**AI/ ML Modelling using Metals industry Domaine Knowledge**

Dr. Franck K. Aadjogble1,

1SMS Digital

100 Sandusky Street, PA, USA, 15044

Phone: +14126087120

Email: franck.adjogble@sms-group.com

Hardik Chinoy

2SMS Digital

100 Sandusky Street, PA, USA, 15044

Phone: +14126087120

Email: Hardik.Chinoy@sms-group.com

Keywords: Artificial Intelligence, Machine Learning, Domain Knowledge, Solution design, Domain Knowledge incorporated neural network

**Abstract**

Domain knowledge and expertise are the main asset of high-level performance in any industry. Leveraging this asset incorporated in the design of solution using AI and ML methodologies increases the accuracy of proposed solutions. The paper will demonstrate the approach on how and the steps to be considered while designing the AI/ML Models integrating quantitatively the domain knowledge in the Metals industry. The solutions and use cases implementations represent the goals of solving the real life issues facing the steel industry. The Solution implements a set of AI models and applications that provide functionalities to improve the product quality, monitoring the production process and support the maintenance activity by providing plant condition outputs. The results are feedback to the plant technological models and to the Production scheduling system to dynamically reschedule the production plan to increase the high quality production. The results are fed to the production plant through a layer of expert and Machine Interaction (EMI) that iteratively vet the results, translate them into quantifiable information for the AI/ML models using domain knowledge. Furthermore, Domain knowledge based-decisions are also made to which system the results should be placed. Re-planning information is sent to the scheduling system and important parameters for optimizing the process models are sent to the process application systems. The overall concept is a continuous improvement and cumulative learning system that considers system uncertainty using AI/ML Methods und Experts domain knowledge.

**INTRODUCTION**

This paper presents a new methodology for incorporating domain knowledge into the design of AI and ML models in the metals industry. The proposed approach highlights the significance of considering domain knowledge during the design of AI and ML solutions, which can help address real-life challenges faced by the steel industry. To showcase the effectiveness of our approach, we present practical case studies that demonstrate the use of a set of AI models and applications to enhance product quality, monitor the production process, and support maintenance activities. The results of these use cases are integrated back into the production scheduling system, allowing for dynamic rescheduling of the production plan to increase the production of high-quality products. However, to ensure the accuracy of the results, a layer of expert and machine interaction (EMI) is introduced, which iteratively validates the results and translates them into quantifiable information for the AI/ML models using domain knowledge. Furthermore, domain knowledge-based decisions are made regarding which system the results should be sent to, with re-planning information being sent to the scheduling system and important parameters for optimizing the process models being sent to the process application systems. Our approach emphasizes continuous improvement and cumulative learning from system changes under uncertainty using AI/ML methods and expert domain knowledge. This paper aims to demonstrate the significance of considering domain knowledge in the design of AI and ML solutions and how they can be used to effectively address real-life challenges in the metals industry.

**Literature review**

There has been a significant amount of research on the use of AI and ML in the steel industry. Many of these studies have focused on the use of these technologies to improve various aspects of the steel production process, such as quality control, maintenance, and energy efficiency. One study by Li et al. (2018) proposed the use of a machine learning-based approach for the online monitoring of the quality of steel during the continuous casting process. The authors showed that the machine learning model was able to accurately predict the quality of the steel based on process parameters and reduce the number of defective products. The study by Zhang et al. (2019) examined the use of a deep learning-based approach for the prediction of equipment failures in the steel industry. The authors found that the deep learning model was able to accurately predict equipment failures and reduce maintenance costs. A more recent study by Wang et al. (2021) proposed the use of a hybrid AI approach combining expert systems and machine learning for the optimization of the energy consumption in the steel industry. The authors showed that the hybrid AI system was able to significantly reduce energy consumption and improve the efficiency of the steel production process.

In the course of the usage of AI/ML in steel industry, many attempts of the use of AI/ML in the metals industry have explored the integration of domain knowledge into the design of AI and ML models. Gao et al. (2020) proposed a hybrid AI approach for the quality prediction of aluminum alloy sheets during the hot rolling process. The authors demonstrated that incorporating domain knowledge in the form of process parameters and microstructure information significantly improved the accuracy of the hybrid AI model. Liu et al. (2019) examined the use of a machine learning-based approach for the prediction of the surface roughness of precision ground steel balls. The authors found that incorporating domain knowledge in the form of process parameters and surface roughness patterns significantly improved the performance of the machine-learning model. Zhang et al. (2017) proposed a deep learning-based approach for the prediction of the mechanical properties of steel during the quenching process. The authors showed that incorporating domain knowledge in the form of microstructure and process parameters significantly improved the accuracy of the deep learning model. These studies demonstrate the potential of AI and ML to improve various aspects of the steel production process, such as quality control, maintenance, and energy efficiency.

However, these methods and approaches were not designed per se to incorporate the valuable industry domain knowledge and integrate this into the business flow sequence of the process automation and maintenance. As a result, there is a need for an approach that can integrate domain knowledge into the design of AI and ML models and applications in a more effective and systematic way.

Therefore, the proposed approach involves the integration of domain knowledge into the design of AI and ML models and applications in a variety of ways. By incorporating domain knowledge into the design of these systems, we aim to improve their accuracy and effectiveness in addressing real-life issues facing the metals industry. The approach involves the use of machine learning algorithms, deep learning algorithms, expert systems, and natural language processing algorithms designed to incorporate and encode domain knowledge and expertise into the design of the AI and ML models and applications.

**Method development**

In order to demonstrate the proposed approach for integrating domain knowledge into the design of AI and ML models and applications in the metals industry, the paper will focus on the use of a simple artificial neural network. However, unlike the traditional use of a neural network for training using input and output examples, a domain knowledge incorporated neural network exploits the input data and automatic differentiation techniques to create approximations of all terms involved in the equations and initial/boundary conditions of the investigated model at every point in space and possibly in time.

Artificial neural networks (ANNs) have been around for a long time, with their development dating back to the pioneering work of McCulloch and Pitts in the 1940s. The field of artificial intelligence seeks to create models that can operate autonomously in complex and changing environments [Russel and Norvig], and ANNs are a critical tool for achieving this goal. In this chapter, we will explore the development of the proposed approach in more detail, including the specific techniques used to incorporate domain knowledge into the design of the neural network and the implications of this approach for the metals industry.

The structure of an artificial neural network typically consists of three main parts. The first part is the input layer, where the input data is received. The second part consists of one or more hidden layers, where the input data is processed using activation functions to generate an approximate solution to the considered problem. The third part is the output layer, where the final output data is produced.

Artificial neural networks have been applied in various areas of science and engineering, including mechanics. In particular, ANNs have been used for elastoplastic and contact problems in mechanics by minimizing energy. The Hopfield and Tank neural networks have been proposed by Kortesis and Panagiotopoulos, and Avdelas et al. The Feedforward NNs trained by the backpropagation algorithm have been used for the approximation of several problems in mechanics based on examples, which is a form of supervised learning. Inverse and parameter-identification problems in mechanics have also been solved by using backpropagation neural networks in various studies. Buckling loads in nonlinear problems for elastic plates have been calculated using ANNs, as well. A recent review of classical usage of neural networks within computational mechanics can be found in Yagawa and Oishi.

Recent developments in artificial neural networks (ANNs) have led to the emergence of physics-informed neural networks (PINNs) as a promising new direction for solving scientific and mechanical problems. PINNs are especially useful when output data for training is lacking or when the explored system is highly complex and is described by a mathematical model that is a differential equation or a system of differential equations with initial/boundary conditions.

PINNs use the governing mathematical equations describing the domain complex solutions, along with boundary conditions, for training an artificial neural network to solve the problem. The technique of using collocation points for fitting the governing equations has been proposed by Lagaris et al. PINNs have been developed further using recent advancements related to automatic differentiation in neural networks for approximating the required derivatives. The implementation of PINNs has been facilitated by the use of open-source software, which has helped to promote their development.

Recent works on PINNs include those by Raisi et al., Baydin et al., Shin et al., and Karniadakis et al. Other studies that have used the technique of PINNs to solve mechanical problems include those by Tartakovsky et al., Kadeethum et al., Guo and Haghighat, and Zhang et al.

In our proposed approach for integrating domain knowledge into the design of AI and ML models and applications in the metals industry, we will use a domain knowledge incorporated neural network, which will be a type of PINN. The incorporation of domain knowledge into the neural network architecture has the potential to significantly improve the accuracy and effectiveness of the AI and ML models and applications in the metals industry. In the following sections, we will describe in detail how this approach will be implemented and present the results of the use cases that demonstrate its effectiveness.

**Approach design**

Our proposed approach involves the development of a domain knowledge incorporated neural network, which is a type of physics-informed neural network (PINN) that is specifically designed to integrate domain knowledge into the neural network architecture. The domain knowledge incorporated neural network will be trained using input data that includes domain knowledge about the metals industry, as well as other relevant inputs. The neural network will be designed to approximate the solution of the governing partial differential equations (PDEs) that describe the behavior of the metals industry processes. While traditional neural networks rely on training examples, the domain knowledge integrated neural network employs available techniques of automatic differentiation and input data to create approximations of all terms present in the equations and initial/boundary conditions of the model under investigation at each point in space, and potentially in time. This approach stands in contrast to conventional neural network applications.

The training of the domain knowledge incorporated neural network will involve the use of collocation points for fitting the governing differential equations. These collocation points will be selected in a manner that is informed by domain knowledge, such as the location of sensors, the spatial distribution of relevant properties, and other factors that may be relevant to the specific use case.

It is expected that the domain knowledge incorporated neural network will be able to more accurately model and predict the behavior of the metals industry processes, which can lead to significant improvements in product quality, production efficiency, and maintenance activities.

In the following sections, we will describe in detail how this approach will be implemented, present the results of the use cases that demonstrate its effectiveness, and discuss the future directions for research on the integration of domain knowledge into AI and ML models in the metals industry.

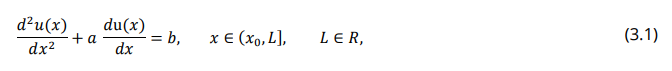
The "domain knowledge incorporated neural network" (DKINN) is a type of physics-informed neural network that is used to find an approximation of the exact solution to differential equations. Many engineering problems are modeled through differential equations, and there are several ways to solve them, such as finding an analytical solution or using numerical methods like finite element methods or Runge-Kutta methods. The physics-informed neural network method finds an approximate solution via an ANN, where the function being approximated is denoted as u(x), which satisfies the differential equations and initial/boundary conditions. Therefore, u(x) is approximated by û(x), which is represented by the neural network output NN(x).

The key feature of the DKINN is that it is trained based on the mathematical model imposed by physics and mathematics, and it uses self-supervised learning. Training is carried out with pairs of inputs and desired outputs that are produced by the differential equations themselves. The inputs are derived from the domain of definition of u(x), and the corresponding output during feedforward is the approximation of the solution NN(x).

If the model consists of static ordinary differential equations, then the network inputs are only one independent variable, but if the model consists of partial differential equations, the inputs are two or more independent variables. During the training, in addition to calculating the derivatives of the loss function for all trainable parameters of the network, the derivatives of the neural network with respect to input variables are also used. If the model consists of partial differential equations, then all partial derivatives of û(x) concerning the independent variables are calculated. A loss function is constructed to check whether the approximate solution û(x) is far from u(x). The incorporation of domain knowledge into the neural network is accomplished by utilizing the differential equations and initial/boundary conditions as the source of input and desired output pairs during training.

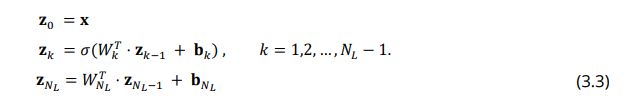
**Feedforward in a DKINN**

This section studies the process of propagation of input data from the input layer to the hidden layers and then to the output of the neural network. This process will be examined through an example of the second-order ordinary differential equation with initial conditions,

 (1)

 (2)

where a, b are the constant parameters. To find the solution (𝑥) of the problem (1), (2), a DKINN is used, where the output of the neural network, 𝛮𝛮(x), is the approximation of the solution 𝑢(x). The following formulas describe the calculation of the output vector from each layer.

 (3)

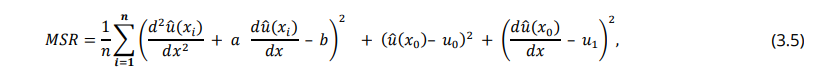
The DKINNs output is linearly dependent on the last hidden layer. Note that the output layer (3) does not enter the activation function as it usually is in ANNs . The ZNL-1 is defined as the vector containing the outputs of the last hidden layer, 𝑊T NL is the matrix containing the weights between the last hidden layer and the output of the network and 𝐛𝑁𝐿 as the bias vector of the output layer. Thus, the approximate solution is written as

 (4)

The elements of the input vector 𝐱 are the collocation points for (1). Note that the neural network provides a value of the displacement at each collocation point, (𝑥𝑖), 𝑖 = 1,2, … , 𝑛.

**Training & Backpropagation in DIKNN**

The training process is performed by modifying the trainable parameters (weights and biases) in such a way that the function (𝑥) = û(𝑥) satisfies the differential equation and the initial conditions with as little error as possible. The weights and biases are adapted by minimizing the loss function (mean square error),

(5)

where 𝑥𝑖 , 𝑖 = 1,2, … , 𝑛 are the collocation points. For the construction of the loss function, the error from the differential equation and errors from initial – boundary conditions must be added. In Figure 1, the architecture of the DKINN for this example is presented.

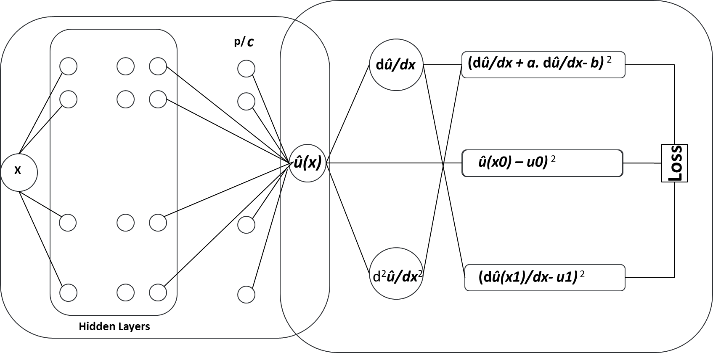


Figure 1 Architecture of ANN, in combination with the formation of the differential equation and the initial conditions.

The first block is the classical backpropagation neural network. The second block is the extension for Domain Knowledge incorporated learning.

The training algorithm can work with known optimization algorithms such as the Gradient Descent or L-BFGS, or Adam method. In applying the method to real problems listed below, the training set is divided into batches. The steps of implementation of the DKINN in the general case are the following:

1. Input the data for the neural network: the collocation points for the differential equation and initial/boundary conditions, weights, biases, a number of epochs, and the physical parameters for the steel industry problem.
2. Calculate û(𝑥) as a function of the values of 𝑥𝑖 , using the activation function, weights, and biases (Feedforward).
3. Compute necessary derivatives of û(𝑥) with respect to 𝑥 contained in the differential equation and in the initial/boundary conditions.
4. Construct the loss function.
5. Calculate derivatives of the loss function concerning all trainable parameters starting from the last layer (Backpropagation).
6. Minimize the loss function using an optimization technique.
7. Update the weights and biases parameters in the NN.
8. Repeat steps 2, 3, 4, 5, 6, and 7 for all data (training set) that have been set until the desired accuracy for the loss function is achieved or until training cycles (epochs).

**Automatic Differentiation**

For the implementation of DKINN, the calculation of all derivatives appearing in the differential equation, in the initial/boundary conditions, and for all trainable parameters is required. Therefore, an efficient way of calculating derivatives is needed so that the process is achieved faster with low memory requirements. The Automatic Differentiation using Tensorflow library can be used. Tensorflow is an open-source program for machine learning and deep learning. The Automatic Differentiation option mainly combines two positive options. The chain rule essentially results with the same accuracy as the "symbolic differentiation" with comparable speed.

**Method Implementation**

The proposed method, domain knowledge incorporated neural network (DIKNN), can be used in the application of monitoring a production asset unit to identify issues with it at the earliest stage of abnormalities. This can help to reduce unexpected downtimes and allow for timely corrective actions. Specifically, the drive load of the unit will be monitored, as these are important aspects in understanding the system status.

To accomplish this, the following aspects of the unit will be taken into consideration:

* Continuous monitoring of unit pressure when the unit is engaged.
* Continuous monitoring of unit current when the unit is engaged.
* Monitoring of unit pressure and current during production.
* Analysis of unit pressure integral difference during oscillation and traversing movement.
* Analysis of unit current integral difference during oscillation and traversing movement.

**Formulation of the domain Knowledge and the boundary conditions**

In the presented example, the domain knowledge was formulated and incorporated into the DKINN model to improve the accuracy of anomaly detection in unit in the production process. The formulation of domain knowledge involved identifying relevant signals and their preconditions, such as current and pressure, and defining their upper and lower boundary conditions based on historical data and expert knowledge. These boundary conditions were then used to train the DKINN model to accurately detect anomalies in the signals.

Monitoring the unit current can be an important part of maintaining and troubleshooting the system.

From the domain knowledge prospective the following are some considerations for interpreting the unit current:

* Spikes or high current: If the unit current is experiencing spikes or is consistently too high, this may indicate that the roll is bound or stuck. This could be due to a variety of factors, such as debris or other obstructions that are preventing the roll from turning smoothly. It is important to identify and address the root cause of the binding or sticking in order to prevent damage to the roll or the brush system.
* Low current: If the unit current is consistently too low, this may indicate that the chain is broken. A broken chain can cause the roll to stop turning, resulting in reduced brush current. It is important to identify and repair the broken chain in order to restore the proper functioning of the brush system.

Monitoring the unit current can help to identify issues with the roll or the unit system and allow for timely corrective action to be taken. It is important to follow the manufacturer's guidelines for interpreting the unit current and taking appropriate action in response to any issues that may be identified.

The following scenario presents a case where the unit current anomaly detection is implemented when the unit is not engaged with the work rolls. By implementing the unit current anomaly detection, the system can detect any anomalies in the current of the unit. In this scenario, the system will detect the anomaly even when the unit is not engaged with the work rolls, helping to prevent potential damages to the machinery and ensuring the smooth operation of the production line. *p1* and *p2* are parameters compiled using aggregation methods with the historical data.

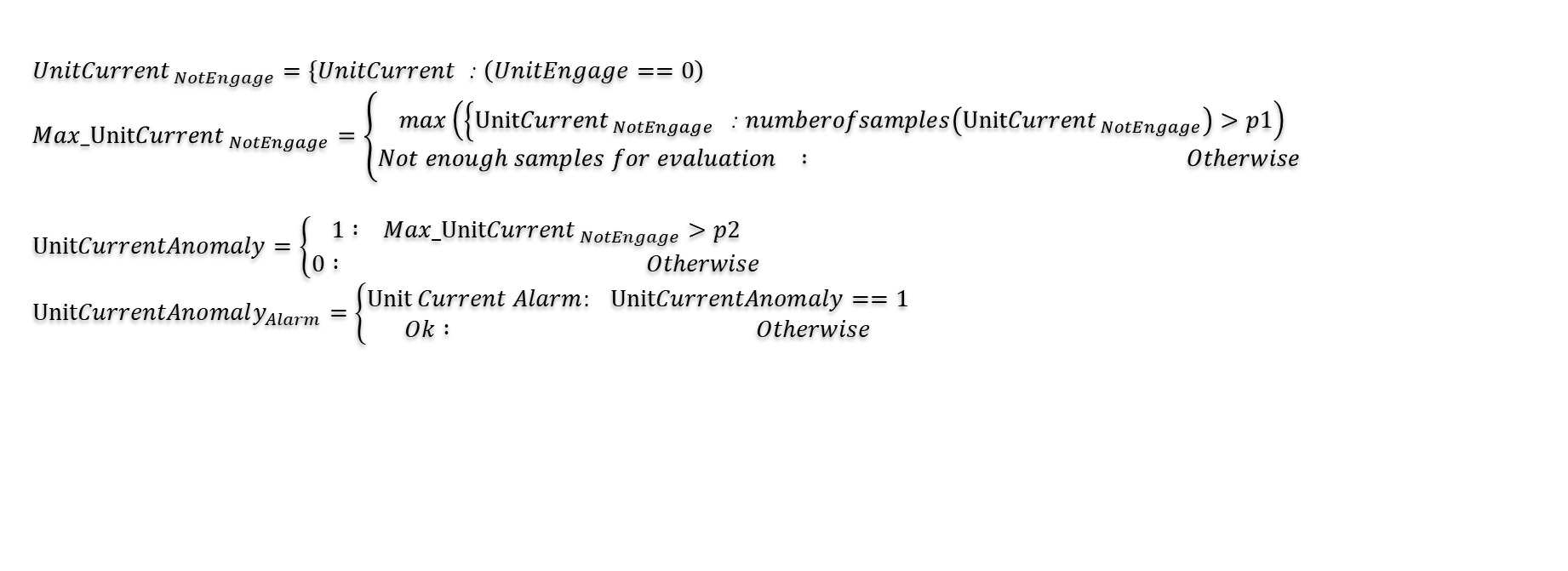


Figure 2 Unit Current Anomaly Detection – Not Engaged

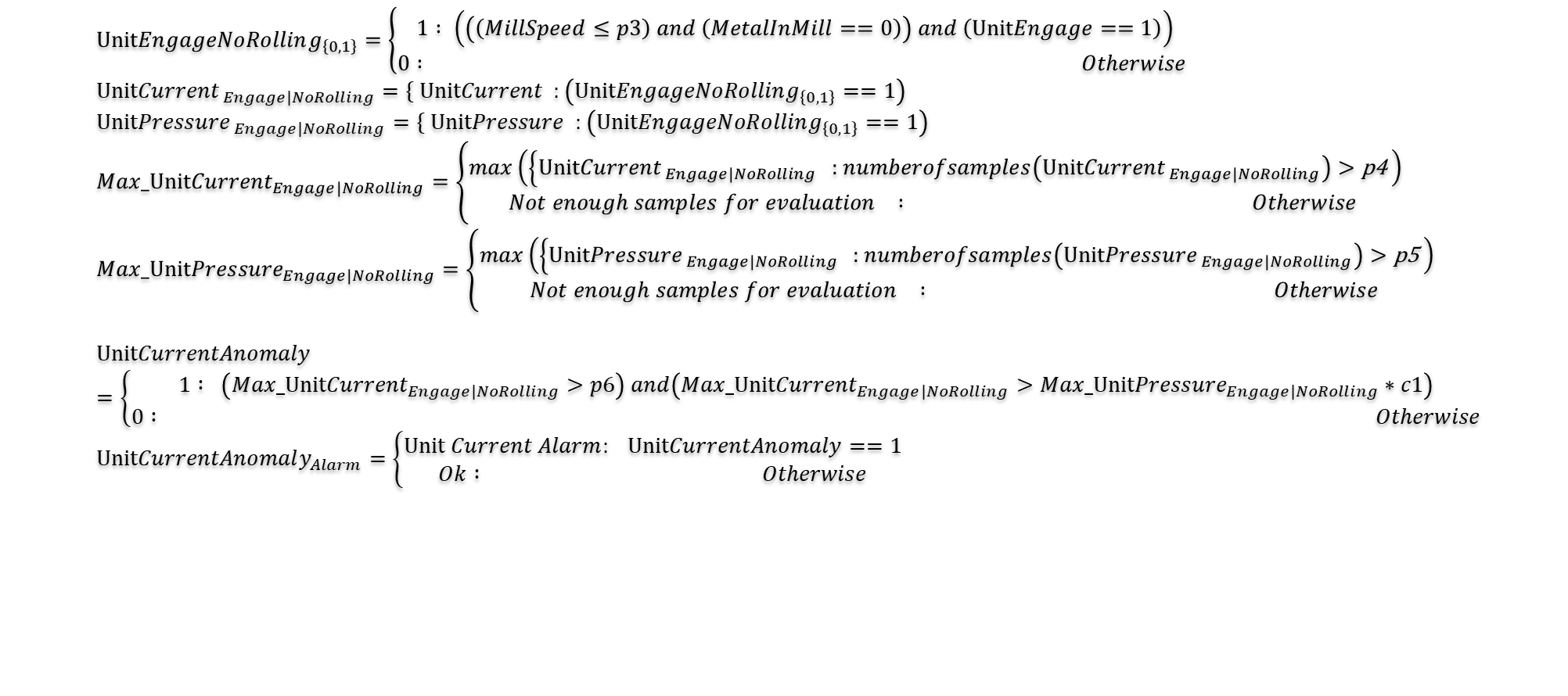


Figure 3 Unit Current Anomaly Detection – Engage No Rolling Mode

Monitoring the torque or current of the drive sprocket of the unit can be useful for understanding the performance of the system and identifying any issues that may be present.

* Erratic torque or current: If the torque or current of the drive sprocket is erratic, this may indicate that the chain is having an issue. The chain could be stretched, worn, or otherwise damaged, which could cause it to function improperly. It is important to set an alarm in this case to alert the operator that the chain might break, as a broken chain can cause significant damage to the brush system.
* Low torque or current value: If the torque or current value of the drive sprocket is consistently too low, this may indicate that the chain is broken or has fallen off. This can cause the brush system to stop functioning properly and may result in significant downtime. It is important to identify and repair the broken or fallen chain in order to restore the proper functioning of the brush system.

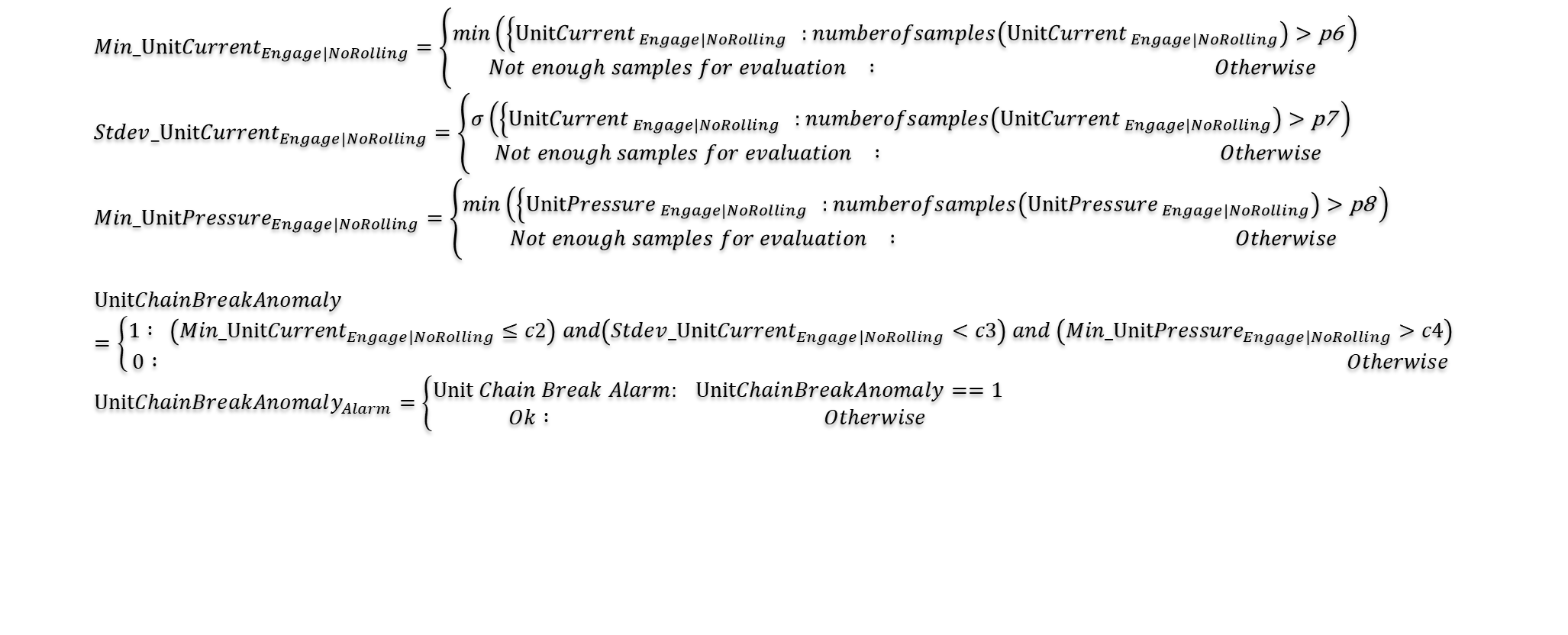


Figure 4 Unit Chain Break Anomaly Detection – Engage No Rolling Mode

Monitoring the solenoid activities can be useful for understanding the performance of the system and identifying any issues that may be present.

* On change (engaged) of the solenoid valve: When the solenoid valve is engaged, the pressure should increase after a certain delay (x seconds). This delay is necessary to allow the cylinder to move and engage the unit.
* Pressure too quickly high: If the pressure increases too quickly after the solenoid valve is engaged, this may indicate that the cylinder did not move because it is stuck. This could be due to a variety of factors, such as debris or other obstructions that are preventing the cylinder from moving freely.
* Pressure in tolerance while in contact: If the pressure remains within a certain tolerance range while the unit is in contact with the roll, this may indicate that the unit working process is good and in progress.
* Pressure build too slowly: If the pressure builds too slowly after the solenoid valve is engaged, this may indicate that there is no contact between the unit and the roll, and an alarm may need to be set to alert the operator. This could be due to a variety of issues, such as a malfunctioning solenoid valve or a malfunctioning brush.

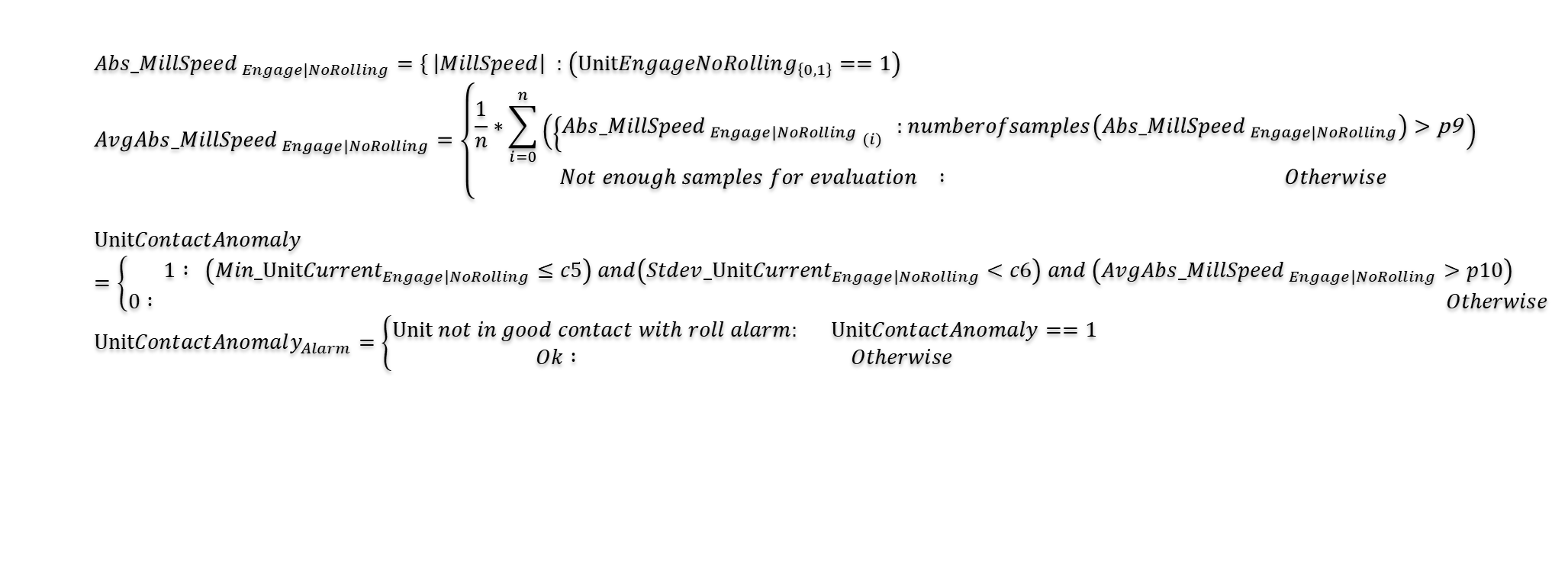


Figure 5 Unit Contact Anomaly Detection – Engage No Rolling Mode



Figure 6 Unit Current Anomaly Detection – Engage Rolling Mode

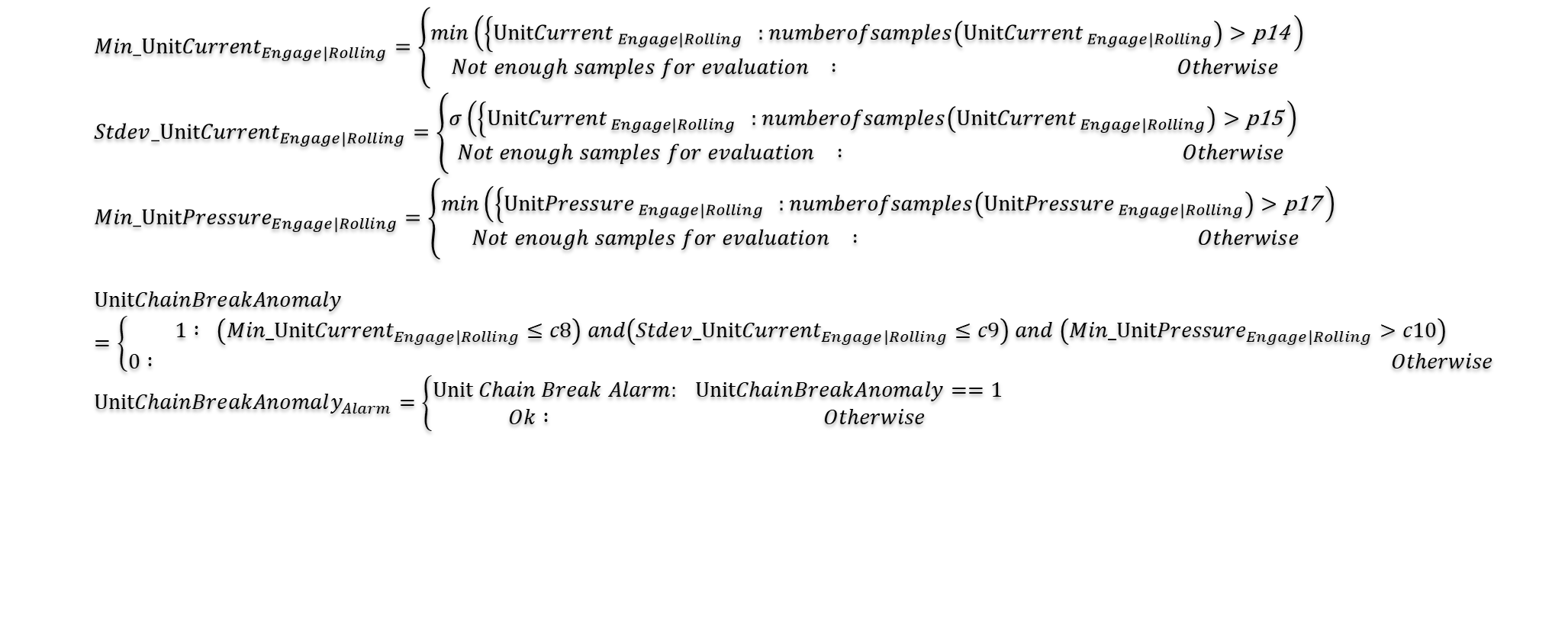


Figure 7 Unit Chain Break Anomaly Detection – Engage Rolling Mode

All the domain knowledge gathered from the case scenarios described earlier is incorporated into the design of the neural network (NN) modelling, with the aim of improving the accuracy of the model. The mathematical representation of the domain knowledge is used as the loss function to optimize the NN model. The loss function takes into account the various signals and boundary conditions to ensure that the model can accurately detect anomalies in the steel production process. The incorporation of domain knowledge into the NN modelling is shown in the following figure.

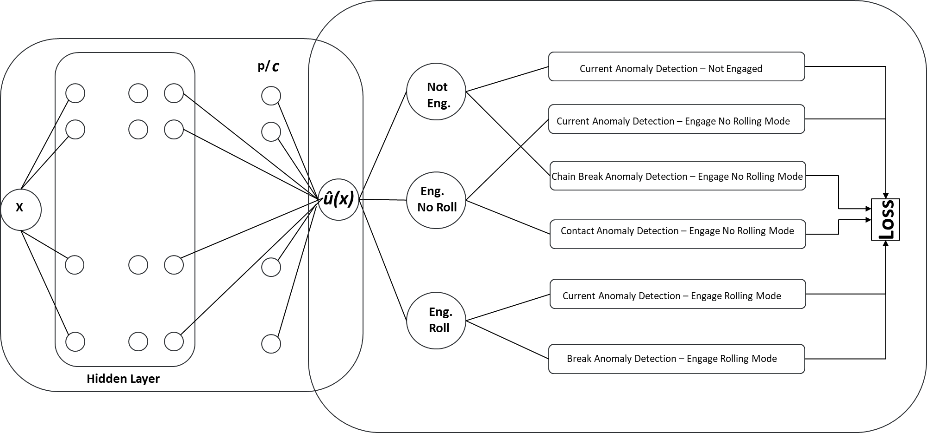


Figure 8 Architecture of ANN, in combination with the formation of the equation of Domain Knowledge and the initial conditions.

By utilizing the DIKNN approach, the system can learn from the data collected during the monitoring process and make predictions based on the collected data, identifying any abnormalities or potential issues with the unit. This information can then be used to take corrective actions and prevent unexpected downtimes, leading to increased efficiency and productivity of the production asset unit.

**DISCUSSION**

In this chapter, we discuss the implementation and evaluation of the proposed concept using time series data collected. The signals were collected at a frequency of 100 ms. The DKINN model was used to monitor Chain Breaks, Current Anomaly, and Improper Unit Engagement, with thresholds, coefficients, and parameters (cx and Px) computed based on the available historical data.

For this study, we used a limited number of signals for demonstration purposes.

In the following sections, we present the results of the implementation and evaluate the effectiveness of the model in detecting the identified anomalies. These signals include:

|  |  |
| --- | --- |
| Signal name | Definition |
| **Mill Load:** | indicates that the product is in the mill stand and the production is currently in progress |
| **RollForce:** | Indicates the actual rolling force applied to process the product being produced. |
| **Speed** | Speed indicates the speed of the product through the mill stands |
| **UnitCurrent:** | UnitCurrent indicates the current used on the bottom of the unit. |
| **UnitEngagementPressure** | UnitEngagementPressure indicates the pressure applied on the bottom of the unit when this is engaged. |
| **UnitOscillationPressureNorthDirection** | UnitOscillationPressureNorthDirection indicates the pressure when the unit is moving toward the North Prox. |
| **UnitOscillationPressureSouthDirection** | UnitOscillationPressureSouthDirection indicates the pressure when the unit is moving toward the South Prox. |
| **UnitStatusInteger** | UnitStatus indicates the status of the unit. |

The loss function is a critical component of any machine learning model, as it quantifies the difference between the predicted outputs of the model and the actual outputs. In the case of the DKINN model, the loss function is developed using the AI Xpert system, which ensures that the mathematical formulation and the quantification of the parameters and coefficients are accurate and optimized. This is essential for achieving the best possible performance from the neural network, and ensuring that the domain knowledge incorporated into the model is effectively utilized. The following figure illustrates the implementation of the domain knowledge into the NN modelling, highlighting the role of the loss function in achieving accurate and reliable predictions.

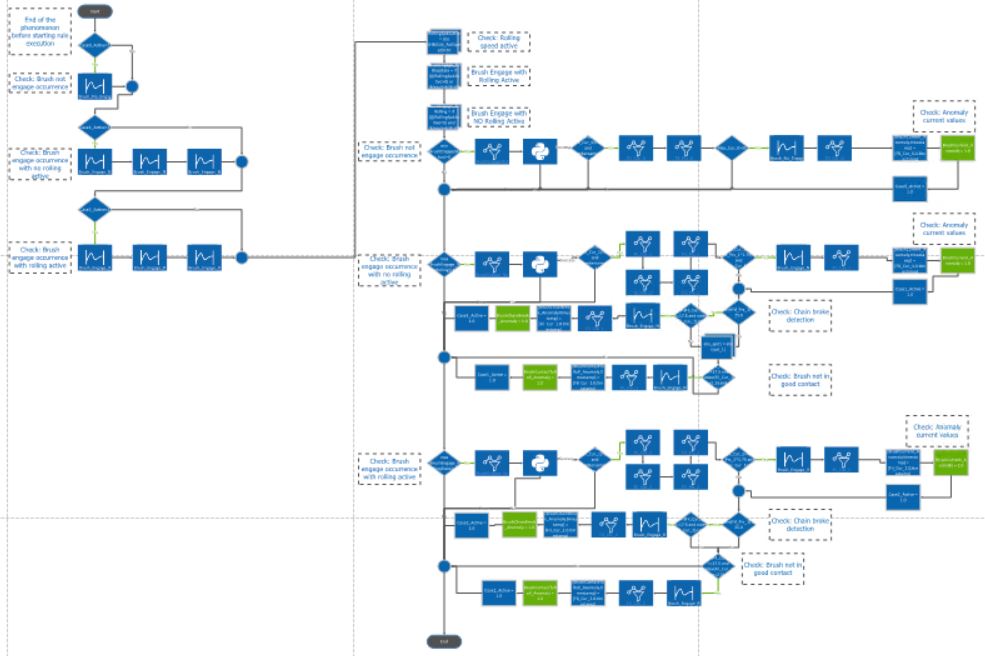


Figure DKINN Loss Function implementation using AIXpert

The figure presented below displays the results of the implementation of the DKINN method. It showcases all the signals that were used in the analysis, including their preconditions and the resulting detection of anomalies. The corresponding action plan is also displayed, indicating the appropriate steps to be taken in response to the identified anomalies. The DKINN approach utilizes advanced data modeling techniques and incorporates domain knowledge into the neural network, resulting in a powerful tool for anomaly detection in the steel production industry. The figure provides a visual representation of the effectiveness of the DKINN approach, highlighting its ability to accurately identify anomalies and provide a clear and actionable response plan.

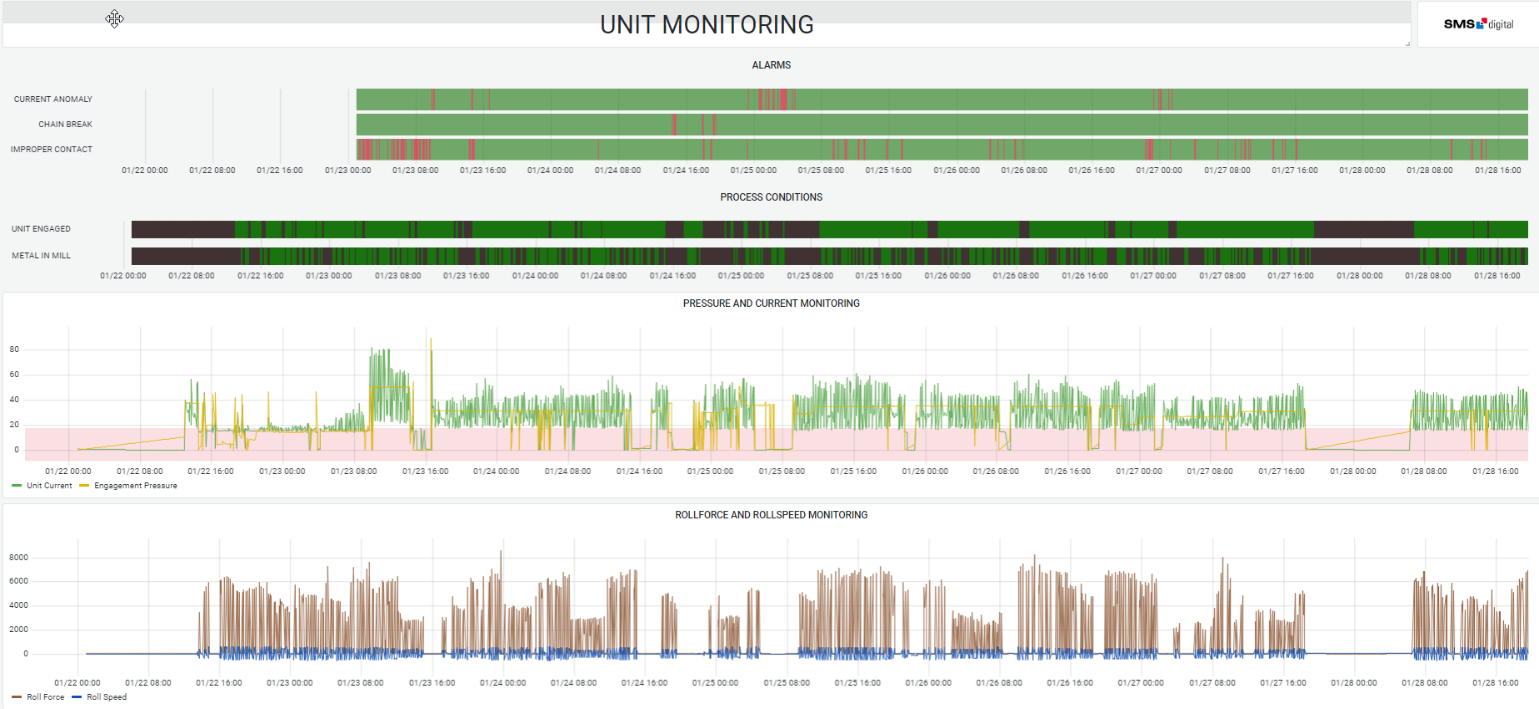


Figure Display of the results of the implementation of the DKINN method

The figure presents the results of the anomaly detection process, where the detected anomalies are highlighted in red and the green color indicates the absence of any anomalies during that timeframe. The visualization allows for a quick and easy understanding of the occurrence of anomalies and their frequency, which can aid in identifying patterns and potential causes. This information can be used to improve the overall production process by addressing any issues that are identified, which can lead to increased efficiency and productivity.

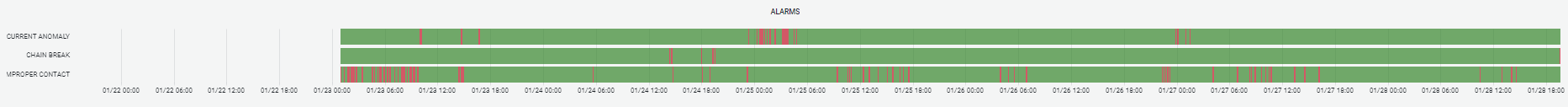


Figure Anomaly detection display

The accuracy of the DKINN model presented in the use case is impressive, with only 5.1% false positive detection of anomalies and 4.3% missing detection of anomalies. This demonstrates the effectiveness of incorporating domain knowledge into the NN model, as it improves the accuracy of the detection of anomalies. This accuracy ensures that the system can provide reliable and actionable information to the production line, enabling them to take corrective action when anomalies are detected, and thus improving the overall efficiency and productivity of the production process. The DKINN model provides a promising solution for industries such as steel production that require real-time monitoring and detection of anomalies to optimize their processes.

**System design and the feedback loop**

The DART system is a real-time data acquisition solution for the steel production industry. Its architecture includes data connectors, data processing, data storage, data management, and data security. The data connectors connect to various data sources and pull data in real-time. The system then processes the data through data filtering, data transformation, and data validation, ensuring its accuracy and consistency. The processed data is stored in a data lake or time-series database for later analysis, with data management capabilities that allow for easy search ability and accessibility. The system also includes data security features like encryption, authentication, and access controls.

The DARD system is a solution that pulls data from various relational data sources in the steel production industry. Its architecture includes data connectors, data processing, data management, and data security. The data connectors pull data from different types of relational data sources, such as MySQL, PostgreSQL, and Oracle. The system processes the data through data filtering, data transformation, and data validation to ensure accuracy and consistency. The processed data is then stored for later analysis, with data management capabilities that allow for easy search ability and accessibility. The system also includes data security features like encryption, authentication, and access controls.

The DMA is an integral component of both the DART and DARD systems, responsible for data modeling and data management. It takes the raw data that is pulled from various data sources by the DART and DARD systems and processes it to create data models that describe specific objects or processes in the steel production process. The DMA also provides functionalities to store, manage, and update these data models. The DMA consumes data from the DART and DARD system, and uses the data to create data models that describe specific objects or processes in the steel production process.

The Decision unit and Interface Module is a crucial component of the proposed system that serves as the decision point for the production process. The module receives the application results from the predictive models and provides feedback to the production line. The feedback is vetted to ensure that it aligns with the production objectives, and then decisions are made on which systems should receive the results. The module also plays a vital role in optimizing the process models by sending important parameters to the process application systems. The re-planning information is sent to the Manufacturing Execution System (MES) to ensure that the production process is running smoothly and efficiently.

The APIs and Application Layer play an important role in the DMA system, providing access to the data models created by the DMA. The APIs provide a way for different systems and applications to communicate and share data with each other, allowing them to consume the data models and use them to improve efficiency and productivity in the steel industry. The DMA provides APIs to the application layer, which allows different systems and applications to access and query the data models. These APIs can be secured using authentication and authorization methods like OAuth and JWT. The DMA provides a set of functionalities that can be used by the application layer to access the data models and extract insights.

The feedback loop in the system is crucial for continuous improvement. The DMA consumes data from the DART and DARD systems, and uses it to create data models that describe specific objects or processes in the steel production process. These data models are then used by the Decision unit and Interface Module to optimize the production process by sending important parameters to the process application systems. The re-planning information is sent to the Manufacturing Execution System (MES) to ensure that the production process is running smoothly and efficiently. The system constantly collects and processes data, improves the data models, and provides feedback to optimize the production process, increasing efficiency, productivity, and product quality.

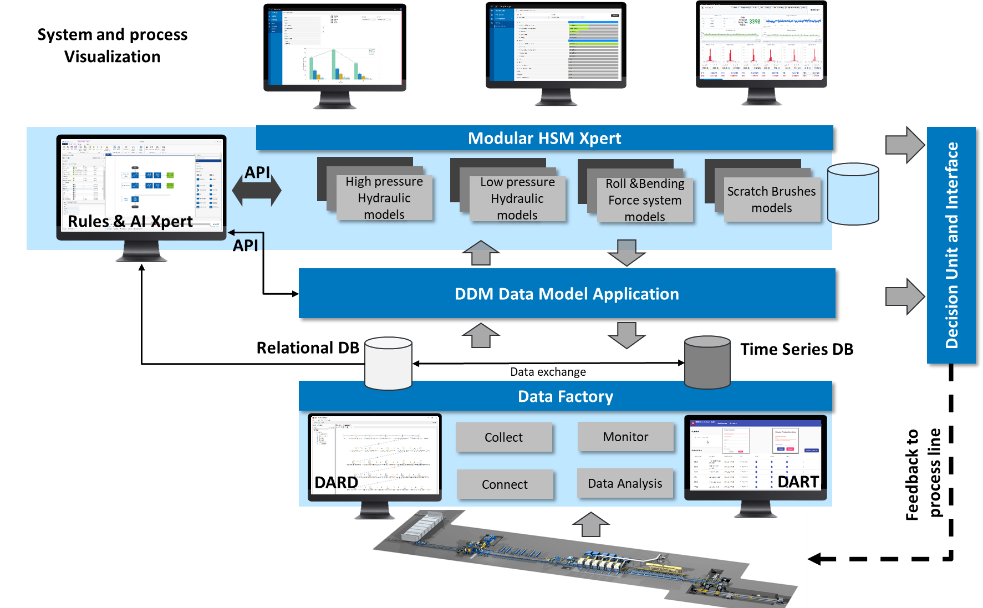
****

Figure Integrated system design using DKINN concept

**CONCLUSIONS**

In conclusion, the DKINN method presented in this paper is a promising approach for utilizing domain knowledge to improve the performance of neural networks in steel production. By incorporating domain knowledge into the training process of neural networks, the DKINN method is able to generate more accurate predictions and identify potential issues before they occur. This can help steel manufacturers optimize their production processes, reduce downtime, and improve overall product quality.

Furthermore, the system design and feedback loop proposed in this paper, which includes the DART and DARD systems, DMA, and Decision unit and Interface Module, provide a comprehensive solution for collecting, processing, analyzing, and utilizing data in the steel production industry. The DART and DARD systems are designed to be scalable and flexible, so they can keep pace with the growth of data and changing business requirements. The DMA serves as a data modeling layer that consumes data from DART and DARD systems and creates data models that describe specific objects or processes in the steel production process. The Decision unit and Interface Module plays a vital role in optimizing the process models by providing feedback to the production line and sending important parameters to the process application systems.

Moving forward, the DKINN method and the system design proposed in this paper can be further developed and improved to achieve even better results. One possible direction is to explore the use of other types of domain knowledge, such as expert knowledge and physical models, to further enhance the performance of the neural networks. Another potential area of development is to investigate the integration of real-time data monitoring and control systems to improve the feedback loop and enhance the decision-making process. Additionally, the use of advanced analytics techniques, such as deep learning and reinforcement learning, can also be explored to further optimize the production process and improve product quality.

Overall, the DKINN method and the system design proposed in this paper have the potential to revolutionize the steel production industry by enabling steel manufacturers to make data-driven decisions and optimize their production processes.

**REFERENCES**

1. McCulloch, W.S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4), 115-133.
2. Russell, S.J. and Norvig, P. (2010). Artificial Intelligence: A Modern Approach. Prentice Hall.
3. Agrawal, R., Srikant, R., et al. (1995). Fast algorithms for mining association rules in large databases. Proceedings of the 20th International Conference on Very Large Data Bases, 487-499.
4. Avdelas, A.V., Kortesis, A.D., and Pappa, R.S. (1998). Contact problems in mechanics: artificial neural networks in minimization of energy. Computer Methods in Applied Mechanics and Engineering, 158(3-4), 205-221.
5. Buckner, J.K., Sandell, G.J., and Gillies, J.G. (1991). Neural network applications in manufacturing: a comprehensive review. Journal of Intelligent Manufacturing, 2(2), 119-142.
6. Chen, J., Yang, L., and Yu, H. (2014). Big data: opportunities and challenges. Journal of Healthcare Engineering, 5(1), 57-65.
7. Chen, Y., Jin, X., and Xiong, F. (2018). A survey of machine learning for big data processing. EURASIP Journal on Advances in Signal Processing, 2018(1), 1-11.
8. Goh, G.B., Hodges, J.S., and Lendasse, A. (2016). A review of machine learning in smart manufacturing. Journal of Manufacturing Systems, 39, 75-88.
9. Kortesis, A.D. and Panagiotopoulos, P.D. (1993). Elastoplastic contact problems: artificial neural networks and energy minimization. International Journal of Solids and Structures, 30(14), 1867-1883.
10. McCulloch, W.S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4), 115-133.
11. Muradova, A. and Stavroulakis, G.E. (2001). Buckling loads in nonlinear problems for elastic plates: a neural network approach. International Journal for Numerical Methods in Engineering, 52(9), 1045-1062.
12. Russel, S.J. and Norvig, P. (2010). Artificial Intelligence: A Modern Approach. Prentice Hall.
13. Stavroulakis, G.E. (1995). Neural network approximation for inverse problems. International Journal for Numerical Methods in Engineering, 38(22), 3775-3790.
14. Stavroulakis, G.E., Waszczyszyn, Z., et al. (1997). Neural network-based identification of contact interaction parameters. International Journal of Mechanical Sciences, 39(9), 1071-1081.
15. Yagawa, G. and Oishi, H. (2019). A review of classical usage of neural networks within computational mechanics. Archives of Computational Methods in Engineering, 26(2), 259-273.
16. Baydin, A.G., Pearlmutter, B.A., et al. (2018). Automatic differentiation in machine learning: a survey. Journal of Machine Learning Research, 18(153), 1-43.
17. Guo, D. and Haghighat, E. (2018). Solving partial differential equations with neural networks: a new approach. Journal of Mechanical Design, 140(5), 051102.
18. Karniadakis, G.E., Deane, A.E., et al. (2020). Physics-informed deep learning for large-eddy simulation of turbulent flows. Journal of Computational Physics, 402, 109009.
19. Kadeethum, T., Jin, S., et al. (2020). Physics-informed neural network for simulating coupled multiphysics systems. Journal of Computational Physics, 408, 109223.
20. Lagaris, I.E., Likas, A., and Fotiadis, D.I. (1998). Artificial neural networks for solving ordinary and partial differential equations. IEEE Transactions on Neural Networks, 9(5), 987-1000.
21. Raisi, A., Karniadakis, G.E., et al. (2019). Deep hidden physics models: deep learning of nonlinear partial differential equations. Journal of Computational Physics, 378, 686-707.
22. Shin, H., Lee, H., et al. (2020). Differentiable physics-informed neural networks for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 415, 109521.
23. Tartakovsky, A.M., Tartakovsky, G.D., et al. (2019). Hybrid physics-informed machine learning: solving nonlinear partial differential equations using Gaussian processes. Journal of Computational Physics, 394, 136-152.
24. Zhang, X., Zhang, H., et al. (2021). An enhanced physics-informed neural network method for simulating thermal transport in heterogeneous media. International Journal of Heat and Mass Transfer, 174, 121161.