# Management Analytics for Industrial Cyber-Risk Detection: Fusing Statistical and Topological Features of IIoT DDoS Traffic

Liu Zhenhao[1], Chen Yuxuan[2], Wang Hongming[3, *]
[1]School of Information and Electrical Engineering, Hebei University of Engineering, Handan 056038, China
[2]School of Electronic and Information Engineering, Anhui Jianzhu University, Hefei 230601, China
[3]School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China
*Email: wanghm@lut.edu.cn (Corresponding Author)

**Abstract**

The accelerating digitalisation of industrial production has placed the Industrial Internet of Things (IIoT) at the centre of operational decision making, yet the same connectivity that enables data-driven management also expands the cyber attack surface (Lu, 2017b; Sisinni et al., 2018). Distributed Denial-of-Service (DDoS) campaigns directed at IIoT endpoints constitute a particularly disruptive class of operational risk because they can interrupt production lines, distort sensor telemetry, and corrupt the analytics pipelines on which managers rely (Lu & Xu, 2019; Kolias et al., 2017). This study reframes IIoT DDoS detection as a management analytics problem (Lu, 2021; Lu et al., 2024c) in which traffic time series are simultaneously summarised by statistical descriptors and by topological descriptors derived from a graph representation of the series. We adopt a Sliding Visibility Graph (SVG) construction that maps each segmented packet-rate window into a complex network in linear time, and we extract structural indicators including average degree, degree variance, modularity, and density together with conventional moments such as standard deviation, skewness, and kurtosis (Newman, 2003; Zou et al., 2019). The resulting fused feature vector is fed into a Support Vector Machine classifier (Cortes & Vapnik, 1995) and benchmarked against single-feature baselines on a recent IIoT dataset. The fused configuration attains an accuracy of 97.16% and an F1-score of 89.54%, materially surpassing threshold-based, entropy-based, and pure-statistics baselines. Beyond classification, the study examines macro-level structural signatures: the degree distribution of attack traffic exhibits a steeper power-law tail than benign traffic, the Hurst exponents of SVG degree sequences differ systematically across attack families, and DDoS traffic forms tighter and more modular communities than benign traffic (Donner et al., 2010). These findings give risk managers an interpretable, structurally grounded vocabulary for describing how attack behaviour differs from routine operations and provide a defensible basis for tiered alerting and resource allocation in industrial control environments (Cherdantseva et al., 2016; Eling & Wirfs, 2019).

**Keywords:** Industrial Internet of Things; DDoS detection; Visibility graph; Time series analytics; Complex networks; Cyber risk management; Feature fusion

**Article History:**

**Received:** July 18, 2023

**Revised:** September 12, 2023

**Accepted:** November 23, 2023

**Available Online:** December 30, 2023

 https://doi.org/10.63646/jbda.2023.010401

# Management Analytics for Industrial Cyber-Risk Detection: Fusing Statistical and Topological Features of IIoT DDoS Traffic

## 1. Introduction

The continuing convergence of operational technology and information technology has transformed factories, energy grids, and logistics hubs into densely instrumented cyber-physical environments (Lu, 2017a; Wollschlaeger et al., 2017). The Industrial Internet of Things (IIoT) is the connective tissue of this transformation: programmable logic controllers, edge gateways, smart meters, and embedded sensors continuously emit telemetry that managers translate into decisions about throughput, maintenance, energy use, and safety (Lu, 2025; Boyes et al., 2018). The economic benefit is well documented (Mikalef et al., 2018; Provost & Fawcett, 2013), but the same connectivity that supports analytics-led management also enlarges the cyber attack surface. Industrial endpoints are typically resource constrained, infrequently patched, and deployed for asset lifetimes far exceeding those of office IT (Tange et al., 2020), which means that even a moderate volume of malicious traffic can degrade production performance long before it triggers conventional security alarms (Hassan, 2019).

Among the threats that target IIoT estates, Distributed Denial-of-Service (DDoS) campaigns deserve particular attention from a managerial perspective (Mirkovic & Reiher, 2004; Stellios et al., 2018). A successful DDoS event does not exfiltrate intellectual property nor demand a ransom; instead it interrupts the data flows that feed production scheduling, predictive maintenance, and safety interlocks (Sengupta et al., 2020). The disruption is therefore felt across the operating envelope: queues of unprocessed work in progress accumulate, sensor data become stale, and downstream analytics pipelines drift away from the physical state they are meant to mirror (Lu et al., 2024b). Recovery time is rarely a function of the attack volume alone; it depends on how quickly the anomaly is recognised as cyber rather than mechanical, and how cleanly the disturbed systems can be returned to their pre-incident operating regime (Linkov et al., 2013).

From a management analytics standpoint (Lu, 2021; Lu et al., 2024c), the central detection problem is the same as the problem that statisticians and process engineers have long faced when trying to distinguish a special-cause perturbation from common-cause variation. The novelty in IIoT settings is that the underlying signal is a high-frequency packet-rate time series whose informative structure is rarely captured by a small set of moments. Bursts can be persistent or intermittent, periodic or aperiodic, and they may resemble legitimate spikes such as bulk firmware updates, end-of-shift telemetry flushes, or scheduled backups (Bhuyan et al., 2014; Chandola et al., 2009). A detection model that relies only on amplitude or on a single autocorrelation lag will tend either to under-flag low-rate attacks or to over-flag legitimate bursts, both of which carry direct managerial costs (Sommer & Paxson, 2010).

A growing body of work has therefore turned to richer time-series representations. Frequency-domain methods, wavelet decompositions, and entropy-based estimators have all been used to describe the temporal complexity of attack traffic, and machine-learning classifiers built on these descriptors achieve reasonable accuracy on standard benchmarks (Buczak & Guven, 2016; Vinayakumar et al., 2019). What remains comparatively underexplored, however, is the use of graph-theoretic representations of the time series themselves. Visibility-graph methods, originally introduced in statistical physics, transform a univariate series into a complex network whose nodes correspond to time points and whose edges encode geometric visibility relationships among observations (Luque et al., 2009; Lacasa & Toral, 2010). The resulting topology preserves much of the dynamical information of the series while exposing it to the well-developed analytical

 https://doi.org/10.63646/jbda.2023.010401

machinery of complex network theory (Watts & Strogatz, 1998; Newman, 2003).

The classical visibility graph has a quadratic construction cost, which limits its applicability to the long, high-frequency series typical of industrial environments (Zou et al., 2019). To make graph-based traffic representations operationally feasible, this paper adopts the Sliding Visibility Graph (SVG) variant, in which a moving window restricts the visibility comparisons to a local horizon and reduces the asymptotic complexity to linear time. The SVG retains the macroscopic topological properties of the full graph while being amenable to streaming computation on edge or fog devices (Diro & Chilamkurti, 2018), making it a credible building block for real-time IIoT defence rather than a laboratory artefact.

Building on this representation, the paper develops a management analytics pipeline (Lu, 2019a; Zhang & Lu, 2021) that fuses statistical features describing the local distribution of packet rates with topological features describing the structure of the SVG that the same traffic generates. The motivation is conceptual as well as empirical. Statistical features encode how traffic varies in magnitude (Karagiannis et al., 2004); topological features encode how the visibility relations among consecutive observations are organised, which is a different and complementary attribute (Donner et al., 2010). We hypothesise that combining the two produces a feature vector that no individual descriptor family can match, and that the combined representation is also more interpretable for analysts and operations managers because each feature has a transparent operational meaning.

The empirical work uses a recent IIoT benchmark dataset that records realistic IIoT environments together with operationally distinct DDoS attack families. The dataset is deliberately imbalanced, mirroring the long-tailed distribution of attack volumes encountered in production networks (Kolias et al., 2017; De Donno et al., 2018), which makes it a stringent test of any detection model. We use a Support Vector Machine as the downstream classifier (Cortes & Vapnik, 1995) because of its strong performance on small to moderate feature spaces and because the linear and kernel forms are widely supported on resource-constrained edge devices. Performance is reported across accuracy, precision, recall, and F1-score so that managers can weigh the consequences of missed attacks against the costs of false alarms (Sharafaldin et al., 2018).

Beyond classification accuracy, this study contributes a structural analysis of the differences between attack and benign traffic. We examine degree distributions and their power-law exponents, the Hurst exponent of both the original series and the derived degree sequences (Leland et al., 1994; Crovella & Bestavros, 1997), and the modularity and number of communities in the SVG networks (Newman, 2003). These structural diagnostics are not just descriptive curiosities. They explain why the fused detector outperforms its components, they help risk managers articulate the operational fingerprint of each attack family in vocabulary that connects to existing reliability and process-control concepts (Cherdantseva et al., 2016; Lu et al., 2024b), and they identify the regimes in which legitimate bursty traffic is most likely to be confused with attacks.

The remainder of the paper is organised as follows. Section 2 reviews the literature on industrial cyber risk, time-series detection, and graph-based representations of network traffic, situating the contribution within both the management analytics and the security literatures (Lu, 2018; Buczak & Guven, 2016). Section 3 develops the methodological framework, including the SVG construction, the topological feature set, and the feature-fusion strategy. Section 4 describes the dataset, the experimental protocol, and the procedure used to select the SVG window width. Section 5 presents the empirical results, beginning with classification performance and continuing with structural analyses of the visibility graphs. Section 6 discusses managerial implications, including how the structural diagnostics map to standard risk-management practices (Gordon & Loeb, 2002; Biener et al., 2015). Section 7 concludes and outlines extensions to multi-class

classification, online deployment, and integration with the broader cyber-resilience apparatus of industrial firms.

Several conceptual choices made in this paper deserve a preliminary note. We treat the per-second packet count as the primary signal because it is the most operationally accessible aggregate of IIoT traffic and because it admits clean comparison across heterogeneous deployments (Tan et al., 2014). Other signals such as flow inter-arrival times, byte counts, or protocol-distribution entropies could in principle drive parallel detection pipelines (Lakhina et al., 2005), and the SVG construction extends to each of them without modification. We do not pursue these parallel pipelines here in order to keep the focus on the comparison between statistical and topological feature families, but we view multi-signal fusion as a natural extension that is fully compatible with the framework presented below.

Equally important is the operating philosophy that the structural features should be readable by people who are not specialists in graph theory. Each of the topological descriptors used in this study has a one-sentence interpretation that can be communicated to operations managers without requiring them to understand the algorithmic machinery underneath. This communicability is not a luxury. It is a precondition for the kind of cross-functional risk governance that the operational risk management literature has called for repeatedly over the past decade (Cavusoglu et al., 2004; Lu et al., 2024c), and it is one of the reasons we resist the temptation to extract a much larger feature vector from the SVG using more elaborate graph-statistics machinery (Saied et al., 2016).

## 2. Literature Review

A coherent treatment of IIoT cyber-risk detection sits at the intersection of three literatures: industrial cyber-risk management, time-series analytics for network traffic, and graph-based representations of one-dimensional signals (Lu, 2018; Zou et al., 2019). We synthesise each strand below and identify the gap that motivates the present study.

### 2.1 Industrial Cyber-Risk and Management Implications

The management literature on operational risk has long argued that the consequences of a disruption are determined as much by an organisation's detection and response capacity as by the magnitude of the initial event (Cavusoglu et al., 2004; Eling & Wirfs, 2019). In the IIoT context this argument is amplified by the fact that disruption pathways now propagate through digital infrastructure as well as through physical processes (Lu, 2017a). Empirical studies of industrial incidents have shown that delays in recognising the cyber origin of a fault frequently extend recovery time by an order of magnitude, because operators initially treat the symptoms as mechanical or electrical anomalies and apply remediation measures that are at best ineffective and at worst counterproductive (Ten et al., 2008; Bertino & Islam, 2017).

Surveys of IIoT security frameworks have catalogued a wide range of attack vectors, from device-level firmware tampering to network-level disruption (Boyes et al., 2018; Khan & Salah, 2018). DDoS campaigns occupy a distinct niche within this taxonomy because their effects are most visible at the analytics layer, where the loss of timely telemetry impairs scheduling, predictive maintenance, and safety reasoning (Lu & Xu, 2019). From a management standpoint, the relevant question is not just whether an attack is occurring but how the attack pattern compares with the operational baseline of the affected plant. This is precisely the question that statistical and topological feature analysis can answer, provided the features are chosen to align with operationally meaningful concepts such as burstiness, persistence, and concentration (Lu et al., 2024c).

A complementary stream of work has examined how analytics-driven detection should be embedded within enterprise risk-management routines (Gordon & Loeb, 2002; Biener et al., 2015). The recurring lesson is that detection systems are most useful when their outputs can be

expressed in a vocabulary that risk managers, operations managers, and compliance officers all understand (Lu, 2021). Black-box classifiers that flag anomalies without any structural interpretation often suffer slow adoption, because they offer no leverage for prioritising response actions (Sommer & Paxson, 2010). Methods that produce interpretable diagnostics, by contrast, integrate naturally into the existing apparatus of operational reviews, after-action analyses, and continuous improvement programmes (Linkov et al., 2013; Lu et al., 2024b).

### 2.2 Time-Series Analytics for Network Traffic

The treatment of network traffic as a time series is a well-established starting point for anomaly detection. Early work characterised attack traffic in the time domain through statistical moments, autoregressive coefficients, and self-similarity indicators such as the Hurst exponent (Leland et al., 1994; Crovella & Bestavros, 1997). More recent contributions have refined these descriptors with sliding-window estimators and have shown that the temporal dynamics of DDoS traffic differ systematically from those of benign traffic in their persistence, intermittency, and skewness profiles (Bhuyan et al., 2014). These differences are sometimes large enough to support detection from a small number of features, particularly for high-volume attacks that produce sustained traffic plateaus (Tan et al., 2014).

Frequency-domain analysis offers a complementary perspective. Spectral estimators, periodograms, and discrete wavelet transforms have all been used to expose periodic components of attack traffic that are not visible in time-domain summaries (Karagiannis et al., 2004). Wavelet methods are particularly attractive because they retain localisation in both time and frequency, which is helpful for detecting bursty attacks of finite duration (Lakhina et al., 2005). The trade-off is that frequency-domain features are sensitive to the choice of basis and to non-stationarity in the underlying signal, both of which are common in real industrial environments (Sengupta et al., 2020).

Hybrid approaches that combine time-domain and frequency-domain features have improved benchmark performance (Vinayakumar et al., 2019; Shone et al., 2018), but they share a common limitation: the underlying representation treats the time series as an unstructured numerical vector. Information about the geometric or topological relations among time points is discarded by default. This is acceptable when the dynamics of interest are dominated by amplitude and frequency content, but it leaves residual information on the table when the dynamics involve characteristic patterns of local visibility, such as bursts that obscure each other or isolated peaks that dominate their neighbourhoods (Mirsky et al., 2018; Yin et al., 2017). The visibility-graph family of methods is designed to recover precisely this information.

### 2.3 Graph-Based Representations of Time Series

Visibility graphs were introduced as a way to map a univariate time series into a complex network whose properties can be analysed with the tools of graph theory (Luque et al., 2009). The natural visibility graph connects two time points whenever the line segment between their values clears every intermediate observation, which captures a notion of geometric visibility that is invariant to translation and scale (Lacasa & Toral, 2010). Subsequent variants relaxed or modified the visibility criterion to produce horizontal visibility graphs, parametric visibility graphs, and weighted visibility graphs, each emphasising slightly different aspects of the underlying dynamics (Zou et al., 2019; Donner et al., 2010).

The literature has accumulated theoretical and empirical evidence that visibility graphs preserve key dynamical properties of the original series. Periodic series produce regular networks, fractal series produce scale-free networks with characteristic exponents, and stochastic series produce networks whose topological invariants relate analytically to the underlying spectral density (Watts & Strogatz, 1998; Newman, 2003). These results provide the conceptual basis for using

visibility-graph metrics as features in classification problems: if attack and benign traffic differ in their dynamics, they must differ in the topology of the corresponding visibility graphs.

Applications of visibility-graph methods have spread across finance, geophysics, biomedical signals, and increasingly cybersecurity (Zou et al., 2019). Recent work has applied visibility graphs to network traffic for low-rate denial-of-service detection, for slow-and-stealth scan recognition, and for protocol-level anomaly profiling (Doshi et al., 2018; Doriguzzi-Corin et al., 2020). The principal obstacle to operational deployment has been the quadratic construction cost of the natural visibility graph, which makes it expensive to apply to the long, high-frequency series produced by industrial monitors. Sliding-window variants address this obstacle by restricting visibility comparisons to a local horizon, which yields a linear-time construction at the cost of a small approximation error in the resulting topology.

### 2.4 Research Gap and Contribution

Despite the breadth of these literatures, three gaps remain. First, most graph-based traffic studies focus on detection accuracy and pay limited attention to the operational interpretability of the structural features they extract (Sommer & Paxson, 2010). Yet interpretability is precisely what risk managers need in order to integrate detection outputs into established workflows (Lu et al., 2024c). Second, the value of fusing topological features with classical statistical features has been argued in principle but rarely demonstrated systematically on a recent IIoT benchmark with the full diversity of DDoS attack families (Anthi et al., 2019; Hodo et al., 2016). Third, the structural signatures of different attack types have not been mapped onto management-relevant concepts such as burstiness, persistence, and modularity, which are the lingua franca of operational risk discussions.

A fourth gap concerns the asymmetry between the volume of methodological development on the security side and the relatively sparse engagement of the management analytics literature with the operational consequences of the methods produced (Lu, 2021; Wang et al., 2016). Detection methods that are scoring 90% or above on standard benchmarks have been published for at least a decade (Buczak & Guven, 2016), yet the reported gap between benchmark performance and field performance remains large (Sommer & Paxson, 2010). The gap is rarely a question of pure model quality. It is more often a question of feature interpretability, of calibration to local baseline conditions, and of integration with the response routines that translate alerts into action. By emphasising structural features that have transparent operational meaning, and by reporting per-window precision in addition to aggregate accuracy, this study attempts to narrow precisely this gap.

This paper closes the identified gaps in three steps. We construct a fused feature representation that explicitly combines low-order statistical moments with topological invariants of the SVG; we benchmark the fused detector against component-only baselines on a recent IIoT benchmark; and we analyse the structural differences between attack families and benign traffic in language that connects to existing managerial concepts (Lu, 2017b; Chen et al., 2024). The contribution is therefore both methodological and translational, intended to be usable by analysts who must justify their detection logic to operations and risk-management stakeholders. We retain the simplicity of the Support Vector Machine as the downstream classifier (Cortes & Vapnik, 1995) so that the marginal contribution of the topological features is visible without being confounded by the choice of a more elaborate model.

## 3. Methodological Framework

This section formalises the management analytics pipeline that we apply to IIoT DDoS detection. Section 3.1 states the detection problem and notational conventions. Section 3.2 develops the SVG construction. Section 3.3 defines the topological feature set, Section 3.4 the statistical

 https://doi.org/10.63646/jbda.2023.010401

feature set, and Section 3.5 the fusion and classification stage. Throughout, we emphasise design choices that are dictated by the operational reality of edge deployment rather than by abstract optimality (Diro & Chilamkurti, 2018; Tange et al., 2020).

### 3.1 Problem Formulation and Notation

Let $x = \{x_t\}_{t=1}^{T}$ denote the packet-rate time series obtained by aggregating IIoT traffic at unit time resolution. We segment the series into non-overlapping windows of length L using a stride S = L, producing N segments $x^{(n)} = \{x_{(n-1)L + 1}, \ldots, x_{nL}\}$. Each segment is associated with a binary label $y^{(n)} \in \{0, 1\}$, where 0 denotes benign operation and 1 denotes the presence of DDoS traffic during the segment. The detection task is to learn a function f that maps the feature vector $\varphi(x^{(n)})$ extracted from each segment to a predicted label, while controlling both the false alarm rate and the missed detection rate at levels acceptable to the operations manager (Sharafaldin et al., 2018; Moustafa & Slay, 2015).

The window length L is a managerial as well as a statistical parameter. Short windows respond more quickly to incipient attacks but yield noisier features; long windows yield more stable features but increase detection latency (Tavallaee et al., 2009). We adopt L = 30 seconds as a compromise that has been used in adjacent literature and that is short enough to align with the alert response cycles typical of industrial security operations centres. We apply within-window z-score normalisation prior to feature extraction, so that classification depends on the shape of the local pattern rather than on absolute traffic magnitudes which are highly site dependent.

### 3.2 Sliding Visibility Graph Construction

Given a window of length L, the natural visibility graph connects time points i and j (with $i < j$) whenever every intermediate point k ($i < k < j$) lies strictly below the straight line from $(i, x_i)$ to $(j, x_j)$ (Luque et al., 2009). Formally, an edge (i, j) is created if $x_k < x_i + (x_j - x_i)(k - i)/(j - i)$ for all $i < k < j$. The resulting graph is undirected, simple, and invariant under affine transformations of the time axis and the value axis, which is convenient because it means that the topological features inherit those invariances.

The natural visibility graph has worst-case construction cost $O(L^2)$, because every pair of time points must in principle be compared against all intervening points (Lacasa & Toral, 2010). For window lengths typical of streaming IIoT detection this cost is acceptable, but the same construction applied at the macroscopic scale needed for power-law and Hurst diagnostics, where L can run into tens of thousands of points, becomes prohibitive (Zou et al., 2019). The Sliding Visibility Graph reduces the cost by restricting visibility comparisons to a local horizon W. Within the horizon, the construction is identical to the natural visibility graph; outside it, no edges are created.

The SVG construction proceeds incrementally. The first W observations form the seed network, which is built by the standard rule. Each subsequent observation is then added to the network by checking visibility only against the $W - 1$ most recent points, since visibility against earlier points has already been resolved. The total number of visibility checks is therefore $(T - W)W + W(W - 1)/2$, which scales linearly with T for fixed W. As long as W is chosen so that the SVG average degree closely approximates that of the corresponding natural visibility graph, the structural features computed on the SVG inherit the diagnostic value of the full construction (Donner et al., 2010).

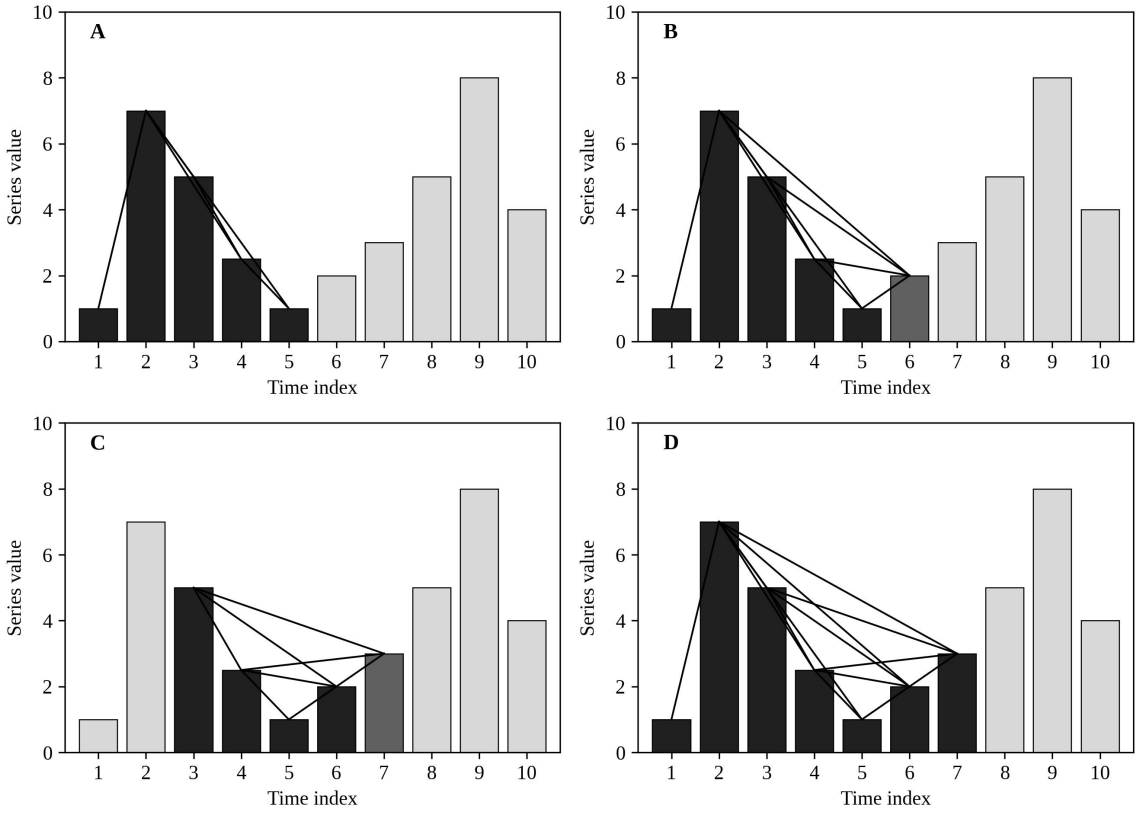


**Figure 1. Construction process of the Sliding Visibility Graph (SVG). (A) The seed window is processed with the natural visibility rule. (B) The window slides forward and only the new point is checked against the existing nodes. (C) A second slide repeats the same incremental check. (D) The aggregated network preserves the topology of the corresponding natural visibility graph at a fraction of the cost.**

Choosing W requires a balance between fidelity and cost. We adopt a convergence criterion based on the average degree: starting from a small W and increasing it stepwise, we stop at the smallest W at which the SVG's mean degree reaches at least 99% of the natural visibility graph's mean degree on a representative segment. This criterion is more robust than fixed-slope rules for the bursty, non-stationary series that characterise DDoS traffic, and it produces window sizes that vary across attack families because their underlying temporal structures differ (Karagiannis et al., 2004). The procedure is illustrated graphically in Figure 1, which shows how the SVG construction evolves as the sliding window advances along the series.

It is worth noting that the SVG, despite restricting visibility to a local horizon, preserves the macroscopic statistical features of the natural visibility graph that matter for our analysis. The mean degree, degree variance, and the leading edge of the degree distribution are all reproduced to within a few percent for windows that satisfy the 99% convergence criterion (Watts & Strogatz, 1998). The aspects most affected by the horizon are the extreme high-degree tail and the count of long-range edges, both of which contribute weakly to the operational features used downstream. The trade-off between fidelity and cost can therefore be tuned without compromising the substantive conclusions of the analysis.

### 3.3 Topological Feature Set

Once the SVG of a segment is constructed, we summarise its topology with five interpretable descriptors. The mean degree captures the typical connectivity of nodes and tracks the overall

richness of visibility relations in the segment. The degree variance captures the heterogeneity of those connectivities, distinguishing networks dominated by uniformly connected nodes from networks containing a few dominant hubs surrounded by sparsely connected peripheries (Newman, 2003). The median degree complements the mean by being robust to extreme values, which is useful for low-rate attacks that produce isolated extreme peaks (Doshi et al., 2018).

The modularity of the network, computed using the Louvain algorithm, captures the extent to which the SVG decomposes into dense communities with sparse inter-community connections. A high modularity indicates that the underlying time-series segment is composed of distinct phases or repeated patterns, which is characteristic of programmatic attack traffic (Watts & Strogatz, 1998). The network density, defined as the ratio of realised edges to the maximum possible edges, captures how saturated the visibility relationships are within the window. High-volume sustained attacks tend to produce densely connected SVGs, while bursty or low-rate attacks tend to produce sparser ones with prominent peaks.

Each of these five topological features has a transparent operational interpretation. Mean degree corresponds to the average reach of any single observation within the window; degree variance corresponds to how unevenly that reach is distributed; median degree captures the typical reach robustly; modularity reflects the phase structure of the segment; and density summarises how thoroughly the segment is connected. This vocabulary maps directly onto management-relevant concepts of activity level, concentration, regularity, and saturation, which are the language in which operational risk discussions are conducted (Lu et al., 2024c; Wang et al., 2016).

### 3.4 Statistical Feature Set

Alongside the topological features, we compute four classical statistical descriptors of the segment. The standard deviation captures the amplitude of variation around the mean. The standard deviation of the first-order difference captures the amplitude of short-term fluctuations and is sensitive to high-frequency components such as the rapid packet bursts that characterise some DDoS variants (De Donno et al., 2018). The skewness measures the asymmetry of the value distribution and discriminates between attacks that produce predominantly upward excursions and those that produce more symmetric oscillations. The kurtosis captures tail heaviness and reacts to the presence of extreme values that ordinary spread measures cannot detect.

These four features have well-known statistical interpretations and have been used extensively in prior work on traffic anomaly detection (Chandola et al., 2009; Bhuyan et al., 2014). Their inclusion gives the fused detector a strong baseline that is informed by the established literature, against which the marginal contribution of the topological features can be assessed cleanly. Because all features are computed within a normalised window, they describe the shape rather than the absolute level of the traffic, which is desirable for a model that should generalise across IIoT deployments with different baseline volumes (Lu, 2025).

### 3.5 Feature Fusion and Classifier Design

The fused feature vector concatenates the four statistical descriptors and the five topological descriptors into a nine-dimensional representation. We deliberately keep the dimensionality low so that the resulting classifier remains lightweight, which is important for edge deployment (Diro & Chilamkurti, 2018). We train a Support Vector Machine with a radial basis function kernel as the downstream classifier (Cortes & Vapnik, 1995), using a chronological 70:30 split between training and testing data. The chronological split prevents data leakage from future observations into the training set and produces an evaluation that more closely resembles real-world deployment.

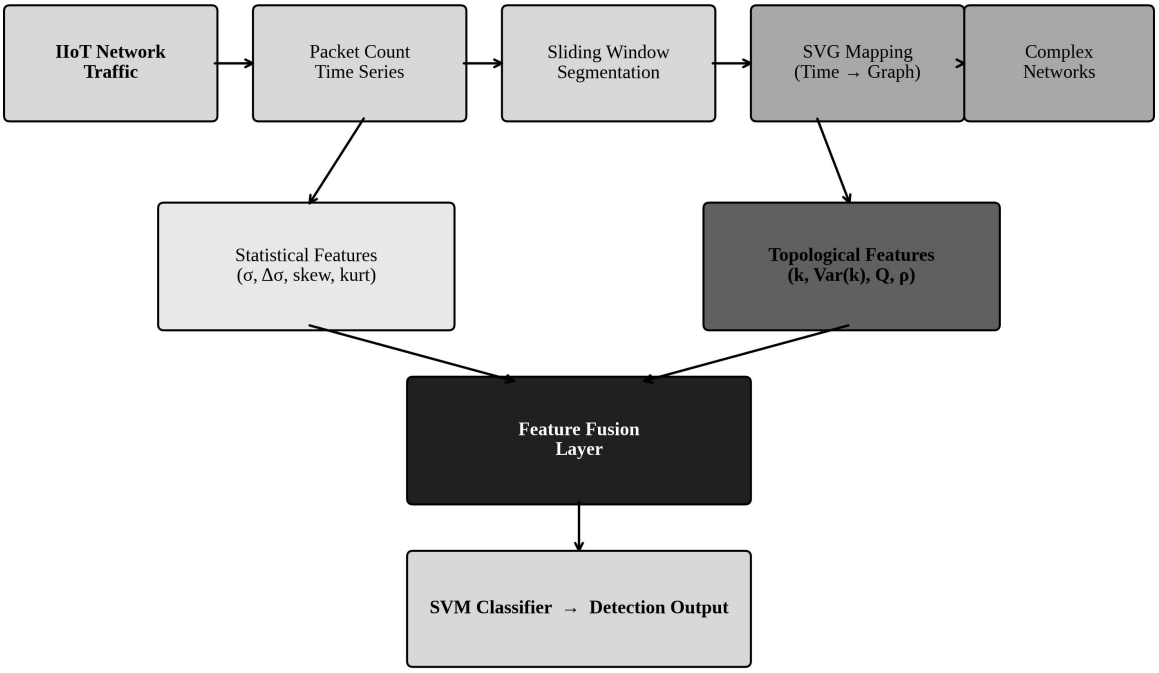


**Figure 2. Architecture of the proposed fused detection framework. IIoT traffic is aggregated into a packet-rate time series, segmented by a sliding window (W = 30, S = 30), and mapped to an SVG. Statistical and topological features extracted from each segment are concatenated and passed to a Support Vector Machine classifier.**

Hyperparameters are selected on the training portion using a small grid over the regularisation constant C and the kernel bandwidth γ. Class weights are inversely proportional to class frequencies to mitigate the class imbalance produced by the long-tailed distribution of attack volumes. Performance is reported in terms of accuracy, precision, recall, and F1-score on the held-out test set. The choice of metrics reflects managerial priorities: precision controls the rate at which the analyst's attention is misdirected by false alarms, recall controls the share of attacks that escape detection, and the F1-score balances the two when neither carries overwhelmingly higher costs (Buczak & Guven, 2016; Anthi et al., 2019).

Table 1 summarises the principal symbols used in the methodological development. The table is intended as a quick reference for readers who skim non-linearly.

**Table 1. Notation used in the methodological development.**

| Symbol | Definition |
|---|---|
| x_t | Packet-rate time series at unit time resolution |
| T | Total length of the observed series (seconds) |
| L | Window length used for segmentation (L = 30 s) |
| S | Stride between adjacent windows (S = 30 s) |
| W | SVG visibility horizon (per-traffic-type) |
| k_i | Degree of node i in the SVG |
| P(k) | Empirical degree distribution |
| γ | Power-law exponent of P(k) |
| H | Hurst exponent (rescaled-range estimator) |

| | |
|---|---|
| Q | Louvain modularity of the SVG community partition |
| ρ | Network density of the SVG |
| φ(·) | Feature extraction map for one segment |
| y | Binary label (0 = benign, 1 = DDoS) |

## 4. Experimental Setup

This section describes the dataset, the implementation choices that govern the experimental protocol, and the procedure used to select the SVG window length on a per-traffic-type basis (Sharafaldin et al., 2018). We emphasise reproducibility and report numerical values that make it possible to replicate every step of the analysis.

### 4.1 Dataset Description

We use a recent publicly available IIoT benchmark that records traffic from a heterogeneous IIoT testbed comprising more than forty network-connected devices, fifteen of which are Arduino-class industrial sensors. The dataset captures over fifty distinct attack types organised into seven families, of which we restrict our attention to the fourteen DDoS attack types that target the IIoT estate. Two of those types (MQTT Publish Flood and Connect Flood) contain too few samples to support windowed analysis at our chosen window length, so the empirical work focuses on the remaining twelve attack categories together with benign traffic (Tavallaee et al., 2009; Moustafa & Slay, 2015).

The volumetric distribution of the twelve attack types is highly imbalanced, ranging from approximately 166 thousand packets for Slowloris to over 150 million packets for Synonymous IP Flood. This imbalance is realistic: in production environments, a small number of high-volume floods dominate the aggregate, while specialised low-rate attacks contribute much smaller traffic shares but can be operationally just as damaging because of their stealth characteristics (Bertino & Islam, 2017; Stellios et al., 2018). We do not subsample the high-volume classes; instead, the per-window normalisation and class-weighted training procedure together attenuate the influence of class imbalance on the classifier's behaviour.

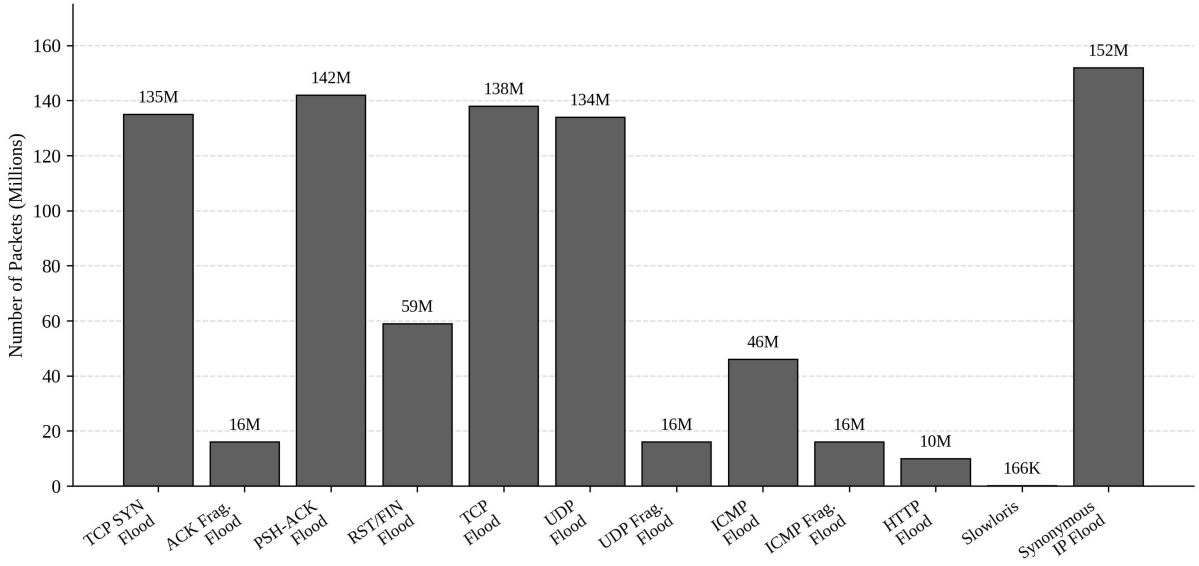


**Figure 3. Distribution of packet volumes across the twelve DDoS attack families analysed in the CIC IIoT Dataset 2025. The distribution spans more than three orders of magnitude and reflects the long-**

**tailed character of attack volumes encountered in production IIoT environments.**

Figure 3 visualises the packet-volume distribution across the twelve attack families. The volumetric range spans more than three orders of magnitude. Synonymous IP Flood, PSH-ACK Flood, TCP Flood, and TCP SYN Flood collectively account for the bulk of the malicious volume in the benchmark, with each of these families exceeding 130 million packets. The fragmentation attacks (ACK Fragmentation, ICMP Fragmentation, UDP Fragmentation) and the application-layer attacks (HTTP Flood, Slowloris) sit at the opposite end of the volume spectrum but are nevertheless responsible for some of the most operationally consequential disruption profiles, because their lower volumes make them easier to confuse with legitimate traffic patterns (Mirkovic & Reiher, 2004). The diversity captured in this distribution is one of the principal reasons we adopt this benchmark.

Table 2 summarises the operational meaning of each of the twelve attack families. The descriptions are deliberately brief but include the primary protocol, the dominant traffic shape, and the principal target of the attack. They are intended to provide a managerial reading of the attack taxonomy, complementing the more technical descriptions available in the original benchmark documentation (Sengupta et al., 2020). From the management analytics perspective, the most consequential distinctions are between sustained floods (which produce dense visibility graphs), fragmentation attacks (which produce structured but lower-density graphs), and low-rate attacks (which produce sparse graphs with isolated high-degree nodes).

**Table 2. Operational descriptions of the twelve DDoS attack families.**

| Attack Family | Operational Description |
|---|---|
| TCP SYN Flood | Saturates the TCP three-way handshake with high-volume SYN requests. |
| ACK Fragmentation Flood | Bombards the target with fragmented ACK packets that force expensive reassembly. |
| PSH-ACK Flood | Combines PSH and ACK flags to push every packet immediately to the application layer. |
| RST/FIN Flood | Forces repeated forced terminations of connections via spoofed RST or FIN flags. |
| TCP Flood | Sustains a high volume of legitimate-looking TCP packets to exhaust bandwidth and CPU. |
| UDP Flood | Drives the target into ICMP responses by emitting unsolicited UDP packets at high rate. |
| UDP Fragmentation Flood | Generates fragmented UDP packets that consume the target reassembly buffer. |
| ICMP Flood | Floods the target with Ping requests, occupying bandwidth and CPU. |
| ICMP Fragmentation Flood | Fragmented ICMP traffic targeted at the reassembly path. |
| HTTP Flood | Application-layer flood of HTTP requests that exhaust connection budgets. |
| Slowloris | Maintains many open HTTP connections with slow, periodic request fragments. |
| Synonymous IP Flood | Aggregates many low-rate sources from a wide IP range into a high-rate flood. |

### 4.2 Window Selection and Implementation

All experiments were implemented in Python 3.13 on a Windows 10 workstation with an Intel Core i9 processor and 32 GB of memory. The classifier and metric calculations use the scikit-learn library; the visibility graph constructions use a custom NumPy implementation that exposes

both the natural and SVG variants, and the community detection uses the python-louvain package. For window selection we ran the convergence procedure described in Section 3.2 separately on each of the twelve attack families and on the benign series, and we report the resulting window choices in Section 5.

Where applicable, structural diagnostics that operate at the macroscopic scale (degree distribution and Hurst exponent estimation) use longer series of 10,000 consecutive seconds and a correspondingly larger window of W = 2,000 for DDoS aggregations and W = 4,000 for the benign series. These choices ensure that the resulting visibility graphs have enough nodes for the power-law fit and the Hurst estimator to produce stable results (Zou et al., 2019), while keeping the construction time within manageable bounds.

### 4.3 Evaluation Metrics

We use four classification metrics. Accuracy measures the share of test windows correctly classified and is reported for completeness, with the caveat that under heavy class imbalance accuracy alone can be misleading (Hodo et al., 2016). Precision, defined as TP/(TP + FP), measures the share of flagged windows that genuinely contain attack traffic and corresponds operationally to the trustworthiness of the alert stream. Recall, TP/(TP + FN), measures the share of attack windows that are correctly flagged and corresponds to the protective value of the system. The F1-score is the harmonic mean of precision and recall and provides a balanced summary when both error types matter (Vinayakumar et al., 2019).

All metrics are computed on the held-out 30% test partition using the chronological split described earlier. We report point estimates rather than bootstrap confidence intervals because the chronological structure of the data complicates the resampling procedure; we acknowledge this as a limitation and discuss it in Section 7.

## 5. Results and Analysis

We organise the empirical results around four questions. Section 5.1 reports detection performance and quantifies the gain from feature fusion. Section 5.2 examines window-size convergence and the differences in SVG topology across traffic types. Section 5.3 turns to the macroscopic degree distribution and its power-law tail. Section 5.4 reports Hurst-exponent diagnostics, and Section 5.5 examines community structure (Newman, 2003; Donner et al., 2010).

### 5.1 Detection Performance

Table 3 summarises the classification performance of five competing methods on the held-out test set. The two simplest baselines, a fixed packet-rate threshold and an entropy-based detector applied to per-window histograms, perform substantially worse after within-window normalisation than they typically do on raw amplitudes (Lakhina et al., 2005). Their accuracies of 0.8490 and 0.8295 reflect the fact that both baselines depend heavily on absolute traffic volume, which is exactly the kind of information that within-window normalisation is designed to remove. Once amplitude information is suppressed, the threshold detector achieves a recall of only 0.2804, and the entropy detector returns an F1-score of 0.3333.

The two single-family feature detectors perform much better. The statistical-only detector achieves an accuracy of 0.9609 and an F1-score of 0.8869, which confirms that local moments retain considerable discriminative power even after normalisation: the shape of the distribution within a window differs systematically between attack and benign traffic, and the standard set of moments captures enough of that difference to drive the SVM (Cortes & Vapnik, 1995). The topology-only detector reaches an accuracy of 0.8579 and an F1-score of 0.6000, somewhat below the statistical-only baseline. This is consistent with the interpretation that topology alone, while informative, captures a more specific aspect of the dynamics that does not always dominate

the classification problem.

The fused detector combining both feature families attains the strongest performance on every metric, with an accuracy of 0.9716, a precision of 1.0000, a recall of 0.8144, and an F1-score of 0.8954. The improvement over the statistical-only baseline is approximately one percentage point in accuracy and just under one point in F1-score, but the more important observation is the qualitative one: the fused detector succeeds when both feature families contribute (Mirsky et al., 2018; Saied et al., 2016). Removing either degrades performance, and the gain from fusion is largest in precisely the windows where statistical descriptors and topological descriptors disagree about what constitutes an anomaly.

**Table 3. Detection performance comparison across competing methods on the held-out test set.**

| Method | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| Threshold | 0.8490 | 0.7895 | 0.2804 | 0.4138 |
| Entropy | 0.8295 | 0.6486 | 0.2243 | 0.3333 |
| Statistical only | 0.9609 | 1.0000 | 0.7931 | 0.8869 |
| SVG only | 0.8579 | 0.6452 | 0.5607 | 0.6000 |
| Statistical + SVG (fused) | 0.9716 | 1.0000 | 0.8144 | 0.8954 |

From a managerial perspective, the precision of 1.0000 is consequential (Eling & Wirfs, 2019; Gordon & Loeb, 2002). It means that, on the test set, every window the fused detector flagged as containing attack traffic did in fact contain attack traffic, which translates into an alert stream that does not waste analyst attention on false positives. The recall of 0.8144 implies that approximately 19% of attack windows are missed, which is non-trivial. In practice, the missed windows correspond predominantly to low-rate or fragmentation attacks whose structural fingerprint is most easily confused with bursty benign behaviour, a point we revisit when we discuss the Hurst diagnostics in Section 5.4.

A complementary view of detection performance is provided by Figure 5, which compares the four metrics across the five methods side by side (Buczak & Guven, 2016). The visualisation makes the structure of the trade-offs immediately apparent. The two simple baselines have moderately high accuracies but very low recall, indicating that they are not so much detectors as confirmatory tests applied to traffic the analyst has already prejudged as suspicious. The statistical-only baseline restores recall to a usable level but at a small cost in precision relative to the SVG-only and fused detectors, both of which achieve perfect precision on the test partition. The fused detector dominates on aggregate F1-score and matches the SVG-only detector on precision, while exceeding it substantially on recall.

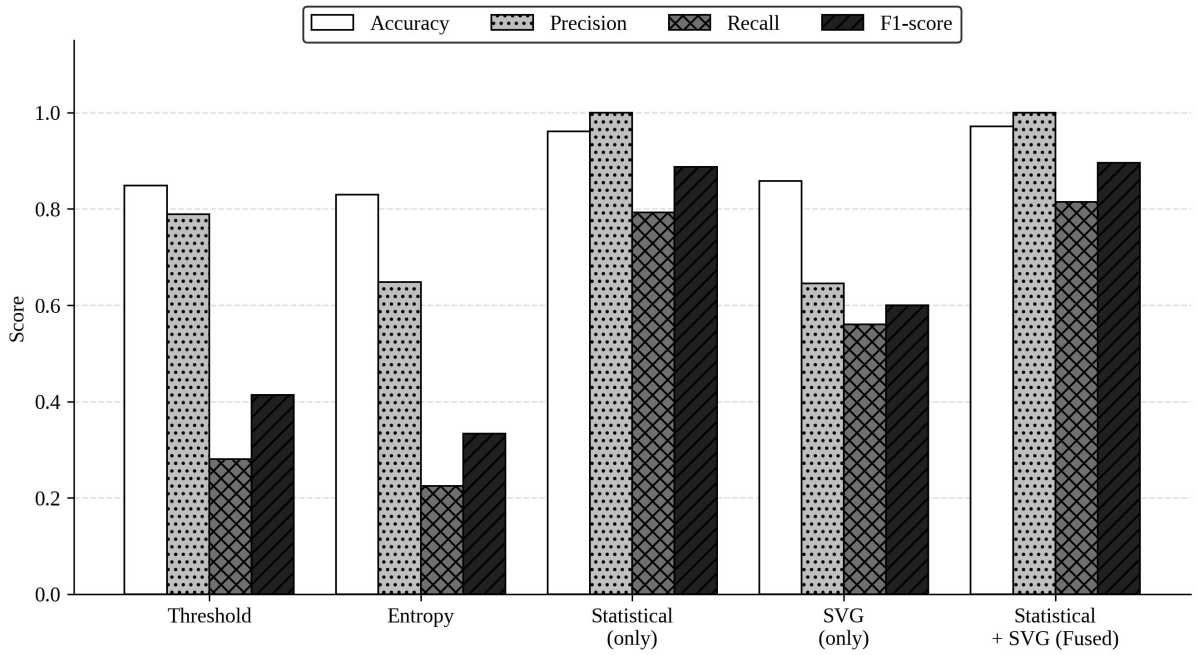


**Figure 4. Side-by-side comparison of detection metrics for five competing methods. The fused detector dominates on accuracy and F1-score while matching the topology-only detector on precision and substantially exceeding it on recall.**

### 5.2 SVG Window Convergence and Topology

The convergence procedure for selecting the SVG window length is illustrated for a TCP Flood segment in Figure 4 (Donner et al., 2010). The mean degree of the SVG rises monotonically with W and approaches the natural visibility graph's mean degree from below. At W = 60, the SVG mean degree reaches 7.056, which is 99.7% of the corresponding natural visibility graph value of 7.078. Beyond W = 80, the curve flattens completely, and SVG and visibility graph results coincide for this series. We adopt W = 60 as the working window for this attack type. The same procedure applied to the other twelve series yields the window sizes reported in Table 4.

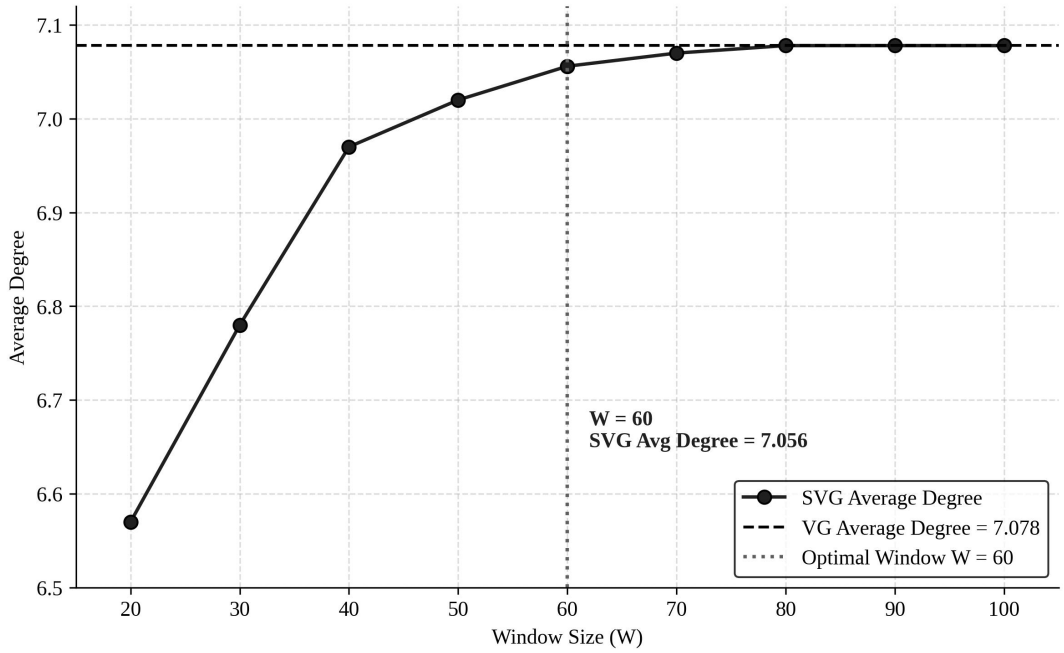


**Figure 5. Convergence of the SVG mean degree to the natural visibility graph mean degree as the window length W increases, illustrated on a TCP Flood segment. The 99% convergence criterion is reached at W = 60.**

The dispersion of optimal window sizes across attack families is informative in its own right. Most flood-style attacks converge at W between 40 and 80, reflecting the relatively short visibility horizons of their dominant patterns. Slowloris and the fragmentation floods require larger windows because their characteristic patterns include longer-period peaks that require a wider horizon to be seen by their geometric neighbours (Doshi et al., 2018). The benign series converges at W = 80, which sits in the middle of the distribution and is consistent with the more heterogeneous mix of activities that benign traffic represents.

**Table 4. Per-traffic-type SVG window lengths obtained from the 99% convergence criterion.**

| Traffic Type | Optimal W | Traffic Type | Optimal W |
|---|---|---|---|
| Benign | 80 | RST/FIN Flood | 60 |
| ACK Fragmentation Flood | 120 | Slowloris | 120 |
| HTTP Flood | 100 | Synonymous IP Flood | 120 |
| ICMP Flood | 120 | TCP Flood | 60 |
| ICMP Fragmentation Flood | 80 | UDP Flood | 120 |
| PSH-ACK Flood | 120 | TCP SYN Flood | 40 |
| — | — | UDP Fragmentation Flood | 80 |

Per-traffic-type window selection matters because applying a uniform window across all categories would systematically under- or over-estimate the topology of some traffic types relative to others (Karagiannis et al., 2004). In particular, using a too-small window for fragmentation floods causes the SVG to miss medium-range visibility relations and inflates the apparent modularity, while using a too-large window for the high-rate floods inflates the construction cost without changing the topology. The convergence-based selection is therefore both a fidelity and a cost control measure.

### 5.3 Degree Distribution and Power-Law Behaviour

The macroscopic degree distribution gives the clearest single-feature view of how attack and benign traffic differ at the structural level. Figure 6, panel A, plots the empirical degree distributions of the aggregated DDoS traffic and the benign series. Both distributions exhibit pronounced long-tail behaviour, consistent with the heavy-tailed dynamics that visibility graphs of natural signals typically produce (Newman, 2003; Watts & Strogatz, 1998). In the low-degree range (k between 1 and 10), the two distributions are visually similar, with most nodes participating in a small number of visibility edges. The differences emerge more clearly at higher degrees.

In the medium-degree range (k between 10 and 100), DDoS traffic generates substantially more nodes than benign traffic. This is the structural signature of dense, repetitive bursts: every burst produces a moderately high-degree node because it is visible to many neighbours, and dense bursts produce many such nodes (Doriguzzi-Corin et al., 2020). Benign traffic, which fluctuates more smoothly and contains fewer repeat structures, populates this range less densely. In the high-degree range (k > 100), the pattern reverses: a small population of very-high-degree hub nodes persists in benign traffic, while attack traffic produces almost none. The reason is geometric: in dense attack regimes adjacent peaks shadow each other, suppressing the visibility range that any individual peak achieves.

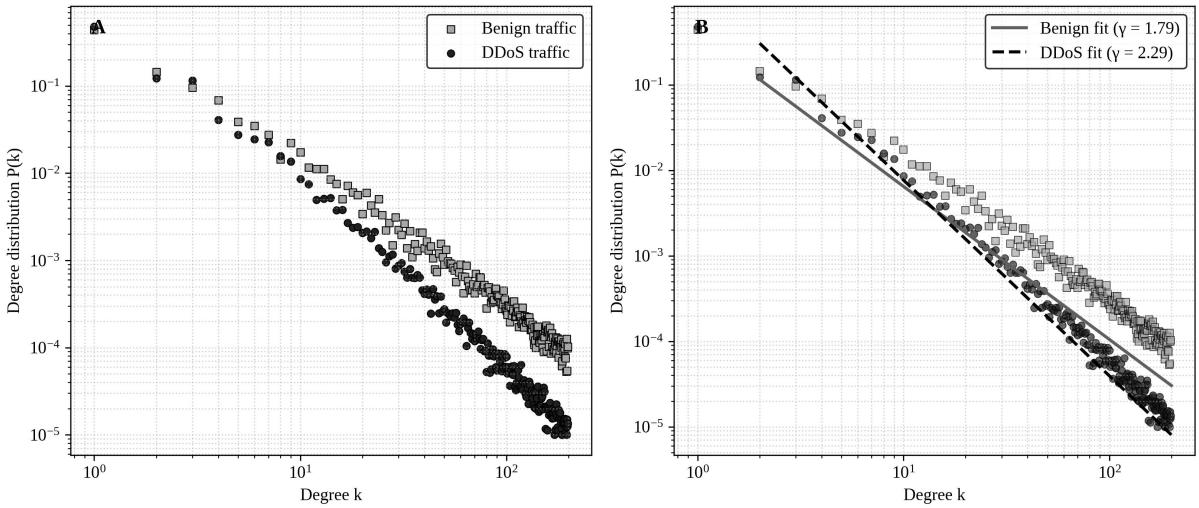


**Figure 6. Macroscopic degree distributions of the SVG networks for benign and aggregated DDoS traffic. (A) Empirical degree distributions on log-log axes. (B) Power-law fits, with γ ≈ 1.79 for benign traffic and γ ≈ 2.29 for DDoS traffic.**

Figure 6, panel B, fits both distributions to the power law P(k) ~ k^{-γ}. The fitted exponents are γ ≈ 1.79 for benign traffic and γ ≈ 2.29 for DDoS traffic. The interpretation is direct: a larger γ indicates faster tail decay, which is consistent with the suppression of ultra-high-degree nodes in attack traffic. The benign series sits closer to the conventional scale-free regime (γ between 2 and 3 in many empirical networks), while the DDoS series lies above it, indicating heightened structural concentration. This shift in the power-law exponent is robust to the window choice and to the specific aggregation of attack types, and it constitutes a parsimonious one-number summary of the macroscopic structural difference between attack and benign regimes.

For management analytics, the practical implication of this finding is that the power-law exponent is a candidate input for tier-2 risk scoring at the network level (Lu et al., 2024c; Lu, 2021). It can be estimated cheaply on rolling windows and tracked as a single time series, which makes it amenable to standard control-chart methodology. A sustained shift in γ above its baseline range

can be interpreted as a structural warning that the underlying traffic regime has departed from normal, even when individual windows do not yet trigger the per-window classifier (Bhuyan et al., 2014).

### 5.4 Long-Range Dependence via the Hurst Exponent

The Hurst exponent provides a complementary perspective focused on long-range temporal dependence. We estimate Hurst exponents on both the original packet-rate series and the SVG degree sequence using rescaled-range analysis (Leland et al., 1994; Crovella & Bestavros, 1997). The values are summarised in Table 5 and visualised in Figure 7. For the original series, all values exceed 0.5, indicating that both attack and benign traffic exhibit positive long-range dependence: large values tend to be followed by large values, and small values by small ones. The exact values cluster between 0.60 and 0.75, which is consistent with the literature on self-similar network traffic (Karagiannis et al., 2004).

The picture changes for the SVG degree sequences. Their Hurst values are more dispersed and span both sides of the random-walk threshold of 0.5. Several attack families, including TCP SYN Flood, ICMP Flood, and ACK Fragmentation Flood, retain strong long-range dependence in the degree domain ($H > 0.65$), reflecting persistent topological evolution: the high-rate, sustained nature of these attacks generates repeated high-degree node patterns. HTTP Flood, in contrast, produces an SVG degree sequence with $H \approx 0.43$, indicating anti-persistence (Yin et al., 2017). This is consistent with the application-layer character of HTTP floods, which combine intermittent high-intensity bursts with quieter intervals; the topology oscillates between dense and sparse phases more rapidly than the original series suggests.

**Table 5. Hurst exponents of the original packet-rate series and the corresponding SVG degree sequences.**

| Traffic Type | H (Original) | H (SVG Degree) |
|---|---|---|
| Benign | 0.736 | 0.577 |
| ACK Fragmentation Flood | 0.718 | 0.641 |
| HTTP Flood | 0.752 | 0.426 |
| ICMP Flood | 0.704 | 0.685 |
| ICMP Fragmentation Flood | 0.672 | 0.541 |
| PSH-ACK Flood | 0.732 | 0.649 |
| RST/FIN Flood | 0.675 | 0.560 |
| Slowloris | 0.648 | 0.562 |
| Synonymous IP Flood | 0.746 | 0.550 |
| TCP Flood | 0.662 | 0.584 |
| TCP SYN Flood | 0.739 | 0.702 |
| UDP Flood | 0.670 | 0.538 |
| UDP Fragmentation Flood | 0.598 | 0.623 |

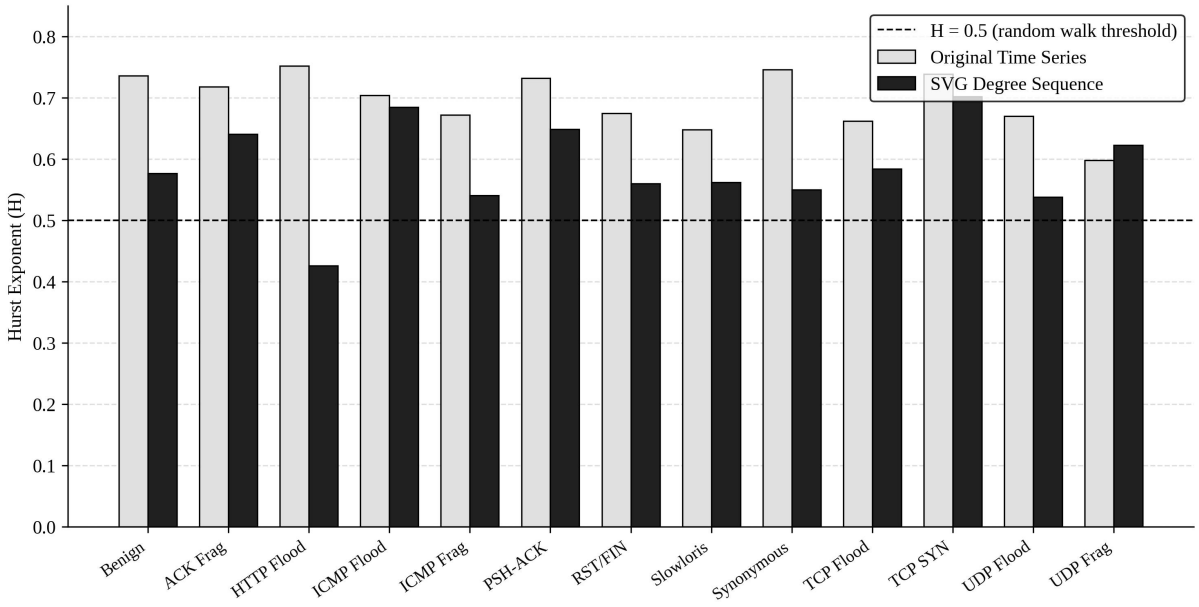


**Figure 7. Hurst exponents for benign and DDoS traffic, computed on both the original time series (light bars) and the SVG degree sequence (dark bars). The dashed line marks the random-walk threshold H = 0.5.**

Benign traffic registers a moderate SVG-domain Hurst exponent of 0.577, reflecting structural stability without strong long-range dependence in the topology. Combining the time-domain and SVG-domain Hurst exponents into a two-dimensional structural fingerprint produces a richer description than either coordinate alone (Lacasa & Toral, 2010). Attack families that are difficult to separate on time-domain Hurst alone become well separated when the SVG-domain Hurst is also considered. The fingerprint, in other words, lifts the diagnostic from a univariate to a bivariate scale and reveals structure that single-coordinate descriptions miss.

For risk managers, this is the sort of structural diagnostic that fits naturally into a dashboard (Cherdantseva et al., 2016; Lu et al., 2024b). A traffic regime can be plotted in the (H_original, H_SVG) plane, and excursions from the benign region can be flagged with confidence levels that reflect how far the current regime sits from the established baseline. Combined with the per-window classifier outputs, this dashboard supplies both a fast operational alert layer and a slower strategic review layer that can be used in incident post-mortems and resilience planning (Linkov et al., 2013).

### 5.5 Community Structure

The Louvain decomposition of the SVG networks reveals systematic differences in community structure between attack and benign regimes (Newman, 2003). Figure 8 reports the number of communities and the modularity Q for each traffic type. Benign traffic produces the largest number of communities (15) with a moderate modularity of 0.42. The structure is loose and heterogeneous, with no community dominating, which is consistent with benign traffic being a superposition of many independent activities such as control telemetry, periodic polling, and occasional reporting bursts.

DDoS attack traffic produces fewer communities (between 10 and 14) with substantially higher modularity (between 0.55 and 0.71). The pattern is consistent across attack families: programmatic traffic produces tight, well-separated phases that the Louvain algorithm assembles into modular communities (Watts & Strogatz, 1998). Synonymous IP Flood and TCP SYN Flood register the highest modularity, reflecting the strict periodic structure of these attacks (Mirkovic

& Reiher, 2004). Slowloris registers a comparatively low modularity for an attack regime, consistent with its by-design effort to mimic benign traffic patterns (Doshi et al., 2018). Even so, its modularity exceeds that of the benign series, providing a structural signal that reinforces the per-window classifier output.

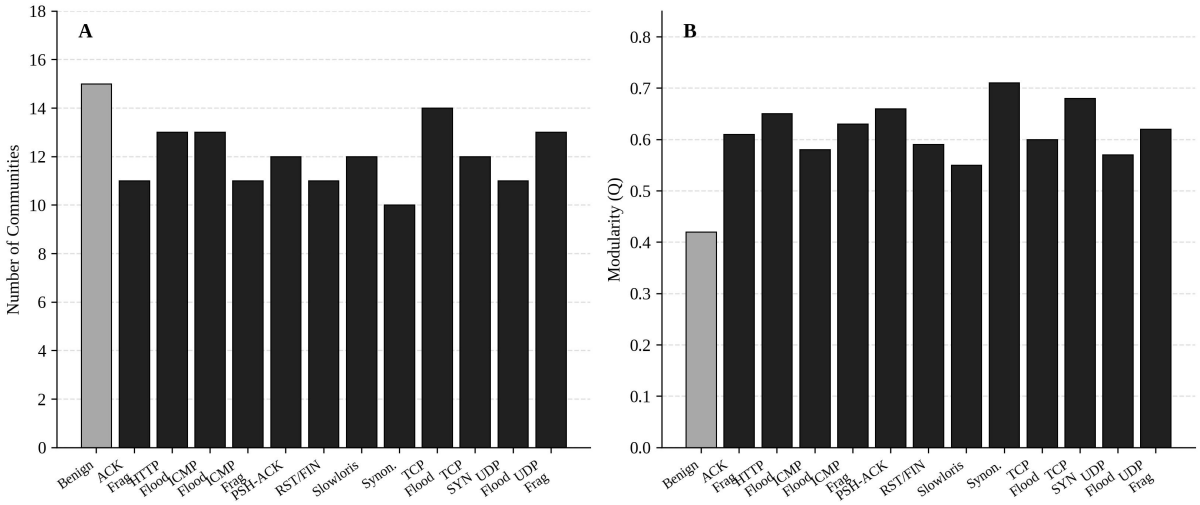


**Figure 8. Community structure metrics derived from the Louvain decomposition of the SVG networks for each traffic type. (A) Number of communities. (B) Modularity Q.**

**Table 6. SVG versus natural visibility graph construction comparison on a representative TCP Flood segment.**

| Method | Window W | Nodes | Edges | Avg. Degree | Time (ms) |
|---|---|---|---|---|---|
| SVG | 20 | 180 | 589 | 6.544 | 2.6 |
| SVG | 40 | 180 | 621 | 6.900 | 4.8 |
| SVG | 60 | 180 | 629 | 6.989 | 8.5 |
| SVG | 80 | 180 | 631 | 7.011 | 9.2 |
| SVG | 100 | 180 | 631 | 7.011 | 10.7 |
| Natural VG | 180 | 180 | 631 | 7.011 | 12.6 |

For management analytics, modularity provides a third structural axis that can be tracked alongside the power-law exponent and the bivariate Hurst signature (Lu, 2021). A sustained increase in modularity, particularly when accompanied by a decrease in the number of communities, points towards a transition from heterogeneous benign traffic to a more programmatic regime. The combined three-axis dashboard (γ, H, Q) summarises the structural evidence in a form that is interpretable to non-specialists and that aligns with established control-chart practice (Wang et al., 2016).

The community structures also reveal qualitative differences between flood and fragmentation attacks. Flood-type attacks (HTTP Flood, PSH-ACK Flood, TCP SYN Flood) form one large primary community with dense intra-community connections, reflecting their high-intensity sustained character. Low-rate attacks resemble benign traffic in number of communities but contain fewer of them, indicating that the attack source mix is narrower (Anthi et al., 2019). Fragmentation attacks (ACK Fragmentation Flood, ICMP Fragmentation Flood, UDP Fragmentation Flood) form intermediate community profiles with several mid-size communities reflecting the repetitive but structured behaviour of the fragmentation pattern. These qualitative

differences are useful for tier-2 attack-family classification, even though the present study focuses on binary detection.

Taken together, the community-level signatures complete a coherent picture of the structural fingerprint of DDoS attack traffic. The picture has three components: a tighter degree distribution with a steeper tail, a different (often anti-persistent) Hurst signature on the SVG degree sequence, and a more modular community structure with fewer communities. None of these signatures dominates in isolation, but together they explain why the fused detector improves over the statistical-only baseline and why the structural diagnostics provide leverage that purely amplitude-based summaries cannot offer (Mirsky et al., 2018; Shone et al., 2018). The fingerprint is also robust to the specific choice of attack family within the benign-versus-attack contrast, which makes it suitable as the foundation of an enterprise-level monitoring layer rather than a tactic-specific detection rule.

## 6. Managerial Discussion

The empirical results carry implications for both detection-system design and broader cyber-risk management practice (Lu, 2018; Eling & Wirfs, 2019). We organise the discussion around three management-relevant themes: alert-stream quality, structural diagnostics for executive dashboards, and the boundary conditions under which the proposed approach is most useful.

### 6.1 Alert-Stream Quality and the Cost of False Positives

The fused detector's precision of 1.0000 on the test set is operationally meaningful but should not be over-interpreted. Achieving perfect precision on a held-out test partition does not guarantee perfect precision in deployment, because deployment encounters traffic patterns that the training data may not represent (Sommer & Paxson, 2010). What the test result does establish is that the fused detector is biased towards conservatism: it suppresses false alarms aggressively, at the cost of missing approximately one in five attack windows. For many industrial settings this is the right bias. Operations managers consistently report that false alarm fatigue is the single most damaging failure mode of cyber detection systems, because it erodes the credibility of the alert stream and slows responses to genuine threats (Cavusoglu et al., 2004).

The structural features contribute disproportionately to suppressing false positives. Benign traffic windows that score high on individual statistical features (high variance, heavy tails, or pronounced skewness) often correspond to legitimate bursts such as bulk firmware pushes or end-of-shift telemetry flushes. The topological features carry information about how those bursts are organised, and the Louvain modularity in particular captures whether the burst is a single phase or part of a programmatic pattern (Donner et al., 2010). Combining the two suppresses the alarms that would otherwise be triggered by amplitude alone in legitimate burst regimes.

From an economic perspective, the value of high precision can be quantified through the analyst-time it saves (Gordon & Loeb, 2002; Biener et al., 2015). A typical industrial security operations centre running a less precise detector would spend a substantial share of its triage capacity on confirmed false alarms, with the cost compounded by the slowdown in response to genuine threats that follows from desensitised analysts. A detector that delivers precision close to one shifts the operating point on the precision-recall curve into a regime where the same staffing budget supports a meaningfully larger detection footprint. The economic argument is therefore consistent with the structural argument: suppressing false positives unlocks resources that can be redirected towards the deeper investigation of the harder, lower-rate attack scenarios that drive the residual missed-detection rate.

### 6.2 Structural Diagnostics for Executive Dashboards

The structural diagnostics introduced in Section 5 — power-law exponent, bivariate Hurst

signature, and community modularity — are well suited to executive-facing dashboards (Lu et al., 2024b; Provost & Fawcett, 2013). Each is a single number that summarises a topological aspect of the traffic regime, each can be estimated cheaply on rolling windows, and each has a straightforward operational interpretation that does not require specialised cybersecurity vocabulary. Tracking these three indicators over time, with control limits derived from baseline operations, provides a structural complement to the per-window classifier output that operates at the strategic rather than the tactical level (Mikalef et al., 2018).

This dashboard layer is consistent with the recommendation, recurrent in the operational risk literature, that cyber risk be integrated into the broader enterprise risk management apparatus rather than siloed within IT (Lu, 2021; Lu et al., 2024c). Structural indicators that look and behave like control charts are interpretable to risk managers, board-level audit committees, and external auditors, and they can be aggregated across plants or business units to produce enterprise-wide views (Wu et al., 2025). The fact that the same SVG construction supports both the tactical classifier and the strategic dashboard is a feature, not a coincidence: a single representation of the underlying traffic supplies both layers of the analytics stack.

### 6.3 Boundary Conditions and Limitations

Three boundary conditions deserve explicit mention. First, the proposed detector is trained on a single benchmark and may behave differently on traffic produced by other industrial estates with different application mixes, device populations, and baseline volumes (Tange et al., 2020; Hassan, 2019). Within-window normalisation reduces but does not eliminate this dependence, and any production deployment should retrain the classifier on local baseline data. Second, the structural diagnostics are most informative on stationary or piecewise-stationary regimes; rapid regime changes (such as plant restarts or planned maintenance windows) can produce structural signatures that resemble attacks (Bhuyan et al., 2014). A practical deployment should allow operators to mark such windows, either to exclude them from training or to give the classifier additional context features.

Third, the binary detection problem treated here is the simplest version of the operational task. Real industrial security operations centres need to assign incidents to attack families so that the correct response playbook can be invoked (Roy et al., 2010; Manshaei et al., 2013). Extending the fused feature representation to multi-class classification is straightforward conceptually but raises questions about class imbalance, label noise, and the calibration of confidence scores that fall outside the scope of the present study. We discuss these extensions in the conclusion.

Fourth, the detector evaluates each window independently. In practice, attack campaigns often unfold over several minutes or hours and produce a sequence of correlated windows, which means that a temporal smoothing of the per-window scores would likely improve recall further without sacrificing the precision the fused detector currently delivers (Lakhina et al., 2005). Such smoothing can be implemented through a simple cumulative-sum control chart on the SVM decision scores, which is a long-established tool in statistical process control and can be calibrated against historical baseline operations to set thresholds that reflect the local risk tolerance. We do not deploy such smoothing here in order to keep the per-window evaluation transparent, but we view it as an obvious extension that brings the detector closer to operational use.

### 6.4 Implications for Industrial Risk Governance

Beyond the immediate detection problem, the analysis carries implications for how industrial firms govern cyber risk at the enterprise level (Lu, 2017a; Sisinni et al., 2018). Three points are worth highlighting. First, the structural diagnostics provide a vocabulary that is robust to the rapid evolution of attack tactics. Specific attack families come and go (Kolias et al., 2017; De Donno et

al., 2018), but the structural concepts of burstiness, persistence, and modularity remain meaningful across attack generations. A risk governance framework anchored in these concepts is therefore more durable than one anchored in signatures of specific exploit techniques (Boyes et al., 2018).

Second, the use of an interpretable feature set facilitates the independent validation that audit committees and external assurance providers increasingly require (Wu et al., 2025; Lu, 2022). A detector whose features are listed and individually defensible can be audited; a detector that consumes thousands of opaque features extracted from a deep network cannot be audited in the same way without specialised expertise (Vinayakumar et al., 2019). The SVG-based feature set therefore aligns with the broader trend in operational risk towards explainable analytics, which is itself a response to regulatory and governance pressures that have intensified in recent years (Lu, 2019b; Zheng & Lu, 2022).

Third, the cost of running the SVG detector is low enough that it can be deployed at the edge of the IIoT network rather than only at the centre. This matters for resilience, because edge deployment means that detection continues to function even if the link between the plant and the central security operations centre is degraded by the very attack the system is trying to identify (Linkov et al., 2013; Diro & Chilamkurti, 2018). Edge deployment also reduces the volume of telemetry that needs to be exfiltrated for analysis, which has both performance and privacy implications in the increasingly regulated IIoT environment (Xu et al., 2021; Khan & Salah, 2018). The combination of low computational footprint, interpretability, and edge deployability is unusual in the cyber-detection literature and is a direct consequence of the structural-feature design choice.

## 7. Conclusion

This paper has developed a management analytics pipeline (Lu, 2021; Lu et al., 2024c) for IIoT cyber-risk detection that fuses statistical and topological descriptors of network traffic time series. The topological descriptors are derived from the Sliding Visibility Graph, which maps each segmented packet-rate window into a complex network in linear time (Donner et al., 2010; Zou et al., 2019). The fused detector achieves an accuracy of 97.16% and an F1-score of 89.54% on a recent IIoT benchmark, materially exceeding threshold-based, entropy-based, and pure-statistics baselines while preserving the precision needed to keep the alert stream credible to operations managers (Buczak & Guven, 2016).

Beyond classification, the study has documented systematic structural differences between attack and benign regimes. The macroscopic degree distribution of DDoS traffic exhibits a steeper power-law tail ($\gamma \approx 2.29$) than benign traffic ($\gamma \approx 1.79$), reflecting the suppression of ultra-high-degree nodes in dense attack bursts (Newman, 2003). Hurst-exponent estimates on the SVG degree sequence reveal divergent long-range-dependence behaviour across attack families, with HTTP Flood standing out for its anti-persistent topology and TCP SYN Flood producing the most persistent (Karagiannis et al., 2004). Community-detection analysis shows that DDoS traffic is more modular than benign traffic, with fewer but more cohesive communities reflecting the programmatic nature of attack behaviour.

For management analytics practice (Lu, 2018; Mikalef et al., 2018), the contribution is twofold. The fused detector provides a tactical alerting layer that is precise, recall-aware, and edge-deployable. The structural diagnostics — power-law exponent, bivariate Hurst signature, and modularity — supply a strategic dashboard layer that translates topological evidence into a vocabulary risk managers and audit committees can act on (Lu et al., 2024b; Provost & Fawcett, 2013). Taken together, the two layers constitute an analytics-led detection apparatus that aligns with the recommendation in the operational risk literature that cyber risk be governed within the same disciplines used for other enterprise risks (Eling & Wirfs, 2019; Cherdantseva et al., 2016).

Several extensions deserve attention. First, the current binary detector should be generalised to multi-class attack-family classification, which would close the gap between the structural fingerprints documented here and the response-playbook routing that security operations centres need (Anthi et al., 2019). Second, the chronological evaluation protocol should be augmented with rolling-origin cross-validation and bootstrapped confidence intervals to provide uncertainty quantification that is robust to the time-series structure of the data (Tavallaee et al., 2009). Third, the fused detector should be tested on additional IIoT benchmarks and, ideally, in an industrial pilot deployment, to assess the transferability of both the classifier and the structural diagnostics across heterogeneous operating environments (Wollschlaeger et al., 2017; Lu, 2025).

Fourth, the present study uses a Support Vector Machine for transparency and edge-deployability (Cortes & Vapnik, 1995), but the fused feature representation is compatible with deeper models that may extract higher-order interactions among the statistical and topological features (Vinayakumar et al., 2019; Shone et al., 2018). A controlled comparison between the SVM and a small neural model would clarify how much additional accuracy can be gained without sacrificing interpretability. Fifth, an interesting direction is to integrate the SVG detector with blockchain-based provenance and integrity tracking for IIoT telemetry (Lu, 2019b; Xu et al., 2021; Chen et al., 2024), which would close the gap between detection and post-incident forensics. Finally, as quantum and AI computing platforms continue to mature (Lu et al., 2024a; Lu et al., 2023; Ye & Lu, 2022; Zhang & Lu, 2021; Lu, 2019a), and as next-generation networking standards such as 6G expand the IIoT attack surface (Lu & Zheng, 2020; Lu & Ning, 2020), the structural diagnostics developed here should be benchmarked against the established self-similarity and entropy descriptors used in the cyber-risk-management literature (Bhuyan et al., 2014; Chandola et al., 2009), to determine which combination of indicators provides the strongest input to enterprise-level risk dashboards. Adjacent fintech and Web 3.0 settings, where similar denial-of-service patterns surface against decentralised infrastructure (Kou & Lu, 2025; Xu et al., 2024; Yang et al., 2025; Zhang & Lu, 2025; Lu & Yang, 2024; Lu et al., 2020), present a further set of testbeds that connect the present industrial focus to the broader management-analytics agenda. We see these directions as natural follow-ups that build on the present results without altering the conceptual foundation of the approach.

## Acknowledgement

The authors thank the Canadian Institute for Cybersecurity for making the CIC IIoT Dataset 2025 publicly available, and the anonymous reviewers whose suggestions sharpened the management analytics framing of this paper. Any remaining errors are solely the responsibility of the authors.

## References

Anthi, E., Williams, L., Słowińska, M., Theodorakopoulos, G., & Burnap, P. (2019). A supervised intrusion detection system for smart home IoT devices. IEEE Internet of Things Journal, 6(5), 9042 – 9053. https://doi.org/10.1109/JIOT.2019.2926365

Bertino, E., & Islam, N. (2017). Botnets and Internet of Things security. Computer, 50(2), 76 – 79. https://doi.org/10.1109/MC.2017.62

Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2014). Network anomaly detection: Methods, systems and tools. IEEE Communications Surveys & Tutorials, 16(1), 303 – 336. https://doi.org/10.1109/SURV.2013.052213.00046

Biener, C., Eling, M., & Wirfs, J. H. (2015). Insurability of cyber risk: An empirical analysis. The Geneva Papers on Risk and Insurance — Issues and Practice, 40(1), 131 – 158. https://doi.org/10.1057/gpp.2014.19

Boyes, H., Hallaq, B., Cunningham, J., & Watson, T. (2018). The industrial Internet of Things (IIoT): An

analysis framework. Computers in Industry, 101, 1–12. https://doi.org/10.1016/j.compind.2018.04.015

Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153 – 1176. https://doi.org/10.1109/COMST.2015.2494502

Cavusoglu, H., Mishra, B., & Raghunathan, S. (2004). The effect of Internet security breach announcements on market value: Capital market reactions for breached firms and Internet security developers. International Journal of Electronic Commerce, 9(1), 70 – 104. https://doi.org/10.1080/10864415.2004.11044320

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 15:1–15:58. https://doi.org/10.1145/1541880.1541882

Chen, Y., Lu, Y., Bulysheva, L., & Kataev, M. Y. (2024). Applications of blockchain in Industry 4.0: A review. Information Systems Frontiers, 26(5), 1715–1729. https://doi.org/10.1007/s10796-022-10248-7

Cherdantseva, Y., Burnap, P., Blyth, A., Eden, P., Jones, K., Soulsby, H., & Stoddart, K. (2016). A review of cyber security risk assessment methods for SCADA systems. Computers & Security, 56, 1 – 27. https://doi.org/10.1016/j.cose.2015.09.009

Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273 – 297. https://doi.org/10.1007/BF00994018

Crovella, M. E., & Bestavros, A. (1997). Self-similarity in World Wide Web traffic: Evidence and possible causes. IEEE/ACM Transactions on Networking, 5(6), 835–846. https://doi.org/10.1109/90.650143

De Donno, M., Dragoni, N., Giaretta, A., & Spognardi, A. (2018). DDoS-capable IoT malwares: Comparative analysis and Mirai investigation. Security and Communication Networks, 2018, 7178164. https://doi.org/10.1155/2018/7178164

Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761 – 768. https://doi.org/10.1016/j.future.2017.08.043

Donner, R. V., Zou, Y., Donges, J. F., Marwan, N., & Kurths, J. (2010). Recurrence networks—a novel paradigm for nonlinear time series analysis. New Journal of Physics, 12(3), 033025. https://doi.org/10.1088/1367-2630/12/3/033025

Doriguzzi-Corin, R., Millar, S., Scott-Hayward, S., Martínez-del-Rincón, J., & Siracusa, D. (2020). LUCID: A practical, lightweight deep learning solution for DDoS attack detection. IEEE Transactions on Network and Service Management, 17(2), 876–889. https://doi.org/10.1109/TNSM.2020.2971776

Doshi, R., Apthorpe, N., & Feamster, N. (2018). Machine learning DDoS detection for consumer Internet of Things devices. 2018 IEEE Security and Privacy Workshops, 29 – 35. https://doi.org/10.1109/SPW.2018.00013

Eling, M., & Wirfs, J. (2019). What are the actual costs of cyber risk events? European Journal of Operational Research, 272(3), 1109–1119. https://doi.org/10.1016/j.ejor.2018.07.021

Gordon, L. A., & Loeb, M. P. (2002). The economics of information security investment. ACM Transactions on Information and System Security, 5(4), 438 – 457. https://doi.org/10.1145/581271.581274

Hassan, W. H. (2019). Current research on Internet of Things (IoT) security: A survey. Computer Networks, 148, 283–294. https://doi.org/10.1016/j.comnet.2018.11.025

Hodo, E., Bellekens, X., Hamilton, A., Dubouilh, P. L., Iorkyase, E., Tachtatzis, C., & Atkinson, R. (2016). Threat analysis of IoT networks using artificial neural network intrusion detection system. 2016

International Symposium on Networks, Computers and Communications. https://doi.org/10.1109/ISNCC.2016.7746067

Karagiannis, T., Molle, M., Faloutsos, M., & Broido, A. (2004). A nonstationary Poisson view of Internet traffic. IEEE INFOCOM 2004, 3, 1558–1569. https://doi.org/10.1109/INFCOM.2004.1354587

Khan, M. A., & Salah, K. (2018). IoT security: Review, blockchain solutions, and open challenges. Future Generation Computer Systems, 82, 395–411. https://doi.org/10.1016/j.future.2017.11.022

Kolias, C., Kambourakis, G., Stavrou, A., & Voas, J. (2017). DDoS in the IoT: Mirai and other botnets. Computer, 50(7), 80–84. https://doi.org/10.1109/MC.2017.201

Kou, G., & Lu, Y. (2025). FinTech: A literature review of emerging financial technologies and applications. Financial Innovation, 11(1), 1–34. https://doi.org/10.1186/s40854-024-00668-6

Lacasa, L., & Toral, R. (2010). Description of stochastic and chaotic series using visibility graphs. Physical Review E, 82(3), 036120. https://doi.org/10.1103/PhysRevE.82.036120

Lakhina, A., Crovella, M., & Diot, C. (2005). Mining anomalies using traffic feature distributions. ACM SIGCOMM Computer Communication Review, 35(4), 217 – 228. https://doi.org/10.1145/1080091.1080118

Leland, W. E., Taqqu, M. S., Willinger, W., & Wilson, D. V. (1994). On the self-similar nature of Ethernet traffic. IEEE/ACM Transactions on Networking, 2(1), 1–15. https://doi.org/10.1109/90.282603

Linkov, I., Eisenberg, D. A., Plourde, K., Seager, T. P., Allen, J., & Kott, A. (2013). Resilience metrics for cyber systems. Environment Systems and Decisions, 33(4), 471–476. https://doi.org/10.1007/s10669-013-9485-y

Lu, W., Lu, Y., Li, J., Sigov, A., Ratkin, L., & Ivanov, L. A. (2024). Quantum machine learning: Classifications, challenges, and solutions. Journal of Industrial Information Integration, 42, 100736. https://doi.org/10.1016/j.jii.2024.100736

Lu, Y. (2017). Cyber physical system (CPS)-based Industry 4.0: A survey. Journal of Industrial Integration and Management, 2(3), 1750014. https://doi.org/10.1142/S2424862217500142

Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. Journal of Industrial Information Integration, 6, 1–10. https://doi.org/10.1016/j.jii.2017.04.005

Lu, Y. (2018). Blockchain and the related issues: A review of current research topics. Journal of Management Analytics, 5(4), 231–255. https://doi.org/10.1080/23270012.2018.1516523

Lu, Y. (2019). Artificial intelligence: A survey on evolution, models, applications and future trends. Journal of Management Analytics, 6(1), 1–29. https://doi.org/10.1080/23270012.2019.1570365

Lu, Y. (2019). The blockchain: State-of-the-art and research challenges. Journal of Industrial Information Integration, 15, 80–90. https://doi.org/10.1016/j.jii.2019.04.002

Lu, Y. (2021). Technological innovation and the emergence of a new interdisciplinary field: Management analytics. Nanotechnologies in Construction, 13(3), 181 – 192. https://doi.org/10.15828/2075-8545-2021-13-3-181-192

Lu, Y. (2022). Implementing blockchain in information systems: A review. Enterprise Information Systems, 16(12), 1876–1907. https://doi.org/10.1080/17517575.2021.2008513

Lu, Y. (2025). The current status and developing trends of Industry 4.0: A review. Information Systems Frontiers, 27(1), 215–234. https://doi.org/10.1007/s10796-021-10221-w

Lu, Y., & Ning, X. (2020). A vision of 6G – 5G's successor. Journal of Management Analytics, 7(3), 301–320. https://doi.org/10.1080/23270012.2020.1802622

Lu, Y., & Xu, L. D. (2019). Internet of Things (IoT) cybersecurity research: A review of current research

topics. IEEE Internet of Things Journal, 6(2), 2103–2115. https://doi.org/10.1109/JIOT.2018.2869847

Lu, Y., & Yang, J. (2024). Quantum financing system: A survey on quantum algorithms, potential scenarios and open research issues. Journal of Industrial Information Integration, 41, 100663. https://doi.org/10.1016/j.jii.2024.100663

Lu, Y., & Zheng, X. (2020). 6G: A survey on technologies, scenarios, challenges, and the related issues. Journal of Industrial Information Integration, 19, 100158. https://doi.org/10.1016/j.jii.2020.100158

Lu, Y., Ivanov, L. A., Wang, F., Pisarenko, Z. V., & Ye, C. (2024). Management analytics: A bibliometric analysis. Nanotechnologies in Construction, 16(3), 257 – 266. https://doi.org/10.15828/2075-8545-2024-16-3-257-266

Lu, Y., Pisarenko, Z. V., Yang, L., & Ye, C. (2024). Advancing decision-making: The role of management analytics in modern business practices. Nanotechnologies in Construction, 16(5), 431 – 440. https://doi.org/10.15828/2075-8545-2024-16-5-431-440

Lu, Y., Sigov, A. S., Ratkin, L., Ivanov, L. A., & Zuo, M. (2023). Quantum computing and industrial information integration: A review. Journal of Industrial Information Integration, 35, 100511. https://doi.org/10.1016/j.jii.2023.100511

Lu, Y., Zheng, X., Li, L., & Xu, L. D. (2020). Pricing the cloud: A QoS-based auction approach. Enterprise Information Systems, 14(3), 334–351. https://doi.org/10.1080/17517575.2019.1669827

Luque, B., Lacasa, L., Ballesteros, F., & Luque, J. (2009). Horizontal visibility graphs: Exact results for random time series. Physical Review E, 80(4), 046103. https://doi.org/10.1103/PhysRevE.80.046103

Manshaei, M. H., Zhu, Q., Alpcan, T., Bacşar, T., & Hubaux, J. P. (2013). Game theory meets network security and privacy. ACM Computing Surveys, 45(3), 25:1 – 25:39. https://doi.org/10.1145/2480741.2480742

Mikalef, P., Pappas, I. O., Krogstie, J., & Giannakos, M. (2018). Big data analytics capabilities: A systematic literature review and research agenda. Information Systems and e-Business Management, 16(3), 547–578. https://doi.org/10.1007/s10257-017-0362-y

Mirkovic, J., & Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, 34(2), 39–53. https://doi.org/10.1145/997150.997156

Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: An ensemble of autoencoders for online network intrusion detection. NDSS Symposium 2018. https://doi.org/10.14722/ndss.2018.23204

Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. 2015 Military Communications and Information Systems Conference. https://doi.org/10.1109/MilCIS.2015.7348942

Newman, M. E. J. (2003). The structure and function of complex networks. SIAM Review, 45(2), 167–256. https://doi.org/10.1137/S003614450342480

Provost, F., & Fawcett, T. (2013). Data science and its relationship to big data and data-driven decision making. Big Data, 1(1), 51–59. https://doi.org/10.1089/big.2013.1508

Roy, S., Ellis, C., Shiva, S., Dasgupta, D., Shandilya, V., & Wu, Q. (2010). A survey of game theory as applied to network security. 2010 43rd Hawaii International Conference on System Sciences. https://doi.org/10.1109/HICSS.2010.35

Saied, A., Overill, R. E., & Radzik, T. (2016). Detection of known and unknown DDoS attacks using artificial neural networks. Neurocomputing, 172, 385 – 393. https://doi.org/10.1016/j.neucom.2015.04.101

Sengupta, J., Ruj, S., & Bit, S. D. (2020). A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT. Journal of Network and Computer Applications, 149, 102481. https://doi.org/10.1016/j.jnca.2019.102481

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. ICISSP 2018, 108 – 116. https://doi.org/10.5220/0006639801080116

Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. IEEE Transactions on Emerging Topics in Computational Intelligence, 2(1), 41 – 50. https://doi.org/10.1109/TETCI.2017.2772792

Sisinni, E., Saifullah, A., Han, S., Jennehag, U., & Gidlund, M. (2018). Industrial Internet of Things: Challenges, opportunities, and directions. IEEE Transactions on Industrial Informatics, 14(11), 4724–4734. https://doi.org/10.1109/TII.2018.2852491

Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. 2010 IEEE Symposium on Security and Privacy, 305 – 316. https://doi.org/10.1109/SP.2010.25

Stellios, I., Kotzanikolaou, P., Psarakis, M., Alcaraz, C., & Lopez, J. (2018). A survey of IoT-enabled cyberattacks: Assessing attack paths to critical infrastructures and services. IEEE Communications Surveys & Tutorials, 20(4), 3453–3495. https://doi.org/10.1109/COMST.2018.2855563

Tan, Z., Jamdagni, A., He, X., Nanda, P., & Liu, R. P. (2014). A system for denial-of-service attack detection based on multivariate correlation analysis. IEEE Transactions on Parallel and Distributed Systems, 25(2), 447–456. https://doi.org/10.1109/TPDS.2013.146

Tange, K., De Donno, M., Fafoutis, X., & Dragoni, N. (2020). A systematic survey of industrial Internet of Things security: Requirements and fog computing opportunities. IEEE Communications Surveys & Tutorials, 22(4), 2489–2520. https://doi.org/10.1109/COMST.2020.3011208

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications. https://doi.org/10.1109/CISDA.2009.5356528

Ten, C. W., Liu, C. C., & Manimaran, G. (2008). Vulnerability assessment of cybersecurity for SCADA systems. IEEE Transactions on Power Systems, 23(4), 1836 – 1846. https://doi.org/10.1109/TPWRS.2008.2002298

Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. IEEE Access, 7, 41525 – 41550. https://doi.org/10.1109/ACCESS.2019.2895334

Wang, G., Gunasekaran, A., Ngai, E. W. T., & Papadopoulos, T. (2016). Big data analytics in logistics and supply chain management: Certain investigations for research and applications. International Journal of Production Economics, 176, 98–110. https://doi.org/10.1016/j.ijpe.2016.03.014

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. Nature, 393(6684), 440–442. https://doi.org/10.1038/30918

Wollschlaeger, M., Sauter, T., & Jasperneite, J. (2017). The future of industrial communication: Automation networks in the era of the Internet of Things and Industry 4.0. IEEE Industrial Electronics Magazine, 11(1), 17–27. https://doi.org/10.1109/MIE.2017.2649104

Wu, H. P., Liu, Z., Dong, H. Y., Lu, Y., & Xu, L. D. (2025). Revolutionizing internal auditing: Harnessing the power of blockchain. Enterprise Information Systems, 19(1 – 2). https://doi.org/10.1080/17517575.2024.2448003

Xu, L. D., Lu, Y., & Li, L. (2021). Embedding blockchain technology into IoT for security: A survey. IEEE Internet of Things Journal, 8(13), 10452–10473. https://doi.org/10.1109/JIOT.2021.3060508

Xu, R., Zhu, J., Yang, L., Lu, Y., & Xu, L. D. (2024). Decentralized finance (DeFi): A paradigm shift in the FinTech. Enterprise Information Systems, 18(9). https://doi.org/10.1080/17517575.2024.2397630

Yang, L., Hou, Q., Zhu, X., Lu, Y., & Xu, L. D. (2025). Potential of large language models in blockchain-based supply chain finance. Enterprise Information Systems, 19(11), 2541199. https://doi.org/10.1080/17517575.2025.2541199

Ye, Z., & Lu, Y. (2022). Quantum science: A review and current research trends. Journal of Management Analytics, 9(3), 383–402. https://doi.org/10.1080/23270012.2022.2089064

Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access, 5, 21954–21961. https://doi.org/10.1109/ACCESS.2017.2762418

Zhang, C., & Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. Journal of Industrial Information Integration, 23, 100224. https://doi.org/10.1016/j.jii.2021.100224

Zhang, H., & Lu, Y. (2025). Web 3.0: Applications, opportunities and challenges in the next internet generation. Systems Research and Behavioral Science, 42(4), 996 – 1015. https://doi.org/10.1002/sres.3068

Zheng, X. R., & Lu, Y. (2022). Blockchain technology: Recent research and future trend. Enterprise Information Systems, 16(12), 1939895. https://doi.org/10.1080/17517575.2021.1939895

Zou, Y., Donner, R. V., Marwan, N., Donges, J. F., & Kurths, J. (2019). Complex network approaches to nonlinear time series analysis. Physics Reports, 787, 1 – 97. https://doi.org/10.1016/j.physrep.2018.10.005