Fault Detection and Analysis via Latent Space Differences Between the Plant and the Model Representing Normal Operation

Enrique Luna Villagomeza, Hamidreza Mahyarb, Vladimir Mahaleca\*

aDept. of Chemical Engineering, bWBooth School of Eng. Practice and Technology

McMaster University, Hamilton, ON L8S 4LS, Canada

\*mahalec@mcmaster.ca

Abstract

Abnormal plant operations (faults) occur when equipment or an instrument no longer functions well. Detection of such faults is made more difficult by process control applications, which attempt to ameliorate the impact of faults by keeping the production on target. Closed loop controls cause the plant to operate in a narrow region, which in turn limits the validity of data-driven fault detection methods to such a limited scope. This work introduces a fault detection architecture that employs data generated from steady-state simulation models to build a latent space model (e.g., Principal Component Analysis, PCA) of the normal operation, thereby overcoming limitations inherent in the plant's historical data. In real-time, variables from the model and the plant are processed by identical copies of PCA. The pattern of the differences indicates the fault occurrence and gives insight into possible causes.

**Keywords**: fault detection, abnormal operation, differences between normal and abnormal operation, latent space differences

* 1. Introduction

Fault detection (FD) based on plant data usually proceeds by identifying a model representing normal operation from plant data. This model could be a first principles model or a data-driven model that is then deployed in real-time to detect the occurrence of faults. The methodology of parity equations (Iserman, 2005) compares the model variables with their corresponding plant measurements to identify the occurrence of faults. Conversely, a data-driven model, usually a reduced dimensionality space model ( e.g., PCA or kernel PCA) measures the deviation from the normal operating region using statistical techniques to detect the occurrence of abnormal operation, i.e., faults (Yoon et al., 2001). The recurrent neural network data-driven model, introduced by Sun et al., (2020), characterizes dynamic behavior during normal operation, offering the possibility to detect and identify the faults based on comparing the measured process variables and the likely values of these variables in a normal operation.

Continuously operating plants operate in relatively narrow regions due to closed-loop controls. When disturbances enter the process, the resulting deviations from desired operating process outputs are rapidly corrected by the control actions. Data representing such plant operations is confined to a relatively narrow region. Consequently, fault detection and identification methods based on data-driven models built from the narrow normal operating region data cannot discern whether a fault (e.g., catalyst poisoning or heat transfer rate decrease due to fouling) forces the control system to move the plant outside the normal operating window in multimodal operations.

Over the last four decades, many first principles process models have been built for design and plant improvement applications. These models represent the behavior of the real plants reasonably well, but they are not perfect. Model parameters (e.g., heat transfer coefficients or reaction rates) are estimated from correlations or first principles equations, resulting in process models that may have considerable error (in the order of 5%) in heat transfer coefficients, reaction rates, or a similar relative error in predicting the concentration of product impurities. Nevertheless, these models adequately represent process behavior over various operating conditions, even if nonlinearity exists.

* 1. Fault Detection via latent space differences

This work relies on the ability of the process models to generate data over a wide operating region (wider than the data available from normal plant operation). Even if such data are generated from the models with incorrect model parameters, these data can be used to train unsupervised learning models (e.g., PCA or Autoencoder) representing (somewhat inaccurately) the normal operating conditions, NOCs. In the proposed FD approach, a trained unsupervised learning model, e.g., PCA, is deployed in real-time to have data from the plant as its inputs, while another instance of the same PCA model has the variables from the simulation model as its inputs. The simulation model takes feed, disturbances, and operating targets as inputs, enabling it to determine the values of the manipulated (control loop) variables. Differences between the latent space variables of these two instances of the latent space model are zero in normal operation if the model matches the plant perfectly. When a fault occurs, these differences are non-zero. Figure 1 depicts this fault detection architecture.

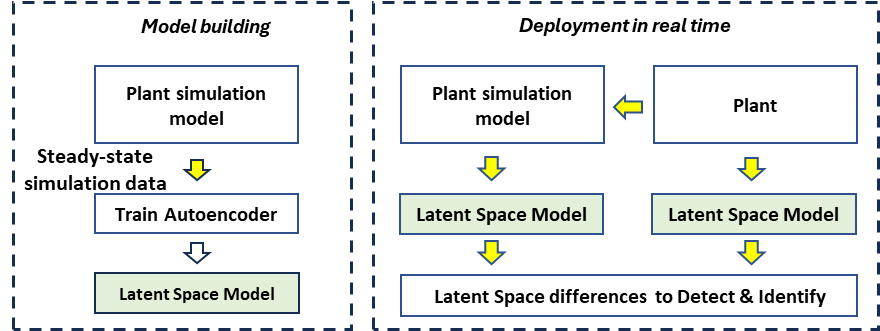


Figure 1. Fault detection and analysis via latent space differences.

This architecture has the potential to enable PCA in multimodal operations as an alternative to cluster-based techniques. For example, in the event of a setpoint change, this can be incorporated into the plant reference model, thereby ensuring that the latent space spanned by the PCA adjusts accordingly to maintain the differences close to zero. Notice that if plant/model mismatch is present, the differences in normal operation have a bias (i.e., they are not zero). Therefore, the bias needs to be corrected before proceeding with the fault detection task.

* 1. Experiments in fault detection

The proposed fault detection architecture has been tested on the CSTR example studied by Yoon et al. (2001). Steady-state data from a model of that CSTR has been used to train a PCA model. The PCA model explains 98% of the variance in the original dataset with two components. A mismatch between the plant and the model has been considered by creating another copy of the original CSTR model. This duplicate represents the plant under conditions where the reaction rate is either overestimated ( or underestimated (. To further explore these scenarios, we conducted experiments simulating two specific events: a sudden catalyst poisoning (where the reaction rate reduces from 100% to 90%) and a drift (ramp) in the readings from the analyzer measuring the concentration of reactant A in the reactor. Figure 2 depicts the two latent variables and their differences when catalyst poisoning occurs at time = 1,000 min.

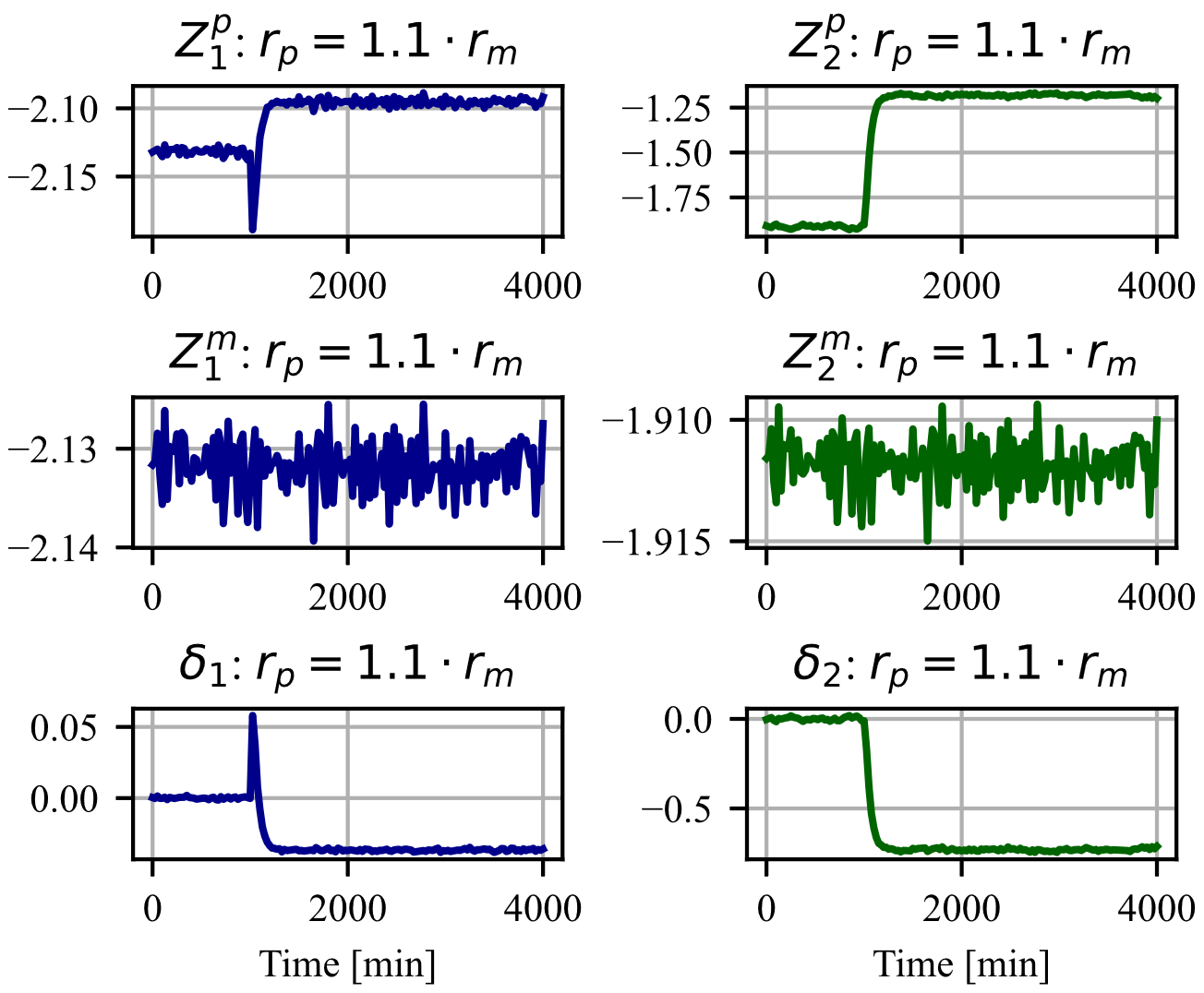
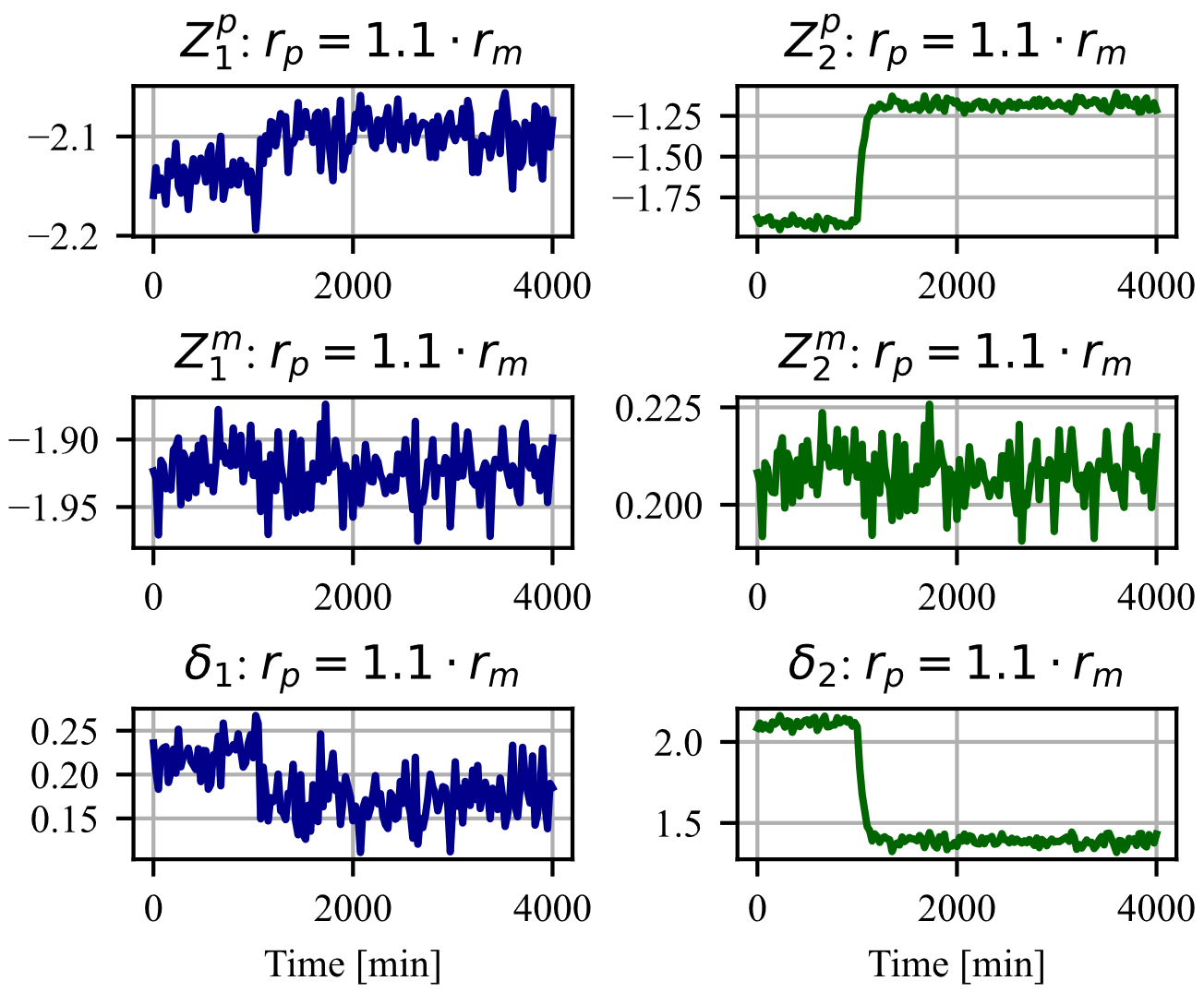


Figure 3. Filtering and bias adjustment of the latent space differences

Figure 2. Catalyst poisoning at t=1,000;

.

Since process noise makes it difficult to discern the shape of the difference plots, the data have been filtered (arithmetic moving average filter, 40 observations), and the differences have been offset by their biases that occur in normal operation (time up to 1,000 min), as shown in Fig. 3. Filtering and bias adjustment of the differences (to correct for the plant-model mismatch) opens a possibility that the plant operators could use the difference plots to detect visually occurrence of the faults ("difference is not zero = fault is in place").

Figure 5 shows the two filtered latent variables and their bias-adjusted differences when the concentration sensor drift starts at t = 1,000 min until t = 3,000 min, after which the error remains constant. The difference plot's slope depends on the analyzer drift's slope. As long as the analyzer error keeps increasing, the difference also increases. Figure 4 depicts the unfiltered and unadjusted data of the same case for reference purposes.

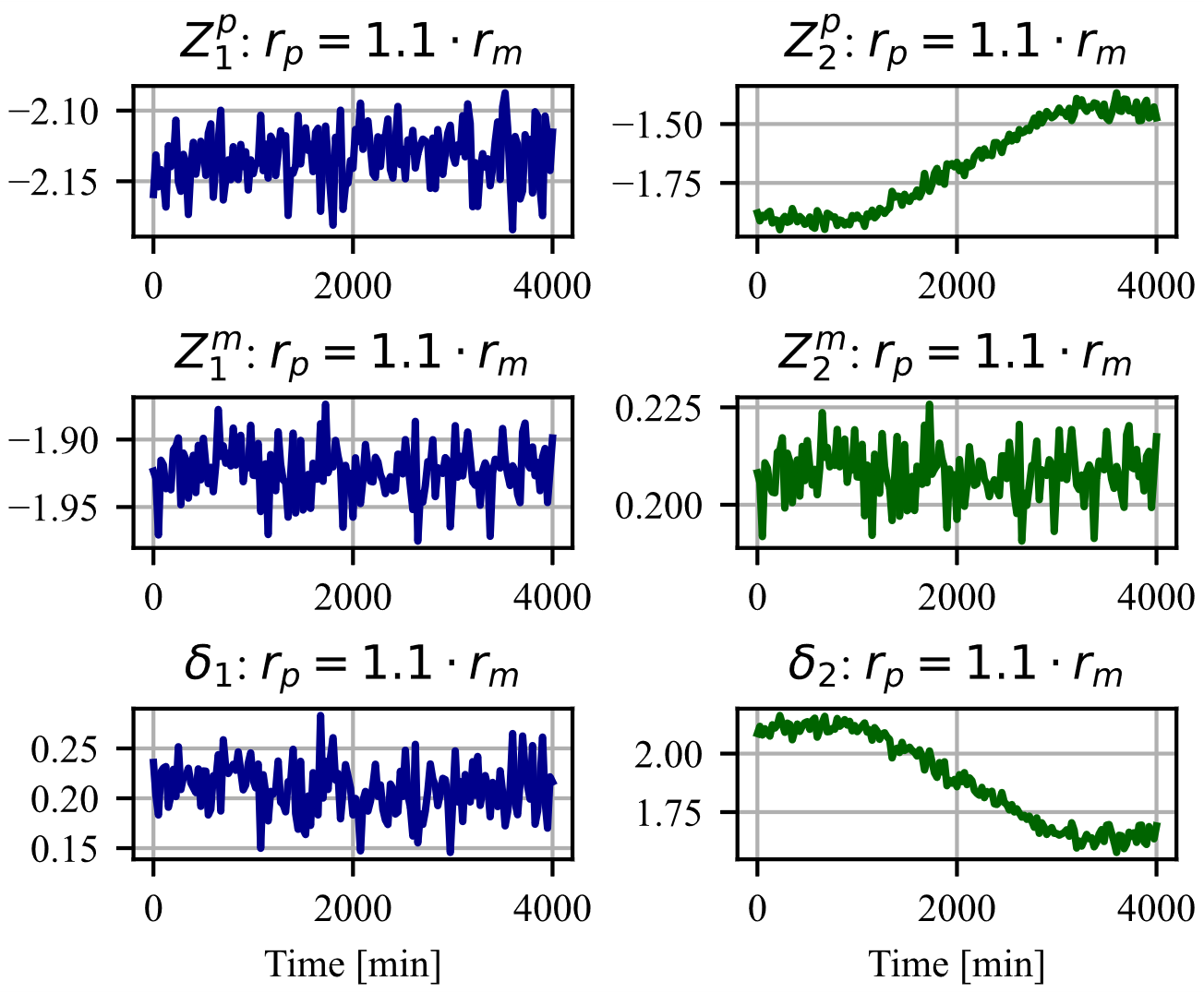


Figure 4. Concentration analyzer drift at t=1,000; .

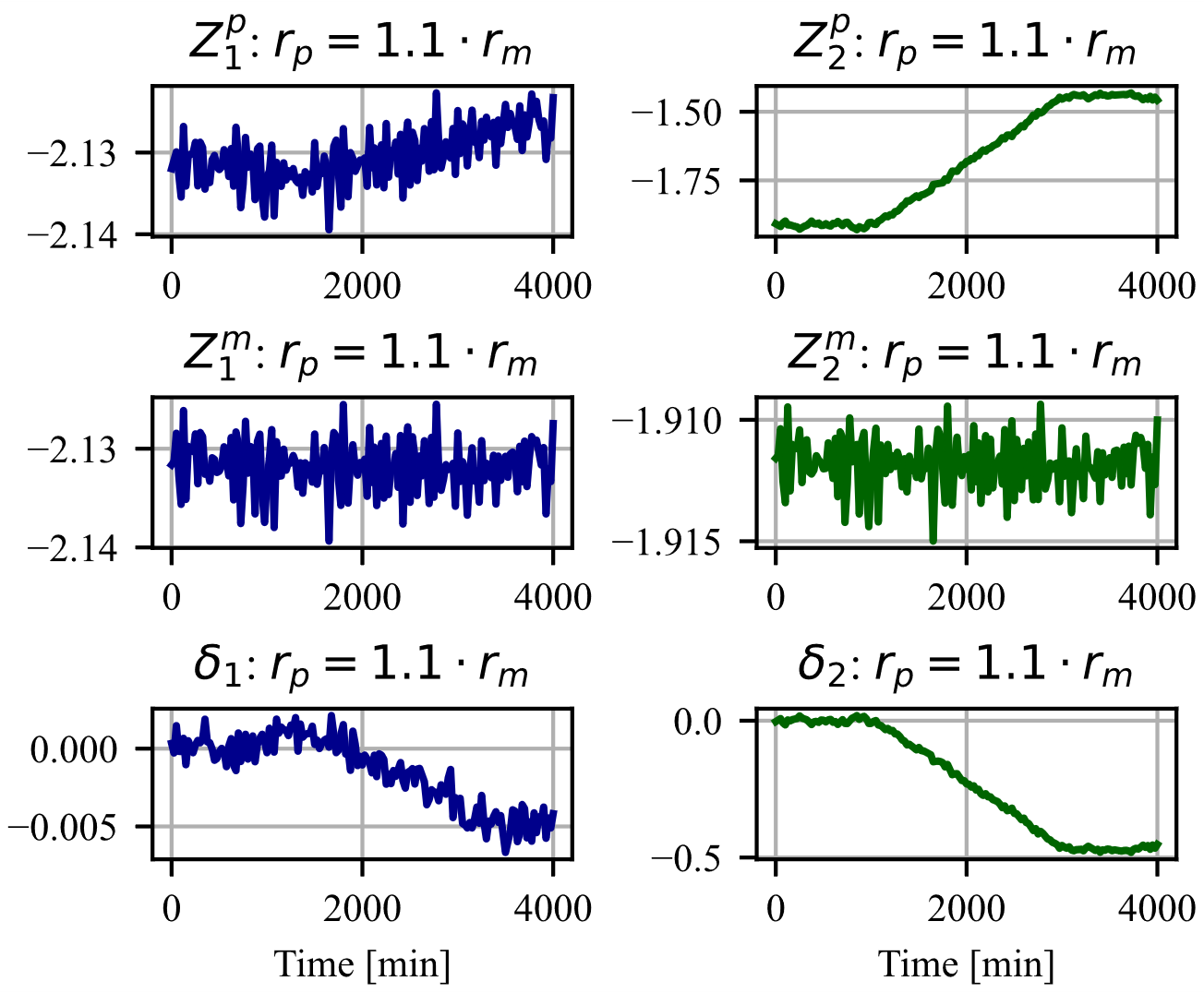


Figure 5. Concentration analyzer drift: filtered latent variables and their differences.

Figures 6 and 7 depict the latent space differences corresponding to the analyzer failure and the catalyst poisoning. For each of these two cases, there are two mismatched reaction rates and a case when the reaction rates in the plant and the model are identical.

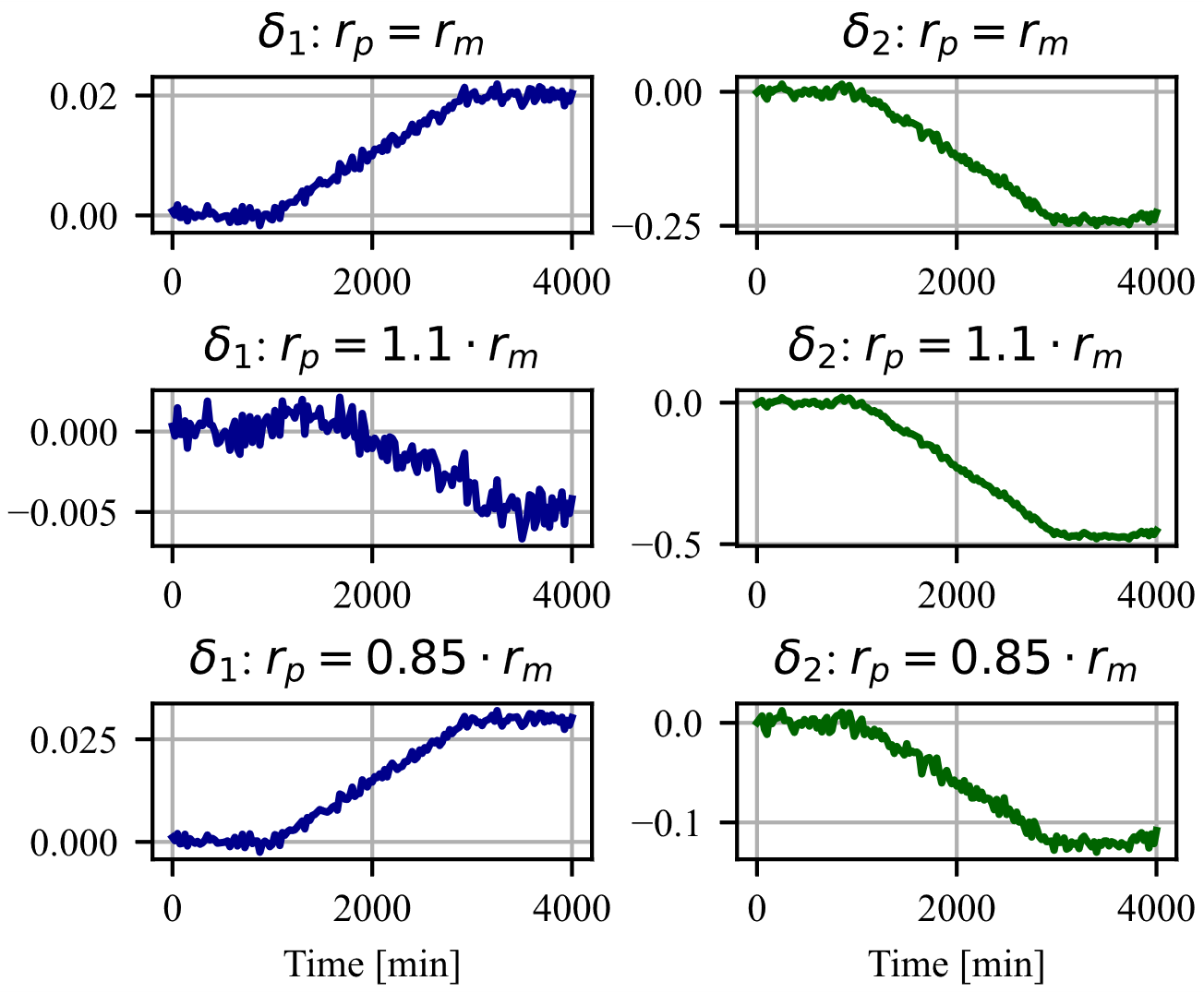


Figure 6. Analyzer drift starting at t=1,000 until t=3,000; latent space differences at different levels of reaction rates mismatch.

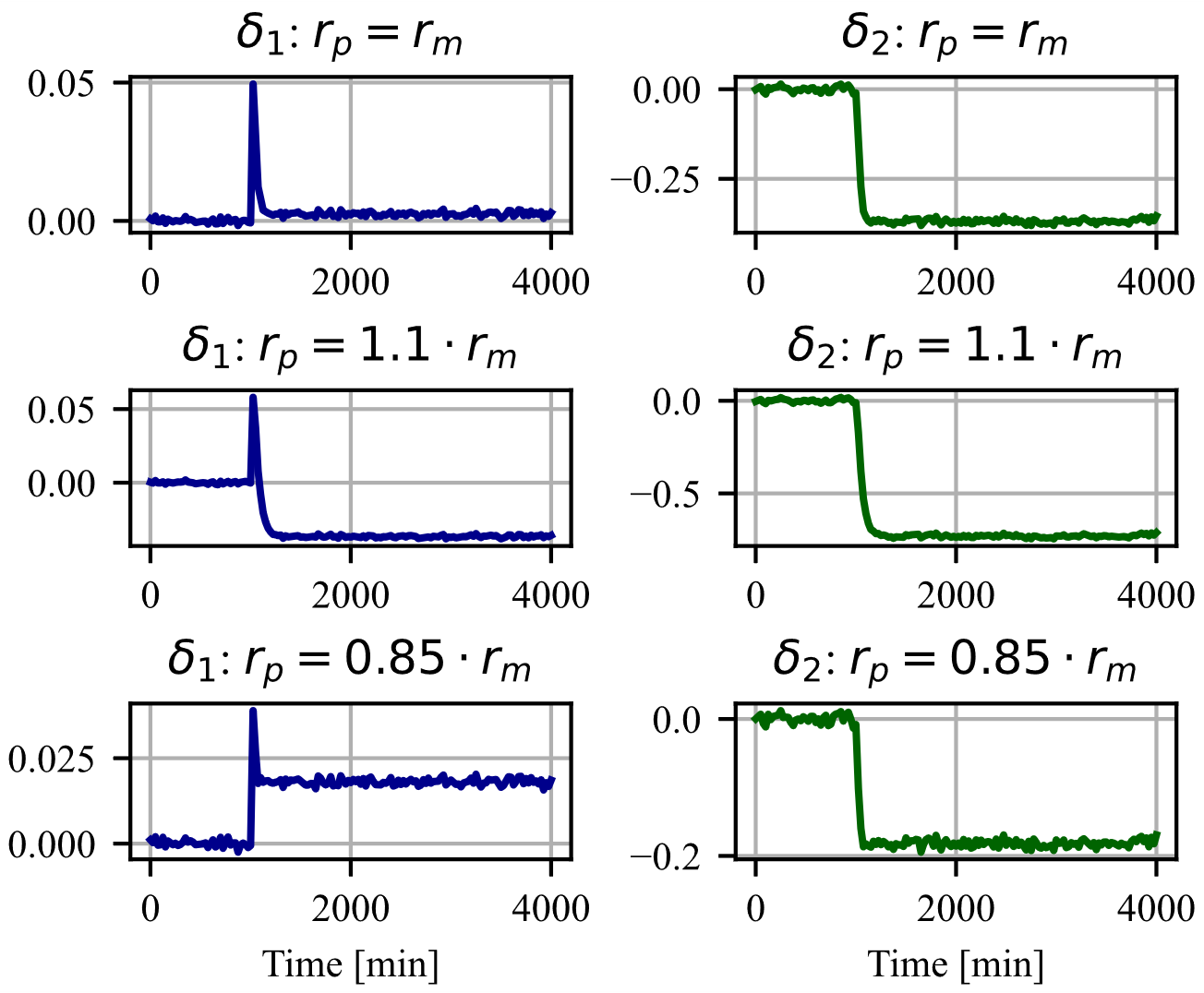


Figure 7. Catalyst poisoning occurs at t=1,000; latent space differences at different levels of reaction rates mismatch.

As seen from Fig. 6, when reaction rates in the model and the plant are the same, the δ1 difference becomes positive while the δ2 difference becomes negative. The same pattern is observed when the reaction rate in the plant is less than the rate in the model. Conversely, if the reaction rate in the plant is greater than the rate in the model, both δ1 and δ2 become negative.

If catalyst poisoning occurs (Fig. 7), a similar pattern is observed, i.e., if the reaction rate in the plant is greater than the reaction rate in the model, when a fault occurs, both (bias adjusted) differences have the same sign. In contrast, in all other cases, the first difference assumes the values that have the opposite sign of the second difference sign. Further research is needed to verify that this is the case when the mismatch is due to reaction rate.

* 1. Fault detection performance

To evaluate the performance of the proposed FD approach, we conducted comparative tests using classic PCA. PCA models were developed using data exclusively from normal operating conditions to ensure each plant-mismatch level scenario was paired with a corresponding PCA model. In this case, the PCA models were created to explain at least 98% of the variance in the training datasets. Notably, this level of explanation was achieved with just two principal components in all cases. The Square Prediction error (SPE) was selected as the metric to perform the fault detection via classic PCA:

To ensure a fair comparison, the PCA training datasets were filtered (arithmetic moving average filter, 40 observations) before z-scaling. In our proposed FD framework, we have introduced an anomaly score termed 'Square Sum of Differences' (SSD). This score quantifies all the differences in the latent space between the plant and its simulation model. The SSD is defined as follows:

In both methodologies, we employed Kernel Density Estimation (KDE) to fit a distribution to the metric. Subsequently, we established the control limit for each metric at the 99.99th percentile of its respective distribution.

Table 1. Fault detection times and F1 scores

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Case | Fault | td (SPE) [min] | F1 score (SPE) | td (SSD) [min] | F1 score (SSD) |
| Classic PCA | | This method | |
| No Mismatch | Sensor drift | 310.5 | 0.9719 | 191.5 | 0.9811 |
| No Mismatch | Catalyst poisoning | 7.5 | 0.9996 | 16 | 0.9990 |
|  | Sensor drift | 322.5 | 0.9712 | 136.5 | 0.9878 |
|  | Catalyst poisoning | 7.5 | 0.9996 | 15 | 0.9991 |
|  | Sensor drift | 310.5 | 0.9722 | 426.5 | 0.9566 |
|  | Catalyst poisoning | 7.5 | 0.9997 | 18.5 | 0.9987 |

The filtering techniques employed in this research were critical in mitigating process noise interference during the fault detection process. Moreover, to reduce the incidence of false alarms, it was determined that a fault alarm should only be activated following 15 consecutive observations exceeding the established threshold for normal operation. The results, presented in Table 1, detail the detection times (**td**) and F1 scores for both metrics.

* 1. Discussion

Latent space difference plots enable visual observation of the faults as they occur and of their progress through time.

Table 1 shows that in cases with no plant/model mismatch, or when the reaction rate is underestimated, the proposed parallel configuration excels in detecting slowly increasing faults (ramp). It also performs satisfactorily in detecting abrupt faults (step), as indicated by a high F1-score. The difference in detection time w.r.t. classic PCA is about 8 to 10 minutes, while at the same time the difference plots offer the insight in the nature of the fault (ramp or step change). We attribute this time difference to the filtering process, which averages down the values in the latent space during the occurrence of step behavior. Further experimentation with different filter types and filter is in progress.

Since plant model reproduces nonlinear behavior of the plant, and the plant model runs in parallel with the plant, it is expected that the proposed methodology will perform better than the classic PCA if a plant behaves nonlinearly, and it moves from one operating region to another.

* 1. Conclusions

The proposed fault detection architecture makes it possible to use the vast number of already existing process equipment simulation models to be used for fault detection. The models need not match the plant perfectly since the method performs well in case of the plant-model mismatch.

When a fault occurs in plant operation, the number of alarms sent to the operator console increases very quickly, which leads to an alarm overload and operators ignoring alarms. Graphical display of the latent space difference plots will allow operators to visually detect occurrence of a fault and eliminate broadcasting of numerous textual alarms for the same fault, which will significantly improve huma-machine interface

In addition to detecting the faults, the latent space differences open a possibility to diagnose the source of the fault by analyzing the pattern of the differences in addition to differences in the process variables between the plant and the model.

References

R. Iserman, 2005, Model-based fault-detection and diagnosis – status and applications. Annual Reviews in Control, 29 (1), 71-85

W. Sun, A.R.C. Paive, P. Xu, A. Sundaram, R.D. Braatz, 2020, Fault detection and identification using Bayesian recurent neural networks, Computers and Chemical Engineering, 141, 106991

Yoon, S., MacGregor, J.F. (2001) Fault diagnosis with multivariate statistical models. part i: using steady state fault signatures. Journal of Process Control, 11, 387–400