Anomaly Detection in Digital Twin Model

Yutong Wang
The State Key Laboratory for Management and Control of Complex Systems
Institute of Automation, Chinese Academy of Sciences
Beijing, China
wangyutong2016@ia.ac.cn

Yansong Cao
Macau Institute of Systems Engineering
Macau University of Science and Technology
Macau, China
yscao@maverickvc.com

Fei-Yue Wang*
The State Key Laboratory for Management and Control of Complex Systems
Institute of Automation, Chinese Academy of Sciences
Beijing, China
feiyue.wang@ia.ac.cn

2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI) | 978-1-6654-3337-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/DTPI52967.2021.9540116

***Abstract*—As the simulation model of a physical system, digital twin has been widely used in many complicated control systems. Providing an effective way to perform simulation, digital twin makes the evaluation, prediction and optimization process cheaper and easier than on physical systems. For smart manufacturing, digital twin achieves high productivity with less operation and maintenance cost. However, the advantages of digital twin in anomaly detection during manufacturing are always neglected. Taking a two-speed transmission system as an example, we generate three kinds of faults on this digital twin model. After extracting features from the system output, we train an anomaly detection model on these features to classify each type of fault. By this digital twin model, we can find the fault in the whole system at the very beginning, and reduce the time and cost of debugging and diagnosing.**

***Index Terms*—digital twin, anomaly detection, transmission system**

I. Introduction

Industry is an important sector of the nation economy, and it determines the national modernization level. Industry is the production of highly mechanized and automatized goods, and it evolves four stages throughout history. These four stages are named four industrial revolutions, as shown in Fig. 1. The first industrial revolution is a transition from human labor to mechanization through the exploit of steam power. The second industrial revolution is an emergence of electricity and mass production. The third revolution is a rise of communication technologies. The fourth revolution, also known as "Industry 4.0" [1], is a use of smart, inter-connected cyber-physical systems, such as internet of things (IoT), artificial intelligence (AI) and big data. Each revolution leads to a paradigm shift, accompanied with a giant heap in productivity.

Industry 4.0 was first proposed by German government in 2011, it was a strategic program to promote smart manufacturing. The main purpose of Industry 4.0 is improving manufacturing automation, flexibility, productivity and efficiency while reducing operation and maintenance cost. To achieve this goal, researchers believe that large-scale intelligent networking of machines and processes is essential. However, only the communication is far from enough, smart manufacturing needs to manage all these complicated machines and processes in a unified way.

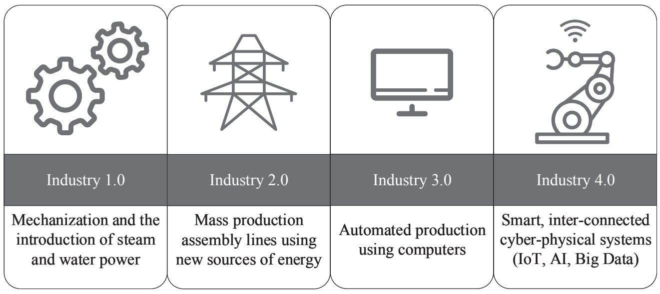

Fig. 1. Illustration of four industrial revolutions.

For complex control problems, if we accurately know the model, we could use digital twin [2]. If we cannot acquire the accurate model, we use parallel control [3]–[6]. Serving as an exact counterpart of physical system, digital twin is widely used in manufacturing since the proposal of Industry 4.0. Digital twin was first proposed by NASA in 2010 for improving physical model simulation of spacecraft. The principle of digital twin is using sensors gathering data from real-world system to establish an accurate virtual model, conducting simulation and machine learning on this model to facilitate decision-making in real-world system. Digital twin has three major advantages:

1) real-time monitoring. For human labor, real-time monitoring is tedious and error prone. But for digital twin, we can use preset rules to monitor the whole manufacturing process.
2) fast simulation and troubleshooting. For real-world experiments, the cost for testing and troubleshooting is high. For digital twin, we do not need to wait until the whole process is conducted to verify the manufacturing process and find out faults. With the help of powerful computing resources, we can shorten the experiment cycle and find defects at the very beginning, avoiding machine damage and manufacturing stop.
3) bottleneck identifying. For real-world manufacturing,

* Corresponding author

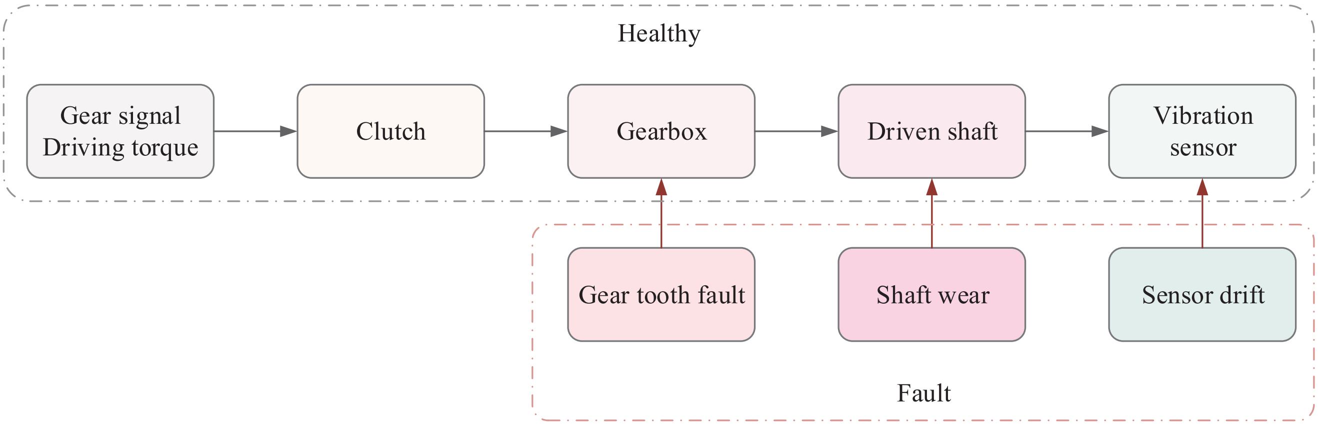

Fig. 2. Block diagram of two-speed transmission system.

TABLE I
CLUTCH STATE FOR THE TWO-SPEED TRANSMISSION SYSTEM

Transmission State	Brake Clutch State	Low Gear Clutch State	High Gear Clutch State
Neutral/Braked	Disengaged/Locked	Disengaged	Disengaged
Low Gear	Disengaged	Locked	Disengaged
High Gear	Disengaged	Disengaged	Locked

searching the bottleneck when taking time, money and other important factors into consideration is difficult. For digital twin, we could easily find out the bottleneck by conducting simulation on the model.

In this paper, we use an example to show the effectiveness of digital twin in smart manufacturing about predictive maintenance. During production, the fault occurrence will severely affect production and damage the machine. Predicting the fault and taking actions immediately will save massive amounts of money and assure the production.

But this situation rarely happens, so it is difficult to model and predict the fault. So we model a two-speed transmission system, and define three types of fault on this model. We generate fault data based on each kind of fault. Then we train an anomaly detection model to classify each type of fault, and predict the fault condition. It turns out that, without real-life fault experiments, we can get good anomaly detection results with the digital twin model.

II. Anomaly Detection

A. Transmission System

A transmission system [7] is to transmit speed and torque from the engine to another device. It is most commonly used in automobiles, but it is also used in robots such as mechanical arms. To verify the usefulness of digital twin, we construct a two-speed transmission system [8]. The system includes a driving shaft, clutch, gearbox and driven shaft, as shown in Fig. 2. The input is a driving torque, then the system transfers this torque and the associated angular velocity from the driving shaft to the driven shaft. This system also accepts a gear signal. For two-speed transmission system, there are two planetary gears and two clutches. According to the gear signal, we can set the clutch to two different states and generate different torque and angular velocity ratios.

As an important part of transmission, clutch schedule is shown in Tab. I. It converts the input signal to clutch and brake states. As a third clutch, the brake works on the driven shaft. The signal drives the clutches and brake, causing one to lock and others to disengage. The configuration avoids motion conflicts between gears and determines which gears power flows through. The difference between neutral and brake is that, neutral remains the brake clutch disengaged and the driven shaft is free to rotate. Without a driving torque, damping gradually ceases the driven shaft from rotation. While brake immediately locks the brake clutch and shuts down the driven shaft. Both the driving shaft and the driven shaft consist of a driveline axis restricted by damping and inertia torque. A net torque is transmitted along the driveline.

In our two-speed transmission system, one gear is a low gear, the other a high gear. The low and high refer to the ratio of output to input (follower to base) angular speed. A low gear is used for accelerating the machine from standstill by transmitting a large torque along the drivetrain from the engine. A high gear is used for slower acceleration or coasting once the machine is moving at a certain speed. The acceleration produced by this gear is less than the acceleration that is produced by the low gear.

The system outputs vibration and tacho signal. We use a casing module to model a mass spring damper system and measure its vibration. The casing translates the shaft angular shift to a linear shift. The tacho is measured from the rotation of the shaft.

B. Fault Modelling

We introduce three kinds of faults to the transmission system, gear tooth fault, sensor drift fault, and shaft wear fault [9], as shown in Fig. 2. For gear tooth fault, we inject a disturbance torque at a fixed position in the spin of the driving shaft. We use radians to measure the shaft position, and when the shaft position is within a small range around 0 a disturbance torque is added to the shaft. We use a variable to control the magnitude of the disturbance. For sensor drift fault, we introduce an offset variable in the vibration sensor model. For shaft wear fault, we model a variable system that changes the shaft damping.

By varying these variables, we can output vibration and tacho data for different fault types. Remaining the input gear signal and driving torque unchanged, we use average sampling and random sampling to decide these three input variables. After we run the simulation of all these input variable combination, we create fault labels for each of the input. We preset a threshold for each fault variable, the value below the threshold is regarded as no fault, larger than the threshold is regarded as fault.

C. Feature Extraction

Since the output vibration and tacho data are time series data of large volume, we have to extract the important features from the data and ignore the unimportant part. Then we can use these features to classify different fault types. We use 18 signal statistics which are usually used for feature extraction, including mean, median, root mean squared (RMS) [10], variance, peak, peak to peak. The mean is defined as

$$\mu = \frac{1}{N}\sum_{n=1}^{N} x_n \tag{1}$$

where x is a vector made up of N scalar observations. The root-mean-squared level is defined as

$$x_{\mathrm{RMS}} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} |x_n|^2} \tag{2}$$

The variance is defined as

$$\sigma^2 = \frac{\sum_{n=1}^{N} |x_n|^2 - \mu^2}{N-1} \tag{3}$$

We also use skewness, kurtosis, crest factor, median absolute deviation [11], range of cumulative sum, correlation dimension [12], [13], approximate entropy [14], [15], Lyapunov exponent [16], the peak frequency of the time synchronous average [17] of the vibration signal, the power of the time synchronous average envelope signal, the envelope spectrum [18] of vibration signal and the frequency with maximum spectral kurtosis [19].

D. Fault Classification

We have tried several classification models including k-nearest neighbors, support vector machine [20], [21], decision tree [22], [23] and random forest to learn these feature data and predict the fault type. Among these models, decision tree model achieves the best accuracy. Decision tree model learns simple decision rules inferred from the data and perform piecewise constant approximation.

III. Experiments

A. Experimental settings

We use Simscape Driveline toolbox of MATLAB Simulink to model and simulate the transmission system. The high gear has a ratio of 2:1, while the low gear has a ratio of 5:1. The input gear signal lock the low gear. We generate 208 data using average sampling and random sampling. And we randomly split the data into training set and testing test at a ratio of 4:1. We discard the first 10 seconds of the output data. Each kind of fault has two statuses, True and False. To combine different kinds of faults together, we use following equation to generate fault label

$$\mathrm{FaultLabel} = \mathrm{sF} + 2 \times \mathrm{sV} + 4 \times \mathrm{sT} \tag{4}$$

where sF denotes the status of sensor drift, sV denotes the status of shaft wear, and sT denotes the status of tooth fault.

B. Prediction Results

The vibration and tacho output is shown in Fig. 3 and Fig. 4. We use the classifier to classify the testing set and evaluate the performance of the fault predictions. The confusion matrix is shown in Fig. 5. For the total 41 test samples, our classifier correctly classifies 38 of them, and achieves an accuracy of 92.7%. In the future research, to improve the prediction accuracy, we consider generating more data, increasing feature counts used in the classifier and using deep learning as the classifier model.

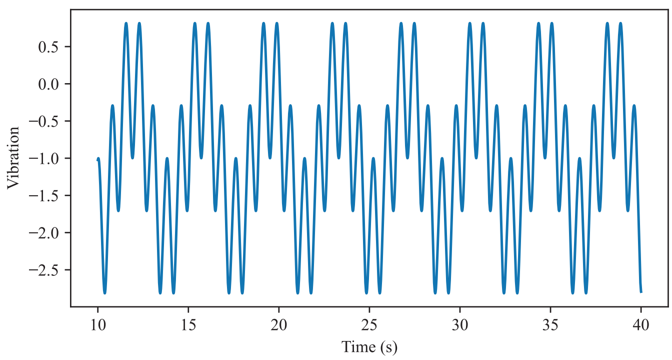

Fig. 3. The temporal vibration output.

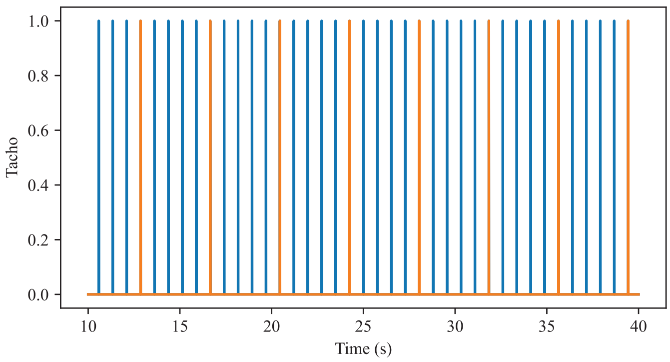

Fig. 4. The temporal tacho output.

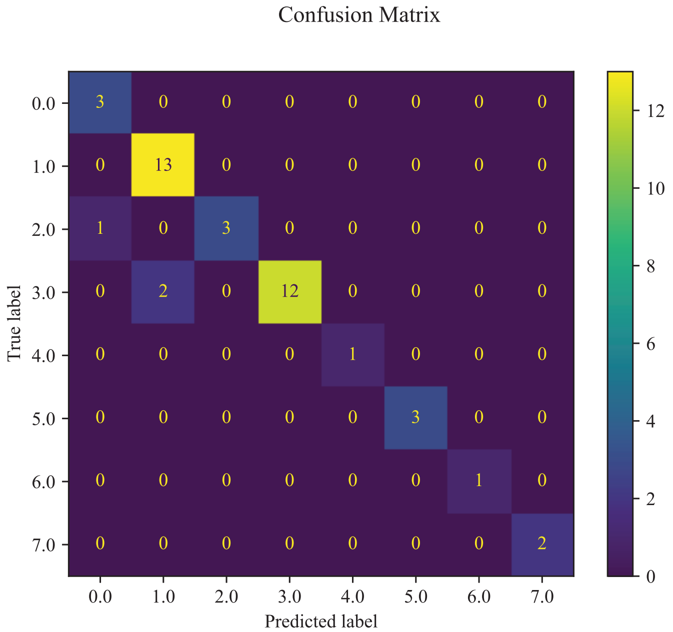

Fig. 5. The confusion matrix of fault classification.

IV. Conclusion

In this paper, we propose an effective method to perform anomaly detection on a digital model. We model a two-speed transmission system as the digital twin of physical two-speed transmission system. we generate three kinds of faults on this digital twin model. We run the simulation according to each fault variable combination and derive the raw output data. Since the time series data is too large to learn, we use feature extraction to extract the important features and reduce the feature dimensionality. After extracting features from the system output, we train an anomaly detection model on these selected features to classify each type of fault. By this digital twin model, even if there is a combination of different types of faults, we can predict the specific faults with a high accuracy.

References

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & Information Systems Engineering*, vol. 6, no. 4, pp. 239–242, 2014.

[2] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *The International Journal of Advanced Manufacturing Technology*, vol. 94, no. 9, pp. 3563–3576, 2018.

[3] Q. Wei, X. Wang, X. Zhong, and N. Wu, "Consensus control of leader-following multi-agent systems in directed topology with heterogeneous disturbances," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, p. 423–431, 2021.

[4] Q. Wei, H. Li, and F.-Y. Wang, "Parallel control for continuous-time linear systems: A case study," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 4, p. 919–926, 2020.

[5] Q. Wei, D. Liu, Y. Liu, and R. Song, "Optimal constrained self-learning battery sequential management in microgrid via adaptive dynamic programming," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 2, p. 168–176, 2017.

[6] Q. Wei, L. Wang, J. Lu, and F.-Y. Wang, "Discrete-time self-learning parallel control," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–13.

[7] B. Saltzberg, "Performance of an efficient parallel data transmission system," *IEEE Transactions on Communication Technology*, vol. 15, no. 6, pp. 805–811, 1967.

[8] https://www.mathworks.com/help/physmod/sdl/ug/two-speed-transmission.html.

[9] https://www.mathworks.com/help/predmaint/ug/Use-Simulink-to-Generate-Fault-Data.html.

[10] N. Paulter, D. R. Larson, and J. J. Blair, "The IEEE standard on transitions, pulses, and related waveforms, std-181," in *Proceedings of the IEEE Instrumentation and Measurement Techonology Conference*, vol. 1, 2003, pp. 110–112.

[11] L. Sachs, *Applied statistics: a handbook of techniques*. Springer Science & Business Media, 2012.

[12] W. Caesarendra, B. Kosasih, K. Tieu, and C. A. Moodie, "An application of nonlinear feature extraction-a case study for low speed slewing bearing condition monitoring and prognosis," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2013, pp. 1713–1718.

[13] J. Theiler, "Efficient algorithm for estimating the correlation dimension from a set of discrete points," *Physical review A*, vol. 36, no. 9, p. 4456, 1987.

[14] S. M. Pincus, "Approximate entropy as a measure of system complexity." *Proceedings of the National Academy of Sciences*, vol. 88, no. 6, pp. 2297–2301, 1991.

[15] U. R. Acharya, F. Molinari, S. V. Sree, S. Chattopadhyay, K.-H. Ng, and J. S. Suri, "Automated diagnosis of epileptic eeg using entropies," *Biomedical Signal Processing and Control*, vol. 7, no. 4, pp. 401–408, 2012.

[16] M. T. Rosenstein, J. J. Collins, and C. J. De Luca, "A practical method for calculating largest lyapunov exponents from small data sets," *Physica D: Nonlinear Phenomena*, vol. 65, no. 1-2, pp. 117–134, 1993.

[17] E. Bechhoefer and M. Kingsley, "A review of time synchronous average algorithms," in *Annual Conference of the PHM society*, vol. 1, no. 1, 2009.

[18] E. P. Carden and P. Fanning, "Vibration based condition monitoring: a review," *Structural Health Monitoring*, vol. 3, no. 4, pp. 355–377, 2004.

[19] J. Antoni, "Fast computation of the kurtogram for the detection of transient faults," *Mechanical Systems and Signal Processing*, vol. 21, no. 1, pp. 108–124, 2007.

[20] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.

[21] J. Platt *et al.*, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in Large Margin Classifiers*, vol. 10, no. 3, pp. 61–74, 1999.

[22] W.-Y. Loh, "Classification and regression trees," *Wiley interdisciplinary reviews: data mining and knowledge discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[23] J. Friedman, T. Hastie, R. Tibshirani *et al.*, *The elements of statistical learning*. Springer Series in Statistics New York, 2001, vol. 1, no. 10.