CONTROL-ORIENTED, PHYSICS-INSPIRED DATA-DRIVEN MODELING AND SIMULATION OF THE CLINKER PRODUCTION PYRO PROCESS

by
MUHAMMAD ASLANIMOGHANLOO

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of Master of Science

Sabanci University
October 2023

CONTROL-ORIENTED, PHYSICS-INSPIRED DATA-DRIVEN MODELING AND SIMULATION OF THE CLINKER PRODUCTION PYRO PROCESS

Approved by:

Asst. Prof. Melih Turkseven .....
(Thesis Supervisor)

Assist. Prof. .....

Assoc. Prof. .....

Date of Approval: October 24, 2023

ABSTRACT

CONTROL-ORIENTED, PHYSICS-INSPIRED DATA-DRIVEN MODELING AND SIMULATION OF THE CLINKER PRODUCTION PYRO PROCESS

MUHAMMAD ASLANIMOGHANLOO

Mechatronics Engineering
M.S. THESIS, October 2023

Thesis Supervisor: Asst. Prof. Melih Turkseven

Keywords: Identification for Control, System Identification, Model Predictive Control, Recurrent Neural Networks, Transformers, Attention Mechanism

The cement industry is one of the critical components of modern society, playing a vital role in infrastructure and building construction. However, it is meanwhile one of the most energy-intensive and pollutant industries in the world. On the other hand, most current cement production plants are controlled and operated manually making them non-optimal. Implementing novel controllers can help to solve these problems. Model Predictive Controllers (MPC) have shown tremendous potential in this regard since they provide optimal controllers and can consider constraints on inputs and process variables. The essential part of MPC is its dynamic predictive model. While efforts have been made to model the cement production process, there is still a lack of suitable models for implementing MPC in cement production processes. Traditionally, physics-based models have been considered for modeling the cement production process. However, these models are typically complex with a huge number of parameters and computationally time-consuming, making them inapplicable to MPC. Recently, data-driven methods such as system identification and machine learning models have been developed for cement production process modeling. In spite of this, the majority of the literature did not take into account the physical principles and internal dynamics of the process. Further, they did not discuss the performance of their models and directly applied them to MPC.

This thesis aims to develop essential predictive models for implementing MPC for the

cement production pyro process by considering both the process's internal dynamics and data-driven methods. Moreover, we investigated the models performance comprehensively with a particular focus on long-term predictions which is essential for the successful implementation of MPC. The first step was to develop a simple and linear control-oriented model suitable for the MPC. Therefore, system identification methods were the focus. Moreover, the first principles of mass and energy conservation laws are considered to discover internal dynamics and inter-component relations in the process. As a result of incorporating these physics insights into systems identification models, a gray-box model has been developed. Next, more sophisticated simulation models were developed to represent the real plant in MPC implementation, since it was not possible to implement the designed controller in the real plant due to high risks and costs. For this purpose, machine learning models particularly sequence modeling machine learning models such as recurrent neural networks, and transformers are used. The selected ML models are modified and implemented on data from the cement production process at the Akcansa Cimento Plant.

Both the control-oriented models and simulation models were used for various prediction tasks on the collected data. Results show that in control-oriented models, the gray-box model performs better than the black-box model in validation data, especially for long-term predictions. This depicts the benefit of considering the internal dynamics and inter-components of the process and integrating them into the data-driven model. The results also reveal that the suggested simulation ML models are capable of modeling the cement production process and predicting its future states. Among the proposed ML models, the transformer outperforms others as it exploits the attention mechanism which overcomes RNN problems and can capture long-term dependencies. It should be noted, however, that the selection of a suitable model is dependent upon the objective task, the problem, and the available data. Accordingly, selecting the right model for their intended use would be the user's responsibility. Lastly, the developed models can be used to design and implement model predictive controllers for the cement production process in the future.

ÖZET

CLİNKER ÜRETİM PİROLİZ SÜRECİNİN KONTROL ODAKLI, FİZİKTEN İLHAM ALAN VERİYE DAYALI MODELLENMESİ VE SİMÜLASYONU

MUHAMMAD ASLANİMOGHANLOO

Mekatronik MÜHENDSLİĞİ
YÜKSEK LİSANS TEZİ, Ekim 2023

Tez Danışmanı: Asst. Prof. Melih Turkseven

Anahtar Kelimeler: Kontrol için Tanımlama, Sistem Tanimlama, Model Öngörülü Kontrol, Yinelemeli Sınır Ağları, Transformatörler, Dikkat Mekanizması

Çimento endüstrisi, modern toplumun kritik bileşenlerinden biridir, altyapı ve bina İnsan hayatında önemli bir rol oynar. Ancak aynı zamanda dünyanın en fazla enerji yoğunluklu ve kirletici endüstrilerinden biridir. Öte yandan, mevcut çimento üretim tesislerinin çoğu manuel olarak kontrol edilmekte ve iletilmektedir, bu da onları optimal olmayan hale getirmektedir. Yeni kontrolörlerin uygulanması bu sorunların çözülmesine yardımcı olabilir. Model Öngörülü Kumanda Sistemleri (Model Predictive Control, MPC), bu konuda muazzam bir potansiyel göstermiştir, çünkü optimal kontrolörler sayesinde girdiler ve süreç değişkenleri üzerindeki kısıtlamaları dikkate alabilir. MPC'nin temel unsuru, dinamik tahmin modelidir. Çimento üretim sürecini modellemek için çaba gösterilmiş olsa da, MPC'yi çimento üretim süreçlerinde uygulamak için hala uygun modellerin eksikliği vardır. Geleneksel olarak, çimento üretim sürecini modellemek için fizik tabanlı modeller düşünülmektedir. Bununla birlikte, bu modeller genellikle çok sayıda parametreye ihtiyaç duyar ve hesaplama açısından zaman alıcıdır. Bu da onları MPC'ye uygulanamaz hale getirir. Son zamanlarda, sistem tanımlama ve makine öğrenimi modelleri gibi veri odaklı yöntemler çimento üretim süreci modelleme için geliştirilmiştir. Buna rağmen, literatürün çoğunluğu sürecin fiziksel prensiplerini ve iç dinamiklerini hesaba katmamaktadır. Ayrıca, mevcut literatürde modellerin performansı tartışılmamış, modeller doğrudan MPC'ye uygulanmışlardır.

Bu tez, çimento üretim piroliz süreci için MPC uygulanmasına yönelik gerekli tahmin modellerini hem sürecin iç dinamiklerini hem de veri odaklı yöntemleri dikkate alarak geliştirmeyi amaçlamaktadır. Ayrıca, MPC'nin başarılı bir şekilde uygulanması için gerekli olan uzun vadeli tahminlere özel bir odaklanma ile modellerin performansı kapsamlı bir şekilde incelenmiştir. Çalışmamızdaki ilk adım, MPC için uygun, basit ve doğrusal bir kontrol odaklı model geliştirmektir. Bu nedenle, sistem tanımlama yöntemlerine odaklanılmıştır. Ayrıca, süreçteki iç dinamikleri ve bileşenler arası ilişkileri keşfetmek için kütle ve enerji korunumu yasaları ilkeleri dikkate alınmıştır. Bu fiziksel içgörüleri sistem tanımlama modellerine dahil etmenin bir sonucu olarak, bir gri kutu modeli elde edilmiştir. Ardından, yüksek risk ve maliyetler nedeniyle tasarlanan kontrolörü gerçek tesiste uygulamak mümkün olmadığından, MPC uygulamasında gerçek tesisi temsil etmek için daha sofistike simülasyon modelleri geliştirilmiştir. Bu amaçla, özellikle tekrarlayan sınır ağları (recurrent neural networks, RNN) ve transformatörler gibi makine öğrenimi (machine learning, ML) modelleri kullanmıştır. Seçilen ML modelleri, Akçansa Çanakkale Çimento Fabrikası'ndaki üretim sürecinden elde edilen veriler üzerinde geliştirilmiş ve uygulanmıştır.

Hem kontrol odaklı modeller hem de simülasyon modelleri, toplanan veriler üzerinde çeşitli tahmin görevleri için kullanılmıştır. Sonuçlar, kontrol odaklı modellerde gri kutu modelinin, özellikle uzun vadeli tahminler için doğrulama verilerinde siyah kutu modelinden daha iyi performans gösterdiğini göstermektedir. Bu, sürecin iç dinamiklerini ve bileşenler arası ilişkileri dikkate almanın ve bunları veri odaklı modele entegre etmenin faydasını göstermektedir. Sonuçlar ayrıca, önerilen simülasyon ML modellerinin çimento üretim sürecini modelleme ve gelecekteki durumlarını tahmin etme yeteneğine sahip olduğunu da ortaya koymaktadır. Önerilen ML modelleri arasında, dikkat mekanizmasından yararlandığı ve RNN'lerdeki sorunları aşarak uzun vadeli ilişkiler yakalayabildiği için transformatör diğerlerinden daha üstün performans göstermektedir. Bununla birlikte, uygun bir modelin seçimi, hedeflenen göreve, soruna ve mevcut verilere bağlıdır. Buna göre, amaçlanan kullanım için doğru modeli seçmek kullanıcının sorumluluğundadır. Son olarak, geliştirilen modeller gelecekte çimento üretim süreci için model öngörülü kontrolörler tasarlamak ve uygulamak için kullanılabilir.

ACKNOWLEDGEMENTS

My sincerest gratitude goes out to my advisor, Dr. Melih Turkseven, for providing me with unwavering guidance, patience, and support throughout my master's degree studies. My experience of being mentored by him during my development as a researcher has been invaluable.

I am grateful for the participation of Professor Meltem Elitas and Professor Evren Samur on my defense jury. Their knowledgeable and constructive feedback greatly aided me in the finalization of my thesis, and I am deeply thankful for their contributions to my academic development.

I am grateful to Akcansa Cimento A.S for their support of this research. Their contribution was invaluable and enabled us to complete this project successfully. Their generosity allowed us to focus on our research and make a real difference. We are thankful for their partnership.

I would like to express my appreciation to Sabanci University, the Engineering and Natural Science Faculty, and the Mechatronics Engineering Department for creating an exceptional and supportive atmosphere for graduate students.

Finally, I would like to express my deepest gratitude to my family for continuously inspiring me and providing unwavering support throughout my life. Throughout my academic career, they have consistently encouraged me to pursue my dreams, and without their encouragement, guidance, and motivation, I would not have been able to complete this thesis.

I dedicate my thesis

To myself, for all the hard work, determination, and sacrifice invested in completing this master's thesis. I dedicate this accomplishment to my personal growth and resilience.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Nomenclature

F_{cal}	Calciner fuel feed rate (Kg/sec)
F_k	Kiln fuel feed rate (Kg/sec)
M_r	Raw material feed rate (Kg/sec)
P_{cal}	Calciner pressure (Pa)
P_{ct}	Cooler tertiary air pressure (Pa)
P_{cyc}	Cyclone pressure (Pa)
P_{kh}	Kiln hood pressure (Pa)
P_{ki}	Kiln inlet pressure (Pa)
PR_{cl}	Clinker production rate (Kg/sec)
R	Ideal gas constant ($J.mol^{-1}.K^{-1}$)
S_{cf}	Cooler fan rotation speed (rad/sec)
S_{ef}	Exhaust fan rotation speed (rad/sec)
S_{id}	ID fan rotation speed (rad/sec)
T_{cal}	Calciner temperature (K)
T_{cl}	Clinker final temperature (K)
T_{ct}	Cooler tertiary air temperature (K)
T_{cyc}	Cyclone temperature (K)
T_{kh}	Kiln hood temperature (K)
T_{ki}	Kiln inlet temperature (K)

1. Introduction

According to the International Energy Agency (IEA), the cement industry is one of the largest industries in the world With an estimated production of 4.5 million tonnes in 2021. It is also one of the most energy-intensive and polluting industries, consuming approximately 7% of global industrial energy consumption each year [2] and emitting an increasing amount of greenhouse gases such as carbon dioxide, nitrogen oxide, and methane [3] [4]. Despite these, current cement production plants are not efficient; according to the latest energy analysis, cement production plants have an average energy efficiency of 58% [3]. Consequently, it is imperative to explore the possibility of reducing energy consumption and environmental effects in the cement industry by improving its efficiency. Currently, the majority of cement production plants operate and are controlled manually, which makes them non-optimal [5]. Novel controllers can be implemented to optimize the process and maximize plant efficiency [6]. Among various types of controllers, model predictive controllers (MPC) are the most commonly used and most suitable for controlling cement processes due to their capability to provide optimal controllers and impose constraints on inputs and states [7] [5], [8], [9]. However, model predictive controllers require control-oriented predictive models of the process [10]. A control-oriented model is a mathematical model of a system that is accurate enough for control tasks while keeping its complexity minimum [11]. In light of this, a considerable amount of research has been conducted on the modeling and control of cement production processes, making it an ongoing research problem [12], [13], [14], [5].

1.1 Process Description

A typical cement production process is illustrated in Figure 1.1, which has three major stages: 1- raw material preparation 2- pyroprocessing 3- final product preparation. Almost all new clinker plants use dry powdered raw meals fed to the kiln via a cyclone preheater to achieve high thermal efficiency and low costs. A mixture of limestone, clay, and other minerals is used to make cement. The main component of cement is calcium oxide (CaO), which is derived from the decomposition of limestone, a sedimentary rock made up mostly of calcium carbonate ($CaCO_3$). Other materials that are commonly used in the production of cement include clay, sand, and iron ore. During the cement production process, these raw materials are combined in specific ratios. Typically, for Portland cement 80% of raw material is limestone and the rest 20% is other materials [15], [16].

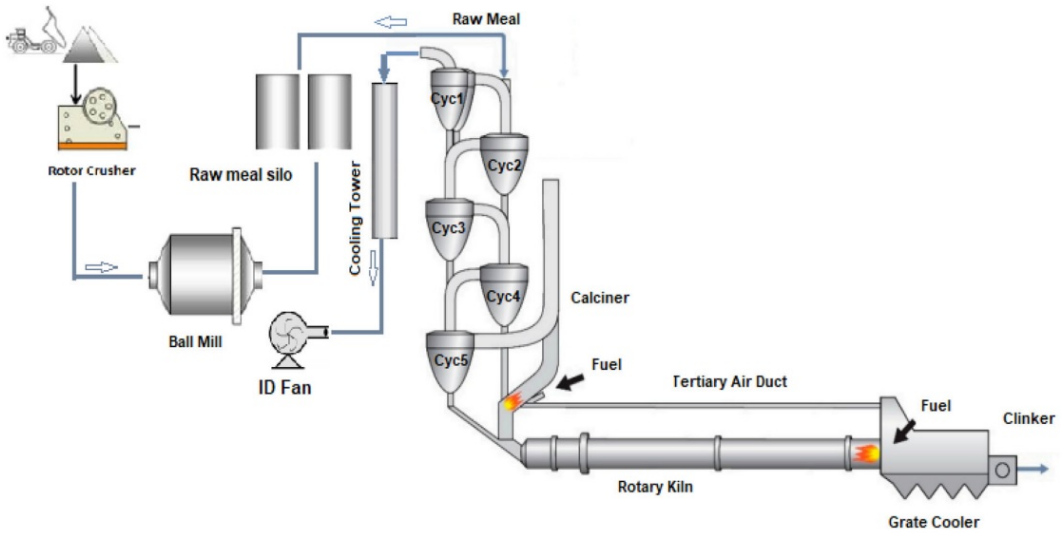


Figure 1.1 Cement production process [1]

The cement production process's main part is pyro processing which includes three major parts: 1- preheating raw materials in the preheater tower 2- calcination and clinker production in the rotary kiln 3- cooling clinkers in the cooler. The pre-heater tower is a large vertical structure consisting of cyclones and a pre-calciner chamber. The preheater tower increases the raw materials' temperature before feeding into the rotary kiln. The raw materials are fed into the top of the tower and heated as they move downward through the cyclones. The preheater tower uses hot gases from the kiln and cooler to heat the raw materials, which helps to minimize the energy required to heat the raw materials in the kiln increasing the overall efficiency of the production plant. The preheater tower also helps to reduce emissions of greenhouse

gases by reducing the amount of fuel needed to heat the raw materials in the kiln [17]. A preheater cyclone is a crucial component in the cement production process, as it facilitates the transfer of heat between the raw materials and the hot gases from the kiln and cooler and separates the raw materials as they move through the tower. The cyclones are cylindrical structures with a conical bottom that tapers to a small outlet. As raw materials and hot gases enter the cyclone, they are spun around by the shape of the cyclone, separating the raw materials from the hot gases. The raw materials then fall to the bottom of the cyclone, while the hot gases are expelled through the outlet at the top.

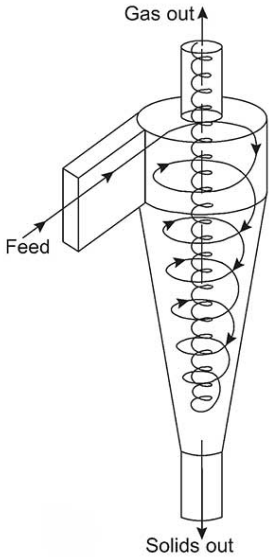


Figure 1.2 Preheater cyclone

A calciner is a device used in the cement industry to heat raw materials to a high temperature in order to partially decompose them and prepare them for the next stage of the production process (rotary kiln). The calciner is typically located in the preheater tower of a cement plant, and it is fed with hot gases from the kiln and cooler as well as raw materials. The calciner burns coal or other fuels to increase the temperature to the required level for calcination, which involves the decomposition of the raw materials into their constituent compounds. The calciner helps to optimize the chemical reactions that occur in the kiln, which in turn helps to improve the efficiency and performance of the cement plant. Typically, 80-90% of calcination happens in the pre-calciner which reduces 8-11% energy consumption in the whole process.

After the raw material is heated and partially calcined in the preheater tower, it enters the rotary kiln, which is the most important and the most complicated part of the cement production process. The rotary kiln is a large, cylindrical tube with

a slight inclination of 2-5 degrees and is rotated by electrical motors at a speed of 0.5-1.5 rpm. The raw material enters the rotary kiln from the tail end and as it moves through the kiln, it is subjected to increasingly high temperatures due to the heat from the burning coal and hot gases coming from the head of the kiln, known as secondary air. At the head of the kiln, a flame is burning, typically fueled by coal, to provide the heat needed for the chemical reactions and burning of the raw materials. The chemical reactions that take place in the rotary kiln are complex and involve the decomposition of limestone, the formation of calcium oxide, and the combination of calcium oxide with other raw materials to form cement clinker. All of these factors make it very challenging to develop an accurate and reliable model for predicting the kiln states. Another challenge in kiln modeling is the difficulty of gathering the necessary and accurate data. For example, it is hard to measure the temperature inside the kiln due to the dusty and hot environment. While heat cameras are often installed in the kiln heads to measure the kiln temperature, their data is not accurate and reliable enough, and they can only measure the temperature of a portion of the kiln, not the full distribution of temperature along the kiln.

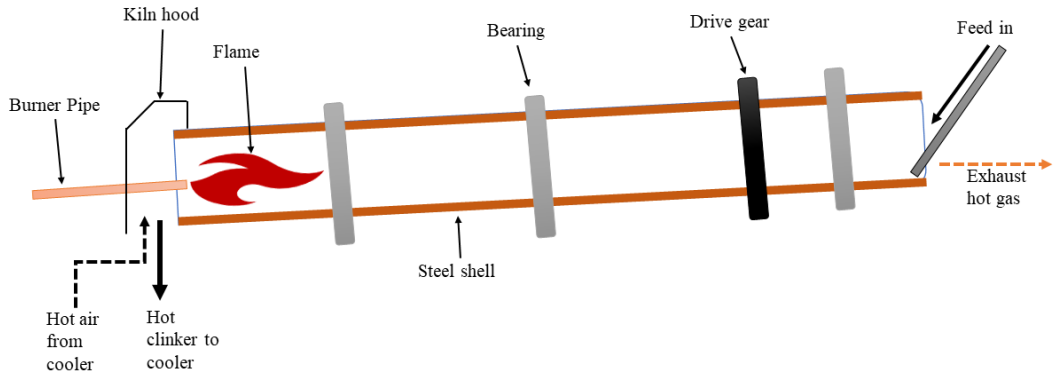


Figure 1.3 Rotary kiln

After the rotary kiln, the hot clinker enters the cooling part, where it cools to about 150 C. The cooler consists of static and moving grates. The static grates are situated at the entrance and the moving grates are located at the following. Under these grates are high-pressure and powerful fans that force ambient air into the under-grate chambers. As the gas goes through the hot clinker, it exchanges heat with the clinker which reduces its temperature. Above the cooler, hot air is collected in two places, one near the kiln head with a temperature around 1000-1200 C, the other near the cooler end with a temperature around 300-400 C, by doing this a part of the energy is recovered and the whole process efficiency increases. Grate coolers are

capable of recovering 75% of the sensible heat of clinker into secondary and tertiary air. A schematic of typical coolers is shown in Figure 1.4. Consequently, the cement production process is a highly complex process with interconnected modules that include numerous physical and chemical phenomena. Thus, modeling of cement production process is a challenging task that requires a deep understanding of the process, and industrial setting as well as modeling knowledge.

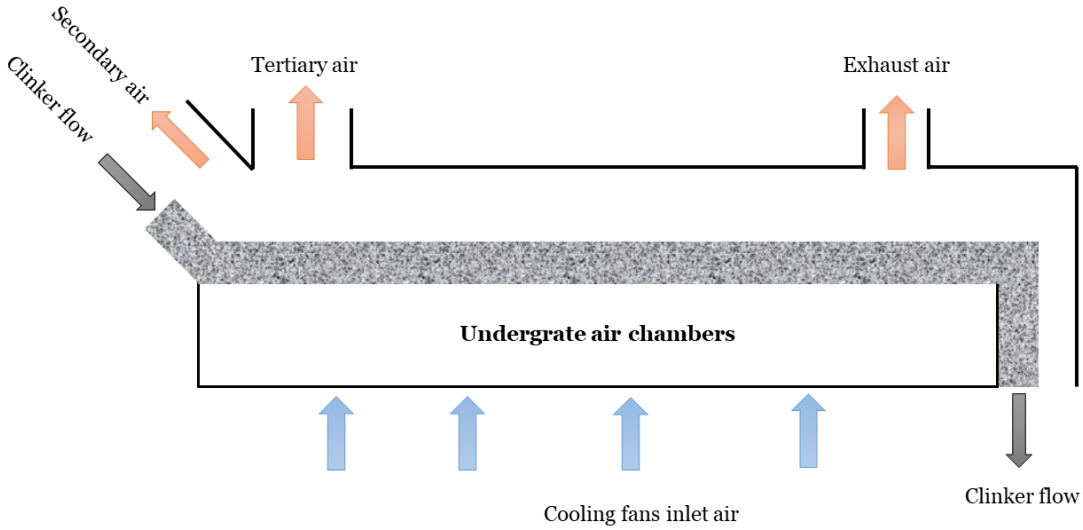


Figure 1.4 Grate clinker cooler

1.2 Physics-based models

Traditionally, the majority of attempts in modeling of cement production process have been devoted to physics-based modeling in which the governing equations are discovered using first principles. In this method, physical laws such as mass, energy, and momentum conservation laws are applied to various chemical reactions and heat transfers in order to obtain dynamic mathematical models [18] [19], [20], [21], [22]. As one major example of this modeling method, [4] provides a physics-based model for the entire cement manufacturing process. A separate model was developed for the pre-heater, the calciner, the kiln, and the cooler. After combining these models, a dynamic model of the entire process was developed. The developed integrated model was used to investigate the effects of different design and operation parameters on net energy consumption (NEC), such as the number of preheaters, rotational

speed of the kiln, tilt of the kiln, and speed of the clinker cooler grate. A dynamic first principle model focused on control for the kiln has been developed in [6]. Authors reduced modeling computation by considering 5 segments for the kiln, and assuming that the mass distribution for gas and solid is homogeneous inside each segment. Moreover, they considered average values for parameters like heat transfer coefficients and linearized the equations. Although these assumptions and simplifications decrease model accuracy compared to simulation models, the accuracy is still sufficient for control-related tasks. Finally, they used this reduced order model to implement an MPC controller for kiln burning zone temperature. For the cooler, physics-based models were developed primarily to determine the temperature profile of the clinker and air along the cooler. For instance, [23] implemented energy analysis and energy balance with a large number of assumptions and simplifications for the parameters. They were able to accurately predict the clinker output temperature, secondary and exhaust gas temperatures, and pressures. This model was used to evaluate the effects of parameters such as air mass velocity and solid mass velocity on the production of entropy. Different simulation software was also used for modeling the cement production process. As an example, Ansys and Fluent software were utilized to develop a physics-based model for the cooler in [24]. The model simulated both gas and solid temperature distributions along the cooler, as well as transient temperatures. It was then used to investigate the effects of various parameters, such as the effect of the average diameter of the clinker particles on the pressure drop in the clinker layer. Based on convective heat transfer equations, [25] developed a mathematical model for the cooler part that takes into account the effect of moving grates, which had been overlooked by most researchers up to that point. As a result of solving discretized difference equations, they determined the temperature distribution along the cooler. Afterward, they used their model to investigate the effect of different parameters on clinker cooling, such as the effect of clinker speed or clinker thickness on secondary air temperature. They later improved their model by incorporating real plant experiments in addition to simulations [26]. They also demonstrated that using exhaust gas as the first and second grates' input air could increase secondary and tertiary air temperatures. Physics-based models are highly adaptable due to their clear physical meaning and easy interpretation of their parameters. However, because of the complexity, nonlinearity, and dynamic nature of the clinker production process, physics-based models tend to be exceedingly complex, of a high order, with an excessive number of parameters. In most cases, it is necessary to simplify the model, resulting in a reduction in accuracy. Furthermore, adjusting many parameters can be a challenging process. Consequently, physics-based methods are not appropriate for developing models for controlling complex systems such as the cement production process [14].

1.3 Data-driven Models

An alternative is to use statistically based methods, also known as data-driven methods or black-box models which use statistics to discover direct relations between inputs and outputs. Thus, they can avoid the complexity and high fidelity of physics-based models. Consequently, data-driven methods have become increasingly popular for modeling complex processes such as cement manufacturing [27], [28], [29], [30], [31]. A Locally Linear Neuro-Fuzzy (LLNF) network was used to model the kiln system in [32]. Using the identified model, they were able to identify abnormal conditions within the kiln. To simulate the kiln process, a multilayer perceptron neural network was designed and trained by [33]. [34] constructed a cement kiln model based on BP and Elman networks. In [35], a dynamic model of the rotary kiln was developed by integrating a process mechanism and a recurrent neural network. Using a time delay mechanism, the kiln's residence time is estimated, and a long-short-term memory model (LSTM) that combines an attention mechanism and an ordinary differential equation solver is proposed for capturing the time-varying and nonlinear characteristics of the kiln process. [36] used ARMAX to model kiln tail temperature, and designed an MPC controller based on this identified model. A nonlinear ARMAX model was considered in [29] as the model for the rotary kiln and a nonlinear MPC implemented with time-varying weights on the identified model. For modeling the cooler, neural networks are the most commonly used data-driven approach. For instance, [37] considered Elman neural networks, while [38] utilized MLP networks. Using multi-modal fusion neural networks, [39] predicted cooler grate pressures. However, black-box models are not without challenges. It can be challenging to understand how a black-box model makes its predictions or decisions due to the lack of transparency or interoperability. Thus, errors and biases can be harder to identify and fix in black-box models [40], [41]. Additionally, their performance is limited by utilized data quantity and quality which is sometimes difficult to acquire in industrial settings [42]. On the other hand, the majority of data-driven models for cement production process control did not discuss their models' performance, particularly multistep predictions which is essential for successful MPC design and implementation. These models were directly used for the controller design and implementation. Analyzing and verifying the models' reliable and satisfactory prediction performance can also enhance the control process. Moreover, these data-driven models mostly did not consider useful knowledge from the physics of the problem while it can be helpful in developing more reliable, generalizable, and accurate models [43].

1.4 Thesis Contributions

This research aims to develop two essential models for model predictive control of the cement production process, namely the control-oriented model and the simulation model. First, a control-oriented gray-box model utilizing classical system identification and first principles was developed. In this model, the first principles were utilized to discover inter-module relations helping to consider the internal dynamics of the process. A comprehensive analysis of the control-oriented model in short-term and long-term predictions demonstrated its superiority over the black-box system identification model. Additionally, the gray-box (physics-based) model was more generalizable, making it less dependent on data, which is particularly beneficial when it is difficult or expensive to collect data.

For a successful design of an MPC controller, it should be implemented on the real plant, however, due to the costs and risks it is not reasonable and possible to do so. Thus, a model that can mimic the real plant behavior is required. As a result, more sophisticated nonlinear models for simulating the process are developed utilizing state-of-the-art machine learning models. In this work, we used sequence modeling machine learning models including Recurrent neural networks, Encoder-decoder recurrent neural networks, and Transformers. Models based on machine learning were very successful in predicting system variables, especially in the case of multi-step predictions. In the end, both proposed hybrid and machine learning models have the potential to improve the accuracy and robustness of the cement production process modeling. This can lead to better process control utilizing MPC controllers, which could ultimately increase efficiency and reduce the environmental impact of cement production.

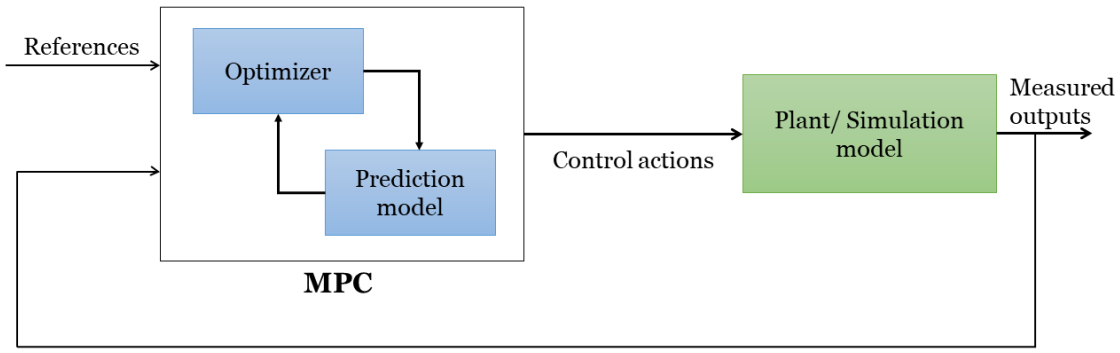


Figure 1.5 Model Predictive Controller Structure

In Chapters 2 and 3, we delve into the modeling approaches in this thesis. First, a control-oriented model was developed utilizing system identification methods and considering the first principles. Afterward, simulation models for sequence modeling, including RNNs, encoder-decoder RNNs, and transformers, are presented. Chapter 4 discusses these models problem specific modifications and implementations and prediction performances for various prediction horizons. In Chapter 5, we analyze the results presented in the previous chapter. We also discuss the limitations of the models and suggest ways to improve their performance. Finally, the last chapter concludes with a general conclusion concerning the performance of the developed modeling approaches and suggests future research directions.

2. Control-oriented modeling

In this section, we discuss the development of the proposed control-oriented model. First, a simplified version of the process is described and presented which helps to consider the physics of the problem. Afterward, considering first principles, the interactions between different components of the process are studied. Lastly, these insights are integrated into a system identification model to develop a gray-box model of the process.

The cement manufacturing process can be simplified as Fig. 2.1. All components are connected to each other through gas and solid flows. These components can be

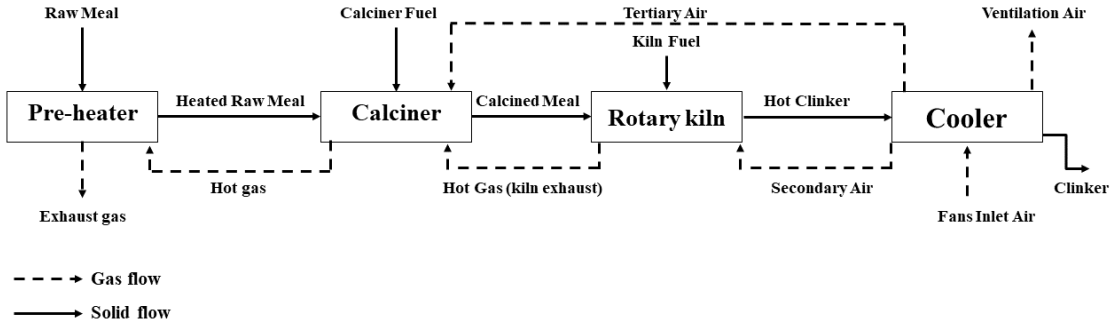


Figure 2.1 Cement production process simplified schematics

considered as individual control volumes connected together. Afterward, they can be analyzed through mass and energy conservation. A single control volume and its interactions with neighboring control volumes can be represented in Fig 2.2.

The mass conservation law can be applied to the control volume:

$$\dot{m}_{cv} = \dot{m}_{in} - \dot{m}_{out} \tag{2.1}$$

It is well known that the volumetric flow rate between two control volumes is a function of their pressures ($Q_{in} = f(P_{cv_{i-1}}, P_{cv_i})$). Considering the ideal gas condition for the control volume, its density can be represented by $\rho_{cv} = \frac{P_{cv}}{RT_{cv}}$. Thus, the mass flow rate between two control volumes can be represented in the

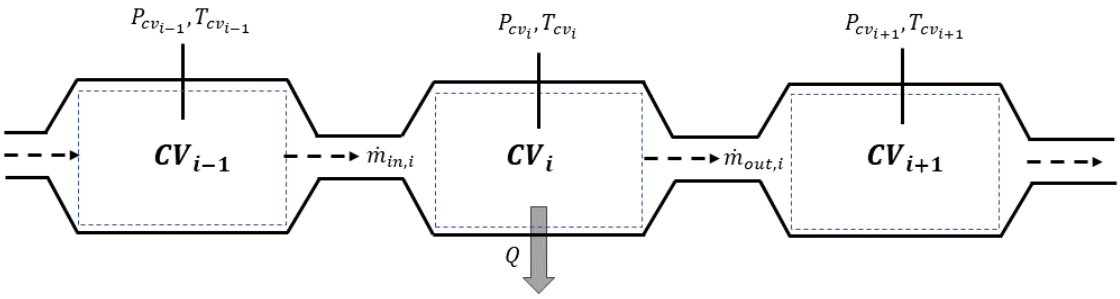


Figure 2.2 A control volume and its interactions with adjacent control volumes

form of $\dot{m}_{in} = \frac{P_{cv_{i-1}}}{RT_{cv_{i-1}}} f(P_{cv_{i-1}}, P_{cv_i})$, which can be written in the general form of $\dot{m}_{in} = f(P_{cv_{i-1}}, P_{cv_i}, T_{cv_{i-1}})$. The same approach can be followed for the outlet mass flow rate from the desired control volume $\dot{m}_{out} = f(P_{cv_{i+1}}, P_{cv_i}, T_{cv_{i+1}})$. Replacing these relations into Eq. 2.1, and applying the ideal gas condition ($m_{cv} = \frac{P_{cv}V_{cv}}{R_{cv}T_{cv}}$), we can get a general function for the desired control volume states dynamics.

$$\dot{P}_{cv_i}, \dot{T}_{cv_i} = f(T_{cv_{i-1}}, P_{cv_{i-1}}, T_{cv_{i+1}}, P_{cv_{i+1}}) \tag{2.2}$$

A similar approach can be followed for applying the energy conservation law for the desired control volume.

$$\frac{d}{dt}(m_{cv_i} C_{p,cv_i} T_{cv_i}) = \sum \dot{m}_{in,i} C_{p,in} T_{in} - \sum \dot{m}_{out,i} C_{p,out} T_{out} + Q_{net} \tag{2.3}$$

By taking into account the mass flow rates relation, and ideal gas assumption Eq. 2.3 can be written as follows:

$$\dot{P}_{cv_i}, \dot{T}_{cv_i} = f(P_{cv_{i-1}}, T_{cv_{i-1}}, P_{cv_{i+1}}, T_{cv_{i+1}}) \tag{2.4}$$

This analyzing method can be applied to each part of the cement production process. For instance, for the preheater cyclones, the same schematics in Fig. 2.2 can be drawn. Afterward, mass and energy conservation laws can be applied like the previous part. The only difference is that this time, the solid flow between cyclones should be also considered. As a result, 2.3, and 2.4 can be written as:

$$\dot{P}_{cv_i}, \dot{T}_{cv_i} = f(T_{cv_{i-1}}, P_{cv_{i-1}}, T_{cv_{i+1}}, P_{cv_{i+1}}, \dot{m}_{s_{in}}, \dot{m}_{s_{out}}) \tag{2.5}$$

$$\dot{P}_{cv_i}, \dot{T}_{cv_i} = f(P_{cv_{i-1}}, T_{cv_{i-1}}, P_{cv_{i+1}}, T_{cv_{i+1}}, \dot{m}_{s_{in}}, \dot{m}_{s_{out}}, T_{m_{sin}}, T_{m_{sout}}) \tag{2.6}$$

where $\dot{m}_{s_{in}}, \dot{m}_{s_{out}}$ are the inlet and outlet solid mass flow rates to the cyclone. However, the solid particle resident time inside cyclones is approximately eight seconds [44], while the sampling time in our data is 1 minute. Thus, the solid

particles amount inside the cyclone can be considered constant meaning that $\dot{m}_{s_{in}} = \dot{m}_{s_{out}} = \dot{m}_s$. Also, the solid and gas phases can be considered to have the same temperature inside a cyclone due to heat equilibrium between them. As a result the mass and energy conservation relations 2.5, 2.6 can be written in the following format:

$$\dot{P}_{cv_i}, \dot{T}_{cv_i} = f(T_{cv_{i-1}}, P_{cv_{i-1}}, T_{cv_{i+1}}, P_{cv_{i+1}}, \dot{m}_s) \tag{2.7}$$

Based on the above relations one can conclude that a cyclone's variables (pressure, temperature) are a function of its adjunct cyclones' states. Therefore, cyclone states can be modeled by only considering its adjunct cyclones' variables. This approach has been implemented for the other parts of the cement production process including the calciner, kiln, and the cooler to discover their relations with other components. As a result, the affecting variables and inputs in each component's states dynamic have been identified. A list of the states of interest and their affecting states and inputs are provided in table 4.2. These relations can be integrated into various data-driven modeling methods in order to develop a dynamic model of the process. As a result, a more accurate and reliable gray-box model can be developed. In this research, we integrated these relations into system identification methods by selecting the regressors based on these insights.

Table 2.1 Process variables and their influencing inputs and variables derived from first principles

Desired State	Affecting states	Affecting inputs
P_{kh}	T_{kh},T_{ct}, P_{ct}, T_{ki}, P_{ki}	S_{cf}, S_{ef}, F_k, M_r
T_{kh}	P_{kh}, T_{ct}, P_{ct}, T_{ki}, P_{ki}	S_{cf}, S_{ef}, F_k, M_r
P_{ki}	P_{kh}, T_{kh}, T_{cal}, P_{cal}, T_{ki}	F_k, M_r
T_{ki}	P_{kh}, T_{kh}, T_{cal}, P_{cal}, P_{ki}	F_k, M_r
P_{cal}	T_{cal}, T_{ct}, P_{ct}, T_{cyc}, P_{cyc}	F_{cal}, M_r
T_{cal}	P_{cal}, T_{ct}, P_{ct}, T_{cyc}, P_{cyc}	F_{cal}, M_r
T_{cl}	P_{kh}, T_{kh}, PR_{cl}	S_{cf}, S_{ef}, F_{cal}, F_k, M_r
PR_{cl}	P_{cal}, T_{cal}, P_{ki}, T_{ki}	F_{cal}, F_k, M_r

System identification is a methodology for building mathematical models of dynamic systems using measurements of the input and output signals of the system. In a dynamic system, the values of the output signals depend on both the instantaneous values of the input signals and also on the past behavior of the system. System identification uses the input and output signals of a system to estimate the values of adjustable parameters in a given model structure. A model structure is a mathematical relationship between input and output variables that contains unknown parameters. Some of the most common model structures are the Input-Output

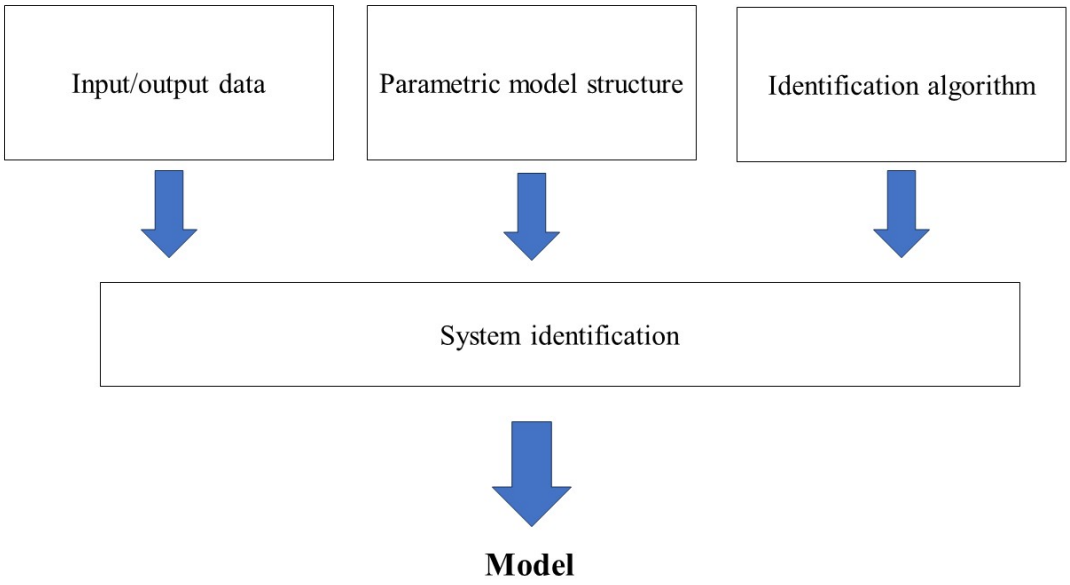


Figure 2.3 System identification process

Polynomial Models. A polynomial model uses a generalized notion of transfer functions to express the relationship between the input, $u(t)$, the output $y(t)$, and the noise $e(t)$ using the equation:

$$\sum_{j=1}^{ny} A(q)y(t) = \sum_{i=1}^{nu} \frac{B_i(q)}{F_i(q)} u_i(t - nk_i) + \frac{C(q)}{D(q)} e(t) \tag{2.8}$$

The variables A, B, C, D, and F are polynomials expressed in the time-shift operator q^{-1}. y_j is the j^{th} output, ny is the total number of outputs, u_i is the i^{th} input, nu is the total number of inputs, and nk_i is the i^{th} input delay that characterizes the transport delay. The variance of the white noise $e(t)$ is assumed to be λ. In practice, not all the polynomials are simultaneously active. Often, simpler forms, such as ARX, ARMAX, Output-Error, and Box-Jenkins are employed. Assuming $C = D = F = 1$ the model becomes ARX.

$$\sum_{i=1}^{ny} A(q)y(t) = \sum_{i=1}^{nu} B_i(q)u_i(t - nk_i) + e(t) \tag{2.9}$$

Polynomial models can be estimated using either time domain or frequency domain data. To estimate a polynomial model, the model order must be specified as a set of integers, where each integer represents the number of coefficients for a polynomial in the selected model structure. The integers na and nb represent the number of coefficients for the A and B polynomials, respectively. The model orders can be selected based on the physics, and statistical relations between output and input data such as autocorrelation and cross-correlations. However, mostly it is helpful to

try various model orders to investigate the effects of increasing and reducing orders from the one suggested by physics or statistics. System identification techniques can be categorized based on the amount of physical knowledge used in the modeling process: black-box models and gray-box models. Unlike black-box models, which are based solely on data, gray-box models incorporate the physics of the problem as well as the statistical methods. The black-box method in our problem can be a direct model between the process inputs and outputs of interest. In this regard, after identifying the process inputs and outputs, a proper model structure with proper orders has to be selected, lastly utilizing parameter estimation methods, the proposed model structure will be fitted to data. However, gray-box models can be developed in this context as well considering the internal dynamics of the process. The first option is to consider not only the process inputs but also all the internal dynamics in the modeling process which would result in a larger model. Another option is to utilize the physical insights from the previous part to identify affecting inputs and states for each process variable. Consequently, the regressors can be selected based on these insights. The matrix format for a multi-input multi-output ARX model with n outputs and m inputs can be written as follows, where p, q indicate the orders of the outputs and inputs.

$$\overbrace{\begin{bmatrix} y_{1,k} \\ \vdots \\ y_{n,k} \end{bmatrix}}^{Y_k} = \overbrace{\begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,p} \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} y_{1,k-1} \\ \vdots \\ y_{1,k-p} \\ \vdots \\ y_{n,k-1} \\ \vdots \\ y_{n,k-p} \end{bmatrix}}^{\hat{Y}_k} + \overbrace{\begin{bmatrix} b_{1,1} & \cdots & b_{1,m} \\ \vdots & \ddots & \vdots \\ b_{m,1} & \cdots & b_{m,q} \end{bmatrix}}^{B} \overbrace{\begin{bmatrix} u_{1,k-1} \\ \vdots \\ u_{1,k-q} \\ \vdots \\ u_{m,k-1} \\ \vdots \\ u_{m,k-q} \end{bmatrix}}^{\hat{U}_k} \tag{2.10}$$

A and B are the coefficients/parameters matrices to be estimated, Y_k is the system's current outputs (to be predicted) and $\hat{Y}_k$, $\hat{U}_k$ are previous inputs and outputs. Integrating relations from table 4.2 into the ARX structure will result in sparse coefficient matrices. This is because, based on these physical models/understandings, only some states and inputs affect the desired state. The parameters in the coefficient matrices associated with non-affecting states and inputs can therefore be equaled to zero. As a result, the number of free parameters in the ARX model is reduced by approximately 50% from 960 to 485. Consequently, the ARX model

coefficients/parameters matrices can be sparified as follows:

$$A = \begin{bmatrix} a_{1,1} & 0 & 0 & \cdots & a_{1,n} \\ \vdots & 0 & 0 & \cdots & \vdots \\ a_{n,1} & 0 & 0 & \cdots & a_{n,p} \end{bmatrix} \quad , B = \begin{bmatrix} b_{1,1} & 0 & 0 & \cdots & b_{1,m} \\ \vdots & 0 & 0 & \cdots & \vdots \\ b_{m,1} & 0 & 0 & \cdots & b_{m,q} \end{bmatrix} \tag{2.11}$$

The implementation of this proposed gray-box (hybrid) model for the intended problem, the Akcansa cement production plant modeling, will be discussed in the models' implementation and results part.

The main advantage of gray-box models is that they can be more accurate than black-box models, which do not use any prior knowledge about the system. This is because the theoretical structure can help to constrain the possible values of the parameters, which can lead to a more accurate model. Another advantage of gray-box models is that they can be less complex than white-box models, which are fully specified by the theoretical structure. This is because the theoretical structure can be simplified or approximated, which can make the model easier to build and estimate. However, there are also some disadvantages to using gray-box models. One disadvantage is that they can be more difficult to build and estimate than black-box models. This is because the theoretical structure must be carefully chosen, and the data must be carefully selected and pre-processed. In addition, the model's accuracy depends on the quality of prior knowledge. An inaccurate or incomplete prior knowledge may lead to an inaccurate model.

Upon selecting a model structure, the identification procedure provides us with a specific model of the selected model structure. Although the identified model may be the best one available, the more critical question is whether it meets the intended purpose adequately. The process of determining if a designated model is suitable is known as model validation. These methods are inevitably problem-dependent, contain several subjective elements, and no conclusive validation procedure can be given. Basically, it is a matter of falsifying a model under the conditions it will be used for and also to gain confidence in its ability to reproduce new data from the system. Some common model validation approaches are:

- Quality of the model: Comparing the model predictions to the measured outputs. In this regard, various comparison methods can be used such as Mean Squared Error, and Normalized Root Mean Squared Error Fitness percentage ($NRMSE - fitness(\%) = 100(1 - \frac{\sum(y-\hat{y})^2}{\sum(y-\bar{y})^2})$.
- Residual analysis: Analyzing autocorrelation and cross-correlation of the model residuals with input data.

- Analyzing model responses such as Impulse and Step Response Plots.
- Price of the model: Compare models using the Akaike Information Criterion or Akaike Final Prediction Error.

For a sufficient model validation task, a combination of these methods is typically required. Nonetheless, the user objectives in modeling, knowledge of the problem, and understanding of the required performance are the essential parts of model validation and system identification as a whole. This is why some people believe that System identification is the art and science of building mathematical models of dynamic systems from observed input-output data [43].

3. Simulation modeling

Every designed controller should be implemented on a real plant in order to study its performance and effects. However, in the case of the cement production process, it is not possible to test the controller on the real plant due to the high risks and costs. A comprehensive model that can simulate the real plant can be a solution to this challenge. This model can serve as a replacement for the real plant to test the controllers. The simulation model does not have most of the limitations of the control-oriented model. Thus it can be more complex, and nonlinear than the control-oriented model. In this section, we will discuss the development of the simulation model with a specific focus on machine learning models. Machine learning can be used to develop models that accurately predict dynamic system output for a given input. There are numerous machine learning algorithms available for modeling dynamic systems. The choice of machine learning algorithm depends on the specific characteristics of the system being modeled and the availability of data. In this work, we will focus on sequential modeling in machine learning due to time dependencies between system variables. Shortly, sequence modeling involves applying machine learning to sequential data. Any data with a natural sequential order can be considered sequential data, such as time-dependent signals, speech, text, or music. It is well noted that system identification is closely related to sequence modeling and that the current trends in this field may have a significant impact on the field of system identification. In this section, we will discuss various common sequence modeling techniques using machine learning. Starting with recurrent neural networks (RNNs) we ended up utilizing the most state-of the art in sequence modeling namely attention mechanism and transformers.

3.1 Recurrent Neural Networks

Recurrent neural networks are a type of artificial neural networks that is particularly suitable for tasks that involve sequential data, such as time series data, natural language processing, and speech recognition. While in standard neural networks, all inputs and outputs are independent of one another, in RNNs, different time steps are processed in the same unit. The fundamental characteristic of RNNs is their hidden state, which remembers certain information about the sequences they process. Due to its ability to remember the previous input, this state is also referred to as a Memory State. To generate the output, RNNs apply the same parameters to all inputs or hidden layers. As a result, the complexity of the model is reduced, as compared to other neural networks.

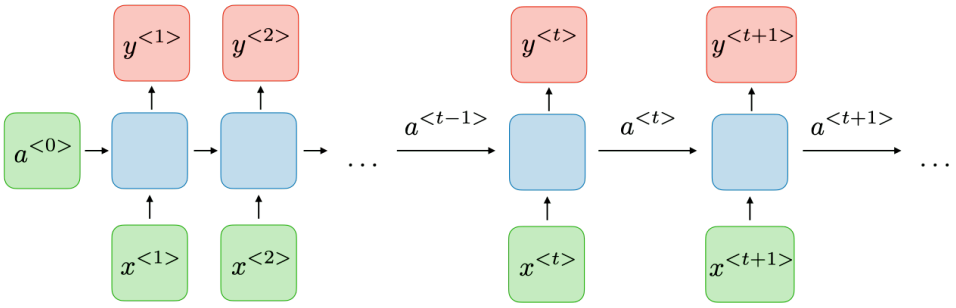


Figure 3.1 Recurrent Neural Networks (RNNs) structure

RNNs follow the same input and output architecture as other deep neural networks. Nevertheless, there are differences in the manner in which information is transferred from input to output. In contrast to deep neural networks, in RNN the weight matrix for each dense network remains the same. A Recurrent Neural Network is composed of multiple fixed activation function units, one for each time step. Each unit has a hidden state that determines its internal state. As a result of this hidden state, the network is capable of retaining past knowledge at any given point in time. As the network learns about the past, the hidden state is updated at every time step. For each time-step t, the activation $a^{<t>}$ and the output $y^{<t>}$ are expressed as follows:

$$\begin{aligned} a^{<t>} &= g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a) \\ y^{<t>} &= g_2(W_{ya}a^{<t>} + b_y) \end{aligned} \tag{3.1}$$

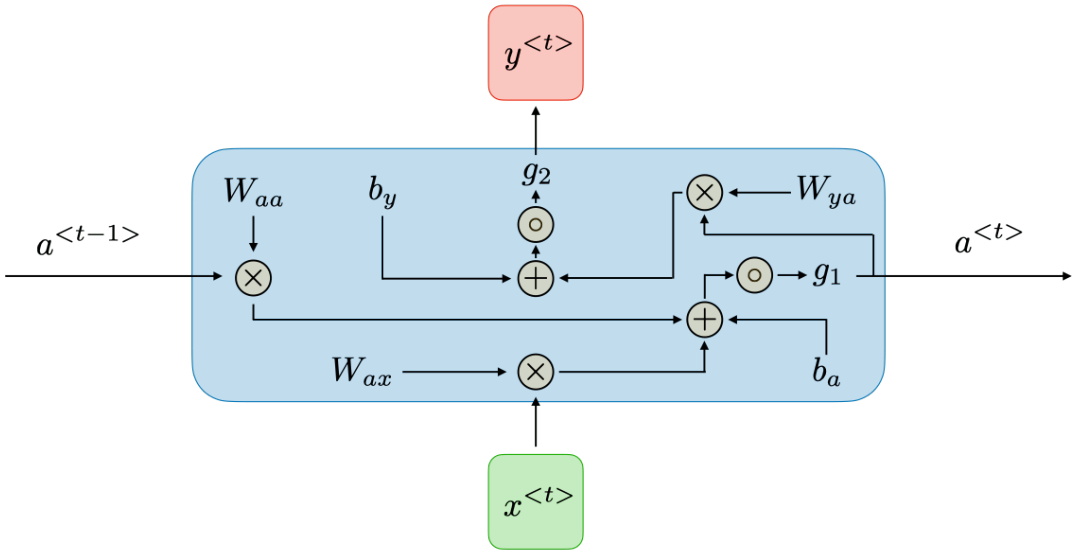


Figure 3.2 RNN cell structure

The parameters are updated using backpropagation. As RNNs work on sequential data, they use updated backpropagation, called backpropagation through time. In a normal RNN, only one input is sent into the network at a time, and only one output is obtained. On the other hand, backpropagation uses both current and prior inputs. This is referred to as a timestep, and one timestep contains multiple data points from the time series entering the RNN simultaneously. Once the neural network has been trained and given an output, its output is used to calculate errors. Afterward, the weights are recalculated and adjusted according to the newly calculated network.

There are four types of RNNs based on the number of inputs and outputs in the network.

- One to One: This type of RNN operates like a standard neural network, often referred to as a vanilla neural network, and it involves a single input and yields a single output.
- One to Many: In this RNN variant, a single input is linked to multiple outputs. One application of this type can be seen in the development of image captions, where an image is input and the network generates a sentence containing various words based on the image.
- Many to One: In this type of RNNs, numerous inputs are taken at various stages, and one output is produced. As an example, it is commonly used in tasks such as sentiment analysis, in which multiple words are used as input and a sentiment prediction is generated for the entire sentence as an output.

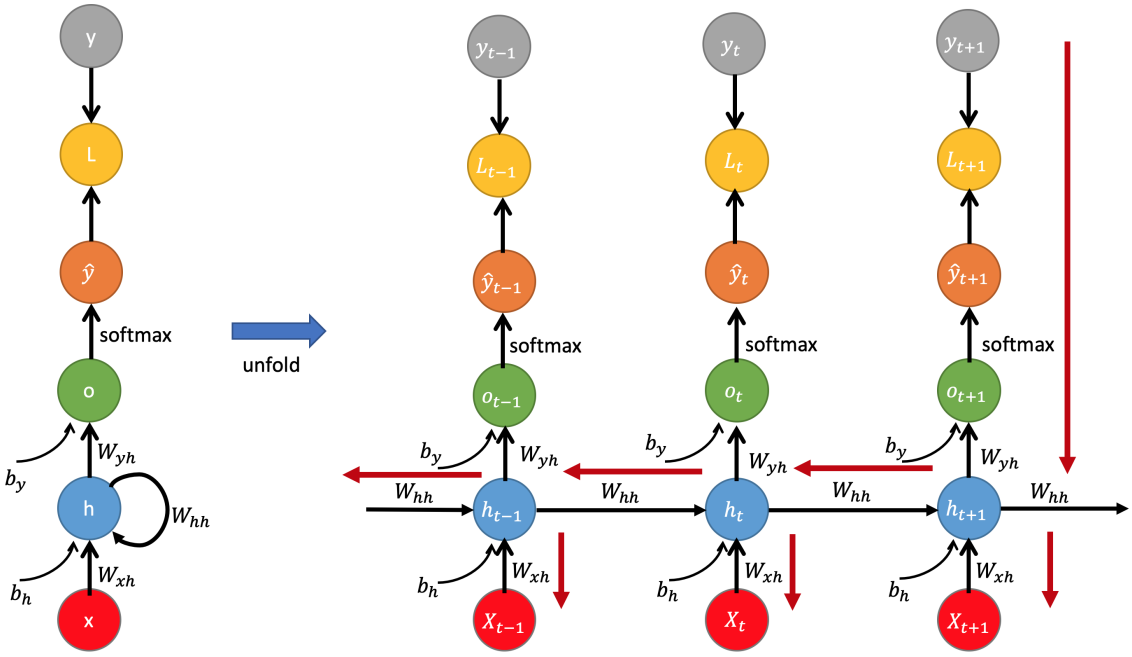


Figure 3.3 Backpropagation through time

- Many to Many: This type of RNN involves multiple inputs and produces multiple outputs. For instance, during translation, it takes several words from one language as input and generates multiple words in another language as output.

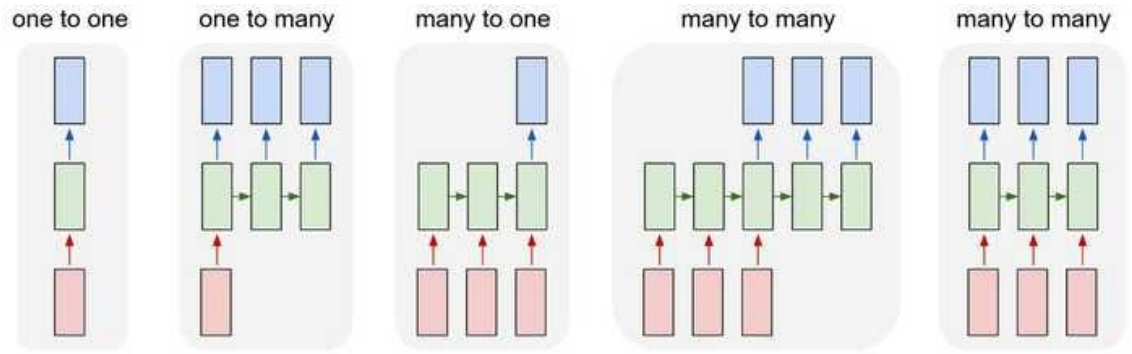


Figure 3.4 Various RNN Types

The main problem with RNNs is that they can be prone to vanishing or exploding gradients. It means that the error signal can become extremely small or extremely large as it propagates through the network, making learning difficult for RNNs. To address this problem, several techniques can be used, including LSTMs (long short-term memories) and GRUs (gated recurrent units) networks. LSTMs and GRUs are specifically designed to address the vanishing gradient problem. They accomplish this by controlling the flow of information through the network using gates. There are three gates in an LSTM:

- Forget gate: The forget gate determines which information from the cell state should be erased.
- Input gate: This gate decides which information to add to the cell state.
- Output gate: An output gate is responsible for determining which information is to be output from a cell.

Gates are implemented as neural networks, and they are updated at every time step. The gates are updated by the current input and the previous state of the LSTM. The gates are also adjusted according to the forgetting factor, which is a measure of how much of the past information should be retained. The gates help to determine which neurons will fire during the next step of the LSTM. This is the key to the LSTM's ability to learn complex tasks and long-term dependencies.

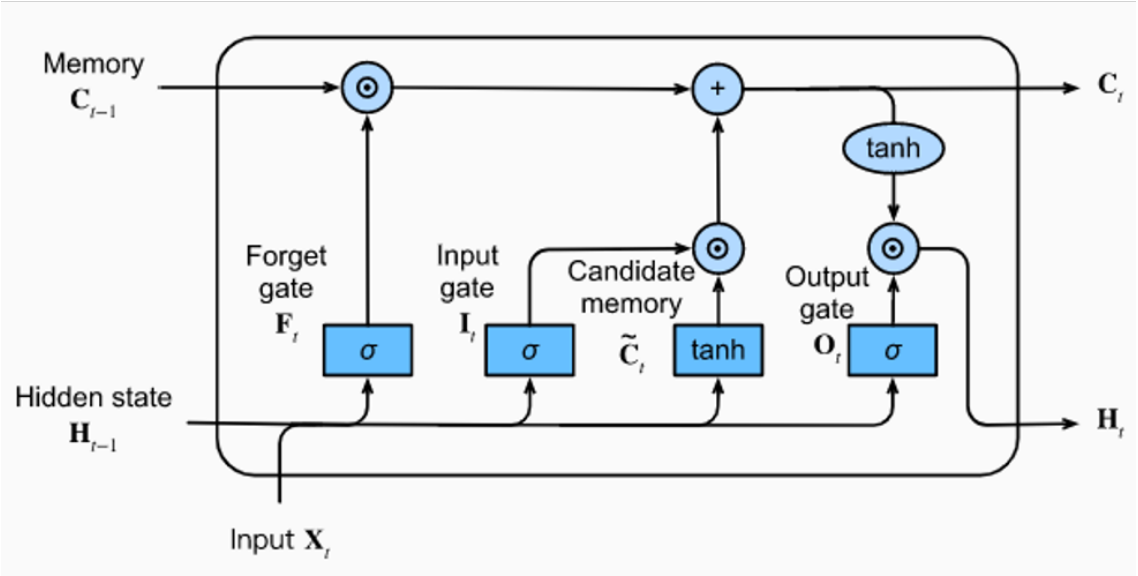


Figure 3.5 Long short-term memory (LSTM) cell structure

3.2 Transformers

Although LSTMs are designed to overcome most of the limitations of traditional RNNs, such as the vanishing gradient problem, they are usually slow to train due to their architectural design, which requires the data to be introduced sequentially to the network. This prevents parallelization of the training process, which is why GPUs are used in deep learning calculations. The LSTM also has limitations in terms of addressing long-range dependencies. This is because the memory capacity

of the LSTM is not large enough to store the time-series data, which prevents it from capturing the long-range dependencies. Additionally, the vanishing gradients problem can also be difficult to resolve since it requires a large number of training iterations, which can take a considerable amount of time. This means that the LSTMs can have challenges in tasks that require long-term predictions, such as forecasting or predicting the future behavior of a system. Transformers are a type of deep learning model introduced in a paper called "Attention is All You Need" in 2017 [45]. Transformers are designed to handle sequential data in a fundamentally different way compared to RNNs and LSTMs. The key innovation of Transformers is the attention mechanism. Instead of processing data sequentially, as RNNs and LSTMs do, Transformers can consider the relationships between all elements in a sequence simultaneously. This is achieved by calculating attention weights for each pair of input elements, and determining how much attention should be given to each element when producing an output. This dynamic attention enables Transformers to capture long-term dependencies and context effectively. This mechanism utilizes three main components, namely the queries Q, the keys, K, and the values, V, then performs the following operations:

1. An average score is computed by matching each query vector, q, against a database of keys. The matching operation is computed as the dot product of the specific query under consideration and each key vector, k_i. The dot product is a scalar value calculated by multiplying the two vectors together. It is a measure of how closely the vectors match, with a higher dot product indicating a closer match.

$$e_{q,k_i} = q.k_i \tag{3.2}$$

2. Weights are generated by passing the scores through a softmax operation: Thus, weights can be used to adjust the importance of features in a model.

$$\alpha_{q,k_i} = softmax(e_{q,k_i}) \tag{3.3}$$

3. Afterwards, the attention vector is calculated as the weighted sum of the value vectors, $V_[k_i]$, with each value vector corresponding to a key:

$$attention(q, K, V) = \sum_i \alpha_{q,k_i} V_{k_i} \tag{3.4}$$

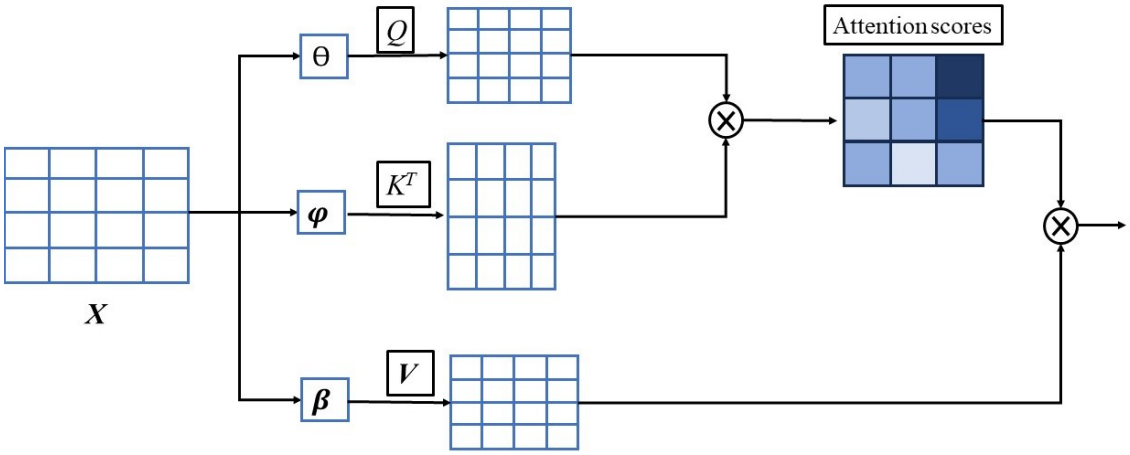


Figure 3.6 Attention Mechanism

Transformers consist of an encoder and a decoder, both composed of multiple layers of self-attention mechanisms and feedforward neural networks. To account for the order of elements in a sequence, positional encoding is added to the input embeddings. Figure 3.7 shows the architecture of the transformer proposed by Vaswani et al., and its main components: encoder and decoder. Inputs for both components are passed through a positional encoder that encodes order information and adds it directly to the vector of input data. The encoder consists of six stacked encoder layers. In each layer, there is a multihead self-attention sublayer and a feed-forward sublayer. The multi-head self-attention sub-layer receives queries (**Q**), keys (**K**), and values (**V**). By using multihead attention, the model can simultaneously attend to information from different representation sub-spaces. In the Vanilla Transformer, the multihead self-attention sublayer consists of six scaled-dot-product heads. Afterward, a residual connection is applied around the multihead attention, followed by layer normalization. The second part of the encoder, referred to as the feed-forward sublayer, contains two dense layers with linear and rectified linear unit (ReLU) activation functions. This layer projects the vector into a larger space, which allows for easier extraction of the required information and then projection back into the original space. Like the multihead self-attention sublayer, a residual connection is employed before applying layer normalization.

The decoder is similar to the encoder consisting of 6 layers. In addition to the multihead self-attention and feed-forward sublayers, the decoder layer includes a third sublayer performing multihead attention at the encoder outputs. Furthermore, the multihead self-attention sublayer is changed to a masked multihead self-attention sublayer, as shown in figure **??**, which is similar to the multihead self-attention sublayer except that masked scaled dot-product attention is used instead of scaled dot-product attention. This prevents later leaks of information by limiting predic-

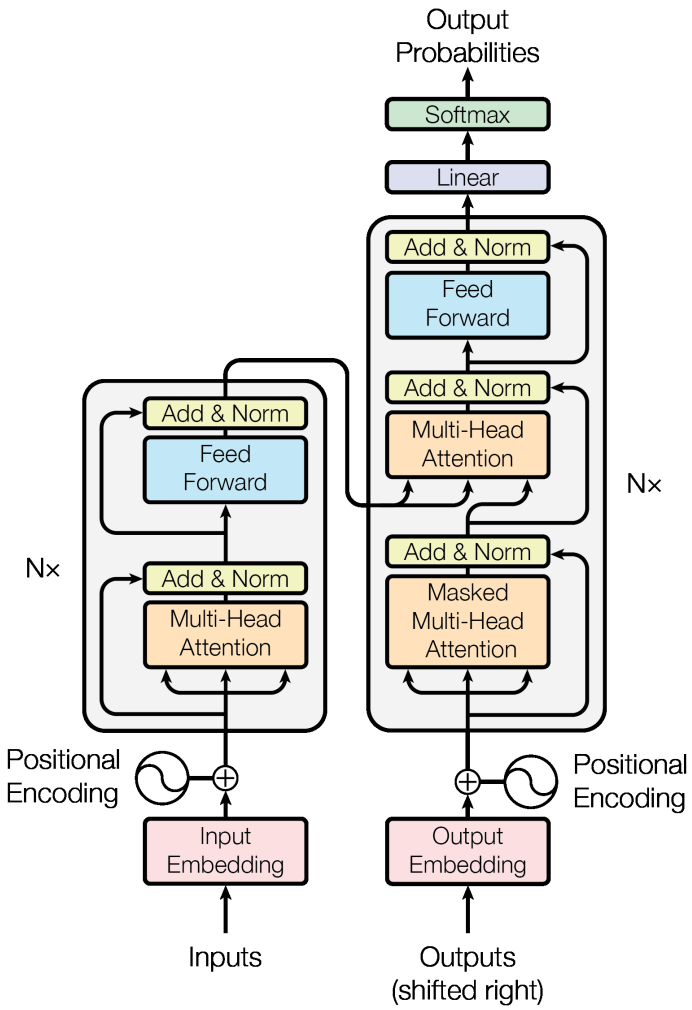


Figure 3.7 Transformer Model Architecture

tions to outputs that are known. When it comes to training these models, one crucial aspect is "teacher forcing." This method is commonly employed during the training phase to teach the model how to generate sequences effectively. In teacher forcing, the model is given the correct target output from the training data at each step of sequence generation. This means that during training, the model receives a ground truth token as input, helping it to understand the relationships between input and output tokens. While teacher forcing accelerates training by providing clear supervision, it can lead to a discrepancy between training and inference, where the model struggles with generating accurate outputs when it doesn't receive ground truth inputs during inference.

Transformers have been shown to be very effective for time series forecasting and dynamic systems modeling tasks. They have achieved state-of-the-art results on a variety of these tasks, such as stock price forecasting, chemical production process modeling, and modeling physical systems. They are also relatively easy to use, with a relatively low training overhead. Transformers are also scalable, and able to handle

large amounts of data. They are also robust, and able to learn and generalize from small amounts of data. Therefore, Transformers offer a strong set of capabilities and advantages over traditional machine learning approaches [46], [47], [35].

4. Model implementation and results

In this section, we will discuss how to apply the models introduced in the previous section to the specific cement production process at the Akcansa cement plant. We will start by explaining how data is collected and prepared. Next, we will present the implementation and results of the physics-inspired or gray-box model and black-box system identification model. Then, we will implement and discuss the results of various machine learning models discussed in the simulation modeling section. Finally, we will draw conclusions about the modeling process.

4.1 Data description and preparation

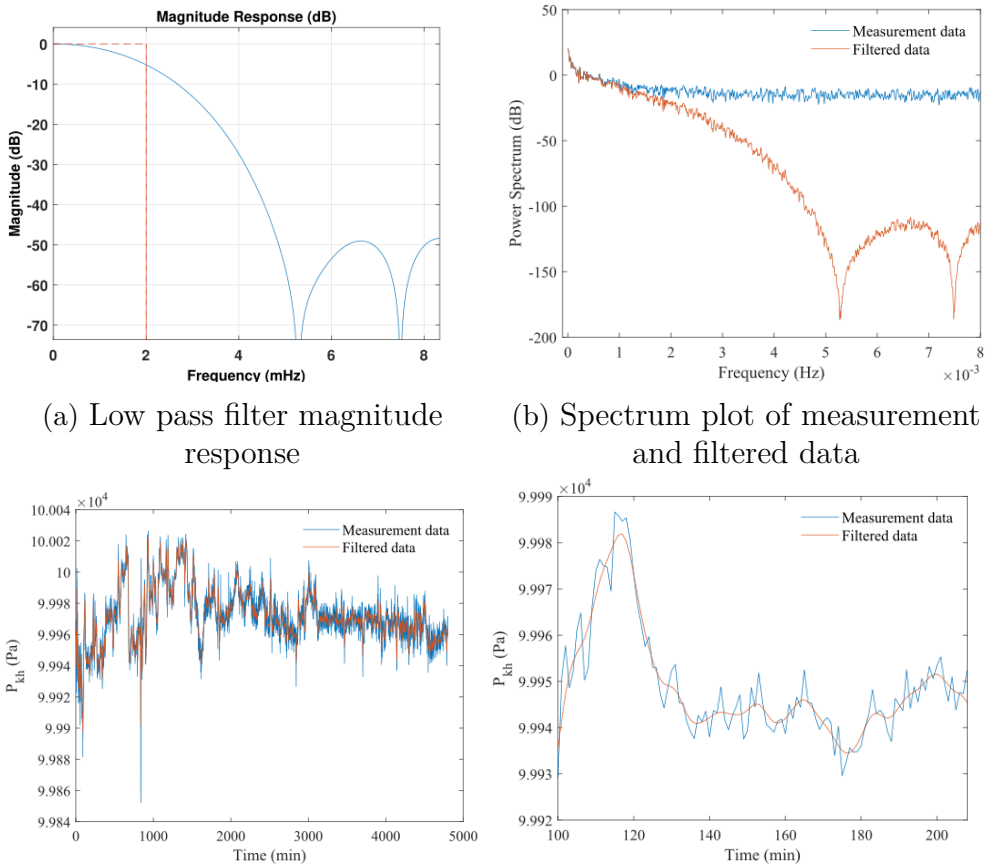
Using process analysis and descriptions, we identify the key variables that must be controlled to achieve optimal process operation. We also pinpoint the internal variables, which represent the internal component states, that will be used to model the process variables. Additionally, we recognize the process inputs that influence and regulate both the process and its variables. For the calciner, the inside temperature and pressure (P_{cal}, T_{cal}) are important states to monitor and control since they directly affect the calcination reaction inside the calciner and the gas flow between various parts of the process. Likewise, in the rotary kiln, the inlet pressure, temperature, kiln hood pressure and temperature (P_{kh}, T_{kh}, P_{ki}, T_{ki}) are critical variables due to the same above-mentioned reasons. Lastly, in the cooler part, the clinker production temperature and rate (T_{cl}, PR_{cl}) are the focus of process control since the clinker temperature determines the mechanical and chemical properties of the clinker. It goes without saying that a cement production plant's clinker production rate is of importance. Considering the process inputs, clearly, the raw material feed rate and fuel feed rate into the calciner and rotary kiln (F_{cal}, F_k, M_r) are the main material inputs to the system. On the other hand, for the gas phase in the process, the fans' speed are the control inputs including the ID fan, cooler undergrate fans,

and cooler exhaust fan [13], [14], [48], [12]. Tables 4.2 lists the process variables, internal variables, and process inputs. A description of each symbol can be found in the nomenclature.

Table 4.1 Process Variables, Internal Variables, and Process Inputs

Process variables:	P_{kh} , T_{kh} , P_{ki} ,T_{ki} ,P_{cal} , T_{cal} , T_{cl} , PR_{cl}
Internal variables:	P_{cyc}, T_{cyc}, T_{ct}, P_{ct}
Process inputs:	F_{cal}, F_k, M_r, S_{id}, S_cf, S_{ef},

Data for the modeling was collected from Akcansa's cement production plant at a sampling rate of 1 minute during plant normal operations. Different data amounts are considered for different modeling approaches as they need various amounts of data. For the physics-inspired part, data from 5 consecutive days were used. While, for the machine learning part, a larger number of 30 days of data is considered. Moreover, since the plant is heavily controlled, most of the time, the process variables are around some set points without much change, while it is imperative to utilize sufficiently rich data for modeling and identification of dynamic systems. Variables in the system are mainly dynamic when the system undergoes changes or if it becomes out of control. Thus, after considering a long period of data (more than 2 years), we attempted to identify the most dynamic areas of the data and to utilize them for modeling tasks. A low-pass filter was then applied to the data to attenuate high-frequency noise. We decided to proceed with a 10th-order low-pass filter with a cutting frequency of 0.002 Hz, noting that the sampling frequency is 0.016 Hz. The power spectrum of the designed low-pass filter and its performance on the kiln hood pressure data as a sample of process data is shown in Fig. 4.1. Finally, in every modeling task, data is divided into two parts, training and validation. Where 70 percent of data is considered for training and 30 percent for validation.



(a) Low pass filter magnitude response

(b) Spectrum plot of measurement and filtered data

(c) Time plot of measurement and filtered data

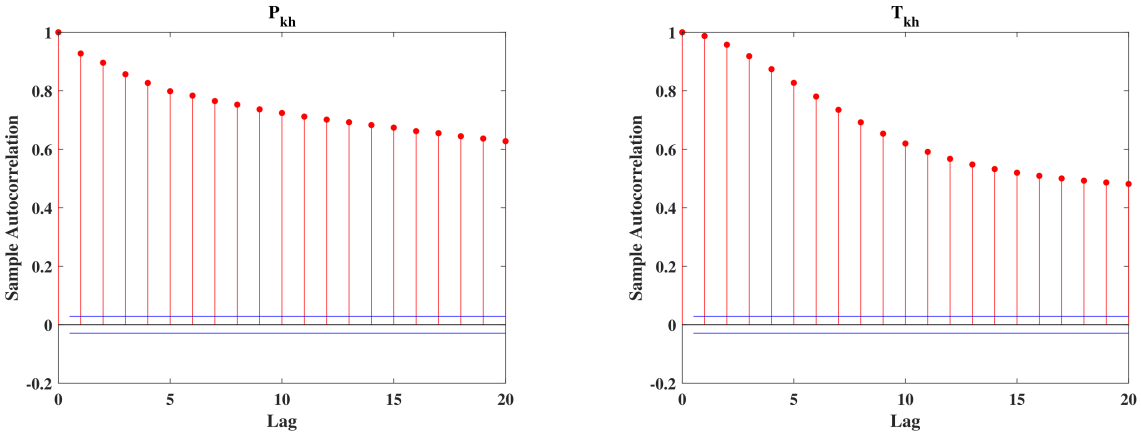
(d) Zoomed Time Plot of Measured and Filtered Data

Figure 4.1 Effect of Filtering on Kiln Hood Pressure: A Sample from Process Data

4.2 Control-oriented model implementation and results

As discussed in the modeling part, system identification requires the selection of a model structure and the implementation of parameter identification algorithms within the model structure chosen. It is generally recommended to select the simplest model first, and then increase the model complexity if the simple model does not provide satisfactory results. Therefore, we have chosen to start with the ARX model structure in this study. Having selected the model structure, the next step involves deciding on the model order(s). In general, the aim should be to not use a model order higher than necessary. In the absence of enough physical knowledge to determine the model order based on them, statistical relationships such as autocorrelation and cross-correlation can be used to determine the model order. However, due to the feedback in the plant, collected data has a correlation with all its previ-

ous values. For instance, the autocorrelation of Kiln hood pressure data is shown in Figure 4.2. Therefore, simple autocorrelation is not informative enough, in this condition, instead partial autocorrelation can be helpful. Partial autocorrelation (PACF) is the correlation between the time series at two different lags while controlling the effect of any intermediate lags. PACF for 4 sample states are plotted



(a) Kiln hood pressure ACF (b) Kiln hood temperature ACF

Figure 4.2 Autocorrelation of Kiln hood pressure and temperature data

in Fig. 4.3. PACF diagrams show that different states (variables) are correlated with different previous values. However, in most cases, the significant correlation is limited to three lags. Consequently, it is reasonable to consider 3 as the number of previous outputs when modeling the current output. A similar approach can be followed for determining the model order for inputs. Investigating the partial cross-correlation between various inputs and outputs reveals that only 1-2 previous inputs are correlated to the current output. Therefore, 2 can be considered for the ARX model external part order. Nevertheless, it is essential to try other model orders that are close to those identified by statistical analysis. In this work, after statistical analysis and trial and error, the model orders were determined to be (5,5). In addition, by setting the model orders to 5 for all states, we are able to reduce the number of design parameters, otherwise, we would have to test a large number of different combinations of model orders. As a result, in this work, we decided to choose an ARX (5,5) for the model structure. As a common black-box model, we can derive a model between the process inputs and process variables without considering any physical insights and internal variables. We name this as 1st black-box model. An alternative is to consider the internal measurements as well which should be included in the model outputs part (we name it 2nd black-box model). However, a gray-box model can be developed by considering the internal variables (measurements) and integrating the relationships between various components discovered by the first principles. One way of achieving this integration is to select the regressors

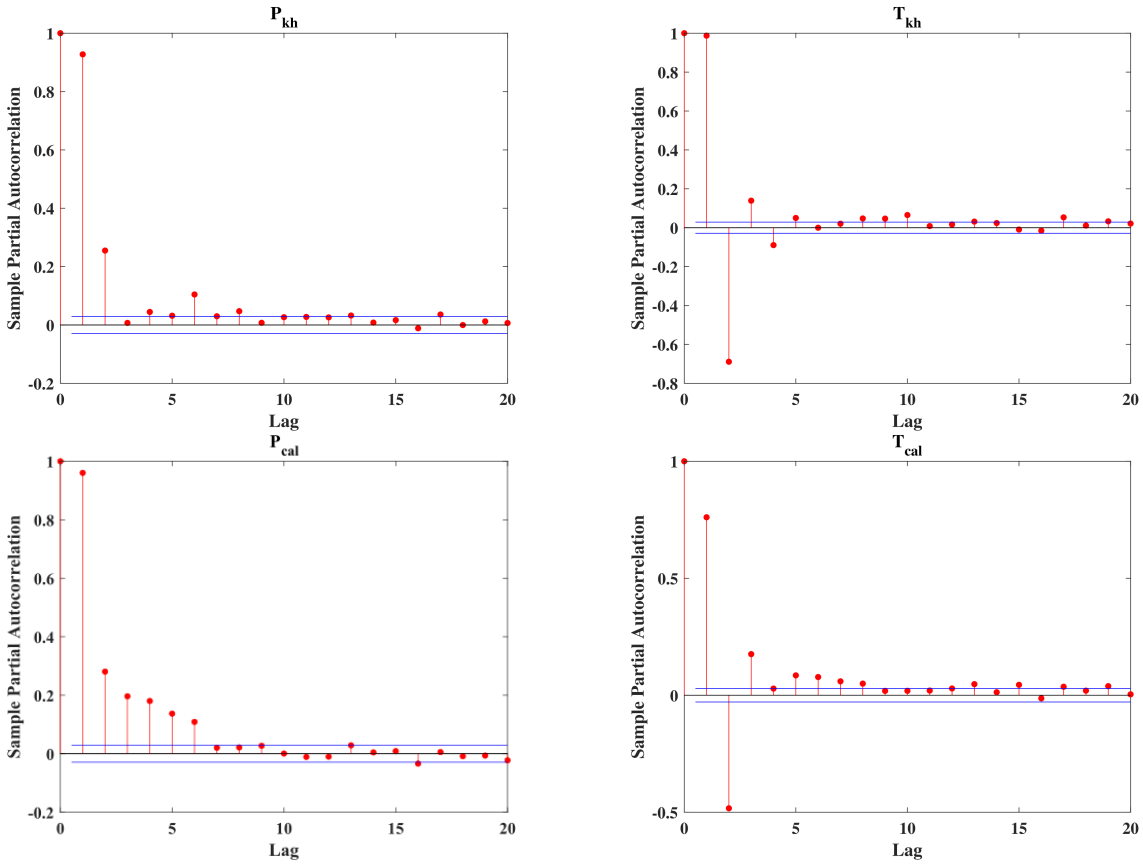


Figure 4.3 Partial Autocorrelation of 4 sample states

for modeling each state to include only the affecting states and inputs. As a result, the black-box and gray-box models can be written in the following matrix format:

$$Y_{k,b} = \overbrace{\begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,p} \end{bmatrix}}^{A_b} \hat{Y}_{k,b} + \overbrace{\begin{bmatrix} b_{1,1} & \cdots & b_{1,m} \\ \vdots & \ddots & \vdots \\ b_{m,1} & \cdots & b_{m,q} \end{bmatrix}}^{B_b} \hat{U}_{k,b} \tag{4.1}$$

Where, $Y_{k,b}$, $U_{k,b}$, $\hat{Y}_{k,g}$, $\hat{U}_{k,g}$ are the black-box model inputs and outputs and their p,q previous values respectively:

$$Y_{k,b} = \begin{bmatrix} P_{kh,k} \\ T_{kh,k} \\ P_{ki,k} \\ T_{ki,k} \\ P_{cal,k} \\ T_{cal,k} \\ T_{cl,k} \\ PR_{cl,k} \end{bmatrix}, \quad U_{k,b} = \begin{bmatrix} S_{id,k} \\ S_{ef,k} \\ F_{ca,k} \\ S_{cf,k} \\ F_{k,k} \\ M_{r,k} \end{bmatrix} \tag{4.2}$$

The parameter matrices A_b, and B_b are estimated by parameter estimation methods like least squares or maximum likelihood method.

The gray-box model has a similar format to the 1st black-box model except that the model outputs matrix includes the process variables as well as internal variables. Moreover, the parameter matrices A_g, and B_g are sparsified based on the relations found in the physical modeling part. In this case, the parameters associated with non-affecting states or inputs for predicting the desired state can be fixed to zero. By doing so, the coefficient matrices in the gray-box model will have fewer free parameters. In this work, the number of process states to be predicted is 8, and the number of inputs is 7. Thus, the selected black-box models have 960, and 1680 free parameters. Upon integrating the physical relations into it, the number of free parameters is reduced to 485, which is approximately 50% and 70% reduction.

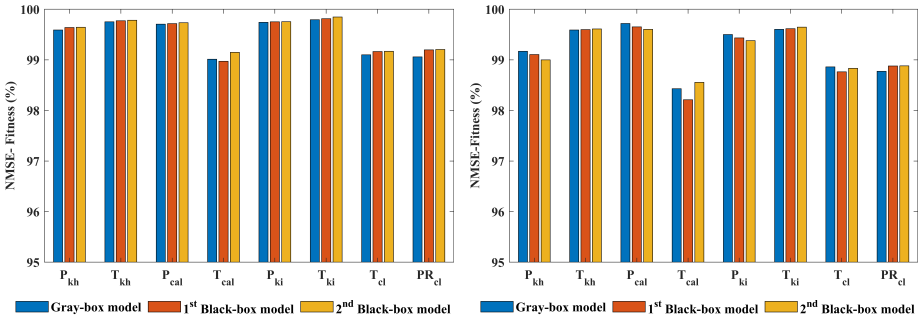
$$Y_{k,g} = \overbrace{\begin{bmatrix} a_{1,1} & 0 & 0 & \cdots & a_{1,n} \\ \vdots & 0 & 0 & \cdots & \vdots \\ a_{n,1} & 0 & 0 & \cdots & a_{n,p} \end{bmatrix}}^{A_g} \hat{Y}_{k,g} + \overbrace{\begin{bmatrix} b_{1,1} & 0 & 0 & \cdots & b_{1,m} \\ \vdots & 0 & 0 & \cdots & \vdots \\ b_{m,1} & 0 & 0 & \cdots & b_{m,q} \end{bmatrix}}^{B_g} \hat{U}_{k,g} \tag{4.3}$$

$$Y_{k,g} = \begin{bmatrix} P_{kh,k} \\ T_{kh,k} \\ P_{ki,k} \\ T_{ki,k} \\ P_{cal,k} \\ T_{cal,k} \\ T_{cl,k} \\ PR_{cl,k} \\ T_{cyc,k} \\ P_{cyc,k} \\ T_{ct,k} \\ P_{ct,k} \end{bmatrix} \tag{4.4}$$

Afterward, MATLAB was used to implement both the black-box and gray-box models, and analyze their performance on both estimation and validation data.

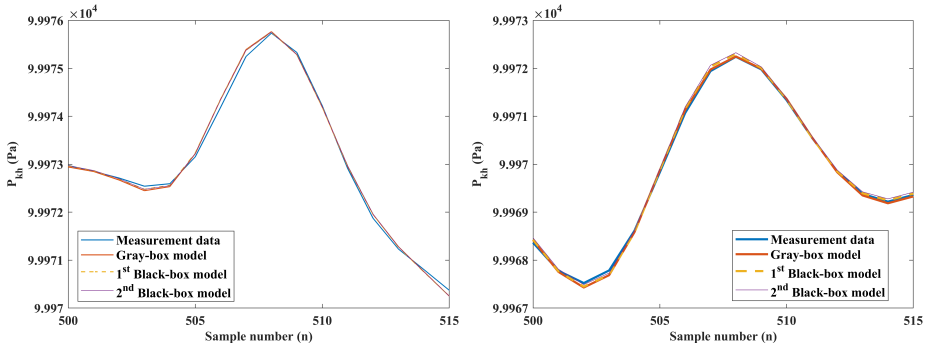
4.2.1 Prediction performance

First, the models' performance is analyzed for short-term predictions. Fig. 4.4 shows models' performance on 1-step ahead prediction on the estimation and validation data. In this case, both models perform very well for all process variables. As expected their performance is slightly better on the estimation data. Model performance on validation data, however, is the primary objective, since the models are intended to predict unseen data (predict the future). In this regard, the gray-box model performed marginally better than the black-box model.



(a) Estimation data (b) Validation data

Figure 4.4 Black-box and gray-box model performance for 1-step ahead prediction on estimation and validation data

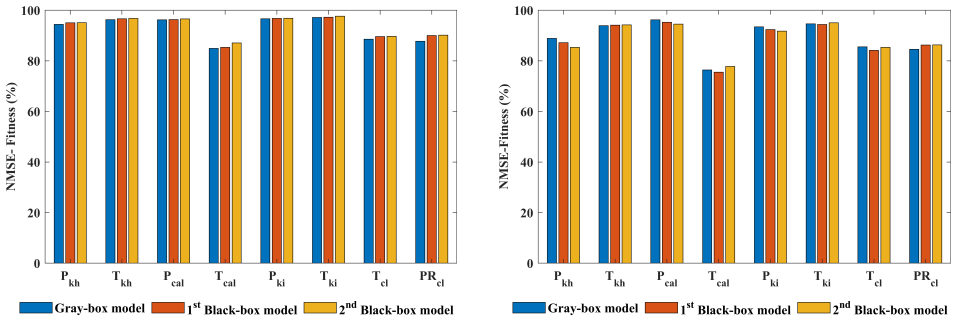


(a) Estimation data (b) Validation data

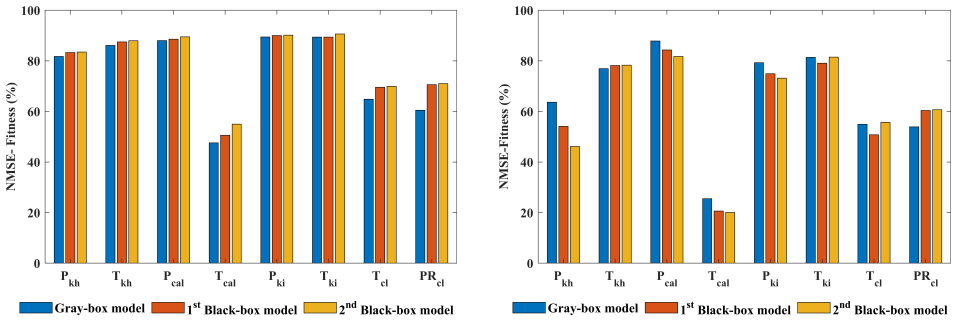
Figure 4.5 Black-box and gray-box model performance for 1-step ahead prediction on estimation and validation data, time plot, kiln hood pressure as a sample data

Modeling for control is primarily concerned with the model's performance on long-term prediction, known as multi-step ahead prediction. A model predictive controller employs a receding horizon algorithm that heavily relies on long-term predictions. In the case of false predictions, controller performance would be poor. As a result, it is imperative to examine the multi-step-ahead performance of every model that is developed for control purposes. Therefore, the developed models are

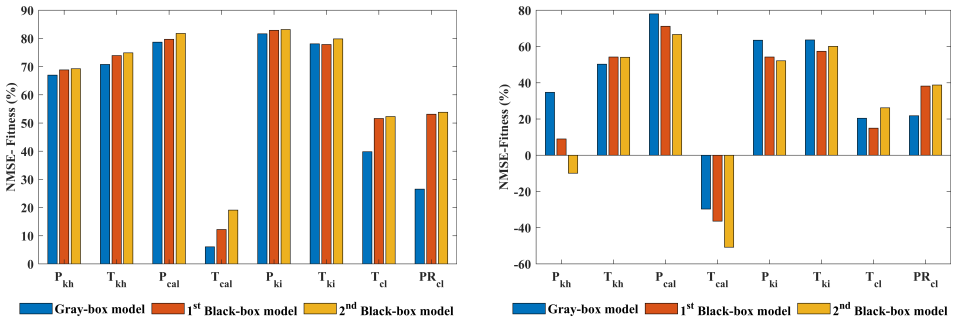
evaluated for various multi-step ahead predictions. Figure. 4.8 shows the models' performance for various multi-step ahead predictions on both estimation and validation data. As the results suggest both models are performing reasonably well on multi-step ahead predictions. However, the gray-box model outperforms the black-box model in validation data which proves the benefits of integrating physical knowledge into black-box models. This better performance in the gray-box model can be also associated with over-fitting. Overfitting is a problem where a model learns the training data too well, including the noise and outliers. As a result, the model fits the training data almost perfectly, but its performance on new, unseen data drops significantly. When models are complex, they tend to over-fit the training data and perform poorly on new, unseen data. The black-box models have more free parameters (1680, 960), thus it is more prone to over-fit on estimation data. On the other hand, the gray-box model has fewer free parameters (485) which makes it more flexible towards new data. It also should be noted that both models' performance deteriorates as the prediction horizon increases which is expected since a longer-term prediction is harder than short-term predictions. This is because both models aim to predict the future, which becomes increasingly uncertain and complex the further ahead they attempt to predict.



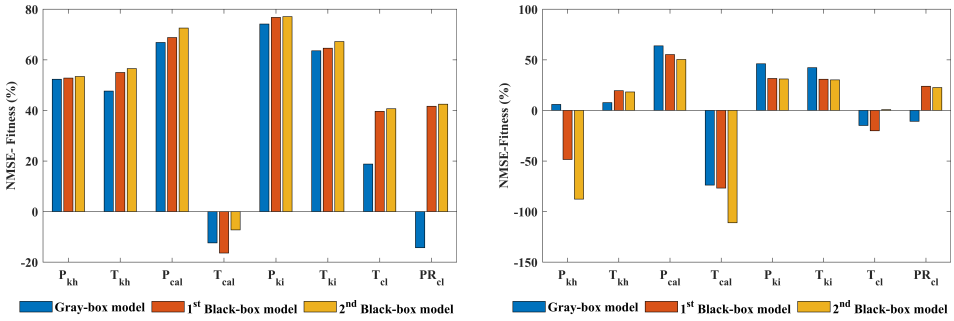
(a) Estimation data (b) Validation data



(c) Estimation data (d) Validation data



(e) Estimation data (f) Validation data

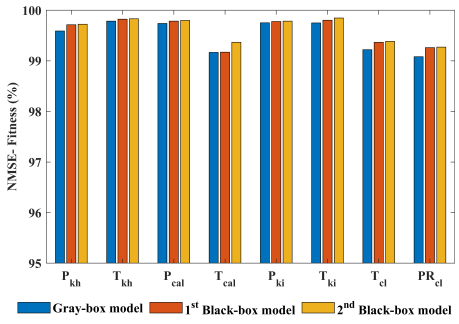


(g) Estimation data (h) Validation data

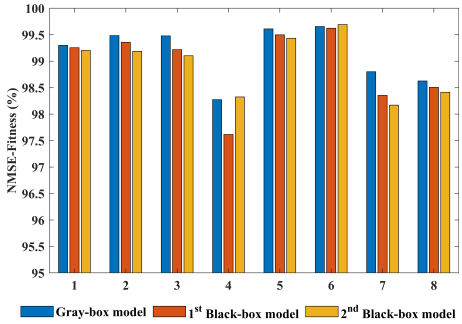
Figure 4.6 Black-box and gray-box models performance for various multi-step (3,5,7, 10 steps) ahead prediction on estimation and validation data

Lastly, in order to ensure that the results are not dependent on the data, two

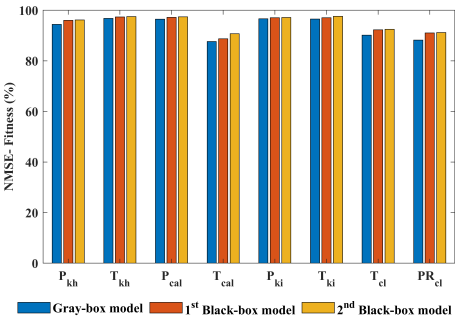
other datasets were used to test the models (gray-box and black-box models). The following are plots illustrating the prediction performance of the models at various steps. In order to save space and time, we have only presented the validation data results.



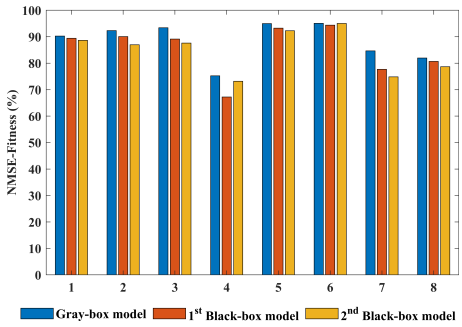
(a) Estimation data



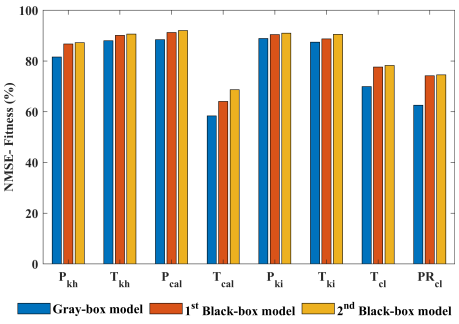
(b) Validation data



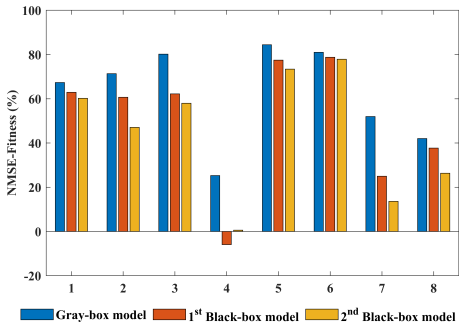
(c) Estimation data



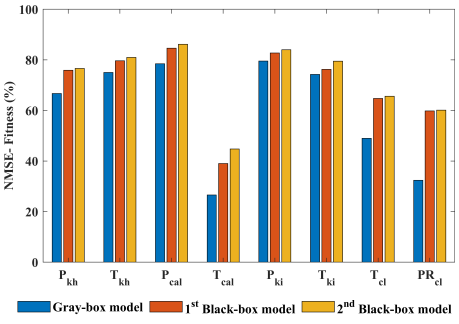
(d) Validation data



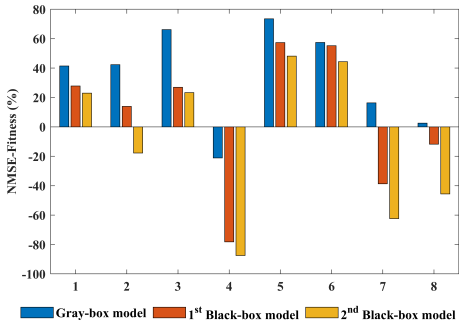
(e) Estimation data



(f) Validation data

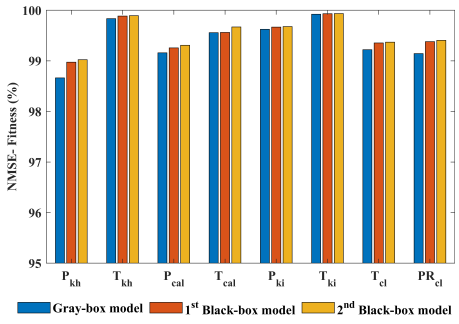


(g) Estimation data

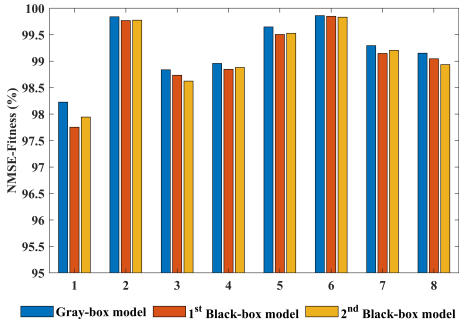


(h) Validation data

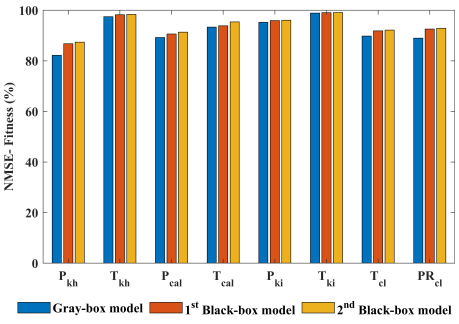
Figure 4.7 Gray-box and black-box models prediction performance on various steps ahead (1,3,5,7) on dataset 2



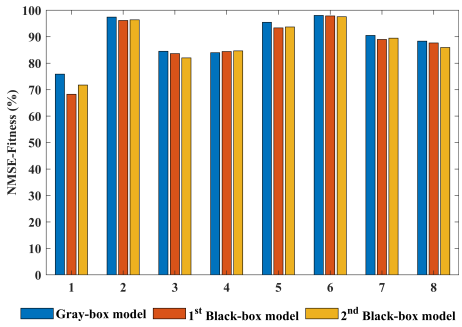
(a) Estimation data



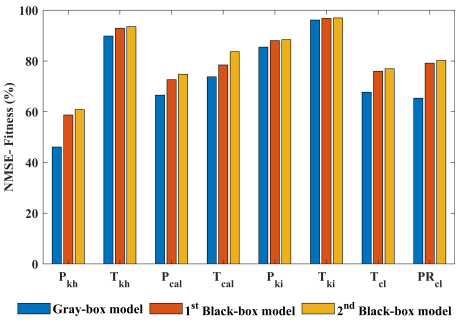
(b) Validation data



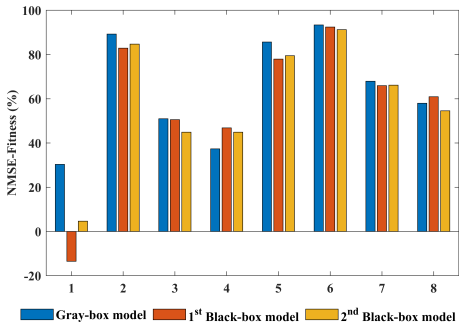
(c) Estimation data



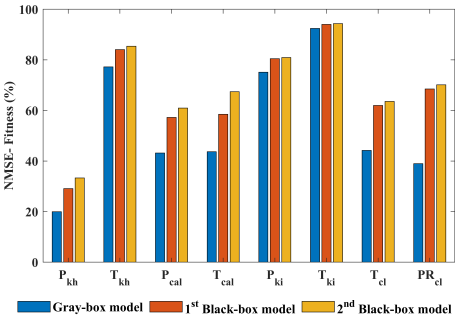
(d) Validation data



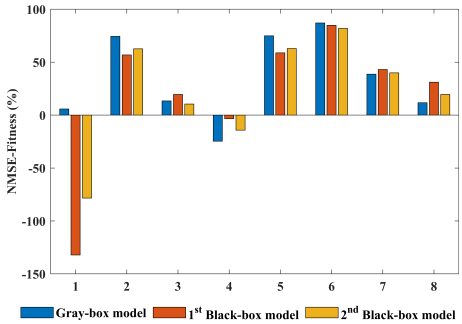
(e) Estimation data



(f) Validation data



(g) Estimation data

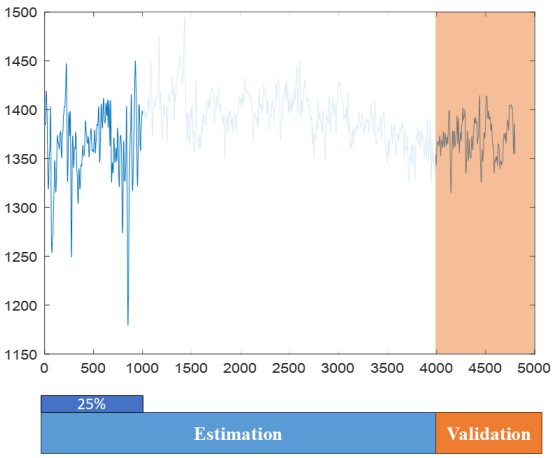


(h) Validation data

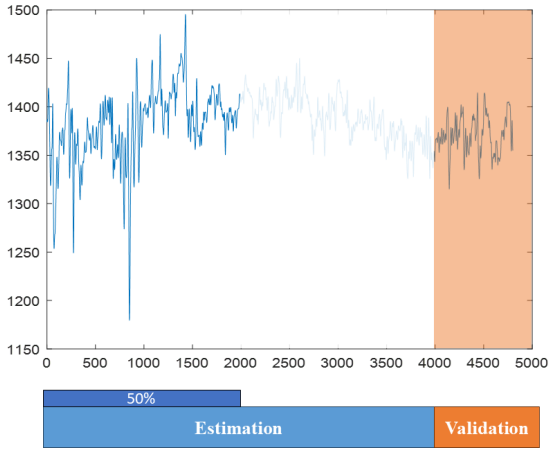
Figure 4.8 Gray-box and black-box models prediction performance on various steps ahead (1,3,5,7) on dataset 3

4.2.2 Model generalizability and effect of data quantity on model performance

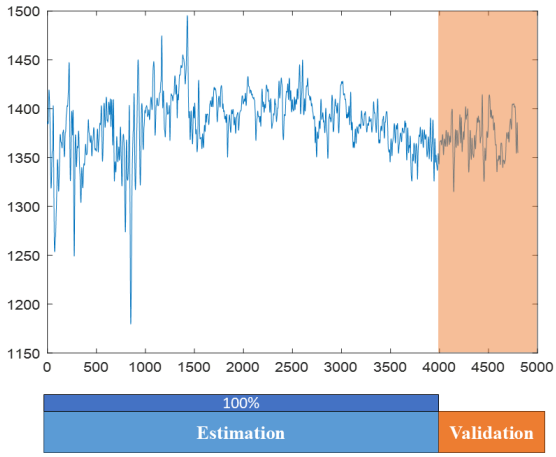
It is well known that data-driven methods heavily rely on data quality and quantity, and without reliable and sufficient data, their performance suffers. In contrast, physics-based methods are less dependent on data since they are based on physical principles. To investigate this issue, we trained both models with different amounts of data, 25%, 50%, and 100%. In Fig. 5.2, the performance of the models on a longer prediction horizon (5 steps ahead) is depicted. Results indicate that the black-box model loses performance when trained on a limited amount of data, whereas the gray-box model retains its performance. As a result of these capabilities, physics-inspired models can be particularly beneficial when there is limited data available or if it is expensive to conduct additional experiments and collect more data. This is because physics-inspired models are able to learn the underlying structure of the data and can use that knowledge to make predictions on larger horizons, even when the amount of data is limited. Additionally, physics-inspired models are also more efficient to run and consume fewer resources, making them an attractive option for applications that require real-time predictions.



(a) Estimation with 25% of estimation data

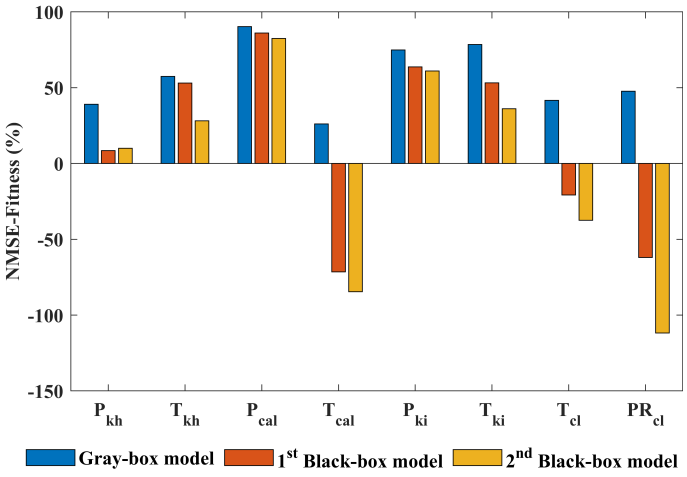


(b) Estimation with 50% of estimation data

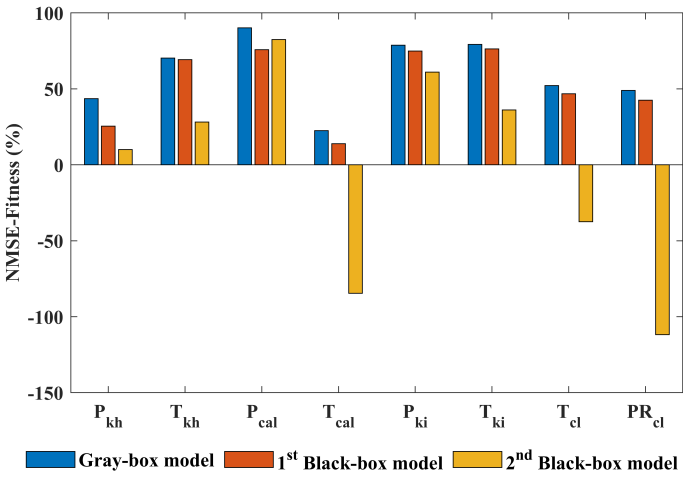


(c) Estimation with 100% of estimation data

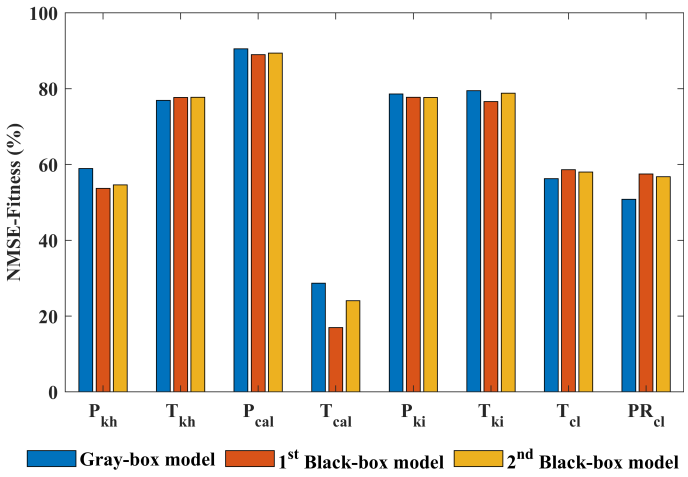
Figure 4.9 Validation Performance of Gray-box and Black-box Models for Multi-Step Ahead Prediction with Varied Training Data Percentages



(a) Trained with 25% of estimation data



(b) Trained with 50% of estimation data



(c) Trained with 100% of estimation data

Figure 4.10 Validation Performance of Gray-box and Black-box Models for 5 steps Ahead Prediction with Varied Training Data Percentages

4.3 Simulation models implementation and results

This section discusses the implementation and results of the machine learning models proposed in the simulation modeling section including RNN, encoder-decoder RNN, and Transformers. First of all each model structure is modified to meet the problem-specific needs. Afterward, the implementation of each method is discussed, followed by the analysis of their results. The models' performance is analyzed in terms of loss mainly the root mean squared value of model errors. Finally, it will discuss the strengths and weaknesses of each model. The results of this analysis will help inform the design of future machine learning models for predicting time series data. In the first place, the data should be prepared like the previous part. The only difference in this part is that the data must be normalized to the range (0, 1). This is essential to ensure that all features are on the same scale and that variables with large magnitudes do not dominate the learning process. Moreover, in this part, in order to make the results concise and more understandable we limited our analysis to the cooler significant states/variables including kiln hood pressure and temperature, clinker production rate, and clinker final temperature simulation. A very similar approach can be applied to all other important states in the process. To provide a baseline for comparing the performance of more complex models, it is beneficial to evaluate the performance of a simple model before building the trainable models. A simple model that can be used as a baseline is one that always predicts "No change". This is a reasonable baseline because the data changes slowly. However, the performance of this baseline will decrease as the prediction horizon increases.

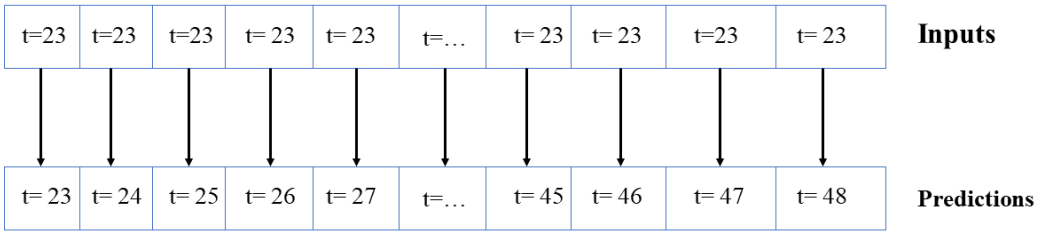


Figure 4.11 Baseline model

4.3.1 Recurrent neural networks (RNNs)

4.3.1.1 Single-shot RNN

Recurrent neural networks are mostly used to get a series of previous inputs and produce the current output. In other words, they are mostly used for 1-step ahead prediction. However, as it was discussed multi-step ahead prediction is required for control tasks, particularly MPC. Therefore, we designed and trained the recurrent neural network model to predict multi-steps ahead. One high-level approach to this problem is to use a "single-shot" model, where the model makes the entire sequence prediction in a single step. In this method, given a series of inputs, the model will predict many steps in the future instead of just one step. A simple schematic of this approach is presented in Figure 4.12 in which 24 previous inputs are fed into the model, and the model predicts the next 24 outputs at the same time. The single-shot RNN model has been designed with 3 LSTM layers with 64, 32, and 8 neurons respectively, and a final dense layer to reshape the model output to the output data shape.

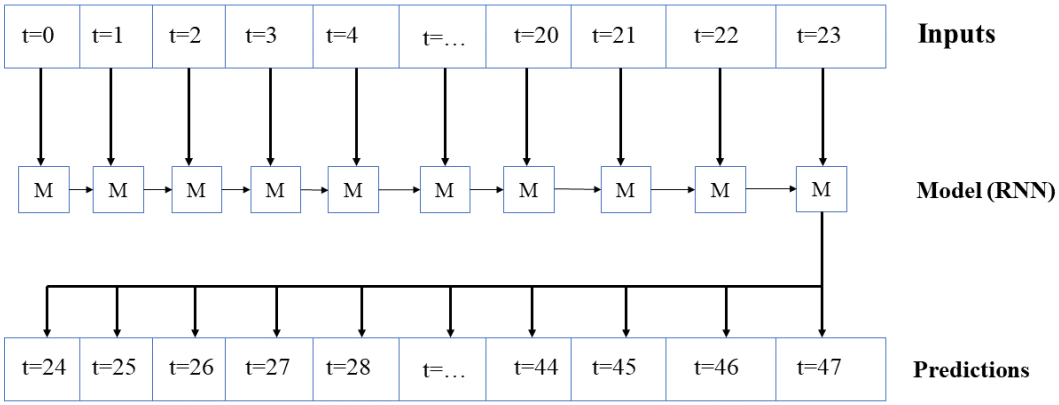


Figure 4.12 Single-shot RNN Model Structure

4.3.1.2 Encoder-decoder RNN

Single-shot RNNs do not consider the previous outputs and they predict the future values for multiple steps at the same time. This method limits their capability and accuracy. Feeding the previous output to the network can increase its performance. Using encoder-decoder architecture can make the models able to consider the previous outputs as well as previous inputs.

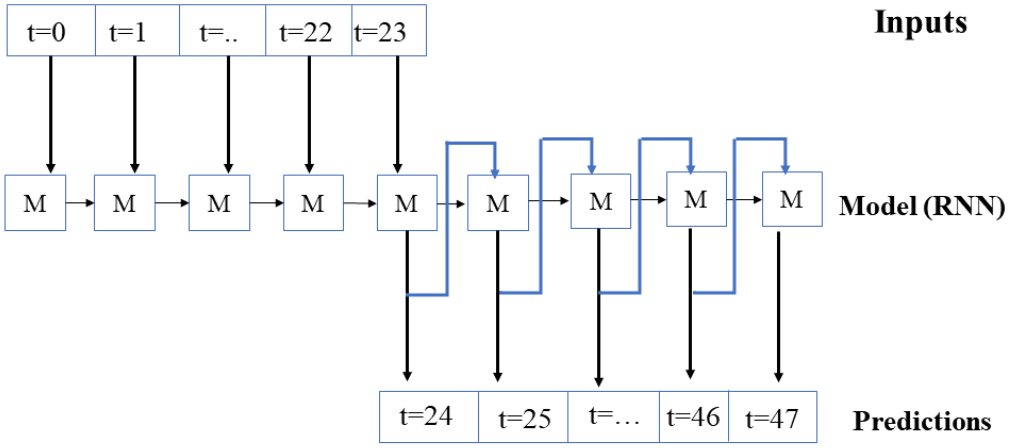


Figure 4.13 Autoregressive Multi-Step Predictor Model Structure

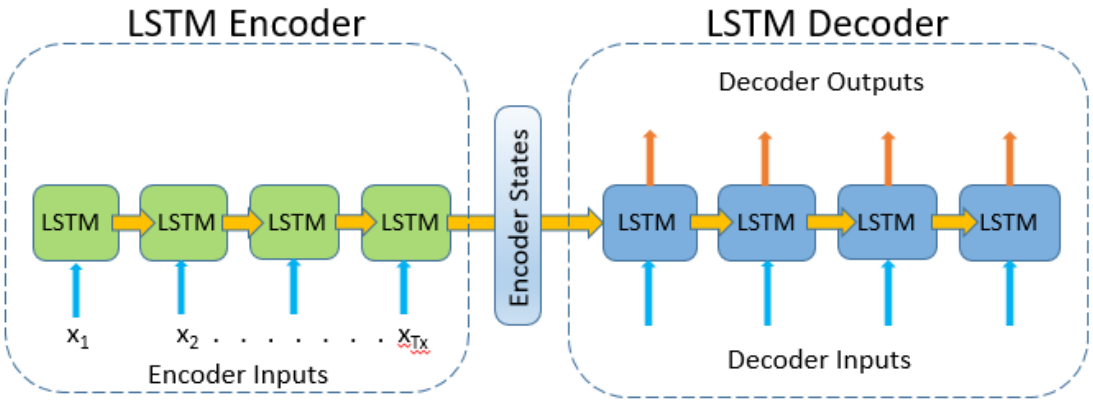


Figure 4.14 Encoder-decoder RNN Model Structure

Encoder-decoders are a type of neural network that consists of two parts: an encoder and a decoder. The encoder compresses the input data into a lower-dimensional representation, known as the latent space. This process allows the model to learn the most important features of the input data. The decoder then reconstructs the output data from the latent space. In the case of encoder-decoder-based RNNs, the encoder processes the input sequence data and extracts the context from it. This context is then fed into the decoder, which also receives the previous outputs. This allows the decoder to consider both the input sequence and the previous outputs when generating the next output. Due to its different structure, the encoder-decoder RNN training process is also different from classical ML models. Teacher-forcing is the most effective training method for encoder-decoder RNNs. It works by feeding the ground truth (actual) output from the previous time step as input to the RNN at the current time step. This forces the RNN to learn to predict the next output in the sequence, even if it has made mistakes in the past. Teacher forcing significantly improves the efficiency and effectiveness of encoder-decoder RNN training. However, during testing, the model predictions are fed as the previous outputs to the decoder, since the actual output data for the future is unavailable. For this encoder-decoder RNN model, we considered 1 LSTM layer with 128 neurons for each encoder and decoder.

4.3.2 Transformers

Transformers also have an encoder-decoder structure. In the encoder, the input data is processed by the attention mechanism. The output of the encoder is then fed into the decoder, which also receives the previous outputs. In addition to the parameters in typical neural networks, transformers also have other parameters that must be determined, such as the number of heads and the model dimension. In this work, we used a transformer architecture with two layers in both the encoder and decoder. We also set the number of heads and the model dimension to four and 32, respectively. We also used dropout and batch normalization in the transformer. Dropout is used to prevent overfitting, while batch normalization helps to keep the weights of the neural network stable.

All of the proposed models were implemented in Python using the TensorFlow machine learning library. Due to the large number of hyperparameters in neural networks, we decided to use the same hyperparameters for all models whenever possible. The hyperparameters were chosen through trial and error, and the best model for each architecture was selected. We fixed the number of future steps to 30 in the

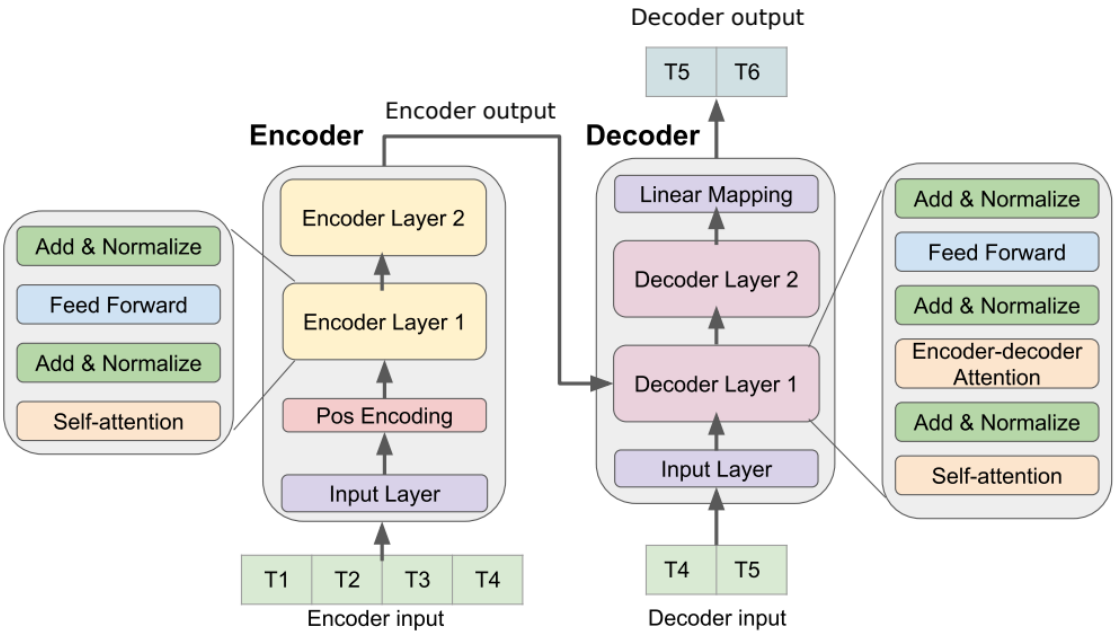
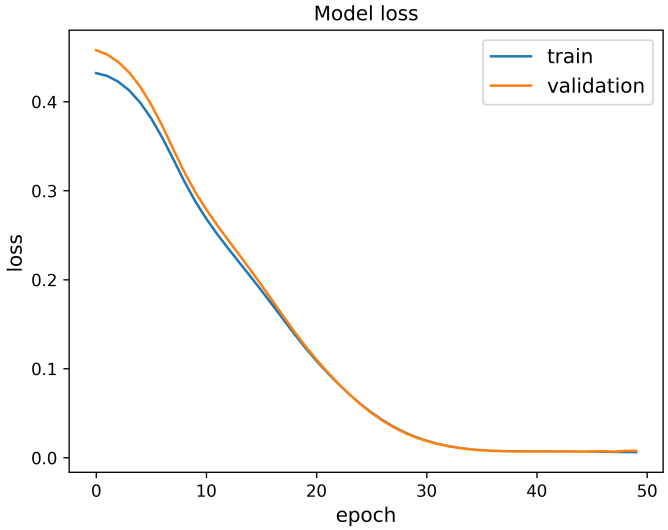
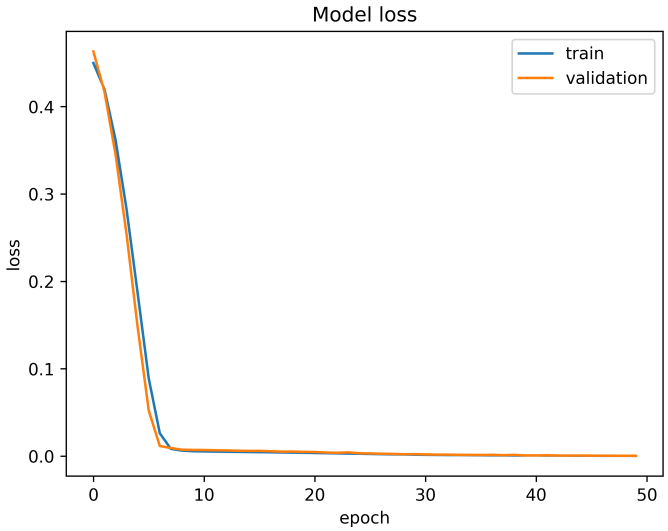


Figure 4.15 Transformers for time-series

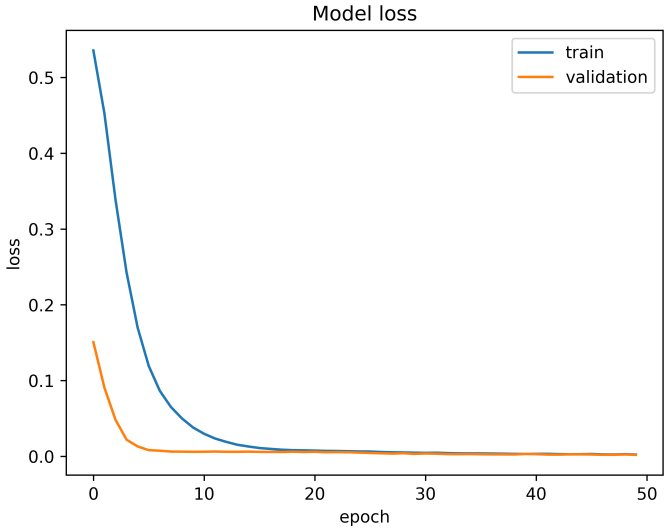
first stage, meaning that the models need to predict the system output from 1 to 30 minutes in the future. We used a dataset of 5,000 data points, splitting it into three sets: 3,500 samples for training, 1000 for validation, and the remaining 500 samples for testing. We trained the models for 50 epochs, and the learning curves (loss vs. epoch) for each model are provided in the figures 4.16. Finally, we evaluated the performance of all models on various steps ahead predictions for different significant states using the root mean squared error (RMSE) criterion.



(a) Single-shot learning curve



(b) Encoder-Decoder RNN learning curve



(c) Transformers learning curve

Figure 4.16 Simulation models learning curves

4.3.3 Prediction performance

After implementing and training the machine learning models, we used them to make various predictions on the test data. Starting with the short-term prediction performance, we present their performance on multi-step ahead predictions. Figures 4.17, 4.18, **??**, **??** show the time plots for the desired states, for 1-step ahead prediction for various models. In 1-step ahead prediction, the baseline model is performing better than ML models. This is mainly because the data is not dynamic and does not change significantly within 1 minute/sample. As a result, the baseline model, which predicts the next output to be the same as the previous output, does not have a large error. However, ML models demonstrate superiority over baseline models in long-term predictions, and as the prediction horizon increases, the baseline model's error also increases significantly, while ML models still maintain accuracy. In the prediction of kiln hood pressure, the baseline model error RMSE increases from 0.0168 in 1-step to 0.1243 in 30 steps (650%), while the encoder-decoder RNN error RMSE increases from 0.0211 in 1-step to 0.0738 in 30 steps (250%). It is also important to note that model performance can vary from state to state depending on the data and its characteristics. As an example, the models' errors in predicting the kiln hood temperature are much lower than in other states. This is primarily due to the fact that data on kiln hood temperature is the least noisy data of all the process states, while it is also changing at a much slower rate than other process states. It is therefore possible for even the simplest model (baseline model) to have small errors. To reveal the full potential of sophisticated models like ML models, suitable dynamic data is necessary [43].

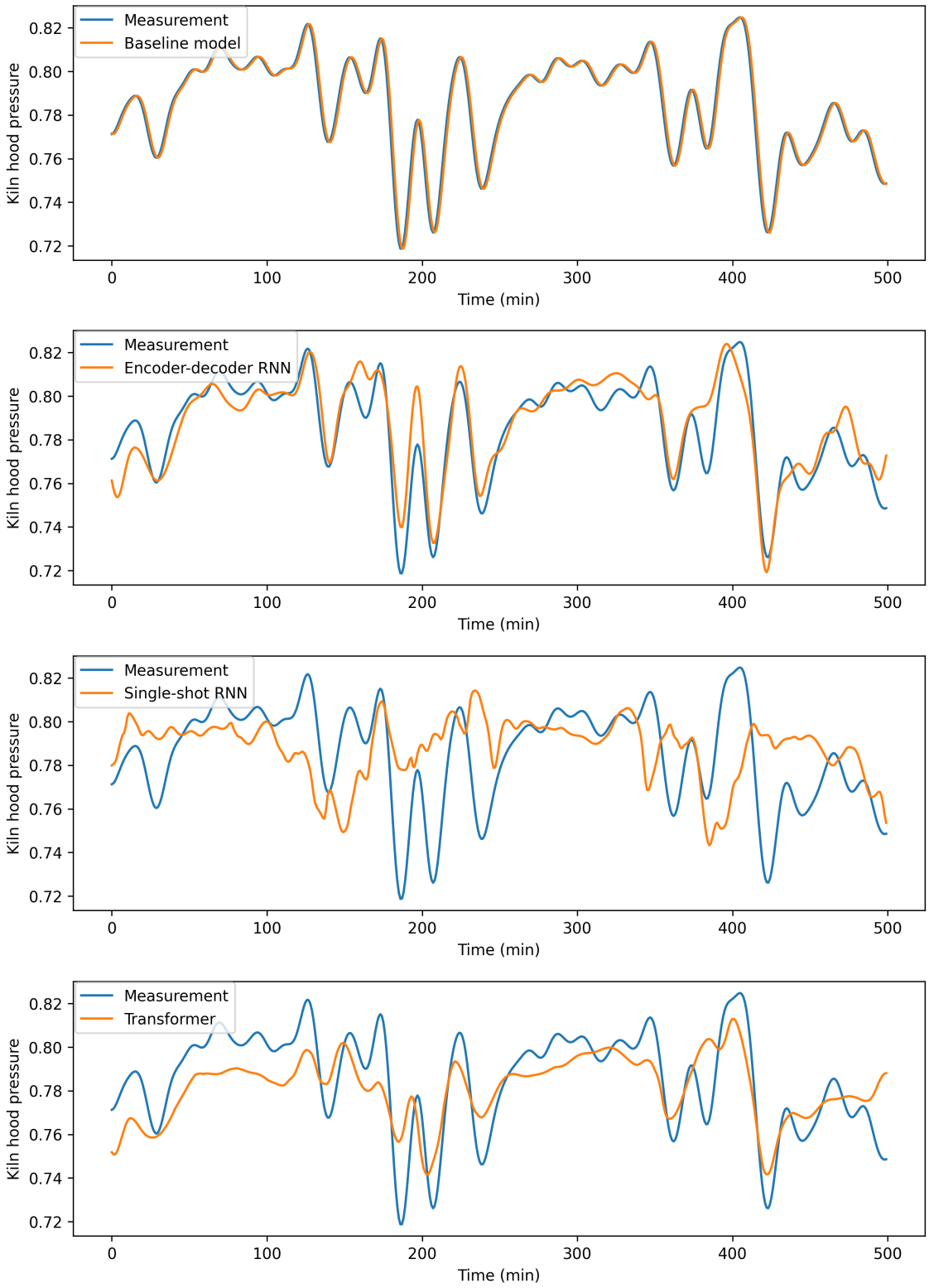


Figure 4.17 Performance of Machine Learning Models for 1-Step Ahead Predictions of Kiln Hood Temperature

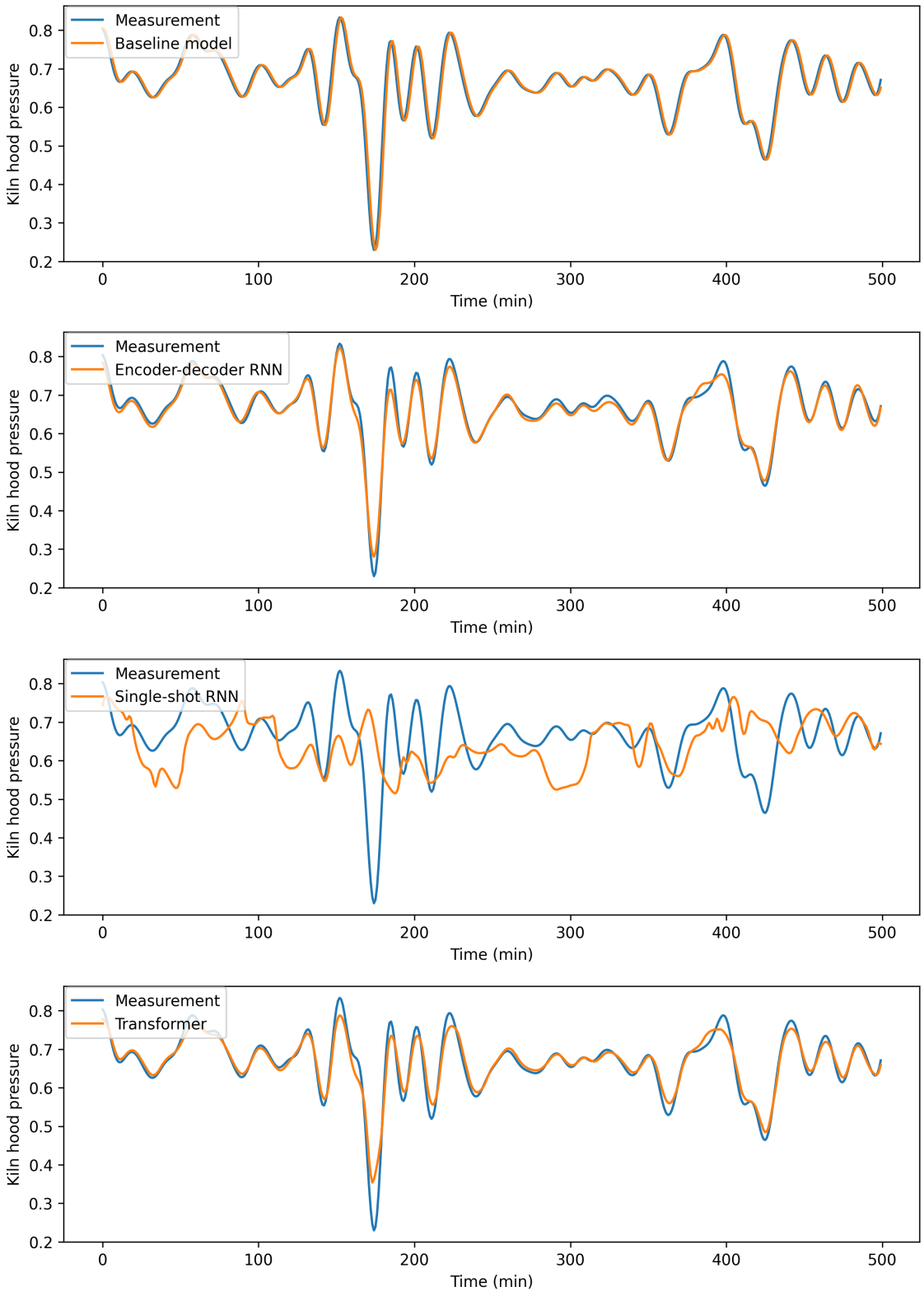


Figure 4.18 Performance of Machine Learning Models for 1-Step Ahead Predictions of Kiln Hood Pressure

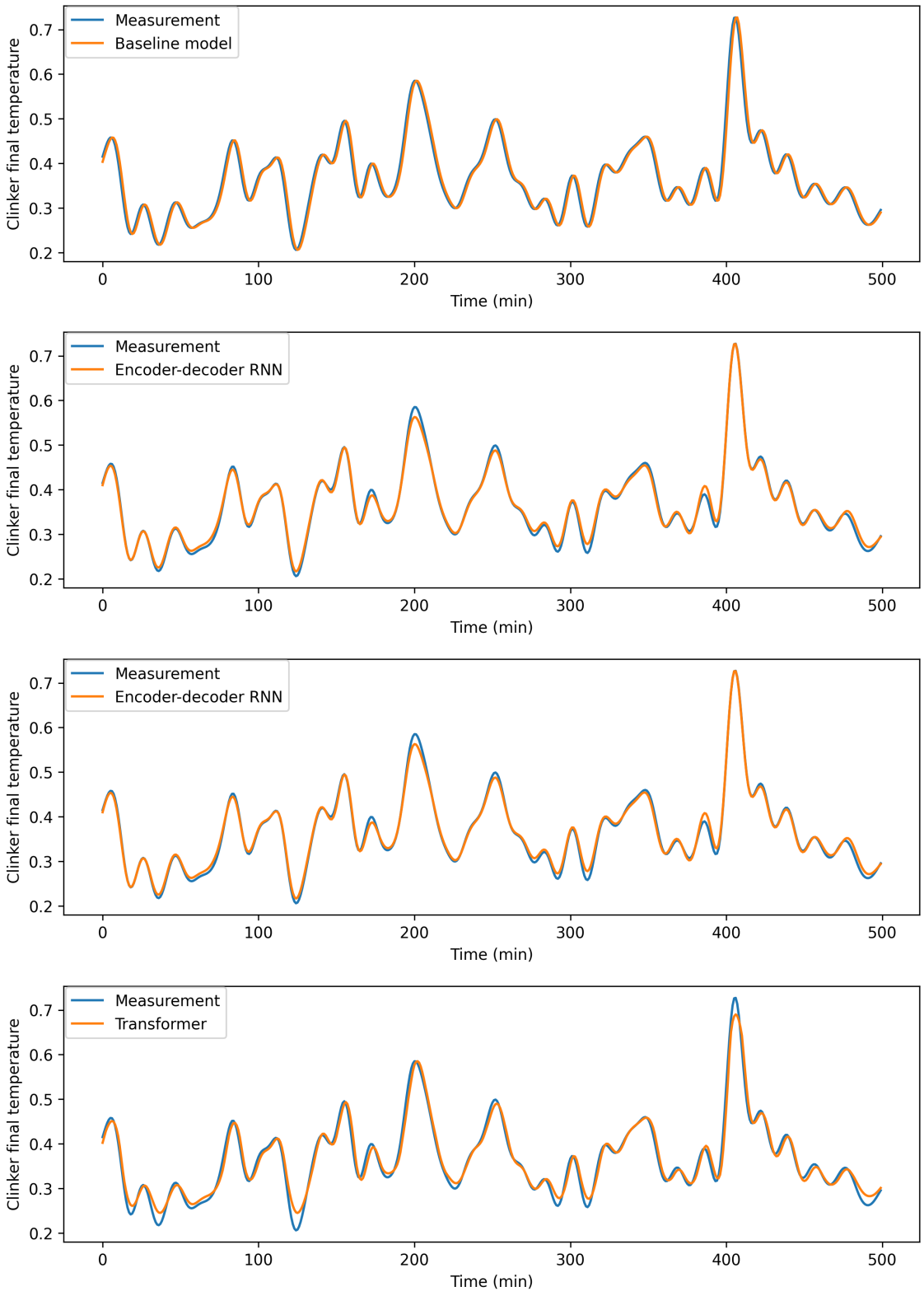


Figure 4.19 Performance of Machine Learning Models for 1-Step Ahead Predictions of Clinker Final Temperature

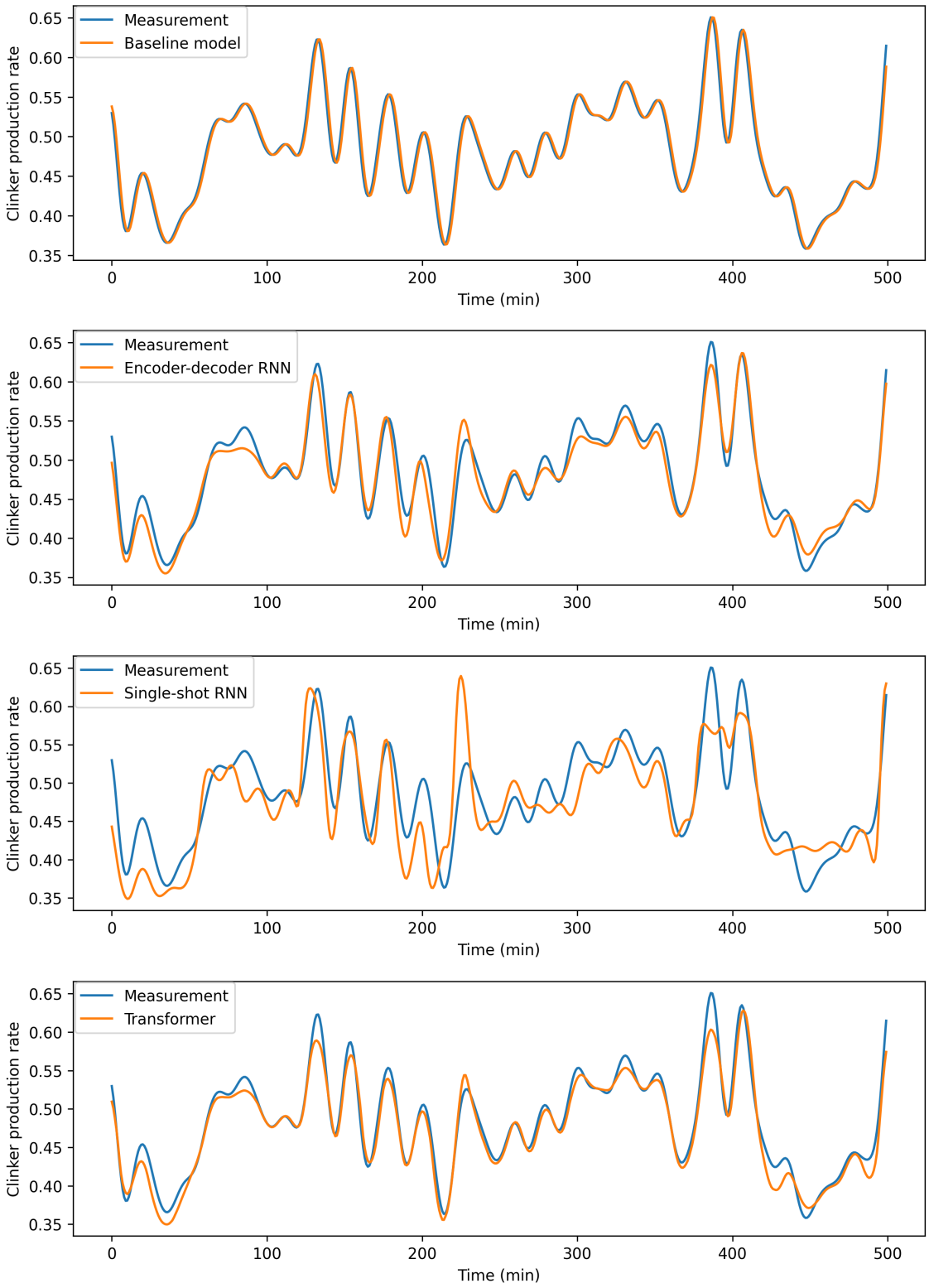
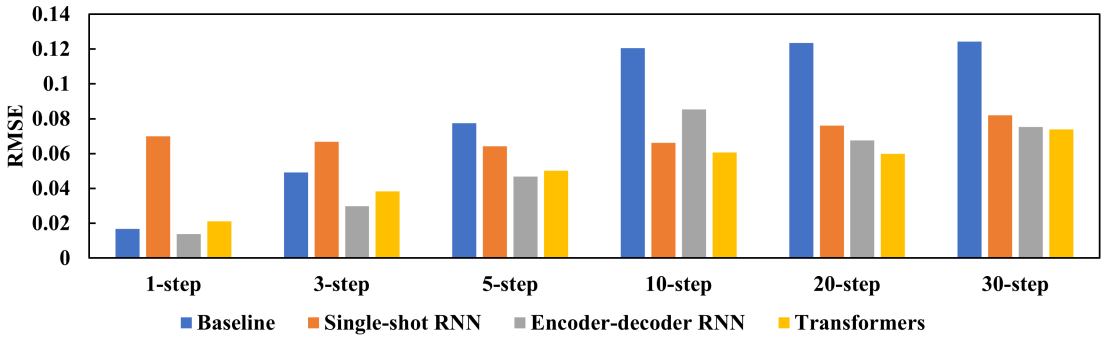


Figure 4.20 Performance of Machine Learning Models for 1-Step Ahead Predictions of Clinker Production Rate

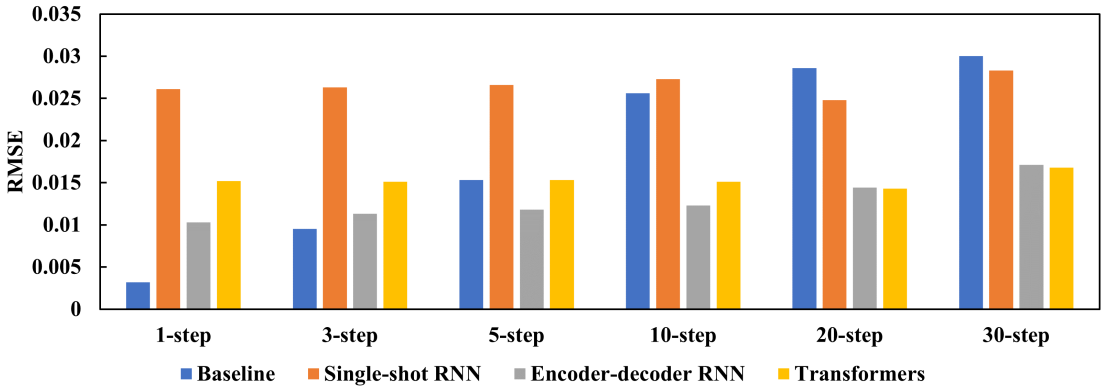
The long-term prediction performance of the ML models is analyzed with various multi-step ahead predictions. Table 4.2 depicts the models' performance on various long-term predictions for different process variables. The single-shot RNN model maintains a similar error level for all multi-step predictions. The performance of this model, however, is not satisfactory for short-term predictions. Encoder-decoder models make better predictions than single-shot models because they exploit the auto-regressive structure in which previous outputs are utilized along with previous inputs. Nonetheless, it makes more errors as the prediction horizon increases. Transformers are presenting the best performance among all ML models considering all process variables and prediction steps although, in some states and steps, it may fall behind other models. The attention mechanism enables the transformers to attend to different parts of the sequence. As a result, they can discover the most affecting features of the previous inputs and outputs for the prediction of future values. Moreover, the attention mechanism overcomes the vanishing gradient problem of recurrent neural networks since the attention mechanism does not process the inputs in sequence. This core element of transformers makes them a better choice for modeling complex and nonlinear systems.

Table 4.2 ML models performance on multi-steps ahead prediction

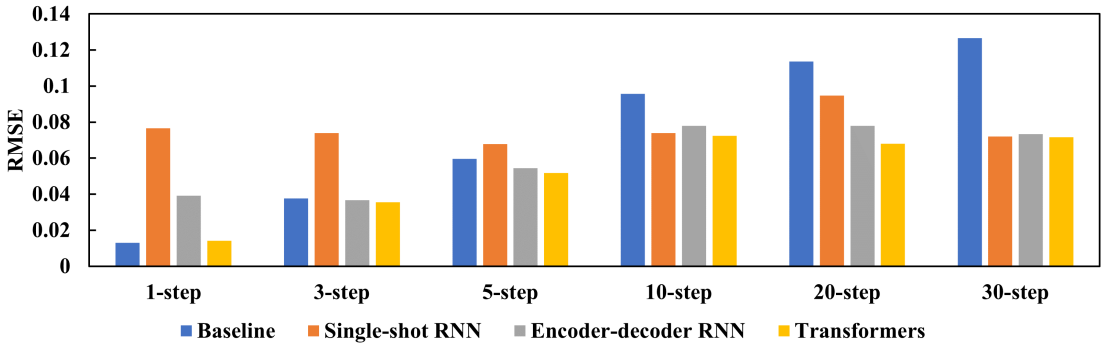
	Kiln hood pressure					
	1-step	3-steps	5-steps	10-steps	20-steps	30-steps
Baseline model	0.0168	0.0491	0.0775	0.1205	0.1234	0.1243
Single-shot RNN	0.0699	0.0667	0.0642	0.0662	0.0761	0.0819
Encoder-decoder RNN	0.0139	0.0299	0.0469	0.0853	0.0676	0.0753
Transformers	0.0211	0.0383	0.0501	0.0607	0.0599	0.0738
	Kiln hood temperature					
	1-step	3-steps	5-steps	10-steps	20-steps	30-steps
Baseline model	0.0032	0.0095	0.0153	0.0256	0.0286	0.03
Single-shot RNN	0.0261	0.0263	0.0266	0.0273	0.0248	0.0283
Encoder-decoder RNN	0.0103	0.0113	0.0118	0.0123	0.0144	0.0171
Transformers	0.0152	0.0151	0.0153	0.0151	0.0143	0.0168
	Clinker final temperature					
	1-step	3-steps	5-steps	10-steps	20-steps	30-steps
Baseline model	0.0129	0.0376	0.0596	0.0956	0.1136	0.1266
Single-shot RNN	0.0765	0.074	0.0677	0.0739	0.0948	0.072
Encoder-decoder RNN	0.0392	0.0366	0.0545	0.078	0.0779	0.0733
Transformers	0.0141	0.0355	0.0518	0.0723	0.0679	0.0717
	Clinker production rate					
	1-step	3-steps	5-steps	10-steps	20-steps	30-steps
Baseline model	0.0093	0.0275	0.0438	0.0731	0.0718	0.0861
Single-shot RNN	0.0382	0.0344	0.0311	0.0377	0.0369	0.0415
Encoder-decoder RNN	0.0155	0.0189	0.0307	0.0377	0.0486	0.0559
Transformers	0.0135	0.0258	0.0342	0.0387	0.0317	0.0325



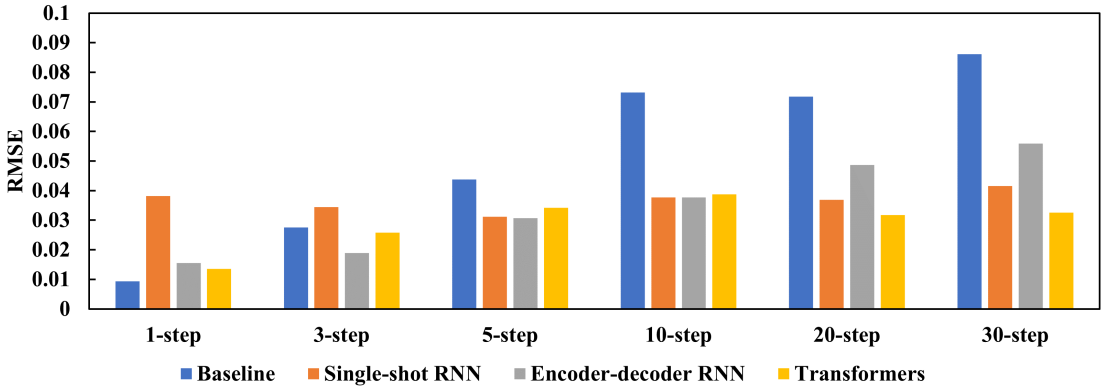
Kiln hood pressure



Kiln hood temperature



Clinker production temperature



Clinker production rate

Figure 4.21 ML Models Multi-steps Ahead Predictions RMSE

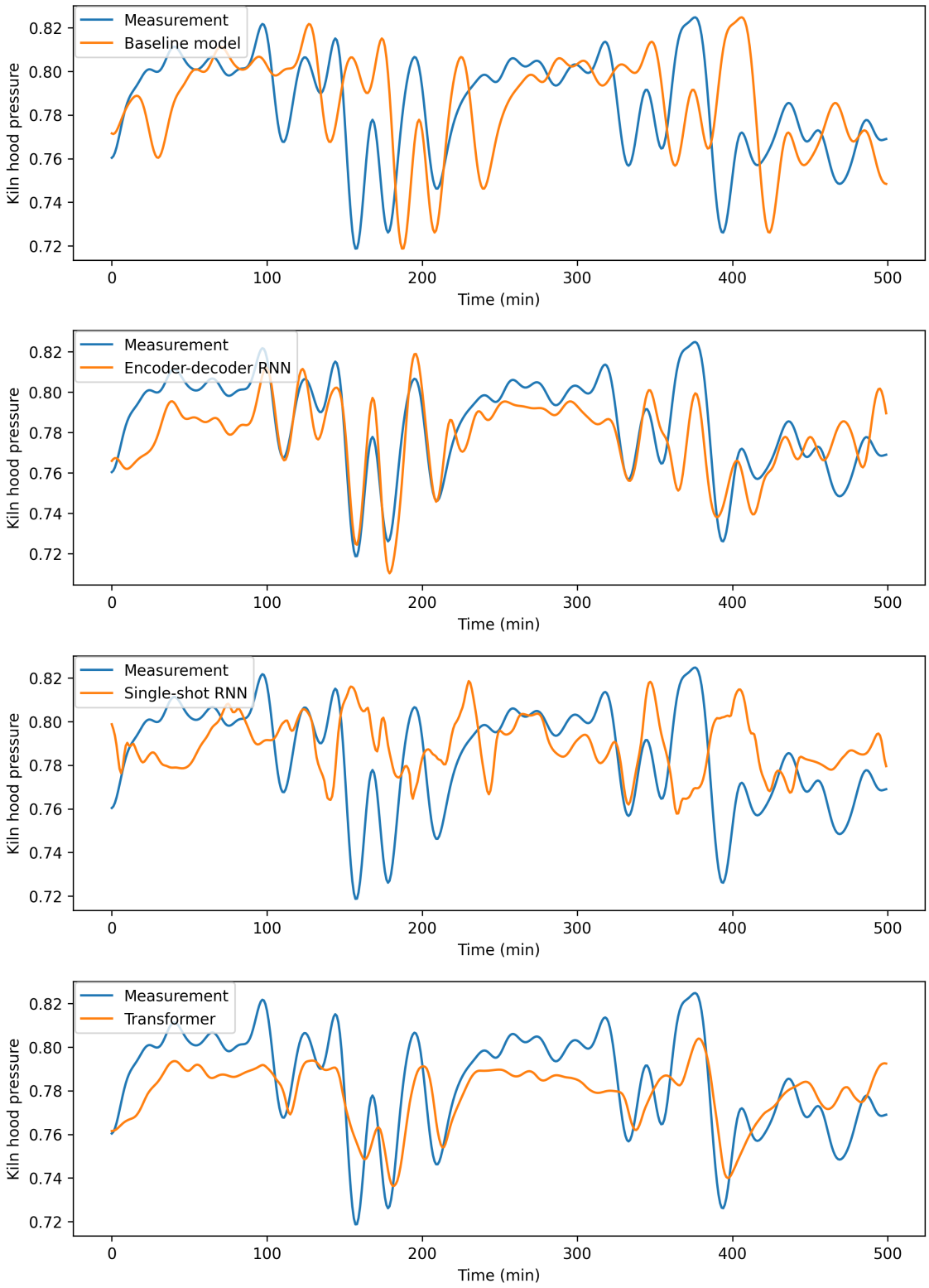


Figure 4.22 Performance of Machine Learning Models for 30-Step Ahead Predictions of Kiln Hood Temperature

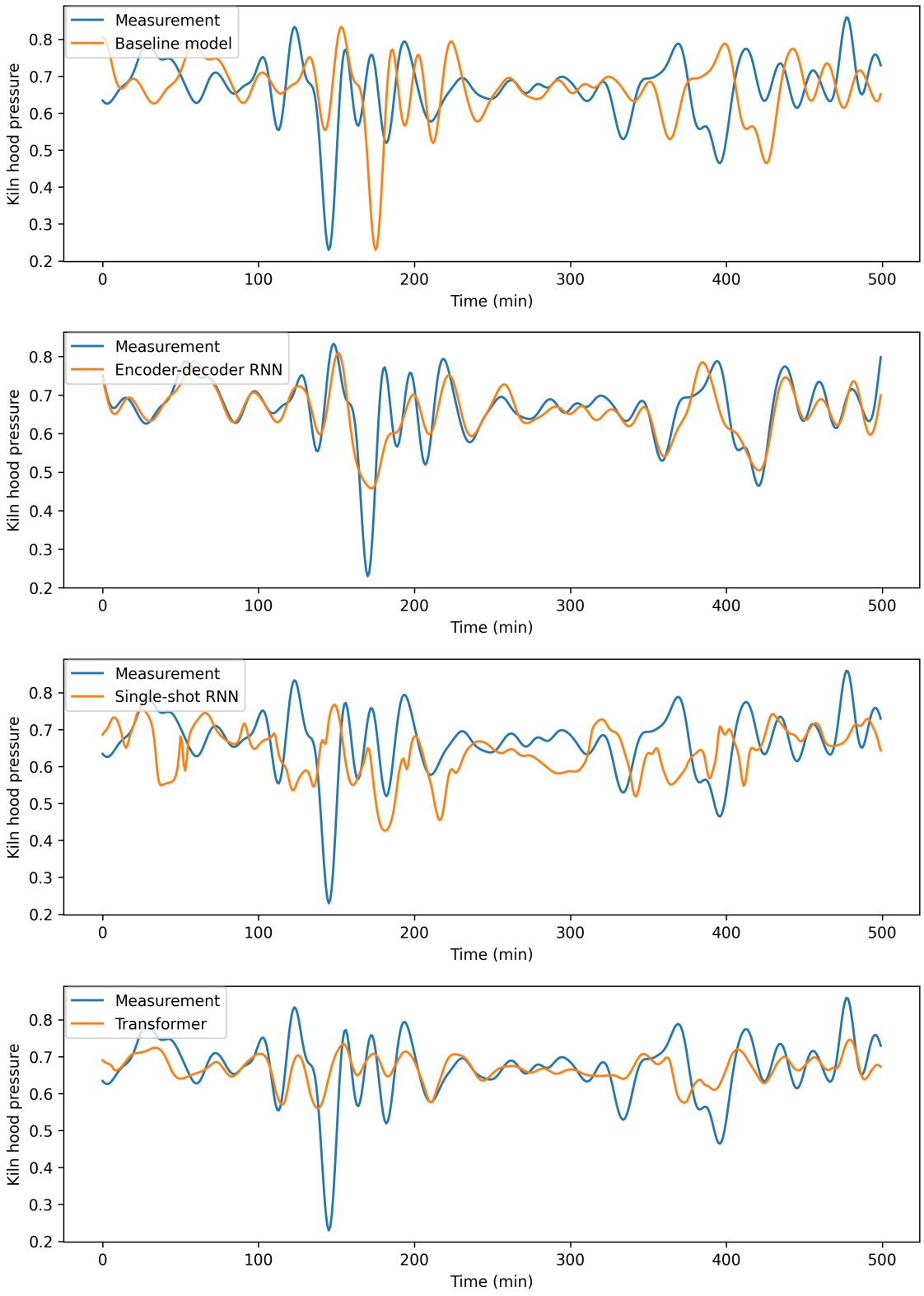


Figure 4.23 Performance of Machine Learning Models for 30-Step Ahead Predictions of Kiln Hood Pressure

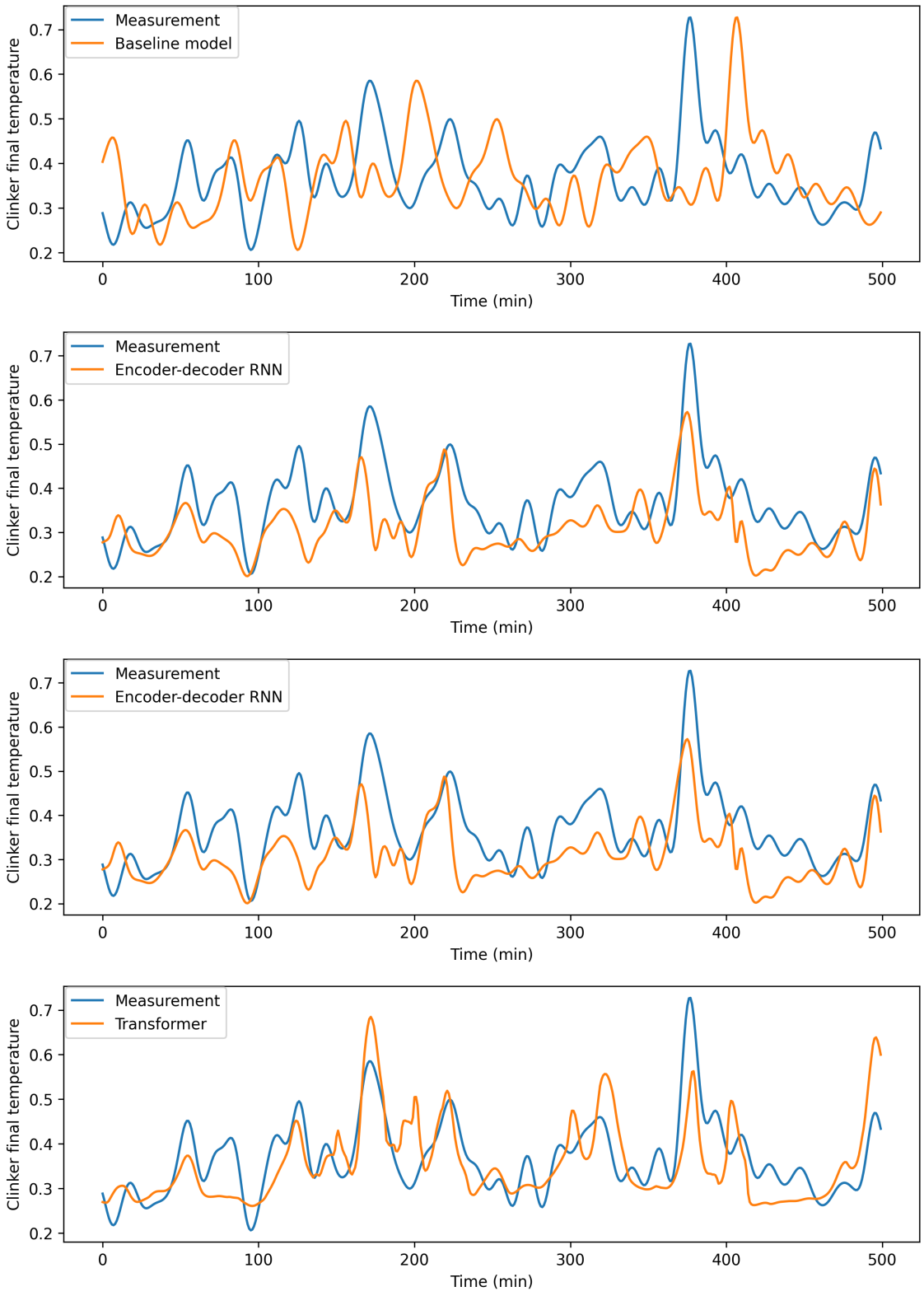


Figure 4.24 Performance of Machine Learning Models for 30-Step Ahead Predictions of Clinker Final Temperature

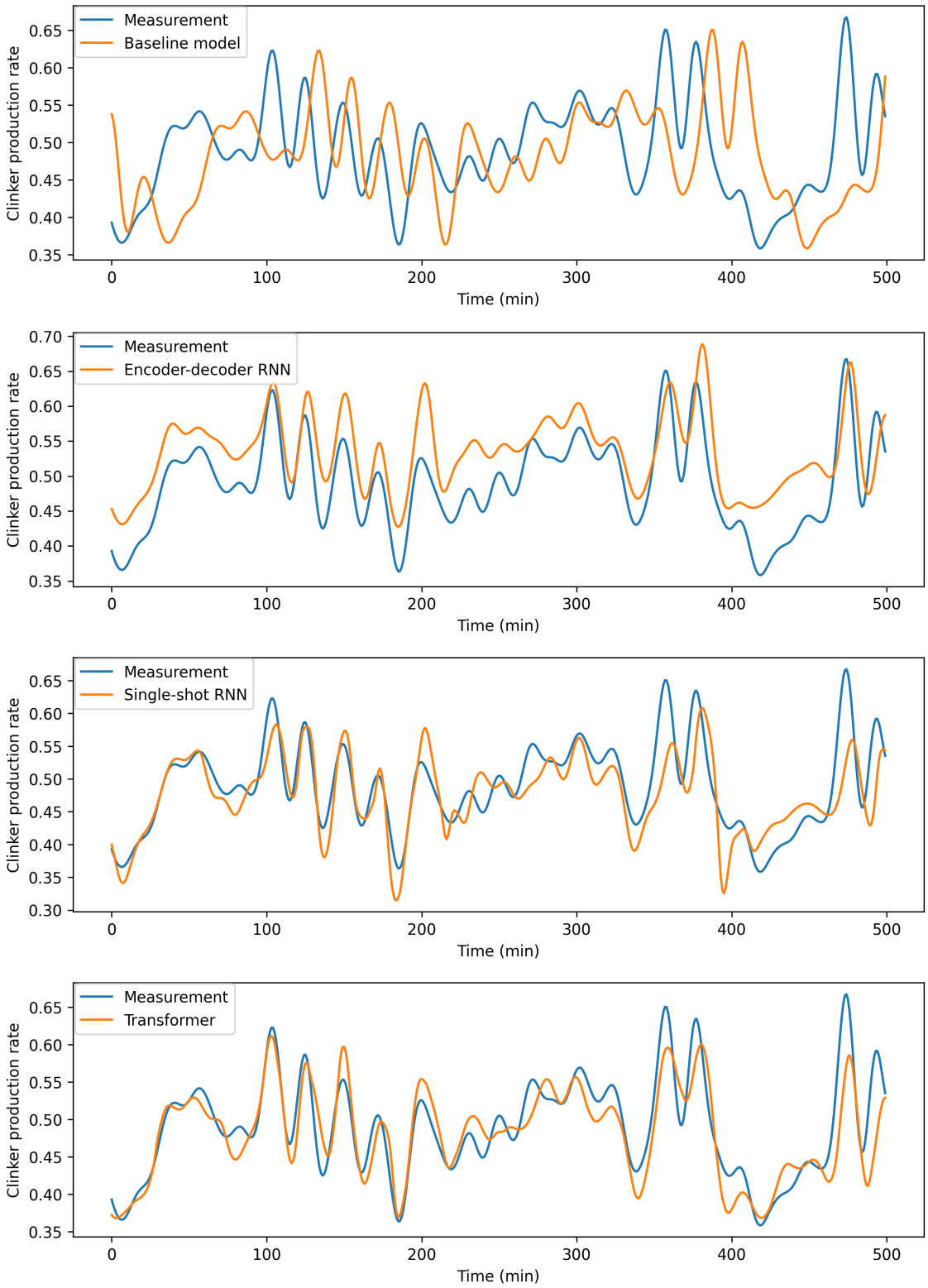
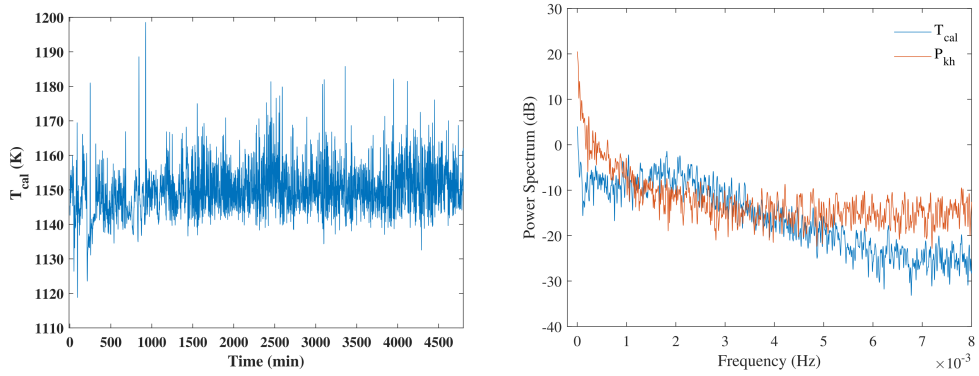


Figure 4.25 Performance of Machine Learning Models for 30-Step Ahead Predictions of Clinker Production Rate

Based on the above analysis and user needs, various machine learning models can be utilized for simulating the cement production process. If the user does not have computational and model complexity limitations, transformers are the most appropriate choice, particularly for long-term predictions. However, if the model needs to be simpler with less computational effort, recurrent neural networks both the simple RNN and encoder-decoder RNN can be utilized. Nonetheless, in every modeling and simulation task, various models should be implemented, and then based on their performance on the available data, the most suitable model can be selected for the intended tasks. All in all, selecting the right model for the intended purpose requires careful consideration of the task, data, and computing resources.

5. Discussion

Simulation and control-oriented models both demonstrate satisfactory performance in predicting the future state of the cement production process. Despite this, the control-oriented model had difficulty predicting the calciner temperature state. Although it displays satisfactory performance in the short term, it loses accuracy in the long-term prediction horizon for the calciner temperature state more than in other states. The reason for this may be related to the quality of the data collected by the calciner temperature sensor. As a result of the significant noise in its measurement signal, this state is the most challenging to model. Furthermore, the calciner temperature is strictly controlled and maintained within a set range during plant normal operation. Consequently, its measurement signals are usually constant and do not provide sufficient excitation to identify its dynamic behavior and response to inputs. A comparison of the calciner temperature signal with the kiln hood pressure signal reveals noise and an absence of information in the signal (Fig 5.1). Another crucial



(a) Calciner temperature time plot

(b) Calciner temperature power spectrum

Figure 5.1 Calciner temperature power spectrum and time plot

issue that should be discussed is the residuals of the control-oriented model. Figure 5.2 depicts the autocorrelation of two sample variables in the control-oriented model which shows that the model's residuals have significant autocorrelation. Although it may seem that the model failed to capture the full dynamic of the system, it should be noticed that the model residual autocorrelation does not imply the failure of the

model, it is more an indication of some missing in the modeling part. Secondly, we believe that this residual autocorrelation is mainly due to the filtering effects. In the case that unfiltered data is used for modeling, the model residual becomes different with almost no autocorrelation between them.

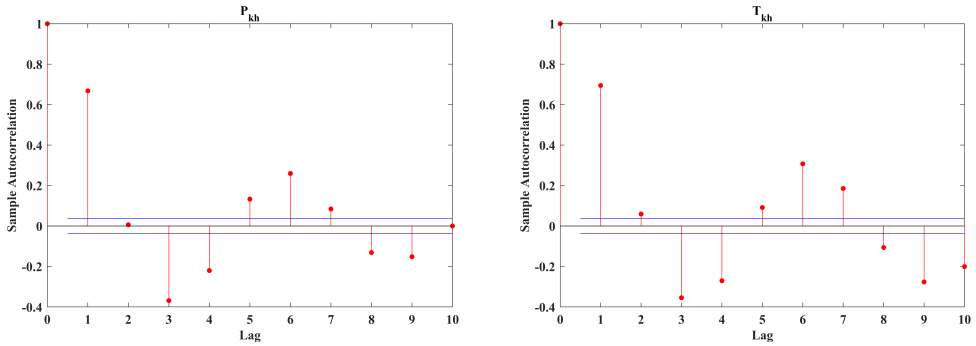


Figure 5.2 Control-oriented model residuals autocorrelation for 2 sample variables

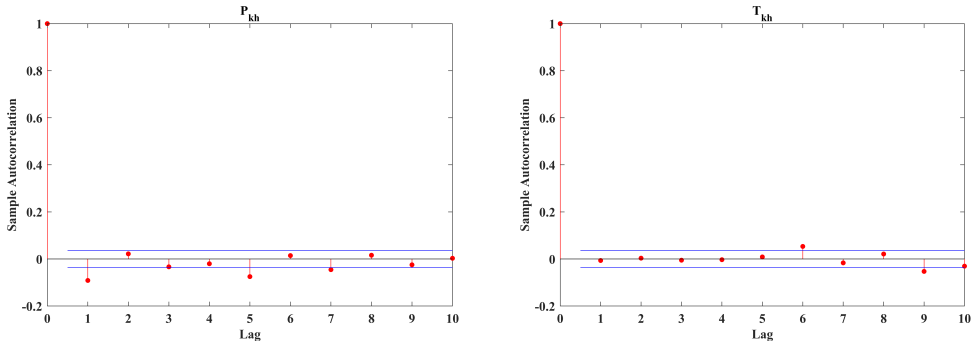


Figure 5.3 Control-oriented model residuals autocorrelation for 2 sample variables when the model estimated with unfiltered data

Among ML models in the simulation part, the encoder-decoder RNN and transformer outperform the common RNN (single-shot) model. This is mainly due to the fact that the encoder-decoder RNN and transformer get not only the previous input but also the previous output. This is particularly beneficial in modeling dynamic physical systems as it considers the current or initial state of the system. Nonetheless, a possible more complex and accurate model could be the encoder-decoder and transformer models which are fed with multiple previous outputs of the system. Transformers demonstrate better performance than the encoder-decoder RNN model in most states in the longer-term predictions. The main reason is the difference in the structure of these two models. Although encoder-decoder RNN considers previous output unlike single-shot RNN, it still exploits the main sequence processing structure in the RNNs. This structure in RNNs is prone to vanishing or exploding gradients, making the encoder-decoder RNN susceptible to this problem. On the other hand, the transformers exploit the attention mechanism which enables them to process a sequence of data simultaneously which increases their ability to

capture long-term dependencies. In this work, the number of prediction steps is limited to 30 due to computational resource limitations, while a longer prediction horizon can reveal complex models' capabilities more clearly. Therefore, future work should focus on exploring longer prediction horizons, with the hope of uncovering more complex models. It's important to emphasize the significance of data quality in the context of data-driven methods. However, in industrial processes, such as cement production, obtaining a sufficient amount of high-quality dynamic data is challenging due to commercial and security constraints. Lastly, it should be noted that model structure selection and development are inevitably problem-dependent, so all these ML models can be suitable depending on the data and problem.

6. Conclusion and Future Work

The cement industry is one of the largest and most polluting industries in the modern era. Therefore, it is imperative to investigate methods to increase their efficiency and reduce their environmental impact. This challenge can be addressed through the use of novel controllers, such as model predictive controllers (MPCs). Nevertheless, model predictive controllers require predictive dynamic models of the plant. In this study, we developed two types of models for the cement production process: Control-oriented models which should be suitable for control design tasks, and simulation models that can replace the real plant in the controller implementation. In the first type of model, the emphasis was on keeping the model simple, and appropriate to the context of the model predictive controller, such as keeping it linear for convex optimization criteria in MPC. For the simulation models, more sophisticated and nonlinear models have been considered since they do not have the limitations of control-oriented models.

In the control-oriented models, in contrast to most of the existing literature, we considered the internal dynamics of the process by using first principles such as mass and energy conservation laws. These analyses enabled us to identify the inter-component relationships in the process. By incorporating these physical insights into the black-box system identification model, we are able to develop a gray-box model of the process. The results indicate that demonstrates that integrating physical understanding by considering internal dynamics and inter-component relations in the process can increase system models' prediction performance and generalizability. The more generalizability capability of the gray-box model makes it suitable for problems in which it is hard to get enough quality data such as industrial plants.

For the simulation models, machine learning techniques including recurrent neural networks (RNNs), encoder-decoder RNNs, and transformers were considered. The models are modified and implemented in a manner that is specific to the intended problem (the cement production process). ML models are trained on actual data from the Akcansa cement plant, and then their performance on both short-term and long-term predictions is evaluated. The results show that all models have accept-

able accuracy for both short-term and long-term predictions. Among these models, transformers are more successful than other models since their ability to capture long-term dependencies and contexts for learning makes them more effective.

This study focused on developing essential predictive models for model predictive controllers for cement production. The next step can be to utilize these models to develop and implement model predictive controllers for the cement production process. In this regard, control-oriented models can be used to design the MPC, whereas simulation models can be utilized to represent the real plant in the controller implementation phase. After the model predictive controllers have been implemented, it is possible to examine their effects on the process in terms of decreasing energy and material consumption as well as environmental pollution.

Due to limitations in computational resources, the ML model dimensions were selected to be small in this study. However, larger ML models, especially in the case of transformers can be more accurate and reliable. In this study, only one previous output value was fed into the autoregressive models, which limits their ability to capture previous states of the system. The performance of these models can be improved when more past output values and inputs are included. Last but not least, a major research line in the modeling part could be the integration of physical concepts into machine learning models to develop physics-informed machine learning models. Current ML models are solely based on data which limits their transparency, reliability, and generalizability. Physics-informed ML models have the potential to address these limitations associated with purely data-driven models.

Bibliography

[1] R. Fatahi, H. Nasiri, A. Homafar, R. Khosravi, H. Siavoshi, S. Chehreh Chelgani, Modeling operational cement rotary kiln variables with explainable artificial intelligence methods–a "conscious lab" development, Particulate Science and Technology (2022). doi:10.1080/02726351.2022.2135470.

[2] A. Cantini, L. Leoni, F. De Carlo, M. Salvio, C. Martini, F. Martini, Technological energy efficiency improvements in cement industries, Sustainability (Switzerland) 13 (7) (4 2021). doi:10.3390/su13073810.

[3] A. Atmaca, R. Yumrutas, R. Y. Yumrutas¸, Analysis of the parameters affecting energy consumption of a rotary kiln in cement industry Production of a thermal insulation material View project Analysis of the parameters affecting energy consumption of a rotary kiln in cement industry (2014). doi:10.1016/j.applthermaleng.2014.02.038.
URL http://dx.doi.org/10.1016/j.applthermaleng.2014.02.038

[4] K. S. Mujumdar, K. V. Ganesh, S. B. Kulkarni, V. V. Ranade, Rotary Cement Kiln Simulator (RoCKS): Integrated modeling of pre-heater, calciner, kiln and clinker cooler, Chemical Engineering Science 62 (9) (2007) 2590–2607. doi:10.1016/J.CES.2007.01.063.

[5] V. Ramasamy, R. Kannan, G. Muralidharan, R. K. Sidharthan, G. Veerasamy, S. Venkatesh, R. Amirtharajan, A comprehensive review on Advanced Process Control of cement kiln process with the focus on MPC tuning strategies, Journal of Process Control 121 (2023) 85–102. doi:10.1016/J.JPROCONT.2022.12.002.
URL https://linkinghub.elsevier.com/retrieve/pii/S0959152422002232

[6] K. S. Stadler, J. Poland, E. Gallestey, Model predictive control of a rotary cement kiln, Control Engineering Practice 19 (1) (2011) 1–9. doi:10.1016/J.CONENGPRAC.2010.08.004.

[7] S. J. Qin, T. A. Badgwell, A survey of industrial model predictive control technology, Control Engineering Practice 11 (7) (2003) 733–764. doi:10.1016/S0967-0661(02)00186-7.

[8] S. M. Zanoli, C. Pepe, M. Rocchi, Improving Performances of a Cement Rotary Kiln: A Model Predictive Control Solution (9). doi:10.18178/joace.4.4.262-267.

[9] Z. Li, X. W. Gao, Survey on rotary kiln process control, in: 2009 Chinese Control and Decision Conference, CCDC 2009, 2009, pp. 4151–4156. doi:10.1109/CCDC.2009.5191870.

[10] B. J. Rawlings, Q. D. Mayne, M. M. Diehl, Model predictive control: Theory, Computation, and Design, Vol. 1, 2018.

[11] Z. Su, Z. Yuan, Q. Zhang, S. Wu, Control-oriented modeling and simulation for calciner outlet temperature based on regression analysis, Proceedings of the World Congress on Intelligent Control and Automation (WCICA) 2015-March (March) (2015) 2636–2641. doi:10.1109/WCICA.2014.7053141.

[12] N. A. Madlool, R. Saidur, M. S. Hossain, N. A. Rahim, A critical review on energy use and savings in the cement industries, Renewable and Sustainable Energy Reviews 15 (4) (2011) 2042–2060. doi:10.1016/J.RSER.2011.01.005.

[13] R. Saidur, M. S. Hossain, M. R. Islam, H. Fayaz, H. A. Mohammed, A review on kiln system modeling, Renewable and Sustainable Energy Reviews 15 (5) (2011) 2487–2500. doi:10.1016/J.RSER.2011.01.020.

[14] Z. Ma, P. Jiang, S. Li, X. Wang, A Review on Multi-objective Optimization of Coordinated Control in Cement Clinker Calcination Process, 2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, CYBER 2021 (2021) 765–770doi:10.1109/CYBER53097.2021.9588328.

[15] H. R. Goshayeshi, F. K. Poor, Modeling of Rotary Kiln in Cement Industry, Energy and Power Engineering 08 (01) (2016) 23–33. doi:10.4236/epe.2016.81003.

[16] Institute of Electrical and Electronics Engineers, Institute of Electrical and Electronics Engineers. Peru Section, IEEE Power & Energy Society, Model Reference Adaptive Temperature Control of a Rotary Cement Kiln.

[17] Y. Zhongqiang, W. Xiaohong, Y. Hongliang, Study on generalized predictive control of cement rotary kiln calcining zone temperature, in: Proceedings of 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference, IMCEC 2016, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1653–1658. doi:10.1109/IMCEC.2016.7867498.

[18] A. V. S. S. K. S. Gupta, P. K. Nag, Prediction of heat transfer coe$cient in the cyclone separator of a CFB, INTERNATIONAL JOURNAL OF ENERGY RESEARCH Int. J. Energy Res 24 (2000) 1065–1079. doi:10.1002/1099-114X. URL https://onlinelibrary.wiley.com/doi/10.1002/1099-114X

[19] C. Csernyei, A. G. Straatman, Numerical modeling of a rotary cement kiln with improvements to shell cooling, International Journal of Heat and Mass Transfer 102 (2016) 610–621. doi:10.1016/J.IJHEATMASSTRANSFER.2016.06.058.

[20] C. Cortés, A. Gil, Modeling the gas and particle flow inside cyclone separators (10 2007). doi:10.1016/j.pecs.2007.02.001.

[21] K. S. Mujumdar, A. Arora, V. V. Ranade, Modeling of rotary cement kilns: Applications to reduction in energy consumption, Industrial and Engineering Chemistry Research 45 (7) (2006) 2315–2330. doi:10.1021/IE050617V/ASSET/IMAGES/LARGE/IE050617VF00009.JPEG.
URL https://pubs.acs.org/doi/full/10.1021/ie050617v

[22] A. Rahman, M. G. Rasul, M. M. Khan, S. Sharma, Cement Kiln Process Modeling to Achieve Energy Efficiency by Utilizing Agricultural Biomass as Alternative Fuels, in: Thermofluid Modeling for Energy Efficiency Applications, Elsevier Inc., 2016, pp. 197–225. doi:10.1016/B978-0-12-802397-6.00009-9.

[23] D. Touil, H. F. Belabed, C. Frances, S. Belaadi, Heat exchange modeling of a grate clinker cooler and entropy production analysis, International Journal of Heat and Technology 23 (1) (2005) 61–68.

[24] W. Shao, Z. Cui, N. H. Wang, L. Cheng, Numerical simulation of heat transfer process in cement grate cooler based on dynamic mesh technique, Science China Technological Sciences 59 (7) (2016) 1065–1070. doi:10.1007/S11431-016-6074-6.

[25] Z. Cui, W. Shao, Z. Chen, L. Cheng, Mathematical model and numerical solutions for the coupled gas–solid heat transfer process in moving packed beds, Applied Energy 206 (2017) 1297–1308. doi:10.1016/j.apenergy.2017.10.011.

[26] W. Shao, Z. Cui, X. t. Ma, Experimental research on cement grate cooler system and numerical simulation of its clinker cooling process, Applied Thermal Engineering 181 (11 2020). doi:10.1016/j.applthermaleng.2020.115904.

[27] M. Negash, G. Alemu, Neural network based data-driven predictor: Case study on clinker quality prediction, Zede Journal 29 (1) (2013) 27–36.
URL https://www.ajol.info/index.php/zj/article/view/92044

[28] P. Kadlec, B. Gabrys, S. Strandt, Data-driven Soft Sensors in the process industry, Computers & Chemical Engineering 33 (4) (2009) 795–814. doi:10.1016/J.COMPCHEMENG.2008.12.012.

[29] A. Wurzinger, H. Leibinger, S. Jakubek, M. Kozek, Data driven modeling and nonlinear model predictive control design for a rotary cement kiln, in: IFAC-PapersOnLine, Vol. 52, Elsevier B.V., 2019, pp. 759–764. doi:10.1016/j.ifacol.2019.12.054.

[30] Z. Qiang, Y. Hongliang, X. Dezhi, Applying Data-driven techniques to online updated PID controller for calciner outlet temperature, in: 2015 34th Chinese Control Conference (CCC), Vol. 2015-Septe, IEEE, 2015, pp. 94–97. doi:10.1109/ChiCC.2015.7259620.
URL http://ieeexplore.ieee.org/document/7259620/

[31] W. Chen, G. Han, W. Qiu, D. Zheng, Modeling of outlet temperature of the first-stage cyclone preheater in cement firing system using data-driven ARMAX models, Proceedings of 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference, IMCEC 2019 (2019) 472–477doi:10.1109/IMCEC46724.2019.8983997.

[32] I. Makaremi, A. Fatehi, B. N. Araabi, M. Azizi, A. Cheloian, Identification and Abnormal Condition Detection of a Cement Rotary Kiln, IFAC Proceedings Volumes 41 (2) (2008) 7233–7238. doi:10.3182/20080706-5-KR-1001.01224.

[33] S. P. Das, Institute of Electrical and Electronics Engineers. India Council., Indian Institute of Technology Kanpur., IEEE Uttar Pradesh Section., Cement Rotary Kiln Control: A Supervised Adaptive Model Predictive Approach (2008).

[34] B. Yang, X. Ma, Temperature prediction and analysis based on BP and Elman neural network for cement rotary kiln, https://doi.org/10.1117/12.888304 7997 (2011) 564–569. doi:10.1117/12.888304.
URL https://www.spiedigitallibrary.org/conference-proceedings-of-spie/7997/79972I/Temperature-prediction-and-analysis-based-on-BP-and-Elman-neural/10.1117/12.888304.fullhttps://www.spiedigitallibrary.org/conference-proceedings-of-spie/7997/79972I/Temperature-prediction-and-analysis-based-on-BP-and-Elman-neural/10.1117/12.888304.short

[35] J. Zheng, L. Zhao, W. Du, Hybrid model of a cement rotary kiln using an improved attention-based recurrent neural network, ISA Transactions 129 (2022) 631–643. doi:10.1016/J.ISATRA.2022.02.018.

[36] W. Chen, Y. Li, B. Chu, H. Zhang, ARMAX Modeling and Control of Cement Rotary Kiln Temperature Based on Improved AIC, Chinese Control Conference, CCC 2022-July (2022) 1473–1479. doi:10.23919/CCC55666.2022.9902608.

[37] E. B. Karikov, V. G. Rubanov, K. V. Korneevich, Construction of a dynamic neural network model as a stage of grate cooler automation, World Applied Sciences Journal 25 (2) (2013) 227–232. doi:10.5829/idosi.wasj.2013.25.02.7056.

[38] M. Seraj, M. A. Shooredeli, Data-driven predictor and soft-sensor models of a cement grate cooler based on neural network and effective dynamics, in: 2017 25th Iranian Conference on Electrical Engineering, ICEE 2017, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 726–731. doi:10.1109/IranianCEE.2017.7985134.

[39] Y. Geng, X. Wang, P. Jiang, Prediction of the Cement Grate Cooler Pressure in the Cooling Process Based on a Multi-Model Fusion Neural Network, IEEE Access 8 (2020) 115028–115040. doi:10.1109/ACCESS.2020.3002768.

[40] Z. S. Hou, Z. Wang, From model-based control to data-driven control: Survey, classification and perspective, Information Sciences 235 (2013) 3–35. doi:10.1016/j.ins.2012.07.014.
URL http://dx.doi.org/10.1016/j.ins.2012.07.014

[41] Z. Ge, Review on data-driven modeling and monitoring for plant-wide industrial proceGe, Z. (2017). Review on data-driven modeling and monitoring for plant-wide industrial processes. Chemometrics and Intelligent Laboratory Systems, Chemometrics and Intelligent Laboratory Systems 171 (October 2016) (2017)

16–25. doi:10.1016/j.chemolab.2017.09.021.
URL https://doi.org/10.1016/j.chemolab.2017.09.021

[42] M. Wang, E. Chen, P. Liu, W. Guo, Multivariable nonlinear predictive control of a clinker sintering system at different working states by combining artificial neural network and autoregressive exogenous, Advances in Mechanical Engineering 12 (1) (1 2020). doi:10.1177/1687814019896509.

[43] L. Ljung, System Identification Theory for the User. (1998).
URL https://books.google.com/books/about/System_Identification.html?id=fYSrk4wDKPsC

[44] C. Maarup, Gas-Solid Heat Exchanger for Cement Production, 2013.
URL https://api.semanticscholar.org/CorpusID:136363394

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[46] N. Geneva, N. Zabaras, Transformers for modeling physical systems, Neural Networks 146 (2022) 272–289. doi:https://doi.org/10.1016/j.neunet.2021.11.022.
URL https://www.sciencedirect.com/science/article/pii/S0893608021004500

[47] J. Park, M. R. Babaei, S. A. Munoz, A. N. Venkat, J. D. Hedengren, Simultaneous multistep transformer architecture for model predictive control, Computers & Chemical Engineering 178 (2023) 108396. doi:10.1016/J.COMPCHEMENG.2023.108396.

[48] G. Bye, P. Livesey, L. Struble, Portland Cement, ICE Publishing, 2011. doi:doi:10.1680/pc.36116.
URL https://doi.org/10.1680/pc.36116