INFORMATION AND MEASUREMENT TECHNOLOGIES IN MECHATRONICS AND ROBOTICS

VIBRATION PARAMETER PREDICTION ALGORITHM FOR OBJECTS BASED ON HISTORICAL DATA FOR PREDICTIVE MAINTENANCE

Oleksandr Ryshkovskyi, PhD, Ass. Professor; Markiian Lukashiv, PhD student,
Lviv Polytechnic National University, Ukraine;
e-mail: markiian.b.lukashiv@lpnu.ua

https://doi.org/10.23939/istcmtm2025.03.

Abstract. The article examines the advantages and disadvantages of an algorithm used to determine the time interval for a monitored object to reach a specific vibration level for use in predictive maintenance in Industry 4.0. The studied algorithm will potentially predict how long before an object fails, helping to reduce downtime and maintain production flow.

Key words: Accelerometer, Industry 4.0, Predictive maintenance, Measurement of vibration parameters.

1. Introduction

In the modern industrial world, where uninterrupted equipment operation is critical for efficiency and safety, vibration monitoring plays a key role in predictive maintenance. Existing software for analyzing vibration parameters can already perform continuous, high-level monitoring, detect anomalies, and alert the operator when set vibration limits are reached. These systems are indispensable for identifying current deviations from the norm and preventing sudden failures by informing about the need for immediate intervention once vibration has reached a dangerous level.

However, despite the high functionality of available solutions, most of them focus on the current state of the equipment. They inform you that a limit has already been reached or exceeded, but do not provide information on when this will happen. This "blind spot" in forecasting creates significant challenges for effective maintenance planning. The inability to predict the time when vibration will reach a critical threshold means that reactive maintenance measures are prioritized over proactive ones [1].

This is where the vital need for developing and implementing algorithms to predict the time it will take to reach a specific vibration threshold becomes important. Such functionality shifts a monitoring system from merely signaling problems to preventing them. The presence of such an algorithm would enable:

- Optimizing maintenance schedules.
- Lowering the risk of accidents and disasters.
- More efficient resource management.

Therefore, developing and integrating an algorithm to predict when a vibration threshold will be reached is not just an enhancement to existing software but a strategic move toward full predictive maintenance, giving companies a significant competitive edge in today's market. This feature addresses the urgent industry need to shift from reactive responses to proactive equipment condition management, ensuring stable, safe, and cost-effective operations.

2. Drawbacks

Modern software for vibration analysis effectively monitors equipment and signals when limits are reached, but its reactive approach does not allow for predicting when a problem will occur. This lack of predictive functionality results in significant drawbacks, negatively impacting a company's operational efficiency, safety, and finances. The main disadvantages caused by the absence of a prediction algorithm are:

- Reactive, not proactive, maintenance. Without forecasting, maintenance remains reactive because decisions are made only after vibration has reached an unacceptable level, making it impossible to plan repairs in advance and prevent unplanned equipment shutdowns.
- Unforeseen downtime and production losses. Sudden equipment failures due to an unpredictable increase in vibration cause unforeseen downtime, which leads to significant financial losses (lost profits, fines, additional costs for emergency repairs) and disruption of production schedules.
- Higher maintenance and repair costs. Emergency repairs are more expensive due to urgent purchases and staff involvement. Ignoring early signs of degradation can lead to cascading failures, and a lack of understanding of the remaining service life leads to inefficient component replacements.

- Reduced safety and increased risks. An uncontrolled increase in vibration is an indicator of serious malfunctions that, without forecasting, can lead to accidents, equipment destruction, personnel injuries, or environmental disasters, as operators do not have the ability to take preventive measures.
- Inefficient spare parts inventory management. Without an accurate prediction of the time of equipment failure, companies either freeze capital in large stocks of spare parts or face shortages during unforeseen breakdowns. A prediction algorithm would allow for optimizing inventory levels.
- Limited opportunities for optimization and improvement. Systems that only signal limits do not provide a deep understanding of wear dynamics, making it difficult to analyze root causes and implement systemic improvements. Forecasting would allow detecting degradation patterns and moving toward predictive and prescriptive maintenance.

3. Goal

The goal is to research the types of algorithms that can be used to predict the time it takes to reach a specified limit, determine their advantages and disadvantages, and implement the algorithm that is most suitable one.

4. Algorithms for predicting future values

4.1. Overview

Predicting future values, or forecasting, is a fundamental task in data analysis that allows making informed decisions in various fields, including finance, manufacturing, healthcare, logistics, and meteorology. This task involves using historical data to identify patterns and trends that can be extrapolated into the future. Effective forecasting is key to strategic planning, resource optimization, and risk minimization.

There is a wide range of forecasting algorithms, classified by data type (time series, cross-sectional data), approach (statistical, machine learning, deep learning), and model complexity. Each approach has its unique advantages and disadvantages, which makes choosing the optimal algorithm critically important. Below is a detailed overview of the main categories and specific algorithms, their operating principles, advantages and disadvantages.

4.1.1. Statistical methods

These methods are the basis for time series forecasting and are well-suited for data with clear trends, seasonality, or cyclicality [4]. They are based on mathematical models and statistical properties of the data, often requiring certain assumptions about data distribution or stationarity. Statistical algorithms include:

- Moving average
  - Advantages: simplicity, clarity, effective noise smoothing, low computational requirements.
  - Disadvantages: significant lag from the trend, does not account for seasonality, poor reaction to sharp changes.
- Exponential smoothing
  - Advantages: adapts well to changes, effectively accounts for trend and seasonality, easier interpretation.
  - Disadvantages: requires parameter tuning (alpha, beta, gamma), sensitive to outliers, limited flexibility for complex patterns.
- Autoregressive integrated moving average (ARIMA) model
  - Advantages: powerful and flexible model, capable of modeling complex time dependencies (trends, seasonality), well-suited for non-stationary series.
  - Disadvantages: requires stationarity, difficulty in selecting parameters, insufficient effectiveness for nonlinear data.
- Prophet
  - Advantages: ease of use, automatic data processing, flexibility for seasonality and holidays, scalability.
  - Disadvantages: can be less accurate for short series or series without clear seasonal/trend components, less transparency.

4.1.2. Machine learning methods

These algorithms process more complex nonlinear relationships and effectively use additional (exogenous) features, turning time series forecasting into a supervised learning task [5]. Machine learning-based algorithms include:

- Linear regression
  - Advantages: simplicity, speed, easy interpretation, a good basic model.
  - Disadvantages: assumes a linear relationship, sensitive to outliers, limited for complex time series without prior "feature engineering".
- Decision trees and tree ensembles
  - Advantages: ability to model nonlinear dependencies, effective use of additional features, high accuracy, processing of large data, robustness to outliers (some implementations).
  - Disadvantages: need for "feature engineering," less interpretable, risk of overfitting, often do not have a built-in concept of time.
- Support vector machines for regression
  - Advantages: works well with high-dimensional data, effective for small samples, flexibility in modeling nonlinear relationships.
  - Disadvantages: sensitive to kernel and parameter selection, can be slow on large data volumes, less interpretable.

4.1.3. Deep learning methods

Deep learning uses neural networks with many layers to detect complex abstractions and hierarchical patterns. These methods are particularly effective for complex time series with long-term dependencies and large volumes of data [6-9]. Deep learning-based algorithms include:

- Recurrent neural networks
  - Advantages: modeling of long-term dependencies, high efficiency for sequential data, automatic feature extraction.
  - Disadvantages: require large volumes of data for training, computationally expensive, less interpretable, long training time.
- Convolutional neural networks for time series
  - Advantages: good at detecting hierarchical patterns, parallelization of computations, fewer parameters.
  - Disadvantages: not always ideal for long-term dependencies without additional layers, need for data transformation.
- Transformers
  - Advantages: powerful, well-parallelizable, high accuracy.
  - Disadvantages: computationally expensive, require significant data volumes, less interpretable.

4.1.4. Combined and hybrid/ensemble methods

Often, the best results are achieved by combining different methods, which allows using the strengths of each approach and compensating for their weaknesses. Combined algorithms include:

- Hybrid models
  - Advantages: combine the strengths of different methods, increase accuracy and reliability, flexibility in creating individual solutions.
  - Disadvantages: Increased complexity of development and support, greater need for expertise, higher computational costs.
- Ensemble methods
  - Advantages: reduced risk of overfitting, increased reliability and accuracy, resistance to errors.
  - Disadvantages: can be computationally expensive, less interpretable, difficulty in tuning.

For further experimental research, the statistical method of Moving Average (MA) will be used. MA was chosen over more complex models like ARIMA or LSTM because it is the easiest to implement. Although MA has lower accuracy than more complex models, it is selected for a number of critical advantages. First, MA is extremely simple to understand and interpret, which makes it ideal for visual analysis and creating transparent, easily explainable basic forecasts. Second, it requires significantly fewer computational resources, which allows it to be used on platforms with limited power. This is an important advantage over LSTM, which requires large amounts of data and high-performance computing (often using a GPU) for effective training. In addition, some studies have found that on certain large datasets, ARIMA can be more accurate than LSTM, which calls into question the absolute superiority of more complex models [10]. Therefore, if the goal is not to achieve maximum accuracy but to get a quick, reliable, and understandable idea of the main trend, MA is the most practical and pragmatic choice and will serve as the foundation for the further development of software for predicting vibration values. VES004 software will be used for vibration monitoring [11]. VES004, in combination with VSE series accelerometers and diagnostic electronics, can be used for vibration monitoring and maintenance of machines and installations [12]. One of its disadvantages is the lack of a feature for predicting the vibration value over a certain period and the inability to send the monitoring history to an email. Only the option to export the recorded data to a local computer is available, which will be sufficient for this experiment.

4.2. Algorithm implementation

VES004 allows you to record monitoring data and export it in three formats: XML, CSV, and XLSX. CSV format was chosen for further work with the input data because the data storage format itself takes up the least amount of space compared to the other two. The output data generated by VES004 is shown in Table.

The algorithm uses the following fields:

- Name: the object's name, since there can be several objects in one file if they were selected for monitoring in VES004.
- Timestamp: time in milliseconds from 00:00:00 January 1, 1970.
- Warning: the "warning" value set in VES004, which serves as an intermediate between a normal state and a state of damage.
- Damage: the "damage" value set in VES004.
- Value: the real-time vibration value of the object.

For the experiment, the Moving Average method was chosen. For the algorithm to work correctly, the recorded and saved monitoring data must first be exported from VES004 into a folder created by the user. All values of all file fields have a "string" type by default, but the library for working with CSV files allows you to choose the type into which the field value should be converted. In the algorithm, the Warning, Damage, and Value values have a real type, and Timestamp has an integer type. The program begins to process each file in chronological order: from oldest to newest. The algorithm processes the file data line by line and saves the data of each object into a separate structure for both its further work and for reference information. The "Name," "Value," and "Timestamp" fields are vital for correct operation, as there can be several objects selected for monitoring in one file, and it is important to store values related to object "A" separately from object "B". As a result of file processing, we get the structure shown in Fig 1.

Table. Fields of the exported file

DeviceID	Device Name	VESID	Type	Object Type	Name	Timestamp	Flags	Unit	Warning	Damage	Value	Speed	Reference Value

The presented structure is a container where the object's name serves as the keys. They have unique values; each key stores an array of data, whose elements are the real vibration value at a certain point in time. At the end of processing each file, the average value of the object and the duration of the recording are calculated, and a similar structure (Fig. 1) is filled. It is initialized one level higher before iterating through all files. The structure stores data about all average values from all processed files (Fig. 2).

After all files have been processed, the average rate of change of the average value between files for one object is calculated (Fig. 3). Then, the difference between the last average value for the selected object and the set threshold value (namely "warning" and "damage") is calculated. The found difference is divided by the rate of change of the average value, and thus the predicted time in minutes to reach the corresponding limit values is found. If the average vibration value in all files has already exceeded the limits, the algorithm immediately signals this. Fig. 4 shows the flowchart of the entire algorithm.

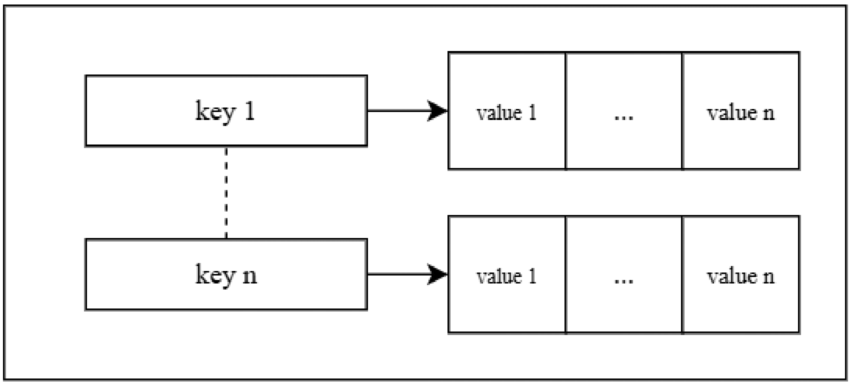

Fig. 1. Structure for storing objects and their values while processing one file.

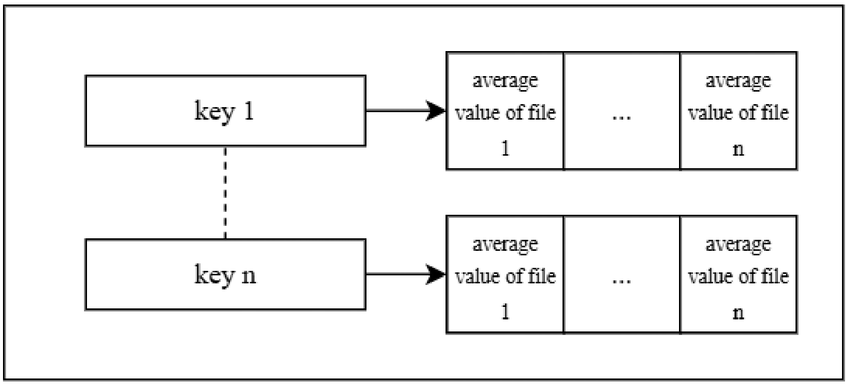

Fig. 2. Structure for storing objects and their average values in each file.

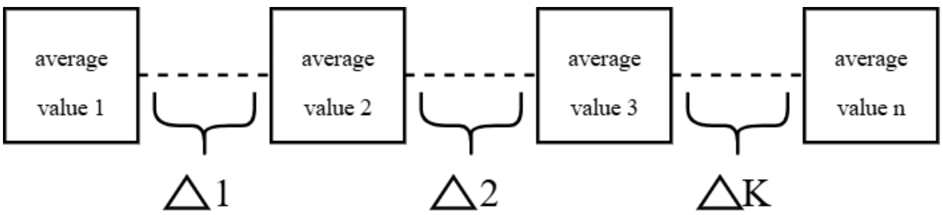

Fig. 3. Determining the change in the average value of one object between files.

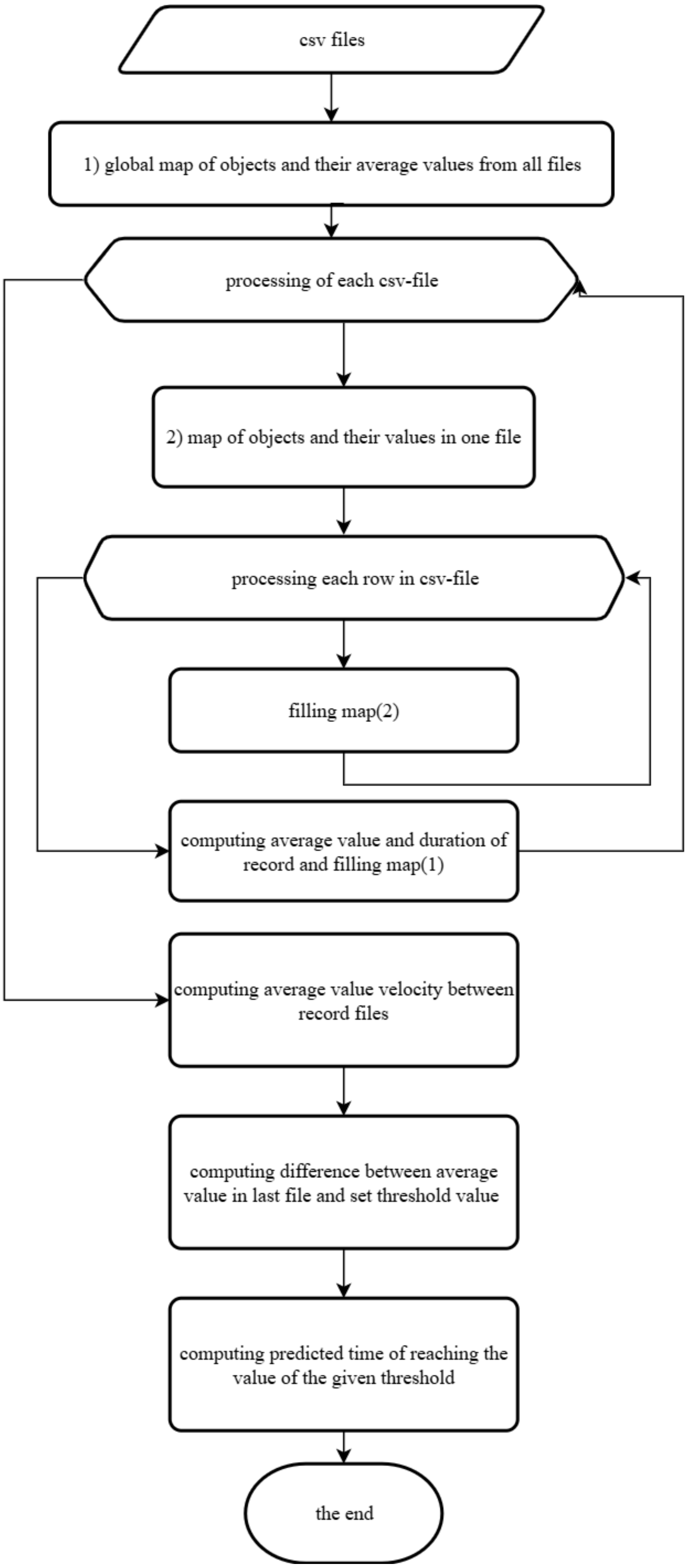

Fig. 4. Flowchart of the algorithm.

4.3. Experimental verification of the algorithm

An experiment was conducted to test the algorithm's performance in real conditions. The essence of the experiment was the degradation of a bearing. Impurities in the form of sand were added to the rolling elements of a new bearing. To speed up the degradation, the seal was dismantled, sand was added, the seal was reinstalled, and thus the bearing's operation in an aggressive environment was simulated To collect the necessary data, a series of recordings of the bearing's operation were made as follows:

1. Adding impurities.
2. Recording for 30 minutes of active bearing operation.
3. Saving the recording.
4. Exporting the recording to a specially created folder.

4.4. Experimental results

After a series of recordings, the set damage limit of the bearing was reached. Fig. 5-6 show the state of the bearing before and after the experiment.

Fig. 7 shows the change in the average vibration value throughout the series of recordings. It can be concluded that the vibration value increases, and after the 15th recording, the average value is constantly above the set limit, namely 0.1 mg.

Fig. 8 shows the relationship between the number of recordings made and the predicted number of minutes to the set limit. A sharp, single increase in the value of minutes is observed. Such an increase signals that at a certain point in the recording, the average vibration value of the object was within the normal range. During the experiment, there were a few cases where, after the introduction of impurities, instead of an increase in the vibration level, an increase was first observed, followed by a decrease in vibration. This happened because the impurity material, namely sand, is fine-grained, and there is a certain space between the bearing seal and the rolling elements designed for lubrication. Accordingly, the sand settled on the walls of the seal during operation, not on the path of the rolling elements. However, with an increase in the proportion of impurities inside the bearing, the vibration value confidently increases, since there is less and less space inside, and the sand will inevitably get on the path of the rolling elements. It is worth noting that if only one recording was made, the algorithm will not show the predicted time. This is because at least two recordings are needed to calculate the change in the average value between them and to make a prediction based on the result. In general, the more data, the more accurate the prediction.

Fig. 5. State before the experiment.

Fig. 6. State after the experiment.

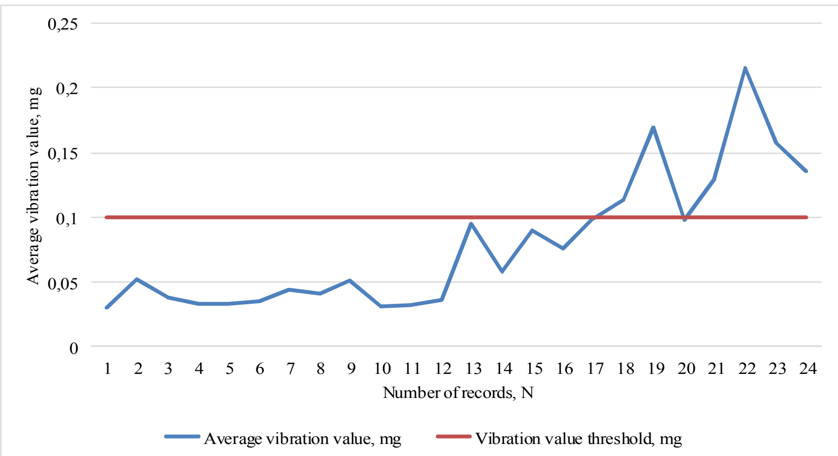

Fig. 7. Change in the average vibration value throughout the series of recordings.

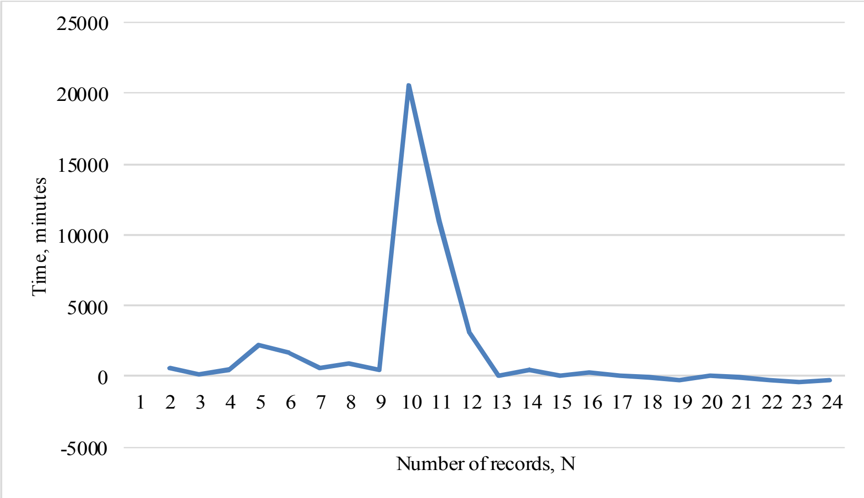

Fig. 8. Predicted number of minutes corresponding to the number of recordings made.

4.5. Algorithm disadvantages

The main disadvantage of this algorithm, as well as the chosen forecasting method, is the smoothing of all values. This means that various value spikes are not taken into account, even if they are generated with a certain frequency. Accordingly, these values may be a sign of incorrect operation of the object, and we will simply not notice it, since they will be smoothed out in the average values of the recordings. One option to account for this disadvantage is to save sharp value deviations in an additional structure for further analysis by the operator. Another disadvantage is the need to manually export the data to the appropriate format.

In the future, to improve the algorithm, the use of neural networks and their integration can be considered. Our specific implementation [13] requires entering the final folder where the exported files are stored each time the program is run. A better option would be to create a service (in Windows operating systems) that would periodically check in the background whether there have been changes in the folder, namely, whether files have been added or deleted. This change would serve as a signal to re-process the files and update the predicted time value.

For wider application of our program, it would be appropriate to support other formats (XLSX, XML, etc.) in the future. It would also be useful to add user configuration options, such as selecting fields that are important for analysis or for displaying additional information. It is also worth exploring the possibility of working directly with sensors, without using specialized programs, if it is possible to connect smart sensors directly to a PC and process data using a defined protocol. This would allow for a wide range of applications for our program, not just in conjunction with VES004.

5. Conclusions

Thus, the developed algorithm, even in its basic form with a moving average, is a strategic step toward implementing full-fledged predictive maintenance in Industry 4.0. Although the algorithm was developed and studied for use in vibration diagnostics, its scope can be significantly expanded after the proposed improvements. Since the moving average method is based on historical data, it can be used in almost any industry where the studied value depends on time and its change is important.

This makes it ideal for initial implementation in small and medium-sized enterprises, research laboratories, or on platforms with limited power, where quickly obtaining clear forecasts is a priority.

Gratitude

We thank the general representative of the “IFM Electronic” company in Eastern Europe, Kamila Mašková.

Conflict of Interest

The authors declare that there are no financial or other potential conflicts related to this work.

References

[1] Russell D. Meller, David S. Kim, “The impact of preventive maintenance on system cost and buffer size”, European Journal of Operational Research, Volume 95, Issue 3, 1996. [Online]. Available: https://doi.org/10.1016/0377-2217(95)00313-4.

[2] Syeda Sitara Wishal Fatima, Afshin Rahimi, “A Review of Time-Series Forecasting Algorithms for Industrial Manufacturing Systems”, Machines, Volume 12, Issue 6, 2024. [Online]. Available: https://doi.org/10.3390/machines12060380.

[3] Nataliia Kashpruk, Cezary Piskor-Ignatowicz, Jerzy Baranowski, “Time Series Prediction in Industry 4.0: A Comprehensive Review and Prospects for Future Advancements”, Applied Sciences, Volume 13, Issue 22, 2023. [Online]. Available: https://doi.org/10.3390/app132212374.

[4] Liu S., "Practical Time-Series Forecasting: Understanding Exponential Smoothing and ARIMA Models.", 2025, [Online]. Available: https://medium.com/kingfisher-technology/practical-time-series-forecasting-understanding-exponential-smoothing-and-arima-models-becac75403b0.

[5] Issar Arab, Rodrigo Benitez, “Benchmarking Time Series Forecasting Models: From Statistical Techniques to Foundation Models in Real-World Applications”, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2502.03395.

[6] Rahul Arulkumaran, Dignesh Kumar Khatri, Viharika Bhimanapati, Lagan Goel, Om Goel, “Predictive Analytics in Industrial Processes Using LSTM Networks.", Universal Research Reports, Volume 10, Issue 4, 2023. [Online]. Available: https://doi.org/10.36676/urr.v10.i4.1361.

[7] Balduíno César Mateus, Mateus Mendes, José Torres Farinha, António Marques Cardoso, "Anticipating Future Behavior of an Industrial Press Using LSTM Networks.", Applied Sciences, Volume 11, Issue 13, 2021. [Online]. Available: https://doi.org/10.3390/app11136101.

[8] F. Fazel Mojtahedi, N. Yousefpour, S. H. Chow, M. Cassidy, “Deep Learning for Time Series Forecasting: Review and Applications in Geotechnics and Geosciences.", Archives of Computational Methods in Engineering, Volume 32, 2025. [Online]. Available: https://doi.org/10.1007/s11831-025-10244-5.

[9] José Manuel Oliveira, Patrícia Ramos, “Evaluating the Effectiveness of Time Series Transformers for Demand Forecasting in Retail”, Mathematics, Volume 12, Issue 17, 2024. [Online]. Available: https://doi.org/10.3390/math12172728.

[10] Deddy Gunawan Taslim, I Made Murwantara, “Comparative analysis of ARIMA and LSTM for predicting fluctuating time series data”, Bulletin of Electrical Engineering and Informatics, Volume 13, Issue 3, 2024. [Online]. Available: https://doi.org/10.11591/eei.v13i3.6034

[11] VES004 - Parameter setting software for vibration diagnostic electronics – ifm. [Online]. Available: https://www.ifm.com/gb/en/product/VES004.

[12] Oleksandr Ryshkovskyi, Markiian Lukashiv, “Instrumental platforms for vibration analysis in predictive maintenance”, ISTCMTM, Volume 85, Issue 2, 2024. [Online]. Available: https://doi.org/10.23939/istcmtm2024.02.021.

[13] Algorithm for predicting the time for the object to reach the specified limit of the vibration value. [Online]. Available: https://github.com/Marki2698/predictionAlgo