Systems & Control Transactions

Original Research Article
Peer Reviewed Conference Proceeding
ESCAPE 36 - European Symposium on Computer Aided Process Engineering
Sheffield, UK. 21-24 June 2026

PSE PRESS

Decentralized Causal Monitoring in High-Dimensional Systems: Revealing the Topological Drivers behind Fault Detection Performance

Rodrigo Paredes[a] and Marco S. Reis[a*]

[a] University of Coimbra, CERES, Department of Chemical Engineering, 3030-790 Coimbra, Portugal
* Corresponding Author: marco@eq.uc.pt.

ABSTRACT

Centralized monitoring methods experience reduced fault detection sensitivity in large-scale industrial systems due to the masking effect arising from the aggregation of many interconnected variables. Decentralized monitoring, where variables are grouped into subsystems, has been shown to effectively address these limitations. However, the performance of this class of methods critically depends on how the network is partitioned, and the role of its structural factors on fault detection remains poorly understood. This work studies how network topology and causal structure affect decentralized monitoring in high-dimensional systems. Using *SimCaNet*, a DAG-based data simulator, where large-scale systems with 100-1000 variables were generated, we rigorously compared the performance of centralized and decentralized causal log-likelihood monitoring methods under process perturbations and sensor bias faults. Network partitioning is performed using the Leiden community detection algorithm and characterized at the network, community, and node levels. Fault Detection Sensitivity (AUC_{NORM}) is quantified per variable, using the normalized area under the curve, which is computed from the true positive rate profiles across fault magnitudes. Results show that AUC_{NORM} is primarily driven by the number and the size of communities across fault types. Smaller communities consistently achieve higher AUC_{NORM} than larger communities, while excessive fragmentation or high modularity degrade performance. Community density and inter-community coupling further enhance fault detection sensitivity. The strength of node-level causal relationships explains most of the variability observed at root nodes for sensor bias faults and their robustness to information loss. Furthermore, ablation studies confirm the robustness of distributed systems to major disruptions in the system: more than 32% of community information can be removed without significantly impacting the fault detection sensitivity. These topology-driven partitioning findings constitute novel and valuable contributions to the theoretical foundations for designing better distributed monitoring systems in complex industrial plants.

Keywords: Fault Detection, Modelling and Simulations, Industry 4.0, Big Data, Decentralized Monitoring, Network Topology, Structural Causal Models, Community Detection

INTRODUCTION

Fault detection in large-scale industrial systems remains a significant challenge with direct implications for safety, efficiency, and operational reliability. As modern processes scale in size and complexity, hundreds or thousands of interdependent variables must be monitored simultaneously. In such high-dimensional settings, small and localized faults are often statistically diluted within global monitoring indices, leading to delayed detection or complete loss of sensitivity [1, 2].

Decentralized monitoring has emerged as a promising strategy to mitigate this limitation, by decomposing large systems into smaller groups monitored locally. By reducing dimensionality and focusing on localized dynamics, decentralized schemes can detect faults before they propagate across the network [3, 4]. However, the effectiveness of decentralization critically depends on system partitioning, where poorly designed partitions can replicate the limitations of centralized monitoring or

introduce excessive false alarms.

Fault propagation and its proper diagnostics are inherently governed by causal relationships between process variables. This motivates the use of causal network representations for monitoring, where directed interactions encode how disturbances propagate through the system [5-7]. While causal monitoring frameworks provide a principled basis for understanding fault propagation, their performance in high-dimensional systems remains limited.

Recent advances in network science have shown that the topological structure of complex systems shapes information flow [8, 9]. Properties such as community structure, internal connectivity, modularity, and node centrality have been shown to influence system dynamics. Fault detection sensitivity in large-scale systems is not only a statistical problem, but also a structural one. Despite this, the role of network topology in decentralized fault detection has not been systematically addressed or characterized. It is still unclear which topological properties govern the sensitivity gains and under what conditions decentralization may fail. Consequently, current monitoring strategies lack topology-aware design principles, limiting their application, scalability, and robustness.

To address this gap, in this work we provide a quantitative analysis of how network topology and causal structure jointly affect decentralized monitoring in high-dimensional systems, with the building blocks described next. A graph-based causal data simulator generates networks reflecting structural characteristics commonly observed in industrial processes. Two representative fault classes are considered: process perturbations and sensor bias faults. Network partitions are obtained using the Leiden algorithm, a modularity-based community detection method, allowing controlled variation of partition size and structure [10]. Monitoring performance is evaluated using a Fault Detection Sensitivity metric that captures node-level fault detection across fault magnitudes, enabling fair comparison between centralized and decentralized schemes.

Our results reveal a consistent relationship between fault detection sensitivity and topological descriptors, including community size, internal density, modularity, and node centrality. Decentralized causal monitoring systematically outperforms centralized approaches in large-scale networks. However, suboptimal partitions may reduce these benefits, highlighting the need to consider structural trade-offs. These findings provide fundamental insights to the systematic design of topology-aware monitoring and demonstrate that fault detection in high-dimensional systems depends fundamentally on network topology.

METHODOLOGY

This section presents the framework used to assess fault detection sensitivity in large-scale systems. The methodology combines a high-dimensional data simulator, centralized and decentralized monitoring schemes, network partitioning, and a unified fault detection sensitivity metric. The goal is to isolate the impact of network topology, partition structure, and fault type on fault detection performance under fully controlled conditions.

Network Data Simulator and Fault Types

To assess fault detection sensitivity in large-scale systems, a high-dimensional causal data simulator, called SimCaNet, is used. Existing benchmark simulators for process monitoring are typically limited to low-dimensional settings, which hinders systematic analysis of how network topology affects fault detection sensitivity. Although latent-variable models can handle high-dimensional data, they ignore explicit causal relationships, and are therefore less sensitive and have more limited diagnostic capabilities [7, 11].

SimCaNet generates synthetic datasets using Directed Acyclic Graphs (DAGs) implemented as Structural Causal Models (SCMs) [12]. Each variable is defined as a function of its causal parents, ensuring understandable propagation mechanisms and realistic fault patterns.

Network structures are generated using a preferential-attachment mechanism based on the Barabási-Albert model, producing heterogeneous, hub-dominated topologies closely related to large-scale industrial systems. This choice reflects the non-uniform connectivity and consequently the asymmetric propagation paths, commonly observed in practice. All network structures preserve the DAG property to ensure causal consistency.

Data are generated under Normal Operating Conditions (NOC) following a linear and static SCM formulation, with additive measurement noise to emulate realistic observational conditions. This simplification enables focusing on the identification of critical topological features driving fault detection sensitivity, without the interference of other factors, such as non-linearity and multiscale dynamics. Structural parameters are generated once for each network and kept fixed across experiments, so that observed variations in monitoring performance can be attributed to network topology, partitioning, and fault characteristics.

Two representative fault classes are considered, as summarized in Table 1. Process perturbations are modeled as step changes in root variables (data drift), inducing process disturbances that propagate through the causal network according to its topology. Sensor bias faults are modeled as additive biases affecting only local measurements, without altering the underlying process. Fault severity is controlled through a fault magnitude

parameter proportional to the variability of the faulty variable.

Table 1: Types of faults implemented in SimCaNet.

Type of fault	Meaning	Modified factor
Process Perturbation	A step change in root nodes, representing e.g., abnormal (but real) variation in raw materials	$\mu_i = \mu_{NOC,i} + k\sigma_i$
Sensor bias	Biased readings caused by a fault in the sensors	$X_i = X_i + k\sigma_{e_i}$

Large-scale Causal Monitoring Methods

This section presents the two causal-based Statistical Process Monitoring methods used in this work to evaluate how network dimensionality and structure affect fault detection sensitivity. Both approaches follow the standard SPM pipeline, including phases of process stability analysis, calculation of monitoring statistics, and fault detection of a new observation via control limits.

Centralized Causal Log-likelihood monitoring

The Centralized Causal Log-likelihood monitoring (CNc-MSPC) method is based on log-likelihood indices and was adopted as the benchmark for the fault detection performance study. In ref. [13], this methodology was compared with the classical PCA-MSPC approach using both a simulated dataset and an industrial semiconductor manufacturing dataset. The results showed that the causal approach achieved superior fault diagnosis, and better or at least similar detection performance.

In practice, the causal network structure is inferred through causal discovery methods [14]. Here, the causal graph is assumed to be known, as the purpose is to study the impact of network topology (and not discovery accuracy).

Each variable X_i is modelled through a SCM as a function of its causal parents, $\mathrm{Pa}(X_i)$. These yields, $X_i = f(\mathrm{Pa}(X_i)) + e_i$, where the conditional distribution of $X_i|\mathrm{Pa}(X_i)$ corresponds to the error term distribution, e_i.

A local log-likelihood index is computed for each observation *j*. Assuming a normal distribution, the log-likelihood for an individual observation $L_i^L(j)$ is defined as:

$$L_i^L(j) = -\frac{\ln(2\pi)}{2} - \frac{\ln(\hat{\sigma}_{e_i}^2)}{2} - \frac{(e_i(j)-\mu_{e_i})^2}{2\hat{\sigma}_{e_i}^2} \quad (1)$$

where $e_i(j)$ is the error term of variable X_i conditioned onto its causal parents, and $\hat{\mu}_{e_i}$ and $\hat{\sigma}_{e_i}$ are the sample mean and standard deviation of e_i calculated from a reference dataset corresponding to normal operating conditions (NOC).

A global log-likelihood index $L^G(j)$ is then obtained by aggregating the local indices across all nodes. Leveraging on the chain rule, the joint (or global) likelihood factorizes into the product of the conditional likelihoods. Therefore, applying logarithms, the global log-likelihood, $L^G(j)$, is obtained in terms of the individual log-likelihoods, as follows:

$$L^G(j) = \sum_{i=1}^{n_{variables}} L_i^L(j) \quad (2)$$

Control limits for fault detection are established using the values of L^G obtained from a validation dataset. A kernel density estimation (KDE) is fitted to approximate the probability density function of this global log-likelihood statistic [14]. The cumulative distribution function (CDF) of the KDE is then used to determine the Lower Control Limit (LCL) by solving the following equation for a predefined Type I error rate, α:

$$\int_{-\infty}^{\mathrm{LCL}} \hat{f}_{KDE}(L^G)\, \mathrm{d}L^G = \alpha \quad (3)$$

where, $\hat{f}_{KDE}(L^G)$ is the estimated probability density function. Since high log-likelihood values indicate proximity to the reference distribution, only the Lower Control Limit (LCL) is needed for fault detection. This threshold is used for monitoring future observations.

Decentralized Causal Log-likelihood monitoring

The Decentralized Causal Log-likelihood monitoring (CNd-MSPC) method also starts by inferring the causal network. Then, the variables are divided into disjoint communities (groups or clusters) using a community detection algorithm that considers network topology and modularity. Such grouping into strongly connected communities effectively enhances the fault detection sensitivity. However, the selection of the partition and consequently its quality directly impacts the monitoring performance.

Given a network partition, a global log-likelihood metric (Equation 2) and the control limits are computed for each community (sub-level monitoring), denoted as L_c^G for $c = 1, 2, \ldots, n_{communities}$, where $n_{communities}$ is the total number of communities.

To aggregate information from multiple communities, an OR-gate approach is adopted, meaning that a global warning is issued as soon as at least one community-level out-of-control signal is produced.

Each community's control limits are set independently, with the significance level adjusted to maintain the overall false alarm rate across multiple monitoring statistics. To this end, the Šidák correction is applied, setting the significance level for each community as $\alpha_c = 1 - (1-\alpha)^{1/n_{communities}}$ [15]. This ensures that the global alarm reflects true anomalies while controlling the overall Type I error rate (also known as the Family-Wise Error Rate, FWER). CNd-MSPC is particularly well-suited for

Table 2: Network-, community-, and node-level descriptors used to characterize the structural, statistical, and causal properties of each potentially faulty variable for a fixed network and partition.

Feature	Description	Level
Network Size	Total number of nodes in the network	Network
Modularity	Quantifies the quality of the network's division into communities	Network
Number of communities	Total number of communities in a partition	Community
Number of nodes in the community	Total number of nodes in the community of the faulty variable *i*	Community
Density	Quantifies the intra-connectivity in the community of the faulty variable *i*	Community
Conductance	Measures the fraction of edges leaving the community of the faulty variable *i* relative to all edges within and leaving the community	Community
Cohesiveness	Quantifies the maximum conductance among all connected subsets (size >2) of the community of the faulty variable *i*, capturing the weakest internal connectivity within the community	Community
Separability	Measures the ratio of external edges to the internal edges in the community of the faulty variable *i*	Community
Clustering Coefficient	The average clustering of nodes within the community containing the faulty variable *i*, quantifying the tendency to form tightly connected triplets	Community
Normalized Causal Strength	Quantifies the relative causal influence of the faulty node *i* on its children, normalized by the total parents' influence on each affected child, based on SCM coefficients	Node
In-degree	Number of incoming edges to faulty node *i*, indicating how many parent nodes directly influence it	Node
Out-degree	Number of outgoing edges from node *i*, indicating how many child nodes it directly influences	Node
Is root-node?	Binary indicator for the faulty node *i*	Node
Degree centrality	Fraction of nodes that are connected to the faulty node *i*	Node
Closeness centrality	Measures how easily other vertices can be reached from the faulty node *i*	Node
Betweenness centrality	A measure of how much the faulty node *i* acts as an intermediary in a shortest path between the other node pairs	Node
Eigenvector centrality	Measures the influence of faulty node *i* based on the influence of its neighbors	Node
Page Rank	Measures the influence of faulty node *i* based on the influence of incoming neighbors, adjusted by the damping factor α	Node
Shortest Path to Centroid	Measures how close faulty node *i* is to the three most degree central nodes of the network	Node

high-dimensional networks, where centralized monitoring often becomes statistically insensitive to localized faults. Although the offline phase (causal discovery, causal inference, process stability analysis) scales with network dimensionality, the online monitoring phase is computationally efficient.

Network Partitioning and Topological Descriptors

Partitioning the inferred causal graph is a crucial step in CNd-MSPC, as the resulting communities define the structural units used for decentralized monitoring. To generate these partitions, community detection algorithms are employed, leveraging the graph representation of the process. A broad range of algorithms is

Table 4: Summary of network partitions used in the Fault Detection Sensitivity study for Process Perturbation (PP) and Sensor Bias (SB) faults.

Network	Partition	#Communities	Modularity
Network 1	Centr.	1	-
	#1	4	0.365
	#2	7	0.417
	#3	15	0.397
	#4	21	0.358
	#5	29	0.310
	#6	41	0.258
Network 2	Centr.	1	-
	#1	11	0.374
	#2	31	0.325
	#3	46	0.292
	#4	84	0.244
	#5	144	0.175
	#6	191	0.122
Network 3	Centr.	1	-
	#1	5	0.347
	#2	11	0.364
	#3	28	0.343
	#4	46	0.312
	#5	99	0.257
	#6	179	0.201
Network 4	Centr.	1	-
	#1	7	0.341
	#2	12	0.348
	#3	37	0.322
	#4	60	0.304
	#5	183	0.230
	#6	391	0.157
			Total of tests:

available, including the Girvan-Newman and Louvain methods, which differ in the principles used to uncover community structure. In this work, the communities were generated through the Leiden algorithm.

The Leiden method improves on the Louvain method by addressing its main limitation: after the aggregation step, the Louvain method can yield communities that are internally disconnected or weakly connected. As for the Louvain method, the Leiden method also optimizes modularity, Q, but introduces a refinement step that guarantees connectivity within each community.

$$Q = \frac{1}{2\mathrm{m}} \sum_{\mathrm{i,j}} \left(w_{ij} - \gamma \frac{k_i k_j}{2\mathrm{m}} \right) \delta(c_i, c_j) \qquad (4)$$

In Equation 4, modularity function is represented where *m* is the number of graph edges, γ is the resolution parameter controlling the granularity of identified communities, w_{ij} is the edge weight between nodes *i* and *j*, k_i and k_j are their respective degrees, and $\delta(c_i, c_j)$ is a community indicator function (1 if *i* and *j* belongs to the same community, *i.e.*, $c_i = c_j$, and 0 otherwise).

The Leiden method tends to produce communities that are densely connected internally while minimizing connections between communities.

Topological Descriptors

For a given network and a fixed partition, fault detection sensitivity may vary across fault locations due to the differences in the structural and causal embedding of variables within the network. To systematically analyze these effects, each faulty variable is characterized using a set of network-, community-, and node-level descriptors, summarized in Table 2. These descriptors are subsequently used for exploratory data analysis and pairwise correlation studies, together with the Fault Detection Sensitivity metric, to identify structural properties that drive fault detection under decentralized monitoring.

Fault Detection Sensitivity Assessment Framework

We developed a unified evaluation framework used to compare fault detection performance across all network sizes, partitioning strategies, fault types, and monitoring schemes. The framework ensures that all results are directly comparable, enabling the analysis of the effects of network topology and partitioning on fault

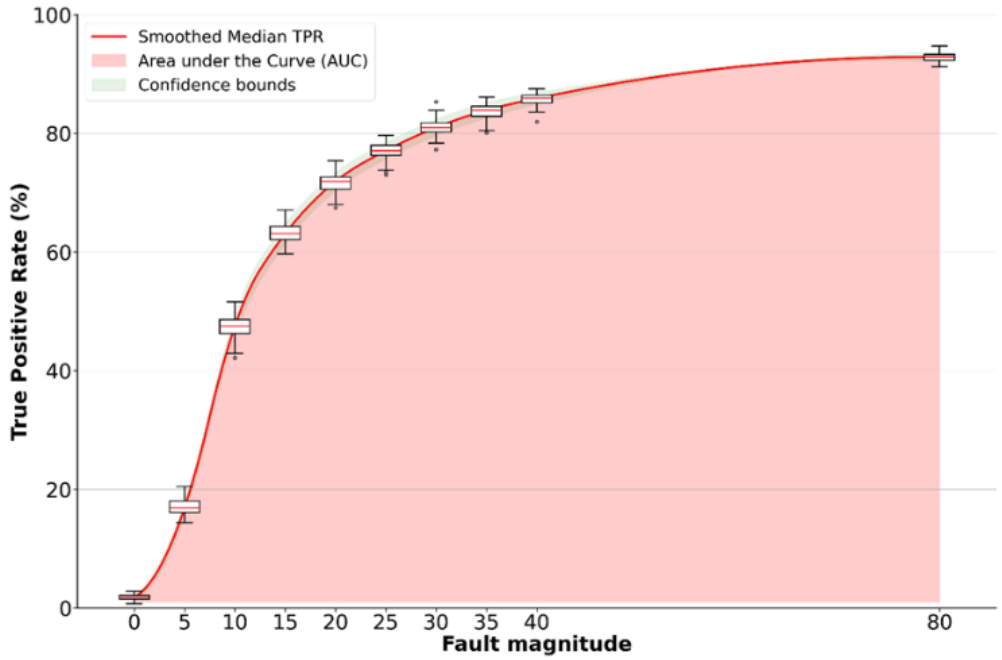

Figure 1. Fault detection sensitivity assessment for a sensor bias fault injected in a single network variable. Box plots show the distribution of True Positive Rate values for each fault magnitude. The red curve represents the median TPR versus magnitude, and the shaded red area corresponds to the AUC.

detection sensitivity.

For each network, both centralized and decentralized monitoring schemes are evaluated. Different partition granularities are obtained by varying the resolution parameter, γ, while centralized monitoring (i.e., no partitioning) is used as a benchmark configuration. Some descriptors of the tested networks are summarized in Table 3.

Table 3: Characterization of the four networks under study.

Descriptor	Network 1	Network 2	Network 3	Network 4
# of nodes	100	300	500	1000
# of edges	212	875	1564	3880
# of root nodes	48	152	277	333
Graph density	0.0214	0.0098	0.0063	0.0039

For process faults, perturbations are injected exclusively at root nodes. Ten fault magnitudes are considered, with a fixed magnitude range across all networks, but that is adapted to each fault type. To account for stochastic variability, each magnitude is simulated 50 times (independent replicates, each consisting of 1000 observations). The signal-to-noise ratio for the additive noise is fixed at 10 dB.For sensor bias faults, the same evaluation pipeline implemented for process perturbation faults is applied, with faults injected independently at each network variable. All monitoring schemes are calibrated to operate at a fixed false alarm rate of α = 0.05 using independent fault-free validation datasets.

Fault detection sensitivity performance is quantified using a Normalized Area Under the Curve (AUC_{NORM}) metric. For a given faulty variable (see the Figure 1 as an example), the True Positive Rate (TPR) is computed for each fault magnitude and replicate. The median TPR across replicates is then calculated for each magnitude, yielding a single TPR vs. Fault magnitude curve. The area under this curve defines the AUC, which summarizes the sensitivity of the monitoring scheme across fault magnitudes. To ensure comparability across configurations, AUC values were normalized using a max-min approach, where the maximum AUC corresponds to perfect detection (TPR = 100% for all magnitudes) and the minimum was equivalent to the false positive rate (FPR) when no faults were introduced.

The evaluation pipeline applies this framework to each network and partition, as illustrated in Figure 2. The workflow starts with network selection and data generation under normal operating conditions, followed by partitioning for decentralized monitoring and control limit estimation. The TPR vs Fault magnitude curve is then computed for each variable under the selected partition, and the corresponding AUC_{NORM} is obtained. This procedure yields a single fault detection sensitivity score per variable, per network, and per partition.

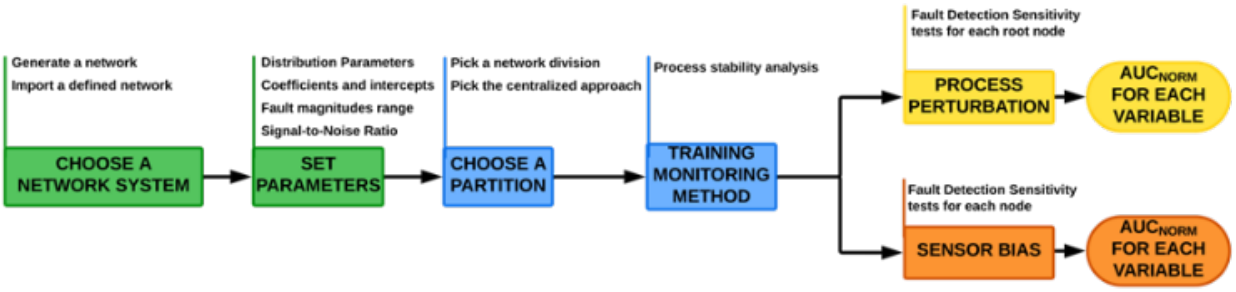

Figure 2. Workflow for Fault Detection Sensitivity evaluation.

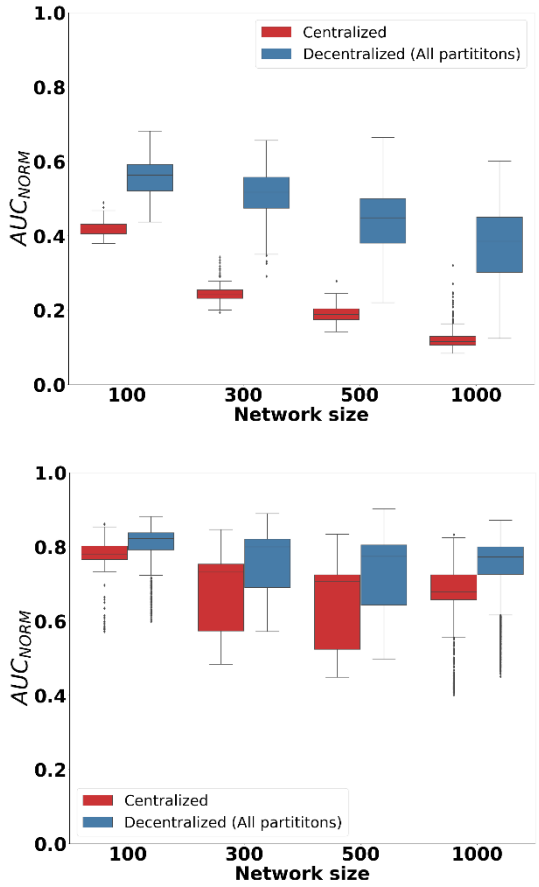

Figure 3. Comparison of AUC_{NORM} metric distributions obtained with centralized and decentralized monitoring schemes across different network sizes for process perturbation (top) and sensor bias (bottom) faults. Box plots aggregate AUC_{NORM} values over all faulty nodes and all partitions.

For each network size, six decentralized partitions are considered in addition to the centralized configuration. These partitions are selected to span a broad range of community sizes and modularity values, allowing a controlled analysis of the effect of the resolution parameter γ on fault detection sensitivity. Table 4 summarizes the resulting network partitions, reporting the number of communities, modularity values, and the number of faulty variables considered for both Process Perturbation and Sensor Bias faults.

RESULTS AND DISCUSSION

This section analyzes how network partitioning and topological descriptors impact fault detection sensitivity (AUC_{NORM}). Rather than focusing on absolute performance gains, the results aim to identify the structural factors that condition and drive fault detectability and that justify poor performances. Specifically, we show that fault detection is primarily governed by the size and number of communities, followed by internal and external connectivity, and finally by node-level causal factors.

Figure 3 compares the AUC_{NORM} distributions obtained with centralized and decentralized monitoring approaches across all network sizes. For both fault types, centralized monitoring exhibits a clear degradation in fault detection sensitivity as network size increases. This behavior reflects the well-known loss of sensitivity to localized faults in high-dimensional scenarios. In contrast, decentralized monitoring consistently attenuates the AUC_{NORM} degradation across all networks and partitions. The gain is more pronounced for process perturbation faults. Sensor bias faults also benefit from decentralization, although with a smaller relative improvement.

Figure 4 illustrates the relationship between AUC_{NORM} and the *number of nodes in the community* containing the faulty variable for both fault types. A strong and consistent trend is observed, with fault detection sensitivity decreasing monotonically as community size increases. This relationship holds across fault types and network sizes, indicating that community size acts as a first-order control parameter for fault detectability.

Figure 5 explores the relationship between partition *modularity* and fault detection sensitivity for process perturbation faults. Contrary to common assumptions, higher modularity does not systematically lead to improved fault detection performance. In several

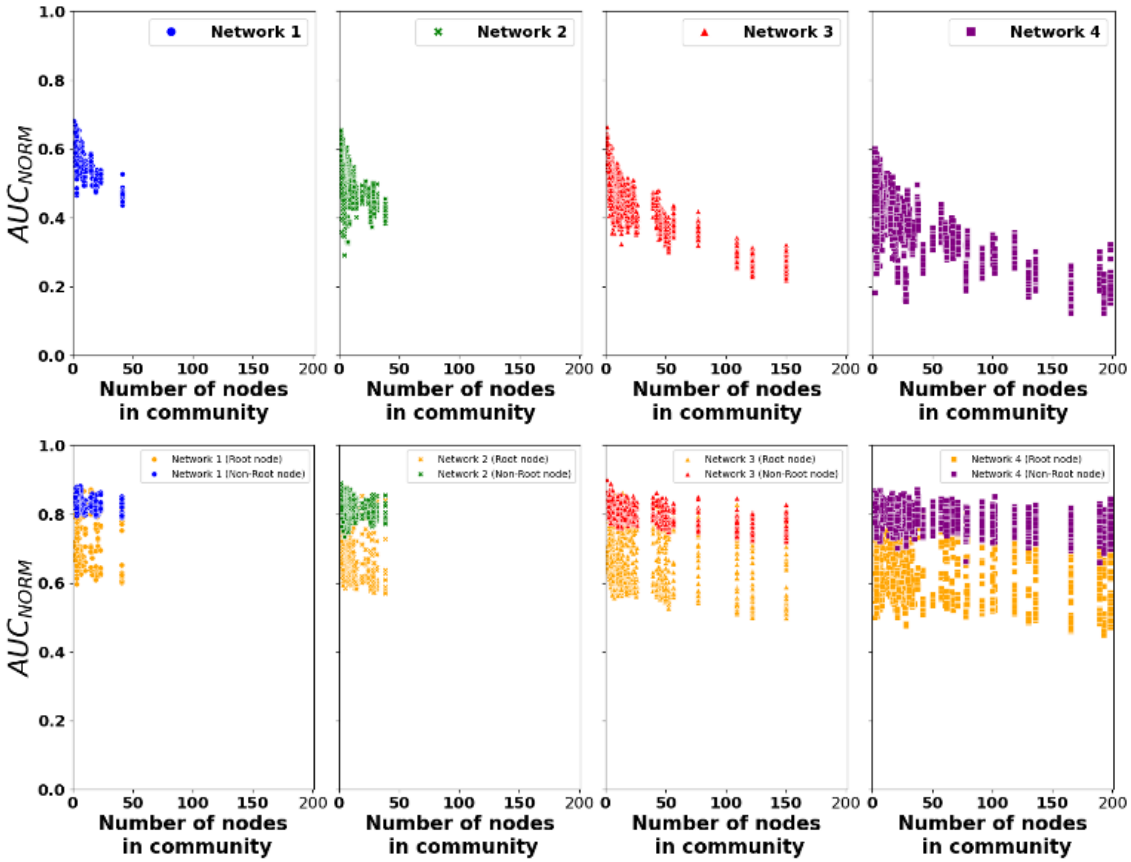

Figure 4. AUC_{NORM} vs. Number of nodes in community containing the faulty node for process perturbation (top) and sensor bias (bottom) faults. Each point represents the AUC_{NORM} for each faulty node.

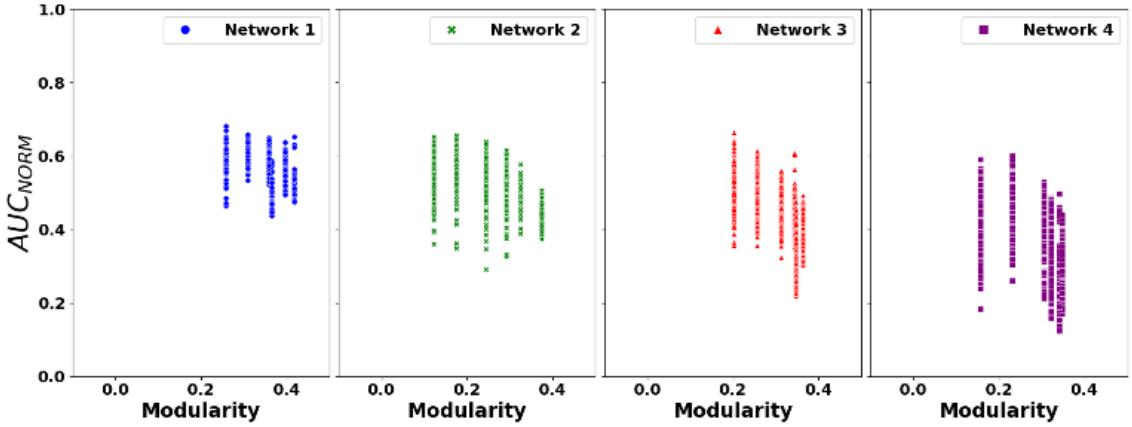

Figure 5. AUC_{NORM} vs. Modularity of the tested partition for process perturbation faults.

regimes, partitions with higher modularity exhibit lower AUC_{NORM} scores. This highlights the existence of an optimal range of modularity, that must be considered when establishing design principles for decentralized monitoring. Maximizing modularity alone may therefore yield structurally isolated communities that are suboptimal for monitoring. Sensor bias follows the same trend but is not present here due to space constraints.

The impact of internal community connectivity is analyzed in Figure 6, which relates AUC_{NORM} to the *density* of the community containing the faulty node for process perturbation faults. Higher internal density is associated with increased fault detection sensitivity, indicating that dense communities better preserve the statistical dependencies. A similar qualitative trend was observed for sensor bias faults. These findings reinforce that effective

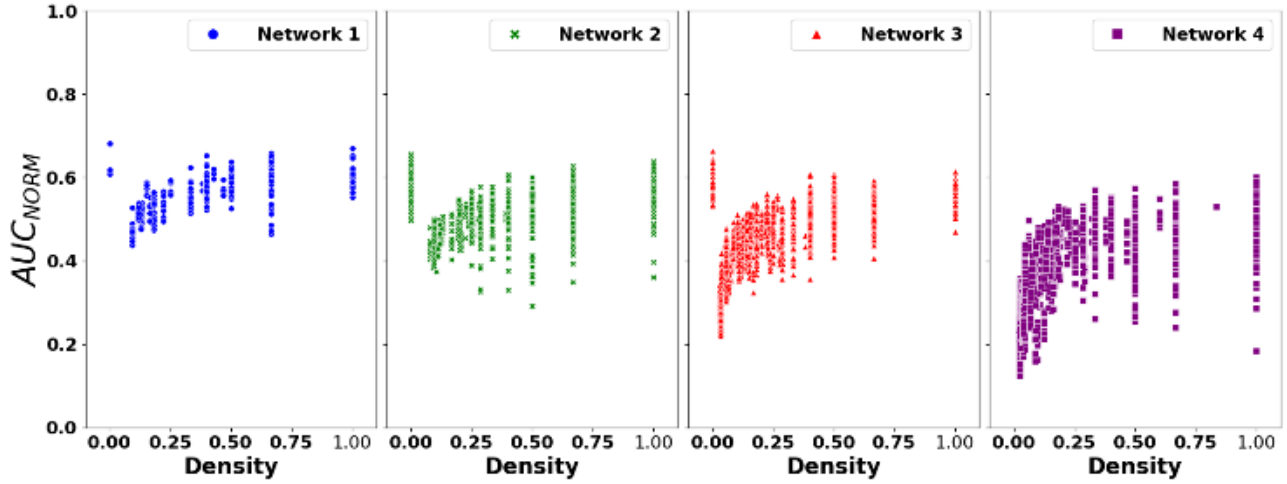

Figure 6. AUC_{NORM} vs. Density in the community containing the faulty node for process perturbation faults.

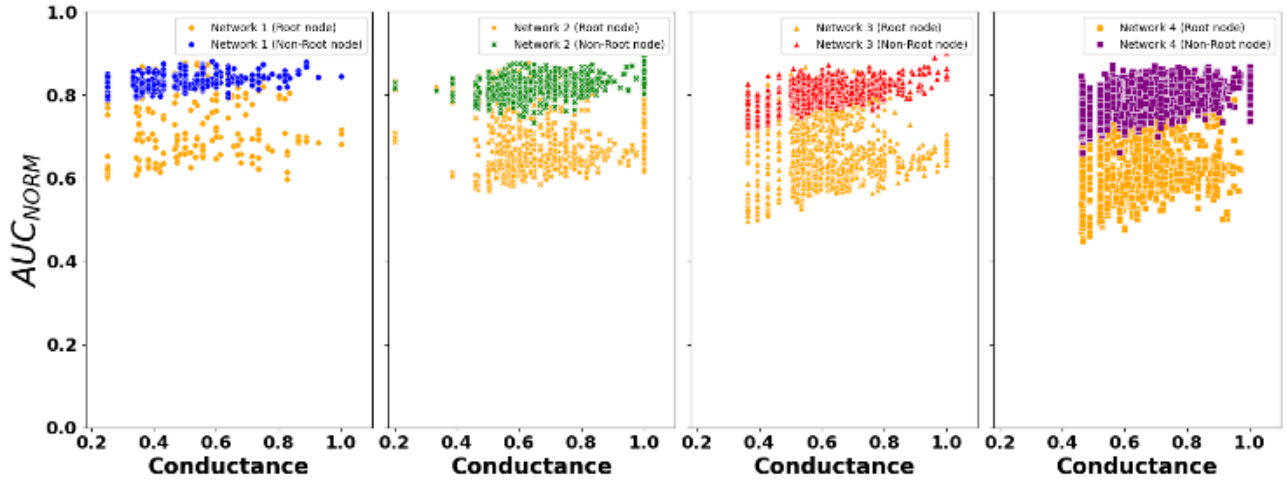

Figure 7. AUC_{NORM} vs. Conductance in the community containing the faulty node for sensor bias faults.

decentralized monitoring requires a balance between dimensionality reduction and sufficient internal connectivity.

Figure 7 focuses on the role of *conductance* for sensor bias faults. Communities with low conductance, corresponding to communities well-isolated from the rest of the network, tend to exhibit lower AUC_{NORM} values. Excessively isolated communities hinder the fault propagation, limiting the monitoring performance. It emphasizes that fault detection is not a purely local problem, but that an information exchange between communities is essential to preserve sensitivity. The same trend is observed in process perturbation faults.

Together, density and conductance reveal a fundamental trade-off: effective decentralized monitoring requires communities that are internally dense while remaining sufficiently coupled to the rest of the network to allow fault information to propagate.

In sensor bias faults, root-nodes lead to a broad range of AUC_{NORM} values, indicating that topological descriptors alone are insufficient to fully explain fault detection performance. Figure 8 relates AUC_{NORM} to the normalized causal strength ($\beta_{NORMALIZED}$) of the faulty node for sensor bias faults. Nodes with stronger causal influence on the network exhibit significantly higher detection sensitivity, even when community-level properties are comparable. This explains why purely topological descriptors fail to fully predict fault detectability and motivates the inclusion of causal metrics in topology-aware monitoring design.

Figure 9 presents an ablation study in which community-level information is selectively removed for a sensor bias fault affecting a representative node. Fault detection performance deteriorates sharply only when the community containing the faulty node is removed, while the exclusion of unrelated communities has a limited impact on sensitivity. For instance, in process perturbation faults, excluding up to 15 non-faulty communities

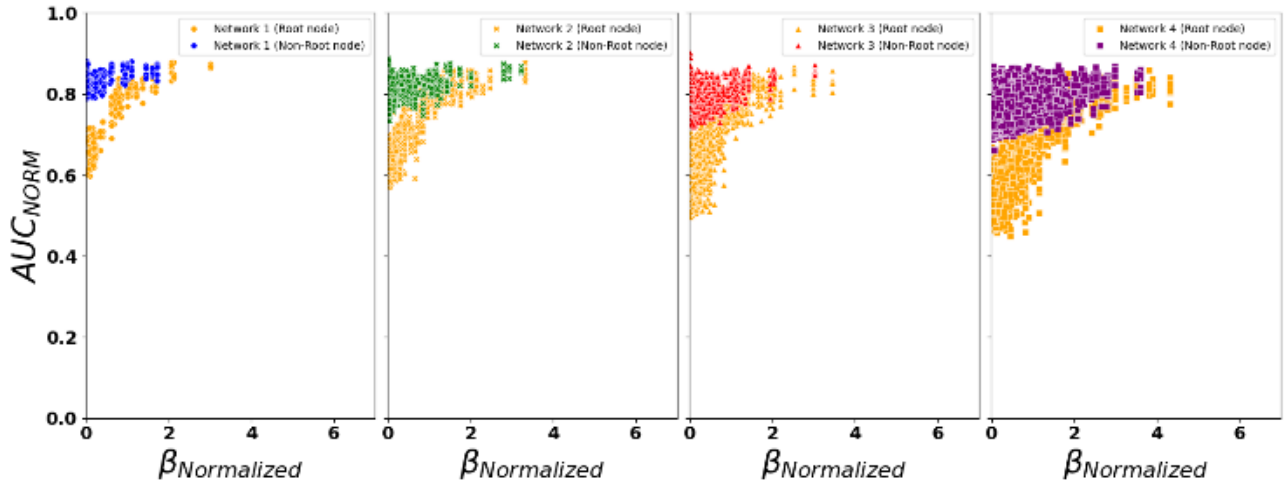

Figure 8. AUC_{NORM} vs. Normalized Causal Strength ($\beta_{NORMALIZED}$)of the faulty node for sensor bias faults.

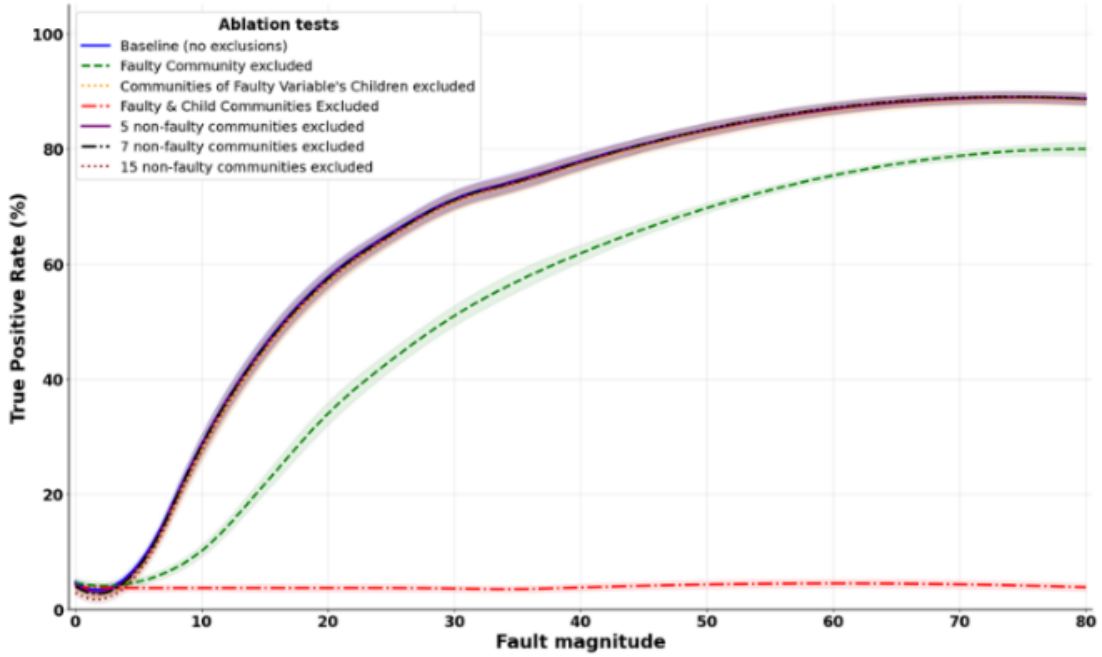

Figure 9. Median True Positive Rate vs Fault magnitude for sensor bias faults affecting node *X226* in Network 3, Partition #4, under different community-level information loss scenarios.

(representing over 32% of network communities) results in only marginal reductions in TPR at low fault magnitudes. This confirms that decentralized monitoring relies primarily on local causal information and is inherently robust to partial observability, a critical requirement for scalable industrial monitoring.

CONCLUSIONS

This study solidly establishes that decentralized causal monitoring substantially enhances fault detection sensitivity in high-dimensional networks compared to centralized approaches. The improvement is statistically significant ($p < 0.00001$) for both process perturbation and sensor bias faults, confirming that network partitioning itself provides a robust mechanism for mitigating the well-known loss of sensitivity in large-scale systems.

Community size and the number of communities emerge as the primary determinants of fault detection: smaller communities preserve sensitivity by limiting aggregation dilution, whereas excessively large communities or overly modular partitions reduce performance. Other topological drivers include internal connectivity and inter-community metrics. Higher community density,

conductance, and separability consistently improve fault detection, while excessive isolation impairs fault detection sensitivity.

Node-level causal influence, quantified by the normalized causal strength ($\beta_{NORMALIZED}$), explains residual variability in the root nodes. For sensor bias faults, root nodes with $\beta_{NORMALIZED}$ < 1.6 exhibit markedly reduced sensitivity, whereas higher $\beta_{NORMALIZED}$ secure high sensitivity through residual propagation to descendants.

Ablation studies further confirm the robustness of decentralized monitoring. Fault detection relies primarily on the local community containing the faulty node. Minimal degradation is observed when unrelated communities, even up to 32% of the network, are removed. For sensor bias, only the simultaneous loss of the faulty community and all descendant communities eliminates fault detection capability, emphasizing the interplay between causal influence and community structure.

Altogether, these results show a range of quantitative design principles for decentralized industrial monitoring: *(i)* maintain community sizes of few nodes (10–25 nodes), *(ii)* balance internal density with moderate inter-community connectivity, *(iii)* avoid excessive modularity (>0.35) and cohesiveness (>0.4), and *(iv)* identify and specially monitor root nodes with weak causal influence.

These guidelines not only support practical implementation of decentralized monitoring but also provide mechanistic foundations for predictive models of fault detection sensitivity, enabling topology-aware, adaptive, and resilient monitoring architectures.

Future work will extend this framework to nonlinear and nonstationary systems, exploring and testing whether the same community-based and causal principles hold in more complex, real-world industrial networks.

ACKNOWLEDGEMENTS

Rodrigo Paredes acknowledges the support from the doctoral Grant (PRT/BD/154267/2022) financed by the Portuguese Foundation for Science and Technology (FCT), under MIT Portugal Program. The authors acknowledge support from CERES funded by FCT (UID/00102/2025) and PRR – Recovery and Resilience Program of the Portuguese Republic (UID/PRR/00102/2025).

AUTHOR IDENTIFIERS

Author ORCIDs:
Paredes, R: 0000-0002-0447-3898
Reis, Marco S: 0000-0002-4997-8865

REFERENCES

1. Isermann R. Process fault detection based on modeling and estimation methods—a survey. Automatica 20:387-404 (1984). https://doi.org/10.1016/0005-1098(84)90098-0
2. Reis M, Gins G. Industrial process monitoring in the big data/industry 4.0 era: from detection, to diagnosis, to prognosis. Processes 5:35 (2017). https://doi.org/10.3390/pr5030035
3. Ge Z, Song Z. Distributed PCA model for plant-wide process monitoring. Ind. Eng. Chem. Res. 52:1947-1957 (2013). https://doi.org/10.1021/ie301945s
4. Peng X, Ding SX, Du W, Zhong W, Qian F. Distributed process monitoring based on canonical correlation analysis with partly-connected topology. Control Engineering Practice 101:104500 (2020). https://doi.org/10.1016/j.conengprac.2020.104500
5. Pearl J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988).
6. Chiang LH, Jiang B, Zhu X, Huang D, Braatz RD. Diagnosis of multiple and unknown faults using the causal map and multivariate statistics. Journal of Process Control 28:27-39 (2015). https://doi.org/10.1016/j.jprocont.2015.02.004
7. Paredes R, Rato TJ, Reis MS. Causal network inference and functional decomposition for decentralized statistical process monitoring: detection and diagnosis. Chemical Engineering Science 267:118338 (2023). https://doi.org/10.1016/j.ces.2022.118338
8. Li Y, He J, Chen C, Guan X. Topology inference for network systems: causality perspective and nonasymptotic performance. IEEE Trans. Automat. Contr. 69:3483-3498 (2024). https://doi.org/10.1109/tac.2023.3303816
9. Newman MEJ. Networks: An Introduction. Oxford Univ. Press (2010) https://doi.org/10.1093/acprof:oso/9780199206650.001.0001
10. Traag VA, Waltman L, van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports* 9:5233 (2019) https://doi.org/10.1038/s41598-019-41695-z
11. Rato TJ, Reis MS. Markovian and non-markovian sensitivity enhancing transformations for process monitoring. Chemical Engineering Science 163:223-233 (2017). https://doi.org/10.1016/j.ces.2017.01.047
12. Paredes R, Yang WT, Reis MS. Decentralized causal-based monitoring for large-scale systems: sensitivity and robustness assessment. IFAC-PapersOnLine 59:127-132 (2025). https://doi.org/10.1016/j.ifacol.2025.07.133
13. Yang WT, Reis MS, Borodin V, Juge M, Roussy A. An interpretable unsupervised bayesian network model for fault detection and diagnosis. Control Engineering Practice 127:105304 (2022). https://doi.org/10.1016/j.conengprac.2022.105304
14. Paredes R, Reis MS. Causality in process systems engineering: fundamentals, applications, and

emerging trends. Computers & Chemical Engineering 203:109345 (2025). https://doi.org/10.1016/j.compchemeng.2025.109345