Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira,
Edoardo Patelli, Hélder S. Sousa, Rui A. Silva and Monica Santamaria
 Published by Research Publishing, Singapore.
doi: 10.3850/ESREL2026061419_esrel26-p26167-cd

Combining Surrogate Digital Twins with Extended Kalman Filters for Model-based Robot Failure Diagnosis

Victoire DENIZOT

HEC Paris, ISAE-SUPAERO, France, victoire.denizot@hec.edu

Zhiguo ZENG

Chair of Risk and Resilience of Complex Systems, Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay, France, zhiguo.zeng@centralesupelec.fr

Data-driven models like deep neural networks have become the state-of-the-art for fault diagnosis. A significant drawback of such models is that they lack explainability and are difficult to generalize. Model-based approaches like the multi-observer approach can address the drawbacks of data-driven models by capturing domain knowledge for fault diagnosis but they remain computationally expensive and time-consuming. In this study, we propose a surrogate digital twin-based system-level fault diagnosis algorithm to reduce the computational burdens of the model-based fault diagnosis methods while keeping their benefits of explainability and generalization capabilities. The developed approach incorporates a Long Short-Term Memory (LSTM) model as a surrogate model for a highly accurate digital twin of the system being diagnosed. While keeping the simulation accuracy, the computational complexity can be greatly reduced. Then, the developed algorithm relies on a Multi-Extended Kalman Filter (MEKF) for multi-class fault diagnosis, in which the state transition function is driven by the surrogate digital twin of the physical entity. Each potential failure mode is modelled in the surrogate digital twin, and an extended Kalman filter is used to estimate the system states under each failure model. The failure mode with the lowest root mean squared error compared to the observation data will be identified as the occurring failure. The proposed hybrid approach also allows leveraging system-level observations to determine the failure states of components. We apply this method to a robotic arm with an end-effector, whose position is determined by 4 rotating motors. Using only the motor control commands and end-effector position measurements, the introduced approach reliably identifies the operational status of the robot, specifying whether it functions properly or if one of its motors is stuck. The framework was trained using data generated by the digital twin and tested with data from the real robot. The experimental results demonstrate an accuracy of 94% compared to 61% obtained with an LSTM only, and a strong computational time reduction, with no observed discrepancies between training and tests.

Keywords: Model-based fault diagnosis, Multi Extended Kalman Filter, LSTM, Digital Twin, Robotic Arm

1. Introduction

Fault diagnosis seeks to identify the root cause of a failure based on the data collected from sensors (Isermann, 2006). It has become a crucial industrial challenge due to the increasing complexity and safety requirements of modern systems. Among existing solutions for fault diagnosis, data-driven approaches like deep neural networks (DNN) are at the state-of-the-art (Mushtaq et al., 2021). However, their performance relies on the availability of large amounts of labeled data, which are costly and difficult to obtain (Lei et al., 2020). To mitigate this issue, digital twins - virtual counterparts of machines and infrastructures - are used to generate failure data (Zhong et al, 2023). While this strategy alleviates data scarcity, data-driven methods still lack explainability and scalability since they rely on statistical correlation rather than physical understanding (Wang et al., 2025). Conversely, model-based approaches, such as Kalman filters and particle filters based on physical or mathematical models, also provide greater interpretability. Their implementation for real-time application remains difficult partly due to their sensitivity to noise and uncertainty. They also require accurate knowledge of system

parameters and often suffer from long computation times (Gao, 2015).

As a result, both techniques often struggle to adapt to real-world operating conditions. To bridge this gap, hybrid methods have emerged, combining the physical insight of model-based approaches with the flexibility of data-driven architectures (Liao and Köttig, 2014). In such frameworks, the computation of residuals, specifically the Root Mean Square Error (RMSE) between estimated model outputs and measured sensor data, serves as a high-fidelity feature for fault diagnosis algorithm, as evidenced by Jain et al. (2020).

In earlier work, Zeng (2024) and Chen et al. (2025) introduced a "digital failure twin" to reduce the amount of real failure data needed to train diagnostic models, notably using Long Short-Term Memory (LSTM) networks or transfer learning with a Domain-Adversarial Neural Network (DANN) architecture. However, these models showed limitations in both accuracy and adaptability when transferred from training on the digital twin to testing on the real robot. Building upon these ideas, our contribution is:

- A data-driven surrogate model for the digital failure twin.
- A system-level -based fault-diagnosis framework for model-based fault diagnosis.

We apply our developed methods to a real robot. The results show that the developed methods achieve better accuracy with lower computational costs as compared to traditional data-driven fault diagnosis methods.

2. A digital-twin-trained surrogate model for robot performance simulation

In this section, we present a data-driven surrogate model for a digital twin of a robotic arm. We present the original digital twin in Sect. 2.1. In Sect. 2.2, we present an LSTM model developed to serve as a surrogate of the original digital twin model.

2.1. Original Digital Twin Modeling

In this paper, we used previously created data to train our surrogate model and fault diagnosis model. This data was generated using the digital failure twin model of a robotic arm developed previously by Zeng (2024) using MATLAB Simulink, as shown in Fig. 1. The robotic arm and its digital twin feature six degrees of freedom including four rotating motors, and an end-effector designed for small-object manipulation.

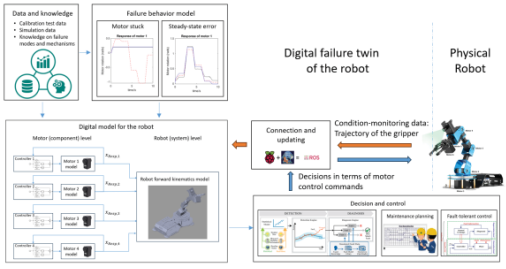

Figure 1. A presentation of the Digital Twin (Zeng, 2024)

We simulated various operational scenarios over ten-second intervals to build a comprehensive training dataset. These scenarios are classified as follows:

- Healthy State: All motors function following the commands (Label 0).
- Failure States: Specific motor "stuck" conditions where a motor ceases to respond to commands (Labels 1–4).

As introduced in previous work, 400 trajectories for each of the five labels were generated. In each trajectory, the robot performs five random movements within the time interval. For each trajectory, the commands on the four motors and the (x,y,z) coordinates of the end-effector are saved in a table for the training of the fault diagnosis model. The simulation is conducted with a step size of 0.01 seconds. Therefore, all the collected features are time series of 1000 points.

The dataset with label 0, i.e., from normal operation, is then used to train a data-driven surrogate digital twin in Sect. 2.2.

2.2. Surrogate modeling through LSTM

To accurately approximate the temporal evolution of motor states, a model capable of capturing non-linear dependencies in sequential data is required. Recurrent Neural Networks (RNNs) are a natural choice for such tasks; however, standard RNNs suffer from the vanishing gradient problem (Hochreiter 1998) , which limits their ability to learn long-term dependencies. To address this issue, the Long Short-Term Memory (LSTM) architecture, as first developed by Hochreiter & Schmidhuber (1997) was adopted.

An LSTM unit is composed of a cell state and three components called gates:

1. The Forget Gate: Determines which information from the previous cell state is discarded.
2. The Input Gate: Updates the cell state with new information from the current timestep.
3. The Output Gate: Decides which part of the cell state is used to produce the hidden state for the next timestep.

The cell remembers values over time intervals, and the gates regulate the flow of information into and out of the cell. Given that our inputs (sensor data and motor commands) are time series, the LSTM's ability to capture long-term dependencies makes it the optimal choice for predicting the robot's state evolution under different conditions.

To capture the internal states of the robotic arm, we designed an LSTM network tailored to approximate the motor positions. The input vector a_k consists of the previous coordinates of the end-effector $x_{k-1}, y_{k-1}, z_{k-1}$ and the current control commands for the four motors $u_{1,k}, u_{2,k}, u_{3,k}$ and $u_{4,k}$.

The LSTM network is structured to map these inputs to the predicted position at the next timestep. The objective of our surrogate model is to replace a computationally expensive part of our model-based fault diagnosis. The model is trained on all the simulation data from label 0.

3. Fault diagnosis based on multi-extended Kalman filter and surrogate digital twins

We design a system-level multi-observer approach for fault diagnosis in Sect. 3.1. In Sect. 3.2, we discuss in detail how to use an extended Kalman filter to model each system state (normal and different failures).

3.1. Multi-observer Framework – System level monitoring and RMSE-based fault diagnosis

To efficiently detect and diagnose a failure, while limiting the number of sensors and therefore data required, a system-level multi-observer approach was designed, as shown in Fig. 2. For any given time step i, the system-level input vector consists of the four motor commands and the end-effector's spatial response up to $i-1$. The model's output provides the estimated position of the motors and end-effector coordinates. At the same time, a sensor placed on the robot provides the actual coordinates of the end-effector.

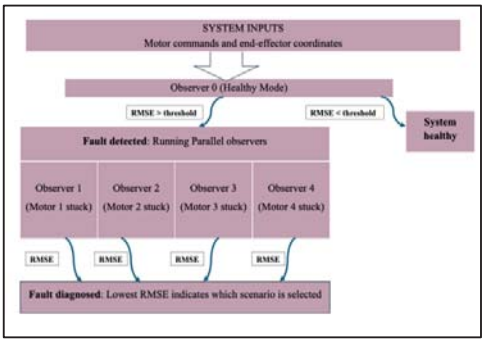

Figure 2. Structure of Multi-observer Fault Diagnosis Model

The fault diagnosis is conducted by passing the input data through a series of observers. Each observer models a given system status (including normal operation and different failure modes). For example, in Fig. 2, a first observer is designed to model the system's nominal behaviour. This observer detects whether the system is healthy or not, by estimating the motor states and the trajectory. A residual mean square error is computed at the end of this first observer:

$$RMSE = \sqrt{\sum_{i=1}^{N} \frac{(\hat{y}_i - y_i)^2}{N}}$$

where $N = 1000$ is the number of points in the input timeseries; $\hat{y}_i$ is the estimated system performance, i.e., the coordinates of the end-effector estimated by the observer at time i; y_i is the observed values of the system prediction at time i.

If the RMSE exceeds a certain threshold, four additional observers are run in parallel to diagnose the failure in action. These filters each simulate a stuck failure on one of the motors (one filter per motor), and estimate the trajectory followed by the end-effector accordingly. In the end, four RMSE are computed, and the lowest one, which signifies the closest estimation of trajectory, determines the scenario occurring.

3.2. Modeling system behaviour through extended Kalman filter

The core of fault diagnosis methods in Fig. 2 is an observer that can estimate system states under nominal and different failure behaviours. In this paper, we rely on extended Kalman filters for this task due to their ability to handle non-linear and dynamic system behaviors, as demonstrated by Zhang, Q. (2018).

Let us consider a system defined by the following non-linear state and measurement equations:

$$x_k = f(x_{k-1}, u_{k-1})$$
$$z_k = h(x_k) + v_k$$

where x_k, u_k, z_k are system state variables, control inputs and observations, respectively, and v_k is the measurement noise at time k. For the multi-observers framework in Fig. 2, we have one model for each state (nominal operation, failure mode 1-4). For the nominal states, the state equation $f(\cdot)$ is derived based on the surrogate model developed in Sect. 2. For failure modes 1-4, as they refer to motors 1-4 getting stuck, respectively, we can simulate the failure by fixing the input u_i to the last valid value before the stuck failure occurs.

Estimating the state variables in an Extended Kalman Filter (EKF) is a recursive process. The process is split into two distinct phases: prediction and correction. In the prediction phase, we use the nonlinear function f to predict where the system will be based on the previous state and any control inputs:

$$\hat{x}_{k|k-1} = f\left(\hat{x}_{k-1|k-1}, u_k\right)$$

and the covariance:

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q,$$

where P is the covariance matrix, F_k is the Jacobian matrix of the nonlinear state equation $f(\cdot)$ and Q is the process noise. Since we approximate the $f(\cdot)$ using a surrogate model, we can also approximate the Jacobian through numerical approximation, as defined by Steihaug and Wolfbrandt (1979) and used by Roumeliotis & Bekey (2000):

$$\partial f / \partial x_{\mathrm{i}} \approx \frac{f(x + \delta \cdot e_{\mathrm{i}}, u) - f(x, u)}{\delta}.$$

Here, we approximate the derivatives of both state transitions and observation functions using finite differences.

Then, when observation data z_k is available, we use it to correct the estimation of state variables and covariance:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k y_k$$

where y_k is the residual between prediction and observation:

$$y_k = z_k - h\left(\hat{x}_{k|k-1}\right),$$

and K_k is the gain of the filter and is decided by:

$$K_k = P_{k|k-1} H_k^T \left(H_k P_{k|k-1} H_k^T + R\right)^{-1}.$$

Here, H_k is the Jacobian matrix of the observation equation.

The covariance matrix P_k is updated by:

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}.$$

By iterating between the prediction and correction phases, we can use the observation data z_k to update the estimation of current state variables. The correction phase allows us to filter out the process and observation noise and obtain an accurate and robust estimation. After updating the estimation, we can use the updated model to predict again the system performance, and then calculate the RMSE error that will be used in the multi-observer framework for fault diagnosis (Sect. 3.1).

4. Results and discussions

We present results obtained when testing the multi-observer algorithm. In Sect. 4.1, we provide a visualisation of the multi-observer's output. The performance of the algorithm is evaluated in Sect. 4.2 before discussing the results in Sect. 4.3.

4.1 Results visualization

The multi-observer framework was trained on simulation data (cf. Sect 2.1.) and tested on 50 datasets generated by the real robot. the end of each run on a ten-second dataset, the multi-observer model prints out two graphs per observer, one for comparison between true vs. estimated trajectory and one for true vs. estimated motor positions.

Figures 3 and 4 below illustrate part of the performance of the proposed algorithm when validated against dataset 17 which simulates a Motor-1-stuck scenario. The first observer detects a fault because the residual between the estimated and nominal trajectory exceeds the defined threshold (Fig. 3). In a second part, four observers run in parallel to determine which motor is failing. The observer for Motor-1-stuck failure has the

lowest RMSE between estimation and true trajectory (Fig 4., the graphs printed from the other observers are not displayed). In the Figures 3 and 4 below, each state (from 1 to 3) corresponds to a cartesian coordinate *(x,y,z)*.

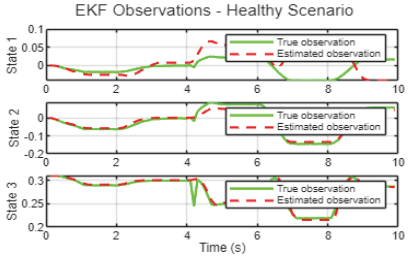

Figure 3: Comparison of True and Estimated Trajectory in Healthy Scenario

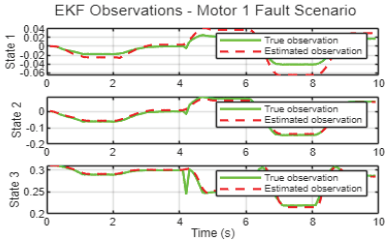

Figure 4: Comparison of True and Estimated Trajectory in Motor-1-Stuck Scenario

4.2 Accuracy and Performance

The confusion matrix, as shown in Fig. 5, was created after testing the fault diagnosis algorithm on 50 datasets generated by the real robot. The algorithm detects the presence of a fault with a theoretical 100% accuracy, within 10 ± 2 minutes. In case of failure, four observers running in parallel diagnose the motor affected by the stuck failure with a 92.5% accuracy.

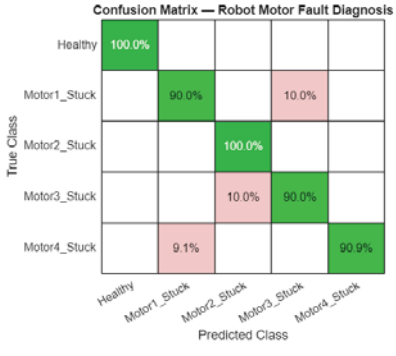

Figure 5: Confusion Matrix for Fault Detection and Diagnosis

4.3 Discussion

Overall, the results show that the proposed multi-observer scheme is able to both detect and diagnose faults in real time, with minimal delay. In addition, the algorithm shows robustness due to its training on multiple datasets, including scenarios with various initial conditions and noise levels.

The misdiagnoses observed mainly correspond to cases where the residual signals of two motors were similar. A fine-tuning of the EKF parameters can certainly increase the precision of classification.

5. Conclusions

This paper proposes a model-based fault diagnosis method based on a multi-observer framework and a data-driven surrogate digital twin. In the multi-observer framework, each state (normal functioning and different failure states) is modeled by a state-space model. In the state space model, the state equation is modeled by a data-driven surrogate digital twin, which keeps the accuracy of the original digital twins but significantly reduces the computational burden. An extended Kalman filter is used to estimate the system state variable and calculate the residuals between the model prediction and observations. Then, the failure diagnosis is conducted by choosing the failure model with the minimal residuals.

The developed methods were applied on a real robot with four failure modes. The results showed that with the help of the surrogate digital twins, the developed models can accurately diagnose different failure modes, while significantly reducing the computational burdens. However, the developed models still face a few challenges. Among them, the current model assumes that the effect of the failure is deterministic. However, in practice, there are a large number of failures that may be uncertain. For example, when the motors have steady-state errors, the exact value of the error is not fixed but a random variable. How to adapt the multi-observer framework to account for failure modes with uncertainty is a challenge that needs to be addressed in the future.

Acknowledgement

This work was conducted during Miss Denizot's internship at CentraleSupélec, with the financial support from the French Research Council under contract number ANR-22-CE10-0004.

References

Chen, Z. et al. (2025). A domain adaptation neural network for digital twin-supported fault diagnosis. *2025 International Conference on Control, Automation and Diagnosis (ICCAD)*.

Gao, Z. (2015). A Survey of Fault Diagnosis and Fault-Tolerant Techniques - Part I: Fault Diagnosis With Model-Based and Signal-Based Approaches. *IEEE Transactions on Industrial Electronics*.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems.*

Isermann, R. (2006). Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance. *Springer*.

Jain, P. et al. (2020), "A Digital Twin Approach for Fault Diagnosis in Distributed Photovoltaic Systems," in IEEE Transactions on Power Electronics, vol. 35, no. 1, pp. 940-956, Jan. 2020

Lei, Y., et al. (2020). Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mechanical Systems and Signal Processing, Volume 138*.

Liao, L., & Köttig, F. (2014). "A Hybrid Framework for Remaining Useful Life Prediction." *Reliability Engineering & System Safety*.

Mushtaq, Shiza, M. M. Manjurul Islam, and Muhammad Sohaib. (2021). "Deep Learning Aided Data-Driven Fault Diagnosis of Rotatory Machine: A Comprehensive Review" *Energies 14, no. 16: 5150.*

Roumeliotis, S. I., & Bekey, G. A. (2000). Numerical Jacobians in extended Kalman filter-based mobile robot localization. *IEEE Transactions on Robotics and Automation*.

Steihaug, T., & Wolfbrandt, A. (1979). An attempt to avoid exact Jacobian and nonlinear equations in the numerical solution of stiff differential equations. *Mathematics of Computation*, *33(146), 521–534.*

Wang, L., et al. (2025). Correlation does not equal causation: The imperative of causal inference.

Zeng, Z. (2024). Use digital twins to support fault diagnosis from system-level condition-monitoring data. *In Proceedings of the 22nd International Multi-Conference on Systems, Signals & Devices (SSD)*

Zhang, Q. (2018). Adaptive Kalman filter for actuator fault diagnosis. *Automatica*, 93, 312–332.

Zhong, D., Xia, Z., Zhu, Y., & Duan, J. (2023). Overview of predictive maintenance based on digital twin technology. *Heliyon*, 9, e14534.