Multivariate Process Monitoring

Fault Detection, Diagnosis, and Prognosis Using PCA, Takens-PCA, CVA, and CVDA

1. ABSTRACT

Industrial plants routinely collect massive volumes of sensor data, yet major incidents still occur because early signs of abnormal behavior are often overlooked. This study demonstrates a unified multivariate monitoring framework for the Tennessee Eastman Process (TEP), combining PCA, Dynamic PCA via Takens embedding (DPCA), Canonical Variate Analysis (CVA), and Canonical Variate Dissimilarity Analysis (CVDA). These methods provide complementary strengths: PCA offers a steady-state baseline, DPCA captures dynamic evolution through time-lagged embeddings, CVA identifies changes in multivariable predictive structure, and CVDA detects subtle trajectory deviations.

The models were trained using strictly normal-operation data and evaluated across 20 fault scenarios. Results show that DPCA provides the highest overall detection performance, particularly for temperature- and composition-related faults, while CVA and CVDA extend coverage to faults that affect state dynamics or produce small mismatches in predicted future states. To demonstrate the full detect → diagnose → prognose workflow, we applied all stages to Fault 2, a composition step disturbance with consistently strong detection. Diagnosis localized the disturbance to the stripper and reactor levels, and prognosis showed that the fault remained stable over time. Overall, the integrated framework provides a practical blueprint for early detection, fault localization, and short-term fault behavior assessment in real industrial environments

2. INTROUCTION

This study investigates how multivariate statistical monitoring methods can detect, diagnose, and forecast abnormal behavior in complex process environments. Modern production systems operate with dozens of interdependent variables that evolve over time, making single-sensor or threshold-based monitoring insufficient. Subtle deviations often emerge first as multivariable patterns or slow dynamic drifts, and effective monitoring must capture these relationships to provide early and reliable warnings.

The Tennessee Eastman Process (TEP) serves as the evaluation platform for this work. The dataset contains three years of simulated plant operation, including normal behavior and twenty engineered fault scenarios. Its dynamic, noisy, and highly correlated structure provides a rigorous benchmark for assessing monitoring tools that aim to identify faults, isolate their sources, and anticipate short-term fault progression.

The objective of this study is to determine how well different multivariate frameworks can answer three operational questions essential for safe and reliable plant operation:

1. **Has a fault occurred?** (Fault Detection)
2. **Where is the fault coming from?** (Fault Diagnosis)
3. **How will the fault evolve over time?** (Fault Prognosis)

To address these questions, we examine four complementary methods: Static Principal Component Analysis (PCA), Dynamic PCA using Takens-style lag embeddings (DPCA), Canonical Variate Analysis (CVA), and Canonical Variate Dissimilarity Analysis (CVDA). PCA provides a steady-state baseline, DPCA captures short-term memory and dynamic structure, CVA models state transitions, and CVDA highlights deviations from expected trajectory evolution.

Together, these approaches form an integrated detect → diagnose → prognose workflow. The study evaluates their performance across all TEP faults and demonstrates the complete framework on Fault 2, which consistently showed strong detection. This provides a clear illustration of how multivariate monitoring can support early warning, fault localization, and short-term behavioral assessment in realistic industrial settings.

3. DATA BACKGROUND

3.1. Data Source

This study utilized the Tennessee Eastman Process dataset which represents a complex chemical production system with strongly interacting variables and multiple feedback loops. The simulation logs span three years of continuous operation from 1970 to 1972, with approximately 525,000 time-stamped records captured at three-minute sampling intervals, yielding a dense multivariate time series.

Each record includes 41 process measurement variables (XMEAS), 12 manipulated variables (XMV), and one status variable indicating the operating condition of the plant. The status variable encodes 0 for normal operation and 1–20 for different fault conditions. All variables are real plant sensor readings obtained from the dynamic simulation, and the dataset is exceptionally clean: there are no missing values and no duplicate rows, which means that the preparation work can focus on modeling instead of data repair.

3.2. Data Dictionary

The following table details the 52 original variables provided, refer to the process flow diagram in the appendices as needed.

Variable	Type	Description
XMEAS(1)	float	A feed flow (stream 1)
XMEAS(2)	float	D feed flow (stream 2)
XMEAS(3)	float	E feed flow (stream 3)
XMEAS(4)	float	Total feed flow (stream 4; A+C feed)
XMEAS(5)	float	Recycle flow (stream 8)
XMEAS(6)	float	Reactor feed rate (stream 6)
XMEAS(7)	float	Reactor pressure
XMEAS(8)	float	Reactor liquid level
XMEAS(9)	float	Reactor temperature
XMEAS(10)	float	Purge rate (stream 9)
XMEAS(11)	float	Product separator (vapor–liquid separator) temperature
XMEAS(12)	float	Product separator level
XMEAS(13)	float	Product separator pressure
XMEAS(14)	float	Product separator underflow / liquid flow (stream 10)
XMEAS(15)	float	Stripper level
XMEAS(16)	float	Stripper pressure
XMEAS(17)	float	Stripper underflow / liquid product flow (stream 11)
XMEAS(18)	float	Stripper temperature
XMEAS(19)	float	Stripper steam flow
XMEAS(20)	float	Compressor work (power)
XMEAS(21)	float	Reactor cooling-water outlet temperature
XMEAS(22)	float	Condenser (separator) cooling-water outlet temperature
XMEAS(23)	float	Composition of A in reactor outlet (stream 6)
XMEAS(24)	float	Composition of B in reactor outlet (stream 6)
XMEAS(25)	float	Composition of C in reactor outlet (stream 6)
XMEAS(26)	float	Composition of D in reactor outlet (stream 6)
XMEAS(27)	float	Composition of E in reactor outlet (stream 6)
XMEAS(28)	float	Composition of F in reactor outlet (stream 6)
XMEAS(29)	float	Composition of A in purge gas (stream 9)
XMEAS(30)	float	Composition of B in purge gas (stream 9)
XMEAS(31)	float	Composition of C in purge gas (stream 9)
XMEAS(32)	float	Composition of D in purge gas (stream 9)
XMEAS(33)	float	Composition of E in purge gas (stream 9)
XMEAS(34)	float	Composition of F in purge gas (stream 9)
XMEAS(35)	float	Composition of G in purge gas (stream 9)
XMEAS(36)	float	Composition of H in purge gas (stream 9)
XMEAS(37)	float	Composition of D in product / bottoms (stream 11)
XMEAS(38)	float	Composition of E in product / bottoms (stream 11)
XMEAS(39)	float	Composition of F in product / bottoms (stream 11)
XMEAS(40)	float	Composition of G in product / bottoms (stream 11)
XMEAS(41)	float	Composition of H in product / bottoms (stream 11)
XMV(1)	float	D feed flow setpoint / valve (stream 2)
XMV(2)	float	E feed flow setpoint / valve (stream 3)
XMV(3)	float	A feed flow setpoint / valve (stream 1)
XMV(4)	float	A and C total feed flow (stream 4)
XMV(5)	float	Compressor recycle valve position
XMV(6)	float	Purge valve position (stream 9)
XMV(7)	float	Separator pot liquid flow (stream 10)
XMV(8)	float	Stripper liquid product flow (stream 11)
XMV(9)	float	Stripper steam valve position
XMV(10)	float	Reactor cooling-water flow valve position
XMV(11)	float	Condenser cooling-water flow valve position
XMV(12)	float	Agitator speed (reactor stirrer)
STATUS	integer	Fault Number = 0 → Normal Operation Fault Number = 1 – 20 → Fault Types

4. METHODOLOGY

A robust multivariate monitoring model requires clean separation between the data used to learn normal behavior and the data used to evaluate detection performance. The first step in our preprocessing pipeline was therefore a careful split of the dataset into normal and faulty subsets based on the STATUS label. Only records with STATUS = 0 were used to estimate the baseline models; all other records were reserved for evaluating fault detection, diagnosis, and prognosis.

4.1. Time Index Construction

The original log file includes a timestamp column that is converted into a proper datetime variable and sorted chronologically. This ensures that the time dimension is correctly aligned before any time series analysis or lag embedding is performed.

4.2. Train – Test Split

The normal operation subset is partitioned into training and test segments using a 70–30 time-based split. The earliest 70% of the normal records form the Normal-Train dataset, from which all baseline models and scaling parameters are estimated. The remaining 30% of normal records constitute the Normal-Test dataset, used exclusively to evaluate false alarm behavior under healthy operating conditions.

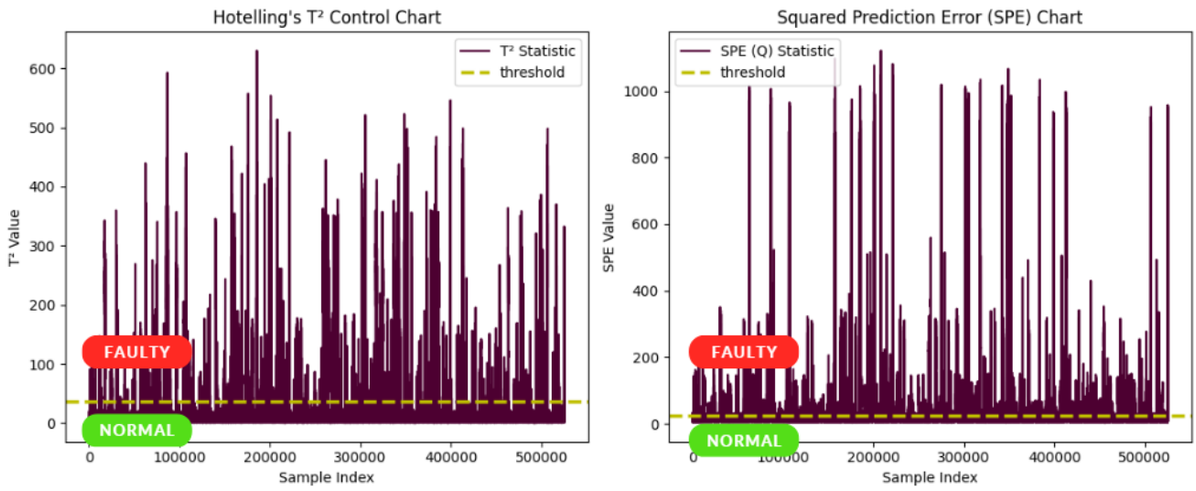
Split	Size	Description
Normal-Train	185.438 rows	Model learns normal behavior
Normal-Test	79,474 rows	Can the model stay quiet when nothing is wrong?
Faulty-Test	260,688 rows	Can the model alarm when something is wrong?

Because the 52 process variables differ in scale, all features are standardized using z-scores. A StandardScaler is fitted only on the Normal-Train subset, and the resulting means and standard deviations are applied to the Normal-Test and each of the 20 fault datasets. This approach prevents information leakage from fault data into the baseline model and ensures that principal components and canonical variates are not dominated by large-scale variables.

4.3. Baseline Model: Static PCA

Static PCA is used as the baseline detector in the monitoring framework. The method decomposes the 52 standardized process variables into uncorrelated latent components that summarize the dominant patterns in the Normal-Train dataset. Only components that capture meaningful steady-state variation are retained, while low-variance directions are treated as noise.

Using this reduced representation, two monitoring statistics are computed for each new observation: Hotelling's T^2, which measures distance within the retained PCA subspace, and the Q or squared prediction error statistic, which measures the residual component not captured by the PCA model. Control limits for both statistics are estimated from the Normal-Train dataset, representing the natural variability of the healthy process. Samples that exceed either limit are considered abnormal.



Static PCA forms the baseline against which the performance of Dynamic PCA, CVA, and CVDA is assessed. For each method, detection performance is evaluated in terms of alarm occurrence, detection delay, and the proportion of faulty samples correctly identified.

This ensures that the time dimension is correctly aligned before any time series analysis or lag embedding is performed.

4.4. Dynamic PCA (DPCA)

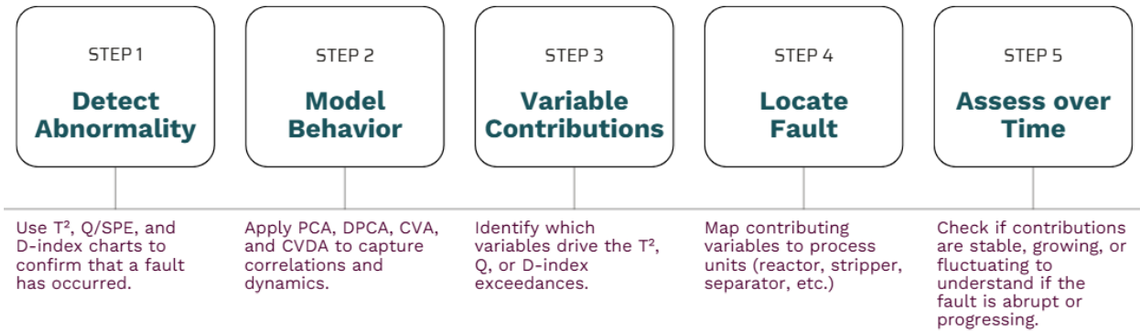
Dynamic PCA extends the static PCA model by incorporating time-lagged versions of each variable. This idea follows Takens' theorem, which states that the behavior of a dynamical system can be reconstructed from delayed observations of its variables. In practice, each variable is expanded into a sequence of past values, allowing the model to learn not only cross-variable correlations but also how past states influence current behavior.

4.5. CVA and CVDA

The study also implemented Canonical Variate Analysis, which models the relationship between past and future data blocks in a multivariate autoregressive framework. By comparing predicted versus actual system behavior, CVA provides residuals that are sensitive to changes in dynamic relationships. Canonical Variate Dissimilarity Analysis (CVDA) further summarizes differences in canonical states through a dissimilarity index D. This index quantifies how different each new observation is from the canonical patterns observed during normal operation. The D-index is highly sensitive to small deviations and is particularly effective for identifying incipient faults with weak observable signatures.

4.6. Unified Workflow

All four detection models follow a common evaluation procedure:

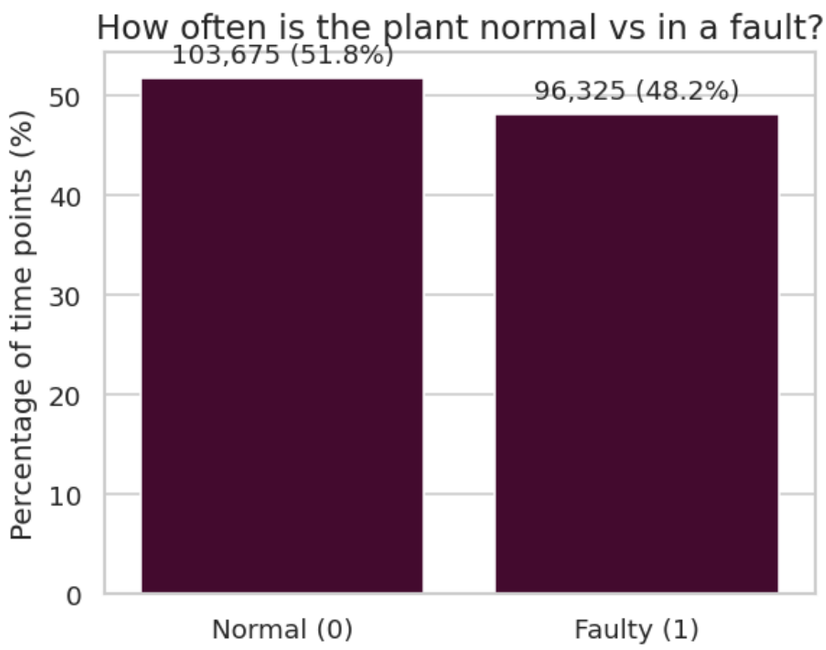


1. Fit the model on the Normal-Train dataset.
2. Apply the Normal-Train scaling parameters to Normal-Test and Faulty-Test data.
3. Compute the corresponding monitoring statistics for each method.
4. Establish control limits using the Normal-Train distribution.
5. Evaluate Normal-Test for false alarms and Faulty-Test for fault detection performance.

This unified workflow ensures fair comparison across methods and supports systematic analysis of which models are best suited for the different types of Tennessee Eastman faults.

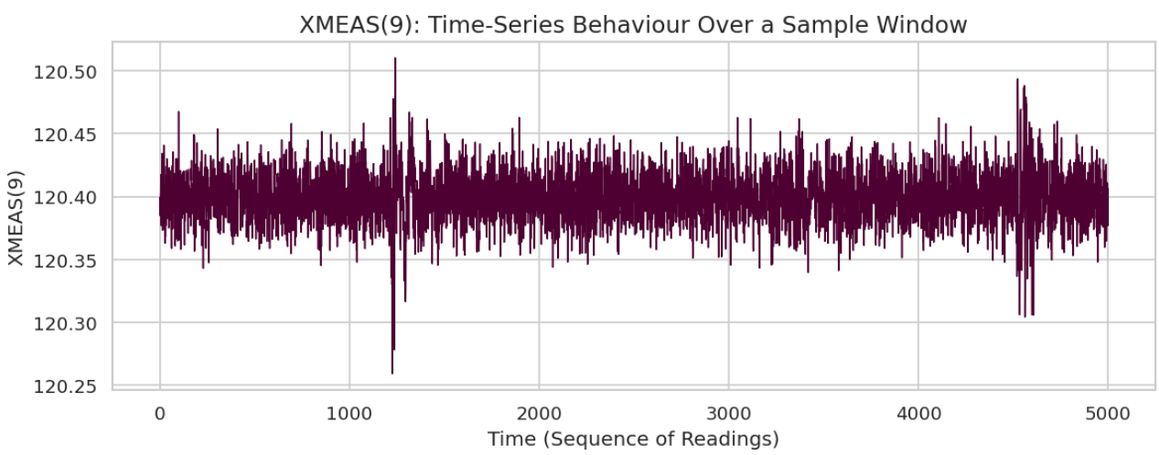
5. BASIC EDA

The process is roughly evenly split between normal and faulty conditions. The bar chart in Figure 1 illustrates this distribution, with counts and percentages annotated for each state. This balanced design is intentional in the TEP benchmark and ensures that the models will be stress-tested across a rich variety of failure modes.

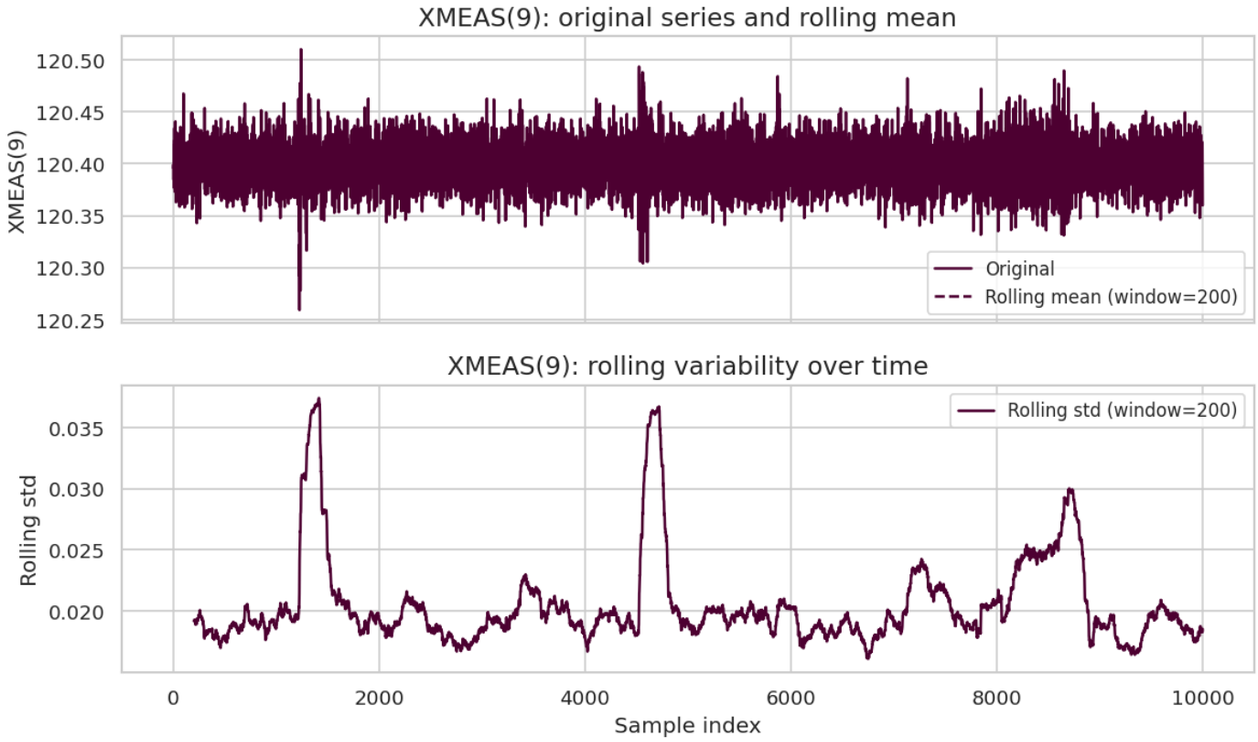


To illustrate time-series behavior, the analysis focuses on XMEAS(9), a reactor temperature measurement used throughout the notebook

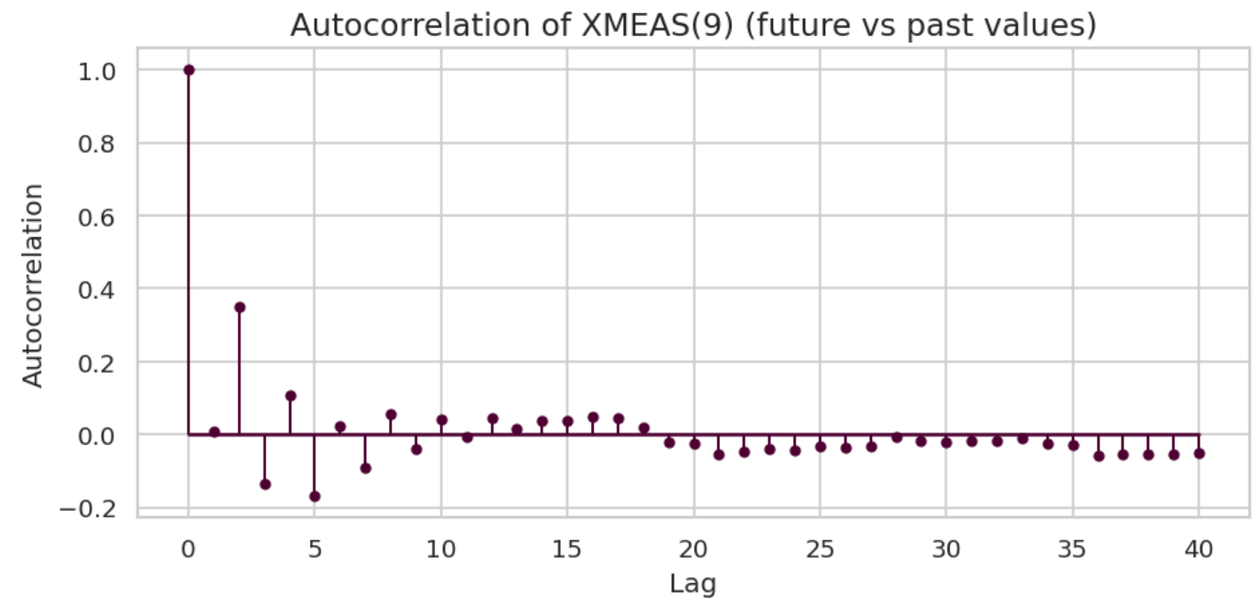
The figure below shows a time-series plot of XMEAS(9) under a representative window of operation. The series exhibits small, steady fluctuations within a narrow band, indicating stable and well-controlled behavior during normal operation.



The figure belowshows the original XMEAS(9) series over a longer horizon, overlaid with a rolling mean, while Figure 4 shows the rolling standard deviation computed over a moving window (e.g., window = 200). At certain intervals, the rolling standard deviation spikes, indicating a sudden increase in variability even before the mean level visibly shifts. These bursts can be interpreted as early warning signs of instability, where the readings start to spread out and the process becomes less tightly controlled.



The figure below presents the autocorrelation function (ACF) of XMEAS(9) up to 40 lags. The strong positive autocorrelation at lower lags confirms that each reading is highly dependent on recent past values. The gradual decay in autocorrelation suggests a process with memory rather than a white-noise sequence, and the small negative dips at longer lags reflect corrective actions by the control system.

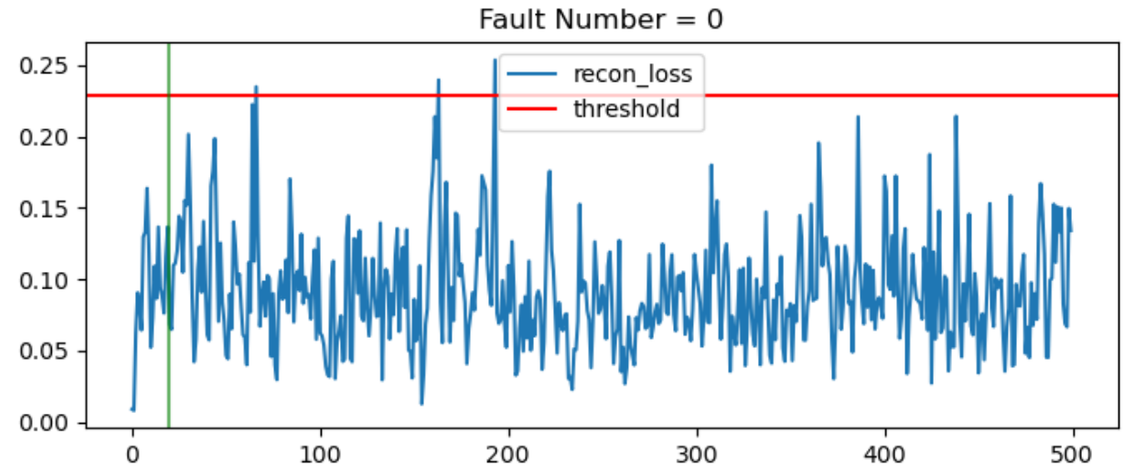


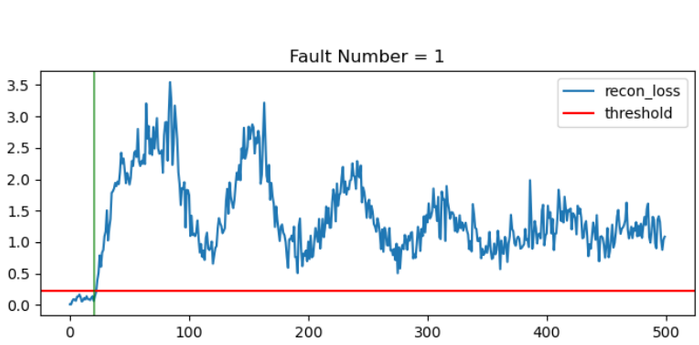
Together, these graphs justify the use of dynamic monitoring approaches such as DPCA and CVA, which explicitly exploit temporal dependencies rather than treating each observation as independent.

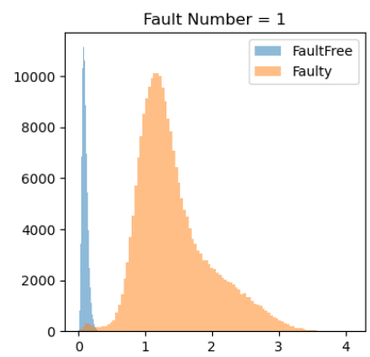
6. DETECTION RESULTS

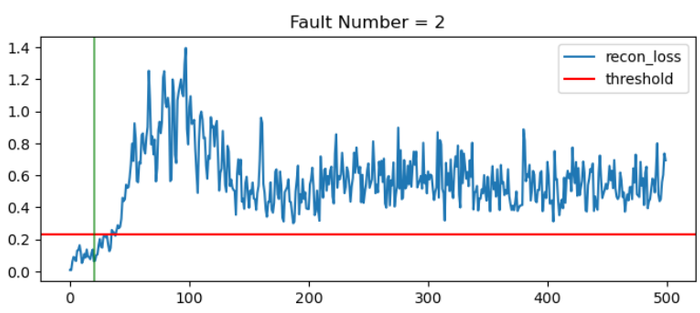
6.1. Baseline Detection

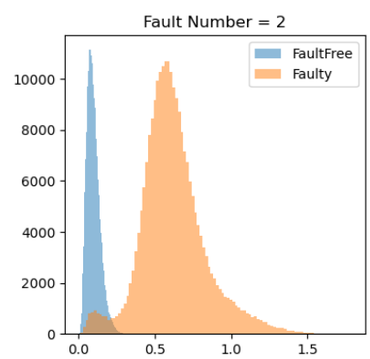
To evaluate how well the baseline PCA model detects faults in real time, we feed the entire test set into the model, including both normal periods and injected fault periods. The PCA model has been trained only on normal operating data, so it knows what healthy behavior looks like. During monitoring, we track the reconstruction loss: the difference between the actual measurements and their reconstruction inside the PCA space. Under normal conditions, this loss stays below the threshold learned from the Normal-Train dataset.



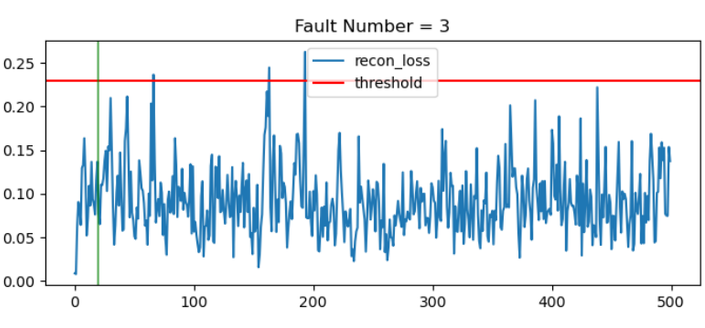


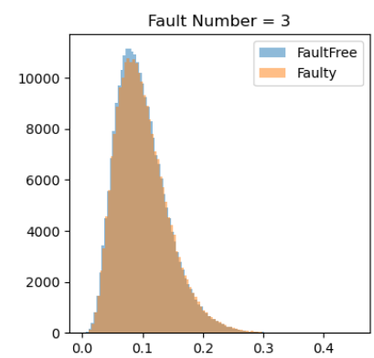






The first plot shows a purely normal segment, where the reconstruction loss fluctuates naturally but remains below the threshold. This confirms that the PCA model stays quiet when nothing is wrong. When a fault is introduced, as shown in Fault 1 and Fault 2, the reconstruction loss increases sharply and consistently crosses the threshold. These exceedances trigger an alarm and indicate abnormal operation.





However, not all faults produce a clear signal. For Fault 3, the reconstruction loss stays within the normal variability range, resulting in no alarm. This illustrates a key limitation of static PCA: faults that are weak, slow, or masked by control actions may not produce enough deviation for PCA to detect. Overall, these examples show how PCA behaves in real time, highlighting both its strengths in flagging large deviations and its limitations in catching subtle or dynamic faults.

6.2. Takens-PCA (Dynamic PCA)

Takens-PCA extends static PCA by embedding short-term process history into each sample. Instead of relying only on current measurements, each observation is augmented with its past values. This captures the temporal dependencies present in continuous chemical operations influenced by recycle flows, control loops, and thermal inertia. In this study, a two-step embedding (L = 2) expanded the 52-variable dataset into 156 dynamic features.

Before selecting L = 2, different lag depths were tested to determine when reconstruction error and detection behavior stabilized. The goal was to find a minimum embedding dimension that captured essential system memory without increasing noise or computational load. The selected lag depth provided stable variance structure and reliable detection performance.

After constructing the dynamic inputs, PCA was trained on normal operation and control limits were derived from held-out normal data. The DPCA model retained 46 components with about 90 percent cumulative variance and maintained low false alarm rates, showing that temporal embedding did not cause excessive sensitivity during healthy operation.

To assess fault detection, the DPCA model was applied to each faulty dataset using the same lags and control limits. Several faults showed strong detection, particularly temperature variations and step changes in feed composition, where deviations evolve over time. These cases benefit from DPCA because the method captures shifts in process trajectory rather than only instantaneous deviations. Other faults, such as mechanical and stochastic disturbances, produced weaker signals. These faults either do not strongly affect the dynamic structure or are masked by active control loops.

Fault No.	Fault Type	Best Stat	Detection Rate
1	Process – Composition Step	T^2	0.947
2	Process – Composition Step	T^2	0.943
4	Process – Temperature Variation	Q	1.000
7	Process – Temperature Variation	T^2	0.998
11	Mechanical – Condenser Valve Stuck	Q	0.602
14	Mechanical – Separator Level Valve Stuck	T^2	0.749
20	Stochastic – Temperature Variation	Q	0.239

The selected faults in the table highlight three main patterns. First, dynamic or temperature-related faults (Faults 4 and 7) show near-perfect detection, confirming that DPCA is well suited for disturbances that propagate over time. Second, some mechanical faults (Faults 11 and 14) are detectable when they influence system dynamics, although with varying strength. Third, stochastic faults (Fault 20) remain difficult to detect, showing the limits of DPCA when disturbances lack clear temporal structure.

Overall, DPCA improves detection for faults that modify the dynamic behavior of the process. By incorporating system memory, the method becomes more sensitive to gradual drifts, temperature propagation, and composition shifts while maintaining low false alarm behavior. Its weaker performance on noisy or weakly structured faults indicates that complementary methods such as CVA and CVDA are valuable for capturing anomalies that do not significantly affect short term dynamics.

6.3.Canonical Variate Analysis (CVA)

Canonical Variate Analysis models the process as a state-space system by separating each observation into past and future blocks and identifying linear combinations that best predict the future from the past. Using only normal-operation data, CVA learns these predictive relationships and defines T^2 and Q statistics that measure deviations from normal state evolution. With past and future windows of size five, the model produced stable canonical variates and low false alarm rates in the normal test set.

When applied to fault data, CVA performed best for faults that disrupt multivariable dynamics rather than only instantaneous measurements. Faults 1, 5, and 6 are the clearest examples. These faults showed strong T^2 and Q detection (Fault 5: Q DR ≈ 0.984; Fault 1: T^2 DR ≈ 0.963; Fault 6: T^2 DR ≈ 0.828), indicating that CVA effectively captures coordinated changes in system state evolution. These are faults where the underlying dynamic relationships between variables shift in a way that PCA-based methods often overlook.

CVA was weaker for faults that remain localized, heavily compensated by control loops, or behave stochastically. Many mechanical faults (Faults 10, 11, 14, 15) and random faults (Faults 16–20) produced very low detection rates because the prediction errors in the canonical space remained small. This pattern shows that CVA is best applied when faults alter the latent state dynamics rather than isolated measurements.

Fault No.	Fault Type	Best Stat	Detection Rate
1	Process – Composition Step	T^2	0.963
5	Process – Temperature Variation	Q	0.984
6	Process – Temperature Variation	T^2	0.828
12	Mechanical – A Feed Valve Stuck	Q	0.283
13	Mechanical – D Feed Valve Stuck	T^2	0.338
17	Random – Cooling Variation	T^2	0.284

Overall, CVA expands the coverage of dynamic faults by detecting changes in predictive structure across variables. Although it cannot capture all disturbances, it provides meaningful detection of faults that shift the coordinated behavior of the process.

6.4.Canonical Variate Dissimilarity Analysis (CVDA)

CVDA builds on the CVA model by quantifying how different the predicted future state is from the actual future state. This difference is summarized by the D-index, which rises when the system trajectory deviates from the learned normal dynamic pattern. A data-driven threshold was computed from held-out normal data, resulting in low false alarm behavior.

The CVDA results show that the method excels in a different subset of faults. The clearest example is Fault 6, where the D-index achieved perfect detection with zero delay. Faults 5, 12, and 16 also showed moderate detection rates (around 0.24), demonstrating CVDA's ability to detect subtle or localized deviations that create mismatches in future-state prediction. These are faults where T^2 or Q statistics often show weak sensitivity, but CVDA's trajectory-based measure captures the anomaly.

At the same time, CVDA shows limited performance for many faults. Composition step changes (Faults 1 and 2), several mechanical faults (Faults 10, 11, 14, 15), and most random faults produced low detection rates and long delays. These faults either evolve smoothly, remain well-compensated by control actions, or do not significantly distort short-term predictive structure.

Fault No.	Fault Type	D-index Detection Rate
6	Process – Temperature Variation	1.000
5	Process – Temperature Variation	0.238
12	Mechanical – A Feed Valve Stuck	0.238
16	Random – Reactor Cooling Variation	0.242
17	Random – Condenser Cooling Variation	0.074
11	Mechanical – Condenser Valve Stuck	0.053

Taken together, the CVA and CVDA results show that state-space methods detect a different class of anomalies from PCA and DPCA. CVA captures coordinated multivariable changes in system dynamics, while CVDA is more sensitive to subtle mismatches in predicted versus actual future states. Their combined strengths complement DPCA, providing broader coverage of the diverse fault behaviors present in the Tennessee Eastman Process.

6.5.Detection Summary

The overall results show that Dynamic PCA using Takens embedding provides the strongest detection performance among all monitoring methods. By incorporating lagged variables, the DPCA model expands the original 52 variables to a 156-variable dynamic feature space and retains 46 principal components that capture more than 90 percent of the variance. This richer temporal representation results in the highest average detection rate across all faults, outperforming static PCA, CVA, and CVDA.

Method	Avg Detection Rate
Dynamic PCA	39.10%
Static PCA	26.00%
CVA	24.70%
CVDA	13.50%

Faults can also be grouped by category. Process faults (F1–F9) are the most detectable, with an average detection rate of about 66 percent. These disturbances typically involve temperature or composition changes that propagate across units, making them suitable for multivariate and dynamic models. Mechanical faults (F10–F15) show moderate detectability at approximately 37 percent, while stochastic faults (F16–F20) remain the most challenging at about 21 percent due to their weak or irregular signatures.

Fault Category	Fault Numbers	Average Detection Rate
Process Faults	F1 - F9	66.20%
Mechanical Faults	F10 - F15	37.20%
Stochastic Faults	F16 - F20	20.70%

To better understand the model behavior across individual faults, Table below summarizes the ranking of faults by their best detection rate and the method responsible for the highest score.

Rank	Fault No.	Fault Definition	Best DR	Best Method
1	F4	Step Feed E comp. (Process)	100.00%	DPCA (Q)
2	F6	Feed B temp var. (Process)	100.00%	DPCA (Q)
3	F7	Feed C temp var. (Process)	99.80%	DPCA (T2)
4	F5	Feed A temp var. (Process)	98.40%	CVA (Q)
5	F1	Step A/C ratio (Process)	97.10%	PCA (Q)
6	F2	Step Feed B comp. (Process)	94.30%	DPCA (T2)
7	F14	Separator valve stuck (Mechanical)	74.90%	DPCA (T2)
8	F11	Condenser valve stuck (Mechanical)	60.20%	DPCA (Q)
9	F13	D feed valve stuck (Mechanical)	42.50%	PCA (Q)
10	F17	Condenser temp (Rand.) (Stochastic)	43.70%	PCA (Q)
11	F12	A feed valve stuck (Mechanical)	43.30%	DPCA (Q)
12	F16	Reactor temp (Rand.) (Stochastic)	24.20%	CVDA (D)
13	F20	Feed A temp (Rand.) (Stochastic)	23.90%	DPCA (Q)
14	F18	Feed A comp (Rand.) (Stochastic)	11.80%	CVA (Q)
15	F8	Feed D temp var. (Process)	4.80%	CVA (Q)
16	F10	Reactor valve stuck (Mechanical)	1.50%	CVDA (D)
17	F9	Condenser temp var. (Process)	0.70%	CVDA (D)
18	F15	Stripper valve stuck (Mechanical)	0.60%	CVDA (D)
19	F3	Step Feed D comp. (Process)	0.50%	CVDA (D)
20	F19	Feed B comp (Rand.) (Stochastic)	0.10%	DPCA (Q)

The ranking highlights several clear patterns. First, DPCA performs exceptionally well on temperature-driven process disturbances such as Faults 4, 6, and 7, each achieving near-perfect or perfect detection. Other process faults like Faults 5, 1, and 2 also achieve high detection using a mix of DPCA, PCA, and CVA, indicating that these faults induce coordinated, system-wide deviations.

Mechanical faults such as Faults 14 and 11 reach moderate detection levels because they influence system pressure and flow dynamics strongly enough to appear in the monitored variables. Among the stochastic faults, only Fault 17 stands out with a detection rate above 40 percent, while the remaining random faults show minimal detectability for all methods.

At the bottom of the ranking are faults that evolve slowly, remain localized, or are heavily compensated by control loops. Reactor and stripper valve failures (Faults 10 and 15), condenser temperature variation (Fault 9), and the subtle step change in Feed D composition (Fault 3) all show very low detection rates even for advanced methods like CVDA.

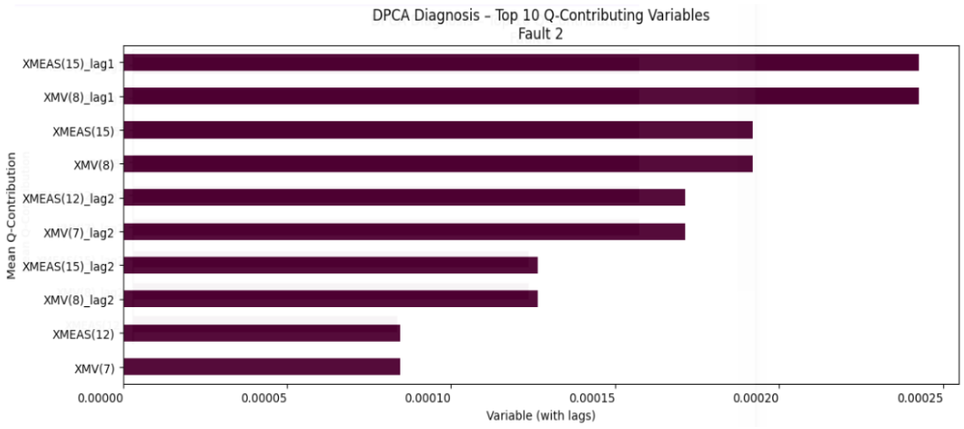
Overall, the results show that no single method can detect every type of disturbance. DPCA provides the strongest general performance, particularly for dynamic and temperature-related faults. PCA and CVA contribute to clearer composition and

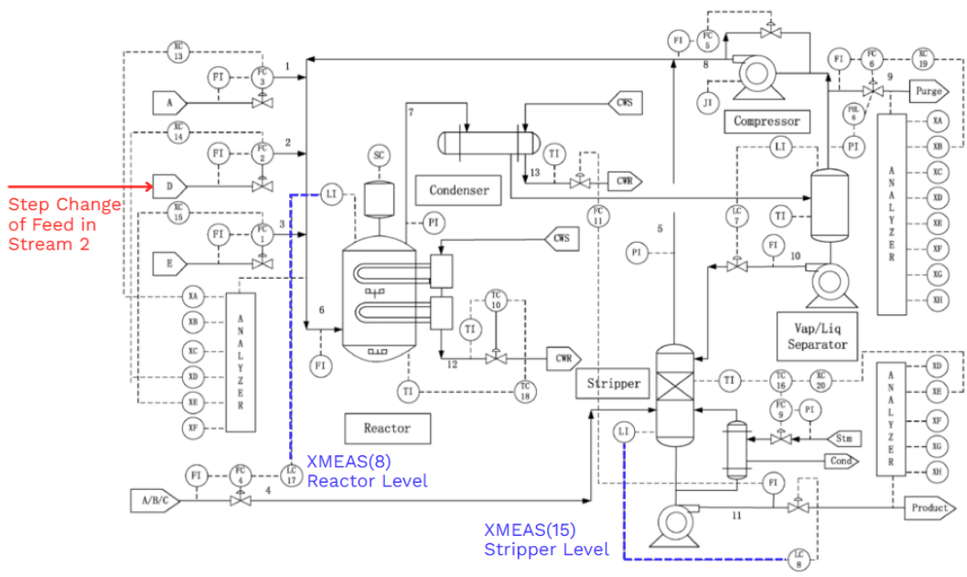
pressure disturbances, while CVDA captures certain subtle deviations in predicted future states. Together, these methods provide complementary strengths for monitoring the diverse fault mechanisms present in the Tennessee Eastman Process.

7. DIAGNOSIS AND PROGNOSIS

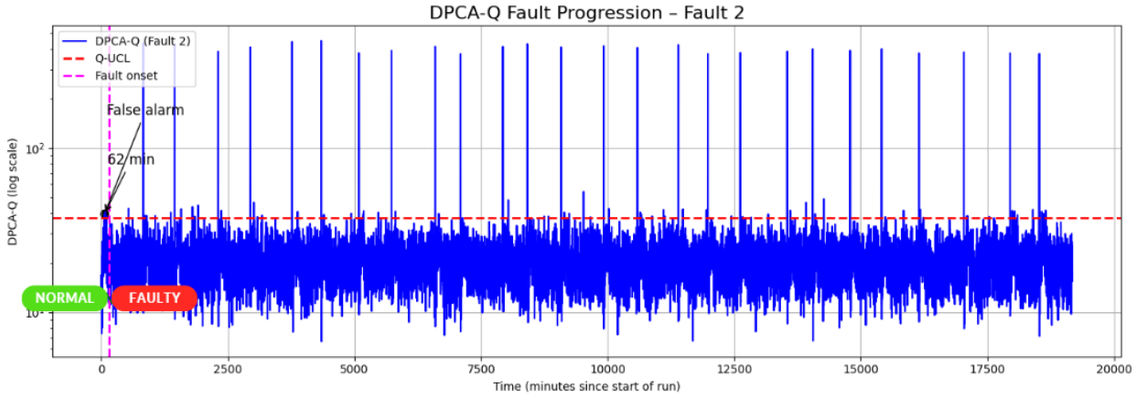
We applied the full detect–diagnose–prognose framework to Fault 2 because it achieved consistently high detection rates across PCA, DPCA, and CVA. Its strong T^2 response made it an ideal case to demonstrate how diagnosis and prognosis complement detection in a real plant setting.

After confirming the presence of Fault 2, we used DPCA Q-contributions to determine where the disturbance propagated inside the plant. The top contributors were XMEAS(15) and XMEAS(8), representing the stripper level and reactor level. These variables repeatedly appeared with and without lags, indicating that the increase in Stream 2 directly pushes material into both units. Overlaying these variables on the process flowsheet highlights the physical pathways of the disturbance.





To evaluate the fault's behavior over time, we examined the DPCA-Q progression chart. The results show that Fault 2 does not intensify or drift. Instead, it remains a stable, abrupt disturbance. This stability suggests that flow and level controllers are actively regulating the impact, preventing escalation. Because no upward trend or spreading behavior is observed, the fault is classified as non-progressing and does not require immediate shutdown.



The combined diagnosis and prognosis stages provide deeper operational insight. Detection tells us that something changed. Diagnosis identifies where the change occurred. Prognosis clarifies whether the change is growing, fading, or stable. For Fault 2, the framework shows a localized and stable disturbance affecting the stripper and reactor levels, with no progression over time.

Finally, to ensure that the workflow can be generalized, we developed a fault behavior library covering all twenty TEP faults. It summarizes fault descriptions, top contributing variables, and operational implications. This library is included in the

appendices of the paper and supports rapid interpretation, risk-based prioritization, and more consistent decision-making by plant engineers.

8. CONCLUSIONS AND FUTURE WORK

This study evaluated four complementary multivariate monitoring techniques: PCA, DPCA, CVA, and CVDA. Using the Tennessee Eastman Process as a benchmark, the results show that DPCA performs best for faults with strong dynamic behavior because it captures temporal patterns through Takens-style embedding. PCA remains effective for large steady-state deviations. CVA improves early detection by modeling state evolution, while CVDA captures subtle inconsistencies in process trajectories. When combined, these methods create a complete detect → diagnose → prognose workflow that offers wider and more reliable coverage than any individual technique.

We demonstrated the entire framework using Fault 2 because it consistently produced high detection rates across the methods. Diagnosis using Q-contributions traced the disturbance to the stripper and reactor levels. Prognosis confirmed that the fault did not grow worse over time. The disturbance remained stable and did not progress into a more severe condition. This example highlights that detection alone is not enough. Real operations require correct localization and an understanding of whether a fault is growing, stable, or self-correcting.

Limitations

- All results depend on the Tennessee Eastman simulator. Real plant data may behave differently and include more noise or unmodeled disturbances.
- Diagnosis using contribution plots becomes more difficult when variables are highly coupled or noisy.
- Prognosis is limited to short-term behavior. Longer-term forecasting may require nonlinear or sequence-based models.
- Domain knowledge was limited. Real operational decisions must be validated by plant engineers and process experts.

Future Work

- Expand the fault-behavior library, which is now included in the appendices, to support maintenance planning and operator training.
- Explore nonlinear methods such as Kernel PCA, autoencoders, and sequence models like LSTMs and Transformers.
- Integrate hybrid physical and statistical models that incorporate plant rules and heuristics.
- Validate the monitoring framework using real industrial data once available.

9. BUSINESS VALUE

Proactive Fault Detection

- Early identification of dynamic and steady-state anomalies reduces the risk of equipment damage, runaway conditions, and off-spec production.
- Prognosis helps determine whether a fault is escalating or stable, giving operators the chance to act before deviations evolve into safety events.
- Consistent early alarms improve product quality and reduce scrap, shutdowns, and lost production.

Process Automation

- Low false alarm behavior prevents alarm fatigue and reduces the monitoring burden on operators.
- Reliable diagnostics support automated workflows and exception-based monitoring, creating a solid foundation for Industry 4.0 and real-time analytics.
- The PCA, DPCA, CVA, and CVDA pipeline functions as a scalable analytics layer that can be integrated into digital twins or advanced control systems.

Operations and Maintenance Planning

- Fast and accurate localization of root-cause variables shortens troubleshooting time and minimizes downtime.
- Stable disturbances, such as Fault 2 in our study, can be safely scheduled for planned maintenance instead of triggering costly unplanned shutdowns.
- A structured fault-behavior library improves consistency across shifts and strengthens long-term predictive maintenance programs.

Emergency Planning

- Prognosis provides clarity on whether a fault is accelerating, holding steady, or self-correcting, supporting faster escalation decisions when needed.
- Better detection and interpretation reduce accident probability by giving teams more time to respond to abnormal operating conditions.
- Improved situational awareness supports safer, more coordinated plant responses during upset scenarios.

Long-Term Operational Excellence

- Integration of the full multivariate monitoring framework enables continuous improvement in reliability, energy efficiency, and quality control.
- The combined methods create a strong foundation for future digital transformation initiatives, from automated reporting to intelligent process optimization.

10. APPENDICES

10.1. References

- Abarbanel, H. D. I. (1996). Analysis of observed chaotic data. Springer.
- Alcalá, C. F., & Qin, S. J. (2009). Reconstruction-based latent-variable fault diagnosis. Journal of Process Control, 19(5), 854–864. https://doi.org/10.1016/j.jprocont.2008.09.006
- Chiang, L. H., Russell, E. L., & Braatz, R. D. (2001). Fault detection and diagnosis in industrial systems. Springer.
- Downs, J. J., & Vogel, E. F. (1993). A plant-wide industrial process control problem. Computers & Chemical Engineering, 17(3), 245–255. https://doi.org/10.1016/0098-1354(93)80018-I
- Jackson, J. E. (1991). A user's guide to principal components. Wiley.
- Kantz, H., & Schreiber, T. (2004). Nonlinear time series analysis (2nd ed.). Cambridge University Press.
- Ku, W., Storer, R. H., & Georgakis, C. (1995). Disturbance detection and isolation by dynamic principal component analysis. Chemometrics and Intelligent Laboratory Systems, 30(1), 179–196. https://doi.org/10.1016/0169-7439(95)00076-3
- Lee, J. M., Yoo, C. K., & Lee, I. B. (2004). Statistical monitoring of dynamic processes based on dynamic PCA and dynamic PLS models. Chemical Engineering Science, 59(7), 1477–1493. https://doi.org/10.1016/j.ces.2003.11.028
- Nomikos, P., & MacGregor, J. F. (1994). Monitoring batch processes using multiway principal component analysis. AIChE Journal, 40(8), 1361–1375. https://doi.org/10.1002/aic.690400809
- Pilario, K. E., Shafiee, M., & Houshmand, M. (2017). Incipient fault detection and diagnosis using canonical variate dissimilarity analysis. Journal of Process Control, 53, 1–14. https://doi.org/10.1016/j.jprocont.2016.12.005
- Qin, S. J. (2003). Statistical process monitoring: Basics and beyond. Journal of Chemometrics, 17(8–9), 480–502. https://doi.org/10.1002/cem.800
- Ricker, N. L. (1996). Decentralized control of the Tennessee Eastman challenge process. Journal of Process Control, 6(4), 205–221. https://doi.org/10.1016/0959-1524(96)00002-5
- Rogers, W. A., & Cooper, D. F. (2010). Fault detection and diagnosis in chemical processes using multivariate statistics. IEEE Transactions on Industrial Electronics, 57(4), 1407–1415. https://doi.org/10.1109/TIE.2009.2024653
- Russell, E. L., Chiang, L. H., & Braatz, R. D. (2000). Data-driven techniques for fault detection and diagnosis in chemical processes. Springer.
- Takens, F. (1981). Detecting strange attractors in turbulence. In D. A. Rand & L.-S. Young (Eds.), Dynamical systems and turbulence (pp. 366–381). Springer.
- Venkatasubramanian, V., Rengaswamy, R., Yin, K., & Kavuri, S. N. (2003). A review of process fault detection and diagnosis: Parts I–III. Computers & Chemical Engineering, 27(3), 293–346. https://doi.org/10.1016/S0098-1354(02)00160-7

10.2. Fault Behavior Library

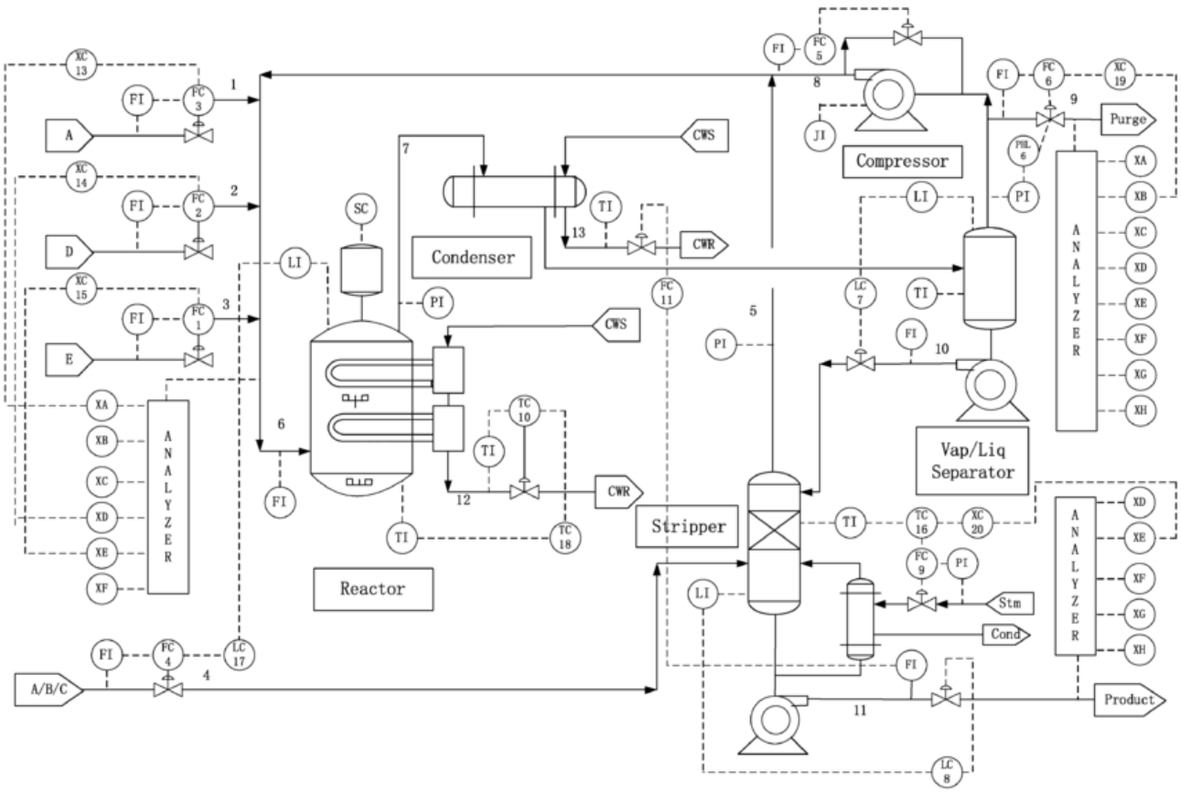
Fault No.	Fault Description	Top Contributing Variables	Implications
1	Feed A/C Ratio Change (Composition)	"XMEAS(23), XMEAS(38) (Product Composition)"	Product Purity is wrong. The final chemical mix is immediately different.
2	Feed B Composition Change	"XMEAS(15), XMV(8) (Stripper Level/Flow)"	The Stripper Level is unstable. The liquid separation unit goes haywire.
3	Feed D Composition Change	"XMV(1), XMEAS(2) (D Feed Flow/Valve)"	D Feed Valve is compensating hard. The controls are frantically adjusting D Feed flow to fix the mix.
4	Feed E Composition Change	XMV(10) (Reactor Cooling-Water Flow)	Reactor is too hot/cold. The main cooling valve has to slam open or shut due to the heat change.
5	Feed A Temperature Variation	XMV(11) (Condenser Cooling-Water Flow)	Condenser is overworked. The main cooling unit's valve works overtime to handle the heat surge.
6	Feed B Temperature Variation	"XMV(3), XMEAS(1) (A Feed Flow/Valve)"	A Feed Valve is compensating. The system is trying to fix the heat issue by changing the A Feed flow.
7	Feed C Temperature Variation	"XMEAS(27) (E Comp.), XMEAS(8) (Reactor Level)"	Reactor Level is moving. The change in density/volume causes the Reactor liquid level to swing.
8	Feed D Temperature Variation	"XMEAS(23) (A Comp.), XMEAS(8) (Reactor Level)"	Reactor Level is moving and the product mix is changing.

9	Condenser Cooling Water Inlet Temp Variation	"XMV(1), XMEAS(2) (D Feed Flow/Valve)"	D Feed Valve is compensating. The system is adjusting D Feed to handle the poor cooling efficiency.
10	Reactor Cooling Valve Stuck (EME)	XMV(10) (Reactor Cooling-Water Flow)	The Stuck Valve screams. The variable that is fixed (XMV(10)) is the clearest sign of its own failure.
11	Condenser Cooling Valve Stuck (EME)	XMV(10) (Reactor Cooling-Water Flow)	Reactor Cooling tries to compensate. The Reactor's cooling valve (XMV(10)) is forced to overwork because the downstream Condenser is broken.
12	A Feed Valve Stuck (EME)	XMEAS(11) (Separator Temperature)	Separator Temperature goes wild. Fixing the largest feed flow destabilizes the separator unit completely.
13	D Feed Valve Stuck (EME)	"XMEAS(16) (Stripper Pressure), XMEAS(23) (A Comp.)"	"Stripper Pressure and Product Mix are wrong. Cannot control D Feed, messing up the separation unit."
14	Separator Liquid Level Valve Stuck (EME)	XMEAS(21) (Reactor Cooling-Water Outlet Temp)	"Reactor Cooling load spikes. The blockage in the separator disrupts the recycle flow, impacting the Reactor's cooling."
15	Stripper Steam Valve Stuck (EME)	"XMV(1), XMEAS(2) (D Feed Flow/Valve)"	D Feed Valve is compensating. The control system adjusts D Feed to counteract the fixed steam/poor separation.
16	Random Variation in Reactor Cooling Temp	"XMV(1), XMEAS(2) (D Feed Flow/Valve)"	D Feed Valve is compensating. The system adjusts D Feed to manage the sudden heat fluctuations in the reactor.
17	Random Variation in Condenser Cooling Temp	XMEAS(21) (Reactor Cooling-Water Outlet Temp)	The Reactor feels the heat. The cooling change at the Condenser comes back via the recycle stream to affect the Reactor.
18	Random Variation in Feed A Composition	"XMEAS(11) (Separator Temp), XMEAS(22) (Condenser C/W Outlet Temp)"	Separator/Condenser are disrupted. The changing mix in Feed A causes temperature swings in the downstream separation units.
19	Random Variation in Feed B Composition	"XMV(1), XMEAS(2) (D Feed Flow/Valve)"	D Feed Valve is compensating. The control system adjusts D Feed to balance the effects of the changing Feed B mix.
20	Random Variation in Feed A Temperature	XMV(5) (Compressor Recycle Valve)	"Recycle Gas Valve is swinging wildly. The temperature change in Feed A changes the gas flow, making the pressure control valve (XMV(5)) react instantly."

10.3. Detection Methods Specifications

Method	Original Variables	Intermediate Variables (After Lags)	Final Components Kept	Explained Variance
PCA	52	N/A	23	90.46%
DPCA	52	156	46	90.36%
CVA	52	260	20	N/A
CVDA	52	N/A	N/A	N/A

10.4.Fault Types



Fault No.	Description	Fault Type
1	Step change in A/C feed ratio (Stream 4 composition)	Process Fault
2	Step change in Feed B composition (Stream 2)	Process Fault
3	Step change in Feed D composition (Stream 3)	Process Fault
4	Step change in Feed E composition (Stream 1)	Process Fault
5	Feed A temperature variation	Process Fault
6	Feed B temperature variation	Process Fault
7	Feed C temperature variation	Process Fault
8	Feed D temperature variation	Process Fault
9	Condenser cooling water inlet temperature variation	Process Fault
10	Reactor cooling water valve stuck	Mechanical Fault
11	Condenser cooling water valve stuck	Mechanical Fault
12	Feed A valve stuck	Mechanical Fault
13	Feed D valve stuck	Mechanical Fault
14	Separator liquid level valve stuck	Mechanical Fault
15	Stripper steam valve stuck	Mechanical Fault
16	Random variation in reactor cooling water temperature	Stochastic Fault
17	Random variation in condenser cooling water temperature	Stochastic Fault
18	Random variation in Feed A composition	Stochastic Fault
19	Random variation in Feed B composition	Stochastic Fault
20	Random variation in Feed A temperature	Stochastic Fault

10.5.Detection Results Summary

Fault	PCA T2 DR	PCA Q DR	PCA T2 Delay	PCA Q Delay	DPCA T2 DR	DPCA Q DR	DPCA T2 Delay	DPCA Q Delay	Enhanced T2 DR	Enhanced Q DR	Enhanced T2 Delay	Enhanced Q Delay	CVA T2 DR	CVA Q DR	CVA T2 Delay	CVA Q Delay	CVDA DR	CVDA Delay
1	0.018	0.971	50	18	0.947	0.785	25	12	0.089	0.98	34	15	0.963	0.622	21	22	0.054	74
2	0.939	0.012	43	64	0.943	0.008	41	62	0.944	0.049	42	33	0.925	0.001	50	16162	0.009	824
3	0	0	–	–	0	0	–	1448	0	0	–	–	0	0	–	–	0.005	945
4	0	0.026	–	87	0	1	–	0	0	1	–	0	0	0	–	–	0.006	746
5	0	0	–	–	0	0.004	–	504	0	0	–	–	0.92	0.984	45	7	0.238	2
6	0.36	0.981	48	2	0.436	1	44	0	0.455	0.998	45	2	0.828	0.775	12	12	1	0
7	0.22	0.007	10	3	0.998	0.997	1	0	0	1	–	0	0.001	0.021	13189	73	0.009	114
8	0.018	0.044	85	55	0.041	0.035	83	74	0.042	0.098	82	54	0.032	0.048	87	322	0.036	196
9	0	0	–	–	0	0.001	–	5437	0	0	–	–	0	0	–	–	0.007	863
10	0	0	–	–	0	0.001	–	1449	0	0	–	–	0	0.001	–	3577	0.015	880
11	0.002	0.189	146	6	0.02	0.602	27	3	0	0.557	–	5	0	0	–	–	0.053	11
12	0.002	0.167	20	11	0.008	0.433	19	9	0	0.23	–	11	0.021	0.283	11	7	0.238	4
13	0.031	0.425	116	35	0.252	0.376	43	31	0.126	0.577	104	34	0.338	0.199	41	339	0.023	338
14	0.531	0.611	1	3	0.749	0.742	0	0	0	0.009	–	22	0	0	–	–	0.007	1607
15	0	0	–	–	0	0	–	4572	0	0	–	–	0	0	–	–	0.006	743
16	0	0	–	–	0	0.001	–	594	0	0	–	–	0	0.151	–	43	0.242	38
17	0.373	0.437	81	79	0.435	0.169	79	92	0.414	0.347	79	89	0.284	0.009	77	151	0.074	88
18	0.047	0.095	162	152	0.073	0.093	154	156	0.081	0.064	153	294	0.069	0.118	155	147	0.075	68
19	0	0	–	5813	0	0.001	–	460	0	0.001	–	5815	0	0	–	–	0.007	631