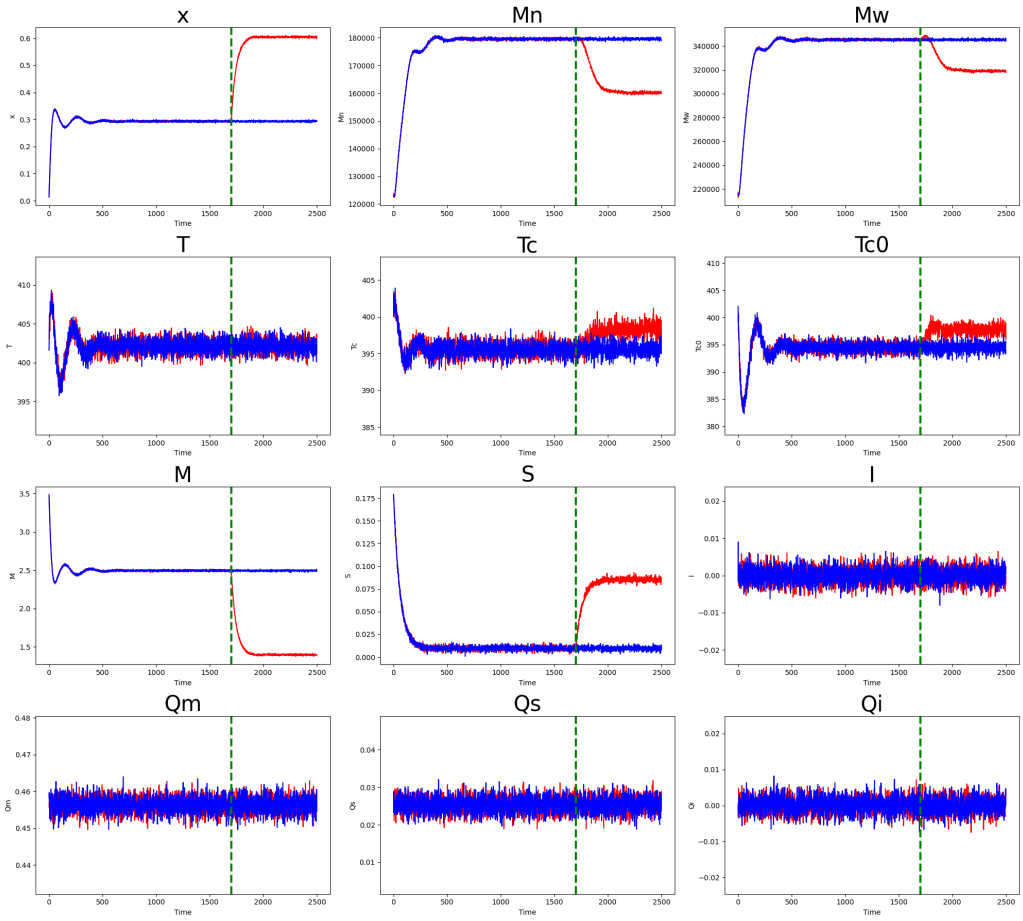

Figure 6.8: First normal process (blue), second faulty process (red) and start of the faulty (green).

Faulty three and four correspond to increase of initiator. Faulty three as in Figure 6.9 is possible to see the response from a higher conversion followed by a slightly increase of reactor temperature that lead to decrease for inlet jacket temperature to control that increase. This accelerates monomer consumption and increases the termination and chain transfer reactions. These effects contribute to a higher number of shorter polymer chains being formed, resulting in decreased average molecular weights.

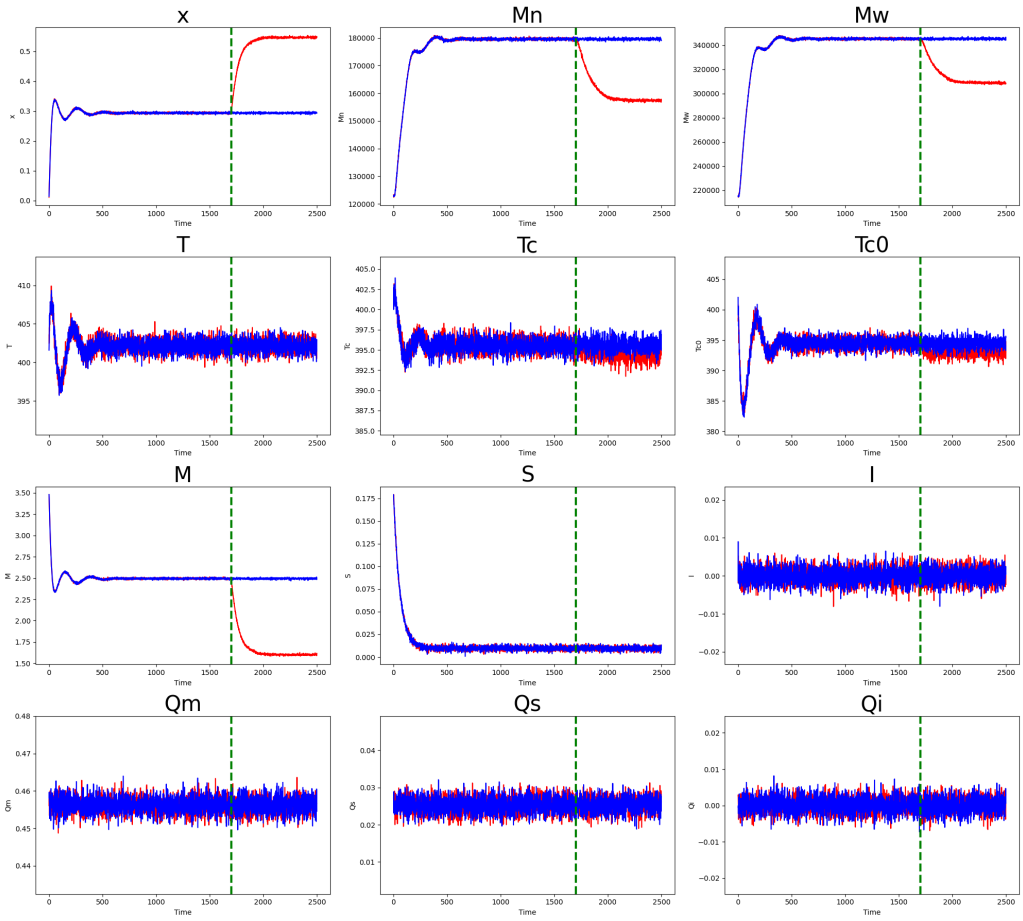

Figure 6.9: First normal process (blue), four faulty process (red) and start of the faulty (green).

As in Figure 6.10 the temperature changes as a step change in the inlet coolant temperature (Tc0) to 405K the initial result is the consequential increase of reactor temperature (T) and outlet coolant temperature (T) followed by higher conversion and Lowe monomer concentration and Mn and Mw concentration. The impact on the molecular weights of the polymer (Mn and Mw) is more nuanced. While higher temperatures can facilitate longer polymer chains by allowing for more extensive propagation, they also increase the rates of termination and chain transfer reactions. If termination and chain transfer become significantly more likely than chain growth, the resulting polymers will be of shorter average length. Thus, despite higher conversion rates, the average molecular weights (Mn and Mw) may decrease. This effect is attributed to the higher probability of premature chain termination or transfer to monomers or solvent molecules, limiting the extent of chain elongation.

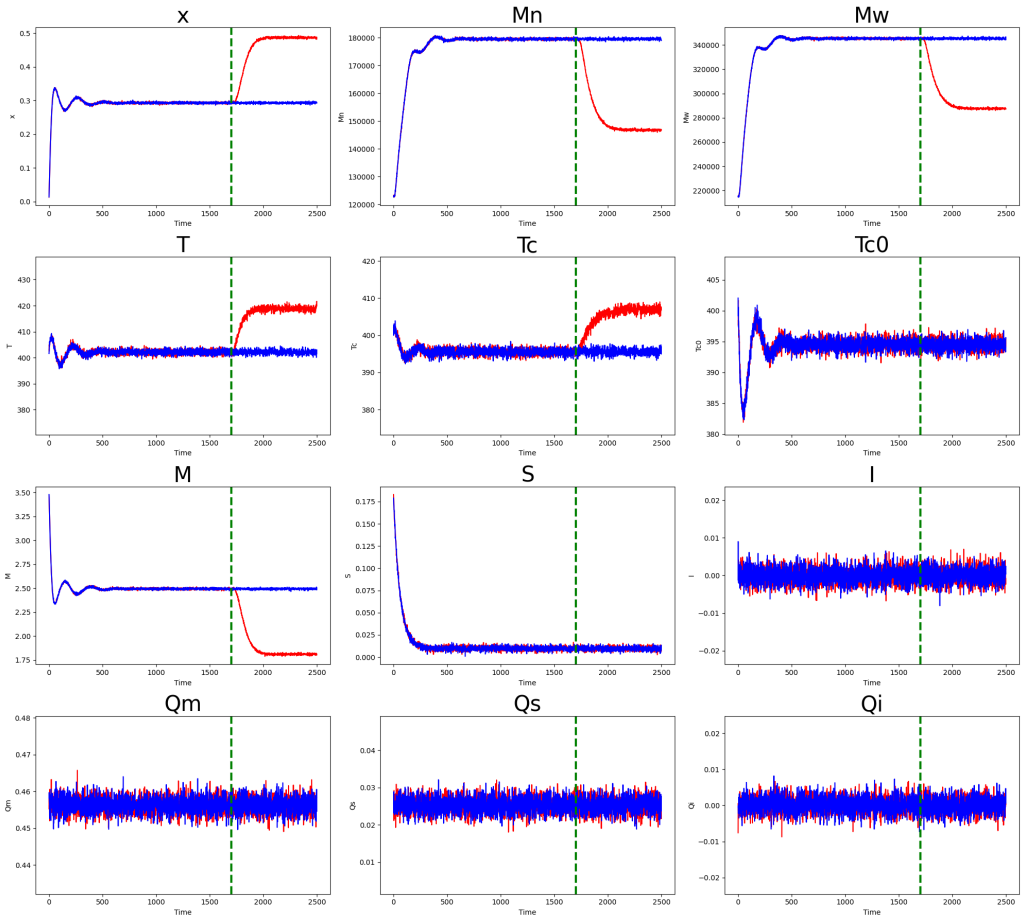

Figure 6.10: First normal process (blue), five faulty process (red) and start of the faulty (green).

Faulty six (Figure 6.11) correspond for an increase to unknown impurity in the system. As its correlated just with chain transfer the main influence is in the molecular weight and distribution of the polymer. The result is the termination of the growth of the current chain and the initiation of a new chain from the point of transfer. This dynamic decreases both Mn and Mw, as the calculations for these averages are sensitive to the presence of lower molecular weight species in the mix.

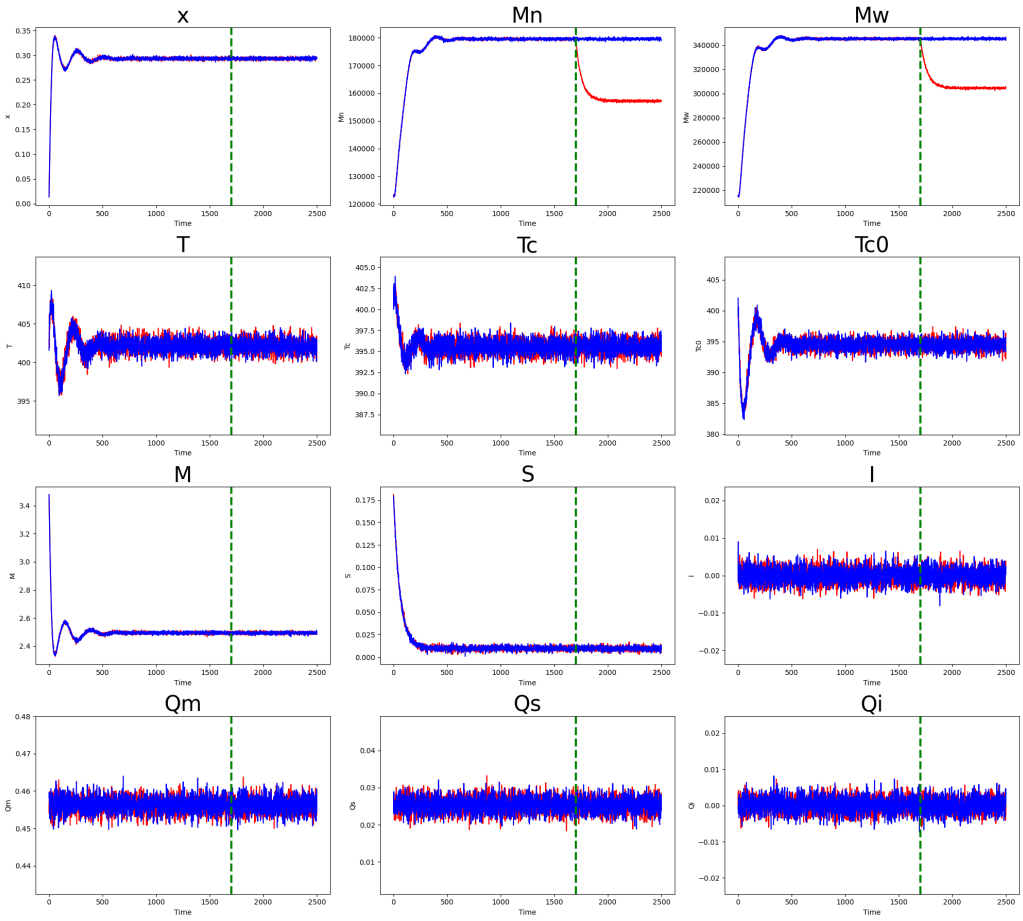

Figure 6.11: First normal process (blue), six faulty process (red) and start of the faulty (green).

This effect occurs in the same why with faulty seven (Figure 6.12) with gradual increase in the ktm. This also leads to a higher frequency of chain termination and initiation events. This dynamic significantly contributes to a reduction in both the number average molecular weight and the weight average molecular weight.

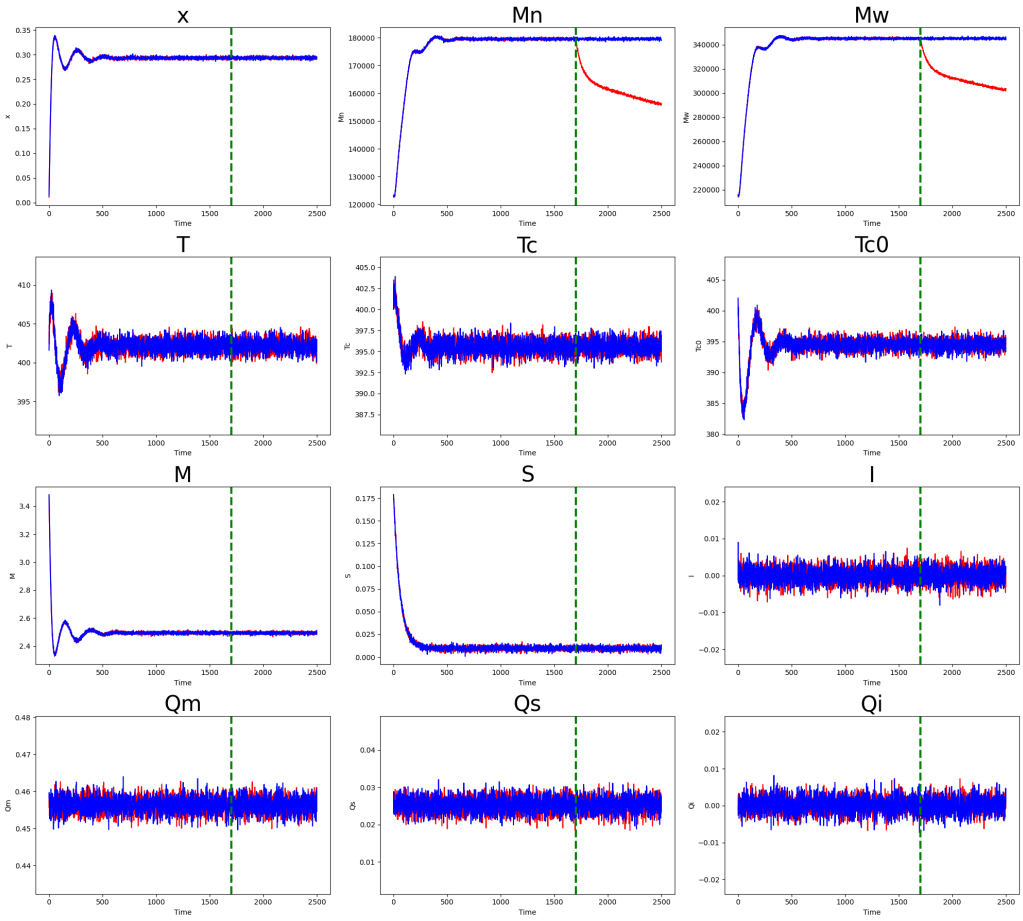

Figure 6.12: First normal process (blue), seven faulty process (red) and start of the faulty (green).

Finally, the last faulty is an gradual increase in ktc, as can see in Figure 6.13. The constant ktc represents the rate of termination by combination in polymerization reactions, where two active polymeric radicals combine to form a longer polymer chain. The rate of this termination mechanism can have significant implications on the polymerization process, particularly when considering the gel effect, which typically occurs at higher conversions and significantly impacts the reaction kinetics and polymer properties.

The gel effect can also lead to a phenomenon known as autoacceleration or the Trommsdorff Effect, where the reaction rate actually increases at high conversions due to the confinement of radicals within the viscous medium, leading to increased probability of propagation over termination. As termination by combination becomes more prevalent, fewer radicals are available to propagate, which could, in theory, lead to a slowdown in the rate of conversion increase. However, the overall conversion could still be high due to the autoacceleration effect, as the reaction nears completion.

While termination by combination generally leads to the formation of longer polymer chains, the overall impact of an increased ktc under the influence of the gel effect. The reduced number of active chains can lower Mn due to fewer chains growing to significant lengths before termination. Mw might also decrease if the formation of very long chains by combination is less than expected due to the reduced mobility and reactivity of chains in a highly viscous medium.

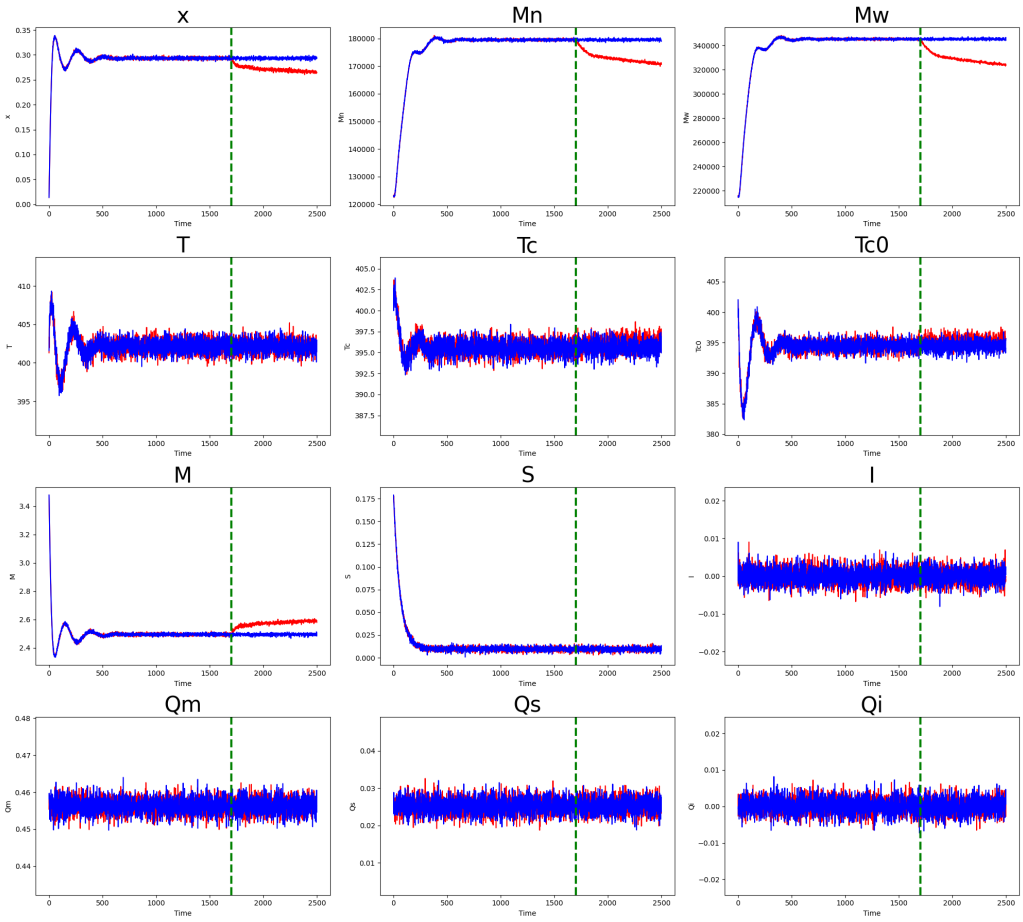

Figure 6.13: First normal process (blue), eight faulty process (red) and start of the faulty (green).

A slower overall polymerization rate due to decreased effective termination by combination at high conversions leads to less monomer being consumed over time consequently, the reactor temperature drop.

6.1 Siamese Neural Network

In this study, aim to prove the efficiency of SNN for the purpose of early fault detection in the polymerization process of styrene. The focus on SNNs arises from their ability to learn from data comparisons, making them

particularly suited for fault detection tasks where the goal is to differentiate between normal operation and abnormal conditions. This section details the results obtained from employing SNNs, using Multilayer Perceptron (MLP) as a benchmark for comparison and LSTM networks as the primary model to encapsulate the dynamic behavior of the chemical process under study.

6.1.1 Benchmark

To validate the effectiveness of LSTM-based model in identifying faults within the chemical polymerization process. The choice of MLP as a benchmark is strategic, allowing us to assess the LSTM-based SNN's performance against a simpler, yet analogous neural network architecture. This comparison is crucial for demonstrating the superior ability of LSTM models to manage the dynamic and temporal complexities inherent in chemical processes, which is often challenging for traditional MLP models. The incorporation of LSTM is based on its recurrent nature, which is particularly beneficial for predicting faults in the chemical industry by capturing temporal dependencies and patterns in process data more effectively than MLP. Results from MLP will appear in the following sections.

6.1.2 Parameter Setting

The journey to optimal model performance began with a meticulous grid search for the perfect combination of hyperparameters. This iterative process was crucial to ensure that the best SNNs models are achived. For the experimental setup, a series of trials were conducted to find the best settings for learning rate, batch size, and the number of epochs among other parameters and results can see in Table 6.1.

Table 6.1: Simulation hyperparams and results.

Schema	Settings Options	Best Settings Options
learning rate	10-3, 10-2	10-2
loss margin	0.2, 0.5, 1, 2	1
batch size	100, 200, 500	100
patience early stop	2, 5, 10	10
patience learning rate	2, 3	2
window size	2, 5, 10, 20	10
moving average	1, 5, 10	1

The net cells architecture was 10 units and accuracy averaged from the 5-fold stratified cross-validation was used to choose the best parameters. The computational setup powered by a 1 GHz CPU and equipped with 8GB RAM, for each model was used 3 kfold and the total model of the grid seach was 3888 total models, culminating in a total training duration around 1080 minutes. Using all database reached 97.55 % of accuracy and the configuration that emerged victorious from this grid search as can see in Table 6.1.

6.1.3 Siamese Neurak Network with LSTM units

The effectiveness of SNNs in fault detection can be significantly impacted by the quality and nature of the database used for training and validation. Using a database containing multiple or a single type of faulty data can lead to good generalization or overfitting. In other words, the SNN model may become too specialized in identifying specific types of faults or fail to generalize well to new, unseen faults. That is why in this section of the results, we will show four types of training data:

- Schema 1: All Normal Databases with All Faulty Databases
- Schema 2: One Normal Database with All Faulty Databases
- Schema 3: All Normal Databases with Each Individual Faulty Databases

The aim is to understand how the SNNs model behaves with different types of datasets, for example, if just one normal process is not good in generalization compared to the ten normal databases, and if doing the same schema for each faulty, the model will be more accurate in detecting the faulty. The training on such an unbalanced dataset could potentially result in a model with a bias toward fault detection, likely leading to a high false positive rate. However, a more balanced dataset would lead to a more generalized model.

Scheme 1 was selected as the initial framework for training, validating and testing the SNNs to determine the most effective model architecture. The experiment was conducted across various neural network configurations, as shown in Table 6.7 with the hyperparams as in 6.1.

6.2 Schema 1: All Normal Databases with All Faulty Databases

In our study, Scheme 1, which integrated all normal and faulty databases for training, was identified as the foundational model. This approach aimed to create a versatile model capable of recognizing multiple fault types in a unified framework. The extensive training under this scheme was hypothesized to yield a model with broad detection capabilities, as evidenced by the results showing high accuracy and MCC values across various network configurations.

Table 6.2: Simulation average results with all normal process from five cross validation process.

Net Type	SNN Configuration	Accuracy	F1-Score	MCC
LSTM	[10]	97.76 %	95.69 %	97.63 %
MLP	[10]	95.80 %	91.57 %	95.52 %
LSTM	[10, 10]	97.76 %	95.69 %	97.63 %
MLP	[10, 10]	97.24 %	94.48 %	97.09 %
LSTM	10, 10, 10]	98.56 %	97.00 %	98.45 %
MLP	[10, 10, 10]	97.56 %	95.19 %	97.42 %
LSTM	[10, 5, 10]	97.24 %	94.61 %	97.01 %
MLP	[10, 5, 10]	97.40 %	94.78 %	97.24 %
LSTM	[10, 2, 10]	92.6 % 4 %	85.13 %	91.81 %
MLP	[10, 2, 10]	97.52 %	94.89 %	97.33 %
LSTM	[10, 10, 10, 10]	98.76 %	97.71 %	98.65 %
MLP	[10, 10, 10, 10]	97.76 %	95.87 %	97.66 %
LSTM	[10, 5, 2, 5, 10]	98.52 %	97.21 %	98.37 %
MLP	[10, 5, 2, 5, 10]	98.64 %	97.10 %	98.59 %
LSTM	[10, 8, 5, 8, 10]	93.20 %	86.23 %	92.42 %
MLP	[10, 8, 5, 8, 10]	98.52 %	96.80 %	98.49 %
LSTM	[10, 10, 10, 10, 10]	73.56 %	49.60 %	68.41 %
MLP	[10, 10, 10, 10, 10]	97.76 %	95.36 %	97.65 %

Specifically, the model configuration with four hidden layers of ten LSTM cells each emerged as the most effective, demonstrating the model's capacity to capture complex patterns in the data, resulting in an impressive training accuracy of 98.76 %, F1-score of 97.7 %, and MCC of 98.65 %.

The benchmark MLP model, characterized by its hidden layer configuration of [10, 5, 2, 5, 10], achieved accuracy of 98.64 %. The benchmark

model, architecture with the configuration [10, 5, 2, 5, 10], achieves a remarkable accuracy of 98.64 %. This high level of accuracy underscores the effectiveness of the chosen network structure in fault detection tasks. This suggests that the model is capable of distilling essential information from the input data and then expanding it to make precise predictions.

That achieved a training accuracy of 98.76 %, F1-score 97.7 % and MCC of 98.65 %. Test accuracy of 97.0 %, F1-score 97.0 %, and loss of 94.13 %, as can be seen in the training. A learning curve indicates if a model learns effectively from the data. The learning curve encountered the final value fast near 50 epochs and showed a small gap between training and validation, indicating that the model did not overfit. The rapid convergence to the final value suggests that the model's architecture and the learning rate are well-suited for the task at hand, allowing it to quickly assimilate the patterns within the data without the need for extensive epochs that could potentially lead to overfitting. Moreover, the small gap between the training and validation performance is a strong indicator of the model's generalization ability. Suggests that the model has learned the underlying patterns of the dataset rather than memorizing it, enhancing its ability to perform well on both training and test data.

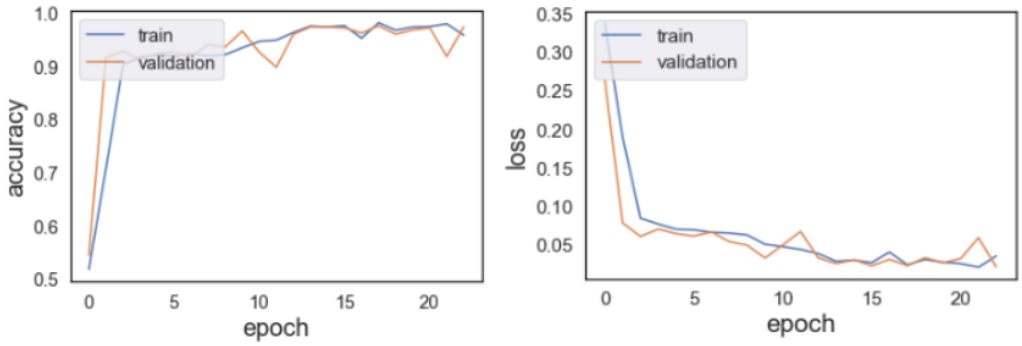

Figure 6.14: Model accuracy process from Schema 1.

What was noteworthy to observe is that the model exhibited higher error to detect faults in #4, and #5, errors of 10.8 % and 7.5 %, respectively, resulting high numbers of false negatives. Further investigation into all faults will be conducted in Schemes 3.

Class Pair	Count	Error Rate
(0, 0)	1973	0.5%
(0, 1)	308	3.3%
(0, 2)	286	1.2%
(0, 3)	279	0.6%
(0, 4)	293	10.8%
(0, 5)	290	7.5%
(0, 6)	284	2.8%
(0, 7)	299	4.7%
(0, 8)	279	2.4%

Table 6.3: Class Pair Error Rates

In our model evaluation, we employ a confusion matrix, a tool that visually outlines the performance of an algorithm by displaying true positives, true negatives, false positives, and false negatives. The matrix enables us to measure the accuracy and precision of classifications in a binary or multi-class problem, providing insights into the model's predictive capabilities.

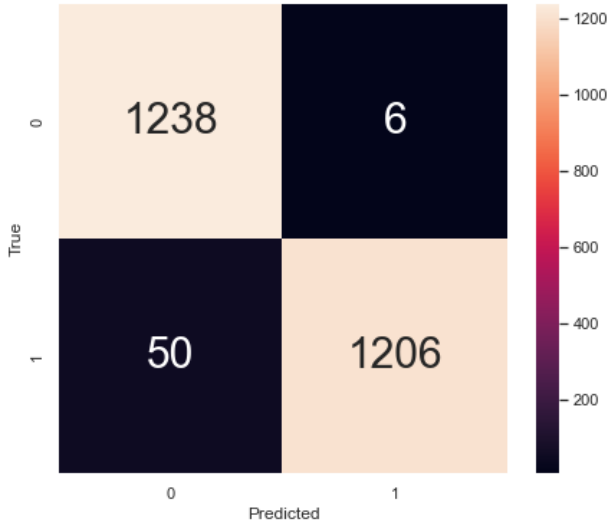

Figure 6.15: Confusion Matrix Schema 1.

The model demonstrates excellent performance with a high degree of precision and accuracy with LSTM units. However, attention should be given to the false negatives to investigate why these instances were missed and to determine if there is a pattern or similarity among them that could be addressed. The absence of false positives is notable and could indicate a very stringent model in predicting a case as positive, which, in

our application, may not be desirable because false negatives can be more dangerous, causing accidents. The model's performance indicated that there is room for improvement in reducing the false negatives.

6.3 Schema 2: One Normal Databases with All Faulty Databases

The examination of different schemas underscored the trade-offs between specificity and generalization in model training. Scheme 2 focused on training with one normal database and all faulty databases, aiming to enhance the model's sensitivity to deviations from a single normal operational state.

When using the normal #1 dataset for training in combination with all the faulty datasets, the expectation is to build a less sensitive model to variations in normal conditions while achieving higher accuracy. The same architecture and hyperparameters from Schema 1 were used to compare properly. It achieved a training accuracy of 98.72 %, F1-score 98.65 % and MCC of 97.43 % and a test accuracy of 98.4 %, F1-score 98.45 % and MCC of 96.83 %.

Table 6.4: Average Results with one normal databases with all faulty databases from five cross validation process.

Net Type	SNN Configuration	Accuracy	F1-Score	MCC
LSTM	[10, 10, 10, 10]	98.72 %	98.65 %	97.43 %
MLP	[10, 10, 10, 10]	98.16 %	98.11 %	96.36 %

The learning curve encountered the final value fast near 50 epochs and showed a small gap between training and validation, indicating that the model didn't overfit, but it is interesting to see that, in the beginning the validation accuracy diminishes faster than the training. The faster diminution in validation accuracy compared to training accuracy can be indicative of beginning of a divergence between the model's performance on seen versus unseen data, potentially that in the overfitting may occurred in the beginning.

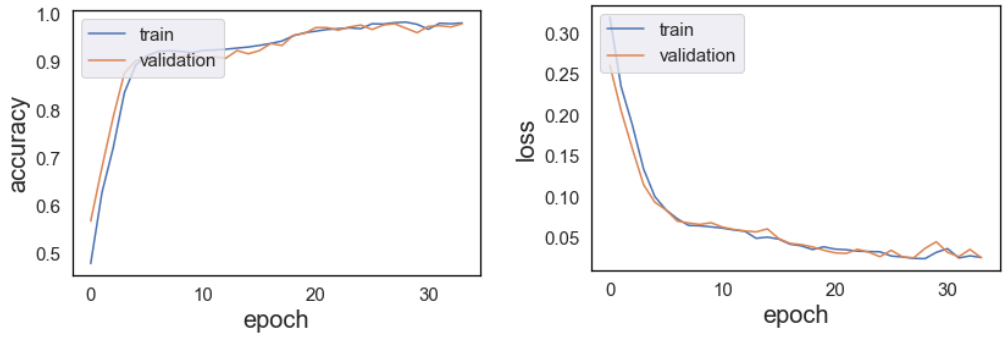

Figure 6.16: Model accuracy process from Schema 1.

When we compare this to Schema 1, it becomes apparent that there has been a general reduction in the magnitude of errors across most fault categories, especially for fault number 4. This indicates that while there is an improvement over the previous schema, there is still potential for refinement. Detailed information on the occurrence count and specific error rates for each fault category can be found below. This granular breakdown provides insight into the model's performance for each class pair and establishes a for targeted improvements.

Class Pair	Count	Error Rate
(0, 0)	1973	0.0%
(0, 1)	308	3.5%
(0, 2)	286	0.0%
(0, 3)	279	0.0%
(0, 4)	293	0.6%
(0, 5)	290	6.9%
(0, 6)	284	2.6%
(0, 7)	299	4.5%
(0, 8)	279	3.2%

Table 6.5: Class Pair Error Rates

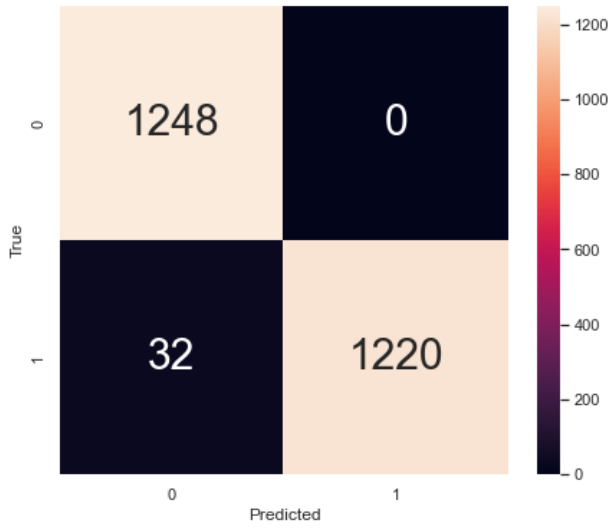

Figure 6.17: Confusion matrix of schema 2.

6.4 Schema 3: All Normal Databases with Each Individual Faulty Databases

The subsequent analysis using Scheme 3, which involved all normal databases with each individual faulty database, highlighted the model's precision in detecting specific faults. The detailed performance metrics for different fault types revealed the model's strengths and areas for improvement.

During the testing phase, the model demonstrated perfect accuracy for identifying faults #1, #2, #3, #4, and #5, with a 100 % success rate in distinguishing these faults from normal operating conditions. However, the test results revealed error rates of 8.4 %, 13.0 %, and 1.5 % for faults #6, #7, and #8 respectively, as detailed in Table 6.6. This differentiation indicates that while the model is highly effective for certain fault types, its testing performance for others, particularly faults #6 and #7, indicates areas where further model refinement is needed.

Table 6.6: Table of percentage of error for each faulty for testing dataset.

Net Type	Faulty	Accuracy	MCC	F1-Score
LSTM	1#	97.80 %	95.96%	97.71 %
MLP	1#	98.60 %	97.71 %	98.47 %
LSTM	2#	99.68 %	99.50 %	99.63 %
MLP	2#	99.68 %	99.501%	99.63 %
LSTM	3#	99.24%	98.50 %	99.17 %
MLP	3#	99.28 %	98.60 %	99.21 %
LSTM	4#	98.12 %	96.13%	98.05 %
MLP	4#	98.32 %	96.72 %	98.24 %
LSTM	5#	96.24%	92.86 %	96.03 %
MLP	5#	97.72%	95.73 %	97.55 %
LSTM	6#	99.32 %	98.70 %	99.27 %
MLP	6#	98.60%	97.43 %	98.49 %
LSTM	7#	98.88%	97.92 %	98.82 %
MLP	7#	98.64%	97.33 %	98.55 %
LSTM	8#	99.32 %	98.80 %	99.22%
MLP	8#	99.48 %	99.10 %	99.43 %

The excellent performance of the model across all faulty instances is evidenced by high accuracy F1-scores and loss, indicating a robust and reliable model across various scenarios. The range of MCC values, from a minimal 99.50 % to 92.86 %, underscores the model's consistent confidence in its predictions, regardless of the fault type. This consistency indicates a well-tuned model that has learned to effectively generalize from its training data.

Despite the overall strong performance, a closer examination reveals that the model does not perform uniformly across all fault types. There were minor deviations from perfect accuracy for faulty instances #1, #5, #4 and #7, manifested as lower accuracy.

The lower accuracy, observed for instance #5, suggesting that the faulty is more complicated to detect. Such a low error rate may be considered acceptable in many industrial applications, but it still represents an opportunity for optimization. To address these errors, it would be prudent to perform a detailed error analysis to understand the underlying causes. This could involve reviewing hyperparameters and the SNN configuration

6.5 Final Results

In Schema 1, the model exhibited a high rate of false negatives, suggesting that while it may efficiently identify normal conditions, it tends to miss many system failures. A high rate of false negatives is a critical concern in industrial settings where failing to detect a fault could lead to costly downtime or safety hazards. Adjusting the decision threshold for fault detection, enhancing feature selection, or providing more representative training data for the types of faults that are being missed could improve the model's detection capability.

Schemas 2 offered a more balanced error profile, with false negatives and false positives better distributed. This balance is essential as it suggests a model that neither over-predicts nor under-predicts the presence of a fault condition. A balanced distribution is preferred in many practical applications, as both types of errors can be costly.

It was particularly noteworthy that Schema 3 performed exceptionally well across all fault types. It suggested that a training approach that includes a broader variety of 'normal' data, alongside data from a single fault type, can enhance the model's fault detection capabilities. This could be due to the model's ability to understand the baseline of normal behavior better, making deviations due to faults more pronounced and easier to detect.

Table 6.7: Highest performance from all schemas from the average cross validation results.

dataset	Accuracy	F1-Score	MCC
Benchmark	98.64 %	97.10 %	98.60 %
Schema #1	98.76 %	97.71 %	98.65 %
Schema #2	98.72 %	98.65 %	97.43 %
Schema #3	99.68 %	99.50 %	99.63 %

In an industrial context, where parameters are continuously monitored, a model with less than 100 % accuracy can still be considered reliable, provided it maintains high precision and recall for critical faults. The slight trade-off in accuracy can be acceptable if the model ensures a high detection rate of actual faults with minimal false alarms depending on the costs of stopping the process.

7
Conclusion

In the modern chemical industry, where operational safety is a critical concern, the development of effective fault detection mechanisms remains a significant challenge. Traditional statistical methods, while powerful, have shown limitations. Our research has addressed this gap by introducing a novel approach employing SNNs for early fault detection in styrene polymerization.

We formulated a polystyrene model using the the method of moments for mass and energy balance in a CSTR reactor, incorporating proportional-integral-derivative (PID) control to reflect realistic industrial processes. This comprehensive framework allowed the introduction of 13 distinctive process faults, feeding the data into the SNNs, 5 normal process and 8 faulty process. The uniqueness of this methodology lies in the first-time application of SNNs for fault detection in polymerization, marking a significant innovation in the field.

Our SNN-LSM and SNN-MPL model demonstrated robustness, efficiency, and versatility. The experimental results, shows an impressive fault detection rate from 99.7% accuracy and 97.9 % MCC, validate the capacity of SNNs in handling complex, dynamic, and non-linear processes.

The study compared various schemas, including training a single model on all available data versus training specific models for each type of fault. The results unequivocally demonstrate that specialized models to individual fault types outperform a generalized model trained on all fault types together.

This finding challenges the conventional approach of seeking a one-size-fits-all model for fault detection. Instead, it underscores the advantages of developing multiple, specialized models, each fine-tuned for specific fault conditions. Such an approach not only enhances the accuracy and efficiency of fault detection but also reduces the likelihood of false positives and negatives, crucial factors in maintaining operational integrity and safety in industrial environments.

Implementing multiple models for fault detection, as recommended by the study's findings, may require additional resources in terms of development time, computational power, and maintenance. However, the benefits, including improved accuracy, reduced downtime, and enhanced safety, significantly outweigh these costs. The adoption of this strategy can lead to more

responsive and adaptable fault detection systems, capable of addressing the diverse and dynamic challenges encountered in industrial processes.

The SNN-Model's adaptability for changing conditions in the reactor and its real-time analysis capabilities make it a valuable asset for modern chemical engineering. Beyond petrochemicals, the principles of the SNN-Model can be extended to various manufacturing processes and abnormal situations, even in other areas.

For future works, build an ensemble approach using multiple SNNs models, each one trained separately with each fault, utilizing for each a proper hyperparameters, classification the fault and explore using it in integration with real data from IoT devices, enhancing its applicability and effectiveness.

7.1
Declaration of Competing Interest

The authors report no declarations of interest.

This work was supported by the National Agency for Petroleum, Natural Gas and Biofuels (ANP) and was supported by the Department of Chemical and Materials Engineering (DEQM) from Pontifical Catholic University of Rio de Janeiro (PUC-Rio).

8
References

[] 3.2

[1] MASEER, Z. K.; YUSOF, R.; BAHAMAN, N.; MOSTAFA, S. A. ; FOOZY, C. F. M.. **Benchmarking of machine learning for anomaly based intrusion detection systems in the cicids2017 dataset**. IEEE Access, 9:22351–22370, 2021. 1, 3.4

[2] GE, X.; WANG, B.; YANG, X.; PAN, Y.; LIU, B. ; LIU, B.. **Fault detection and diagnosis for reactive distillation based on convolutional neural network**. Computers and Chemical Engineering, 2020. 1, 3.4

[3] MELO, A.; CÂMARA, M. M.; CLAVIJO, N. ; PINTO, J. C.. **Open benchmarks for assessment of process monitoring and fault diagnosis techniques: A review and critical analysis**. Computers & Chemical Engineering, 165:107964, 2022. 1

[4] GAO, S.-Z.; WU, X.-F.; WANG, G.-C.; WANG, J.-S. ; CHAI, Z.-Q.. **Fault diagnosis method on polyvinyl chloride polymerization process based on dynamic kernel principal component and fisher discriminant analysis method**. 2016:7263285, 2016. 1, 3.4

[5] CHICCO, D.. **Siamese Neural Networks: An Overview**, volumen 2190 de **Methods in Molecular Biology**. Humana, New York, NY, 2021. 1, 3.5

[6] DRACHE, M.; DRACHE, G.. **Simulating controlled radical polymerizations with mcpolymer - a monte carlo approach**. Polymers, 4:1416–1442, 2012. 3.2

[7] RIVERO, P.. **Calculation method of molecular weight averages in polymerization with chain-length-dependent termination**. Journal of Polymer Research, 11:309–315, 2004. 3.2

[8] HUI, A. W.; HAMIELEC, A. E.. **Thermal polymerization of styrene at high conversions and temperatures. an experimental study**. Journal of Applied Polymer Science, 16:749–769, 1972. (document), 3.2, 4.1, 5.1, 5.2

[9] MACHADO, P. F. D. M. P. B.; LONA, L. M. F.. **Analyzing the real advantages of bifunctional initiator over monofunctional initiator in free radical polymerization**. Wiley InterScience, 2010. 1, 3.2, 4.1

[10] MAAFA, I. M.. **Pyrolysis of polystyrene waste: A review**. Polymers, 132:225, 2021. 1, 3.2

[11] D., G.. **Exact stochastic simulation of coupled chemical reactions**. The Journal of Physical Chemistry, 8125:2340–2361, 1977. 3.2

[11] ERLITA MASTAN, S. Z.. **Method of moments: A versatile tool for deterministic modeling of polymerization kinetics**. European Polymer Journal, 68:139–160, 2015. 3.2, 4.3

[12] S., Z.. **Modeling of molecular weight development in atom transfer radical polymerization**. Macro Molecular Theory and Simulations, 81:29–37, 1999. 3.2

[13] MOORE, E. R.. **The dow chemical company, encyclopedia of polymer and engineering**. Styrene Polymers, 2ª ed. by John Wiley Sons, 16:1–246, 1989. 3.2

[14] A. S. ALMEIDA, K. W.; SECCHI, A. R.. **Simulation of styrene polymerization reactors: Kinetic and thermodynamic modeling**. Brazilian Journal of Chemical Engineering, 25:337–349, 2008. 3.2

[16] CELINE W. S. YEUNG, JERALD Y. Q. TEO, X. J. L.; LIM, J. Y. C.. **Polyolefins and polystyrene as chemical resources for a sustainable future: Challenges, advances, and prospects**. American Chemical Society, 3:1660–1676, 2021. 1

[17] R. R. MILLER, R. N.; POOLEC, A.. **Styrene production, use, and human exposure**. Critical Reiiews in Toxicology, 24:1–10, 1994. 1, 3.2

[18] HAYDEN, P.; MELVILLE, S. H.. **The kinetics of the polymerization of methyl methacrylate. 11. the crosslinked and heterogeneous reaction**. JOURNAL OF POLYMER SCIENCE, 24:215–225, 1960. 3.2

[19] M. A. VILLALOBOS, A. E. H.; WOOD, P. E.. **Kinetic model for short-cycle bulk styrene polymerization through bifunctional**

initiators. Journal of Applied Polymer Science, 42:629–641, 1991. (document), 5.1

[20] K. Y. CHOI, W. R. L.; LEI, G. D.. **Kinetics of bulk styrene polymerization catalyzed by symmetrical bifunctional initiators**. Journal of Applied Polymer Science, 35:1547–1562, 1988. (document), 3.2, 5.2, 5.1

[21] YOON, W. J.; CHOI, K. Y.. **Kinetics of free radical styrene polymerization with the symmetrical bifunctional initiator 2,5-dimethyl-2,5-bis (2-ethyl hexanoyl peroxy) hexane**. Polymer, 33:21, 1992. (document), 5.1

[22] MOORE, E. R.. **Styrene polymers, the dow chemical company, reprinted from encyclopedia of polymer and engineering**. John Wiley Sons, Inc., 16:1–246, 1989. (document), 5.1

[23] MATTHEW J. SCORAH, RAMDHANE DHIB, A. P.. **Recent advances in the study of multifunctional initiators in free radical polymerizations**. Macromolecular Reaction Engineering, 1:209–221, 2007. (document), 5.2, 5.1

[24] FANG LI, JINGQIN XU, Y. W. H. Z.; LI, K.. **Pressure-induced polymerization: Addition and condensation reactions**. Molecules, 26, 2021. 3.2

[26] HOSSEIN RIAZI, AHMAD ARABI SHAMSABADI, M. C. G. A. M. R. .; SOROUSH, M.. **Method of moments applied to most-likely high-temperature free-radical polymerization reactions**. Process, 7:656, 2019. 3.2

[27] BACHMANN, R.. **Extension of the method of moments in nonlinear free radical polymerization**. Macromolecular Theory and Simulations, 26, 2017. 3.2

[29] IBRAHIM KORKUT, F. E.; OZBAY, S.. **Cost-effective control of molecular weight in ultrasound-assisted emulsion polymerization of styrene**. Acta Chimica Slovenica, 69:884–895, 2022. 3.2

[30] O'DRISCOLL, K. F., . H. J.. **The rate of copolymerization of styrene and methyl methacrylate—ii. the gel effect in copolymerization.** European Polymer Journal, 26:643–647, 1990. 3.2

[31] KREJCAR, O., S. I. F. R. . H. M.. **Incremental pic controller with handled limit states and manual settings. 2010 2nd international conference on mechanical and electronics engineering.** 2nd International Conference on Mechanical and Electronics Engineering, 2010. 3.3, 4.2

[32] RENATO APARECIDO AGUIAR, IVAN CARLOS FRANCO, F. L. F. L.. **Fractional pid controller applied to a chemical plant with level and ph control.** Chemical Product and Process Modeling, 2018. 3.3

[33] PATTON, R. J.. **Fault diagnosis in nonlinear dynamic systems via neural networks.** International Conference on Control '94., 1994. 3.4

[34] MOHAIMINUL ISLAM, GUORONG CHEN, S. J.. **An overview of neural network.** American Journal of Neural Networks and Applications, 5:7–11, 2019. 3.4, 3.5

[35] WASSERMANN, A., D. B. J. H. F. D. S. R. H. E. M.-H. E. A. . M. H.. **General discussion.** Transactions of the Faraday Society, 32:69, 1936. 3.2

[36] FLORY, P. J.. **The mechanism of vinyl polymerizations**. Journal of the American Chemical Society, 59:241–253, 1937. 3.2

[37] RUSSO, L. P., . B. B. W.. **Operability of chemical reactors: multiplicity behavior of a jacketed styrene polymerization reactor. chemical engineering science**. Journal of the American Chemical Society, 53:27–45, 1998. 3.2

[38] CHEN, C.-C.. **Continuous production of solid polystyrene in back-mixed and linear-flow reactors.** Polymer Engineering Science, 40:441–464, 2000. 3.2

[39] LING, J., Z. Y. S. Z. . N. J.. **Kinetics simulation of high viscous styrene bulk polymerization system.** European Polymer Journal, 37(12):2407–2411, 2001. 3.2

[40] GUIPING, C., Z. Z. H. L. M. Z. . W. Y.. **Molecular weight distribution of styrene polymerization in a starved feed reactor.** Journal of Polymer Engineering, 19(2):135–149, 1999. 3.2

[41] XIAOYANG LUA, PEIJIE LINA, S. C. Y. L. Z. C. L. W. Q. Z.. **Fault diagnosis for photovoltaic array based on convolutional neural network and electrical time series graph**. Energy Conversion and Management, 196:950–965, 2019. 3.4

[42] SHLENS, J.. **A tutorial on principal component analysis**. Adv Neural Inf Process, 2014. 3.4

[42] JANE BROMLEY, ISABELLE GUYON, Y. L.. **Signature verification using a "siamese" time delay neural network**. Adv Neural Inf Process, 6:737–744, 1994. 3.5

[43] KU, W., S. R. H. . G. C.. **Disturbance detection and isolation by dynamic principal component analysis.** Chemometrics and Intelligent Laboratory Systems, 30(1):179–196, 1995. 3.4

[44] ZHENG, S., . Z. J.. **A new unsupervised data mining method based on the stacked autoencoder for chemical process fault diagnosis.** Computers Chemical Engineering, 106755, 2020. 3.4

[45] GEORGIOS GRAVANIS, IOANNIS DRAGOGIAS, K. P. C. Z. K. D.. **Fault detection and diagnosis for non-linear processes empowered by dynamic neural networks.** Computers Chemical Engineering, 156, 2022. 3.4

[46] CHOPRA, S., H. R. . L. Y.. **Learning a similarity metric discriminatively, with application to face verification.** IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1:539–546, 2005. 4.5

[47] MCCULLOCH, W. S., . P. W.. **A logical calculus of the ideas immanent in nervous activity.** The Bulletin of Mathematical Biophysics, 5(4):115–133, 1943. 3.4

[48] CHADHA, G. S., . S. A.. **Comparison of deep neural network architectures for fault detection in tennessee eastman process.** IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)., 2017. 3.4

[49] NORUZI NASHALJI, M., A. S. M. . T. M.. **Fault detection of the tennessee eastman process using improved pca and neural classifier.** Soft Computing in Industrial Applications, p. 41–50, 2010. 3.4

[50] ROSENBLATT, F.. **Principles of neurodynamics; perceptrons and the theory of brain mechanisms.** Spartan Books, 1962. 3.4

[51] HAYKIN, S.. **Neural networks and learning machines**. Pearson Education, 2nd ed, 1999. 3.4, 5.5

[52] ZHOU, X., L. W. S. S. M. J. . J. Q.. **Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems.** Transactions on Industrial Informatics, 17(8):5790–5798, 2021. 3.5

[53] FERNÁNDEZ-LLANEZA, ULANDER, S. G. D. N. E. Z. H. . T. C.. **Siamese recurrent neural network with a self-attention mechanism for bioactivity prediction.** ACS Omega, 6(16):11086–11094, 2021. 3.5

[54] NAGY, A. M.; CZUNI, L.. **Detecting object defects with fusioning convolutional siamese neural networks**. Science and Technology Publications, 5:157–163, 2021. 3.5

[55] M. BERGMANN, B. C. ALMROTH, S. B. T. D. D. G. S. G. A. K. M. W. T. W.. **A global plastic treaty must cap production**. Science, 2022. 1

[56] ALTINTEN, A., E. S. A. F. H. H. . A. M.. **Application of adaptive pid control with genetic algorithm to a polymerization reactor.** Chemical Engineering Communications, 191(9):1158–1172, 2004. 3.3

[57] ALTINTEN, A., K. F. E. S. H. H. . A. M.. **Self-tuning pid control of jacketed batch polystyrene reactor using genetic algorithm.** Chemical Engineering Journal, 138(1-3):490–497, 2008. 3.3

[58] R. P., MAGHADE, D. K. S. S. Y. . P. S. N.. **A review of pid control, tuning methods and applications.** International Journal of Dynamics and Control, 138(1-3):490–497, 2020. 3.3

[59] MASOUD JALAYER, CARLOTTA ORSENIGO, C. V.. **Fault detection and diagnosis for rotating machinery: A model based on convolutional lstm, fast fourier and continuous wavelet transforms**. Computers in Industry, 125, 2021. 4.6

[60] SCHMIDHUBER, S. H. J.. **Long short-term memory**. Neural Computation, 9(8):1735–1780, 1997. 4.6

[61] ZHU, J.; JIANG, M. L. Z.. **Fault detection and diagnosis in industrial processes with variational autoencoder: A comprehensive study**. Sensors, 22:227, 2022. 1

[62] AMINE NAIMI, JIAMEI DENG, S. R. S.; ARUL, A. J.. **Fault detection and isolation of a pressurized water reactor based on neural network and k-nearest neighbor**. IEEE Access, 10, 2022. 1

[63] TIAGO LEMOS, LUIZ FELIPE CAMPOS, A. M. N. C. R. S. M. C. T. F. T. A. J. C. P.. **Echo state network based soft sensor for monitoring and fault detection of industrial processes**. Computers and Chemical Engineering, 155, 2021. 1

[64] CHIA-YU, W. L.. **Multiple time-series convolutional neural network for fault detection and diagnosis and empirical study in semiconductor manufacturing**. Journal of Intelligent Manufacturing, 32:823–836, 2021. 1

[65] O. I. PROVOTAR, Y. M. L.; VERES, M. M.. **Unsupervised anomaly detection in time series using lstm-based autoencoders**. IEEE International Conference on Advanced Trends in Information Theory (ATIT), p. 513–517, 2019. 1

[66] KAMARIZAN KIDAM, MARKKU HURME, M. H. H.. **Technical analysis of accident in chemical process industry and lessons learnt**. Chemistry Engineering, 2020. (document), 5.4, 5.3, 5.4

[67] PEREIRA, J. O.. **Modelageme e simulação de reatores de polimerização em massa de estireno com inicidores multifuncionais**, 2012. 5.3, 6.0.1.1

9
Appendix