

Network Digital Twins with Concept Drift Detectors: A Comparative Analysis

Marcelo Arbage, Gabriel Sousa, Lucas Matni, Cláudio Modesto and Glauco Gonçalves

Abstract—In a Network Digital Twin (NDT), unsupervised concept drift detection triggers crucial model updates. This paper evaluates Kolmogorov-Smirnov Windowing (KSWIN), Adaptive Windowing (ADWIN), and Page-Hinkley (PH) on a simulated traffic stream. Our sensitivity analysis reveals that window-based detectors are highly susceptible to the alignment between buffer sizes and the stream’s stationary periods. KSWIN and PH showed apparent optimal performance at a 7 100 window merely by synchronizing with the distribution’s duration, but failed completely at 9 000. ADWIN remained overly conservative, risking under-detection. Calibrating unsupervised detectors requires prior knowledge of the network’s stationarity to prevent critical failures in operational NDTs.

Keywords—Network digital twin, Concept drift, Unsupervised detection.

I. INTRODUCTION

Modern communication networks increasingly rely on Machine Learning (ML) to improve monitoring, management, and automation, especially in transport-network scenarios [1]. However, network telemetry is naturally generated as a continuous and non-stationary stream. As a consequence, the joint data distribution may change over time, progressively degrading the performance of deployed models. In contrast to momentary outliers, this longer-lasting distributional change is known as concept drift [2].

In this context, an NDT is a dynamic digital representation of a telecommunication network that replicates its topology, configuration, behavior, and operational state in real or near-real time [1]. By continuously synchronizing the physical domain Physical Twin (PTwin), which generates the raw data, and the virtual counterpart Virtual Twin (VTwin), which consumes this data to predict a certain phenomenon, an NDT enables safer evaluation of control decisions, earlier detection of anomalies, and more robust automation without interrupting the live infrastructure. However, decisions generated by the virtual counterpart must remain aligned with the physical network state. If a concept drift is not detected in time, the VTwin may issue unreliable predictions and trigger inadequate control actions.

Therefore, concept drift detectors are required to indicate when model behavior is no longer consistent with current network dynamics and when retraining should be considered. However, ground-truth output labels are rarely available or are prohibitively expensive to obtain in operational NDT

deployments. Because of this limitation, retraining decisions must be based solely on changes in input features. This specific scenario, where only the marginal input distribution shifts while the conditional relationship remains stable, is referred to as covariate drift.

The relevance of Digital Twin (DT) technology has been widely discussed in the literature. Building on this concept, Modesto et al.[3] proposed an NDT architecture that uses KSWIN as the sole concept drift detector. However, the choice of the detector and window configuration significantly affects alarm behavior, and no comparative study has been conducted for this scenario. This paper aims to evaluate three unsupervised concept drift detectors from the *river* library: KSWIN, ADWIN, and PH, under a univariate distribution-based setup, where alarms are triggered from changes in the input traffic feature stream.

These detectors were chosen because they cover the main paradigms of unsupervised change detection, enabling a comprehensive comparison. First, KSWIN is based on the Kolmogorov-Smirnov (KS) statistical test, which compares two sample windows to identify distributional changes [4]. Its sensitivity is controlled by a significance level α , and the window size sets the amount of historical context available. Second, ADWIN maintains a variable-length window and continuously checks whether two subwindows have statistically different means [5]. When a significant difference is found, the older portion is discarded, indicating a concept drift. A confidence parameter δ controls its test sensitivity. Finally, the PH detector is a sequential change-detection method based on cumulative deviations from the running mean [6]. As a sequential method without a fixed observation window, PH often reacts quickly to abrupt changes but may become more sensitive to short-term fluctuations if not properly tuned.

II. EVALUATION METHOD

Our evaluation follows the NDT workflow proposed in [3], where the VTwin is continuously synchronized with the PTwin. In this environment, network traffic features are streamed and monitored to assess changes in input distribution under non-stationary conditions.

The objective is to analyze how the window size influences the number of detected concept drifts. We compare three detectors from the *river* library under a univariate distribution-based monitoring regime, with configurations as specified in Table I.

All the KSWIN parameters were set to be as they are in [3], the other detectors’ parameters were set to be as

M. Arbage, G. Sousa, L. Matni, C. Modesto and G. Gonçalves
E-mails: marcelo.arbage@itec.ufpa.br, gabriel.vinicius@itec.ufpa.br,
lucas.matni@itec.ufpa.br, claudio.barata@itec.ufpa.br, glauco-goncalves@ufpa.br.

TABLE I: Configured parameters for the concept drift detectors.

Detector	Parameter	Value / Config.
KSWIN	Alpha (α)	0.001
	Window size (w)	Variable [100–9900]
	Stat. size (stat_size)	$w/4$
	Seed	42
Page-Hinkley	Delta (δ)	0.001
	Threshold (λ)	50 (default)
	Min. instances (min_instances)	Variable [100–9900]
ADWIN	Delta (δ)	0.001
	Min. window length (min_length)	Variable [100–9900]

close to equivalent to KSWIN parameters as possible, and the other parameters were set at their default values in *river* for more reliable, general results. We progressively increase the window size by 100 in each verification and record the number of detector alarms at each operating point. Furthermore, it is important to reinforce that the window parameter plays structurally different roles in each method: controlling the KS test window in KSWIN, the minimum subwindow size before splitting in ADWIN, and the minimum number of instances before allowing detection in PH. Therefore, while plotting them on the same axis is useful for behavioral comparison, their equivalence is not absolute; increasing this parameter consistently reduces sensitivity across all three architectures.

In this paper, deviations are identified at timestamps where detectors trigger alarms over this univariate traffic-rate stream, and it is covered by a dataset corresponding to 4 traffic patterns evaluated in the Germany topology scenario from [3], comprising 50 nodes and 224 links with a total stream duration of approximately 6.8 hours. The network data were collected at 1-second granularity, yielding 24 600 discrete instances. The stream portrays three abrupt concept drift events: the first occurs when the traffic pattern distribution switches from Exponential to Poisson, the second when it switches to a Uniform pattern, and the third when this pattern transitions to a Congested form. These events demonstrate the actual concept drifts to better interpret the detector alarm behavior across the evaluated window configurations. In this NDT scenario, the model is designed to predict network performance metrics, such as end-to-end delay, which function as the target labels. As it is in [3], detector inputs are derived from the average flow traffic rate, a covariate feature of the stream, rather than the target performance metrics themselves. This instantiates the covariate drift monitoring regime introduced in Section I.

This controlled comparison ensures that observed differences in alarm counts are attributable to detector logic and window configuration. To evaluate the performance at each operating point, two metrics are analyzed: the total number of triggered alarms and their temporal alignment. The total alarm count indicates the detector’s overall sensitivity across the stream. Meanwhile, temporal alignment is evaluated by cross-referencing the alarm timestamps with the three known abrupt concept drift events, distinguishing actual concept drift

detections from false alarms. To formalize this, an alarm triggered at timestamp t_a is considered a True Positive (TP) if it occurs within a bidirectional tolerance margin $[t_d - \Delta t, t_d + \Delta t]$, where t_d is the actual drift index. Based on the system’s operational constraints, we set $\Delta t = 200$ instances. Repeated alarms within the same margin for a single drift event are treated as a single TP to avoid double-counting. Alarms outside this margin are considered False Positives (FPs), and actual events without corresponding alarms within the margin are computed as False Negatives (FNs).

III. RESULTS

Applying the parameter sensitivity analysis, a consistent trend is observed across all methods: as the window size increases, the number of detector alarms decreases, as shown in Table II. This behavior reflects that smaller windows react to short-term fluctuations, raising alarm counts well above the number of known concept drift events, which translates to a high rate of FPs. In comparison, larger windows suppress transient alarms at the cost of slower reaction, increasing the risk of FNs. The timestamps of the concept drift detected by the detectors are shown in Figure 1.

TABLE II: Number of Concept Drifts detected for each Window size at a 200 bidirectional margin.

Window Size	KSWIN (TP / FP / FN)	Page-Hinkley (TP / FP / FN)	ADWIN (TP / FP / FN)
100	3 / 143 / 0	3 / 164 / 0	3 / 51 / 0
1000	1 / 23 / 2	1 / 23 / 2	1 / 9 / 2
2500	0 / 9 / 3	0 / 9 / 3	0 / 4 / 3
5000	0 / 4 / 3	3 / 4 / 3	0 / 2 / 3
7100	3 / 0 / 0	3 / 0 / 0	0 / 1 / 3
9000	0 / 2 / 3	0 / 2 / 3	0 / 1 / 3

While the general trend suggests alarm suppression, Table II reveals critical non-monotonic behaviors. At window 2 500, all detectors produce counts well above the expected minimum (9, 9, and 4 for KSWIN, PH, and ADWIN), indicating a high False Positive (FP) burden that would trigger unnecessary retraining in an NDT deployment. Above window 5 000, detectors begin accumulating FNs. However, the empirical KSWIN window of 7 100 yields 3 detections, effectively capturing all True Positives (TPs) while minimizing FPs (3 TPs, 0 FPs, 0 FNs). This sudden increase in accuracy contradicts the monotonic degradation expected with larger windows. We attribute this to a structural artifact of the dataset resonating with the window size. The distance between stationary distribution shifts in this stream is approximately 7 100 instances. Consequently, at $w = 7 100$, the algorithms’ buffers saturate almost exactly as the actual concept drift occurs, causing the detectors to act as synchronized timers rather than dynamically reacting to the data.

This synchronization explains the severe degradation at window 9 000. Because the window boundary overflows the stationary traffic blocks, KSWIN and PH fail (0 TPs, 3 FNs). The 2 alarms reported at this scale are strictly FPs, triggered by the internal high-variance noise evaluated within a misaligned

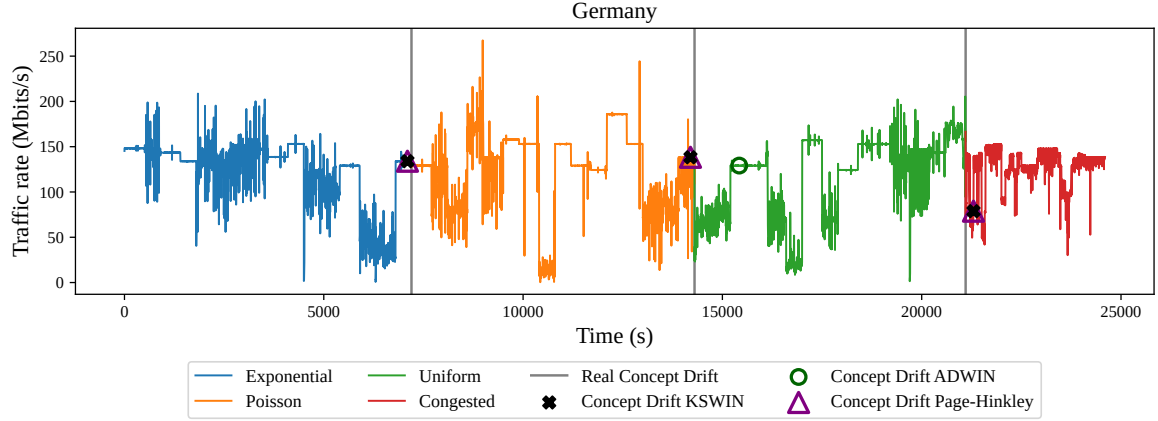


Fig. 1: Alarm events triggered by KSWIN, PH, and ADWIN over the Germany topology average traffic-rate stream on a 7 100 window size.

buffer. Therefore, treating the reduction of alarms at $w = 9\,000$ as a stabilization is misleading, as it actually represents a complete loss of detection capability.

Conversely, ADWIN demonstrates a structurally conservative profile. It produces only 1 alarm at both windows 7 100 and 9 000, indicating its alarm count has already plateaued. As visible in Figure 1, this single alarm appears temporally aligned with the second distribution shift, confirming that the first, second, and third concept drift events were missed FNs. Imposing a minimum subwindow limit of 7 100 forces ADWIN to assimilate structural drifts as normal long-term variance, confirming its risk of under-detection at large windows.

A notable observation is that KSWIN and PH produce nearly identical counts at every tested point, despite relying on fundamentally different statistical mechanisms. This convergence suggests, as a preliminary hypothesis, that the high variance and signal intensity in this stream might temporarily overpower the mathematical differences between the distinct architectures, causing both to be similarly constrained by their fixed buffer sizes. These findings align with the mechanisms described in Section I, highlighting that mindlessly scaling window parameters without considering the physical network’s stationary periods risks critical under-detection in NDT deployments.

IV. CONCLUSIONS

This paper presented an initial comparative analysis of three concept drift detectors from the *river* library in an NDT scenario. The experiments showed that window size is a key control parameter: increasing the window reduces alarms, while smaller windows improve responsiveness but increase sensitivity to normal fluctuations. ADWIN is consistently conservative, producing only 1 alarm from window 7 100 onward, which confirms its risk of extreme under-detection at large window scales, missing multiple concept drift events. KSWIN and PH exhibit nearly identical alarm profiles, which we hypothesize is due to the stream’s high variance overpowering the differences in their statistical mechanisms.

Our formal evaluation revealed that static window scaling introduces severe reliability risks. The apparent optimal performance of KSWIN and PH at $w = 7\,100$ was a structural artifact of the dataset’s specific distribution-shift interval, leading the algorithms to act as synchronized timers. When misaligned ($w = 9\,000$), they missed all structural drifts. For practical NDT deployments, mindlessly scaling window parameters to reduce retraining overhead risks detaching the VTwin from the physical network reality.

A key limitation is that experiments were conducted only on the Germany topology. As future work, we plan to validate these findings across additional topologies and include jitter metrics to achieve a comprehensive evaluation for adaptive parameter-tuning strategies.

ACKNOWLEDGMENTS

The authors would like to thank the National Council for Scientific and Technological Development (CNPq) (process: 408326/2023-9) and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

REFERENCES

- [1] B. R. Barricelli, E. Casiraghi, and D. Fogli, “A survey on digital twin: Definitions, characteristics, applications, and design implications,” *IEEE Access*, vol. 7, pp. 167 653–167 671, 2019.
- [2] D. Lukats, O. Zielinski, A. Hahn, and F. Stahl, “A benchmark and survey of fully unsupervised concept drift detectors on real-world data streams,” *International Journal of Data Science and Analytics*, vol. 19, no. 1, pp. 1–31, 2025.
- [3] C. Modesto, J. Borges, C. Nahum, L. Matni, C. B. Both, K. Cardoso, G. Gonçalves, I. Correa, S. Lins, A. Silva *et al.*, “Towards a Robust Transport Network With Self-Adaptive Network Digital Twin,” *Computer Networks*, vol. 276, p. 111967, 2025.
- [4] C. Raab, M. Heusinger, and F.-M. Schleif, “Reactive soft prototype computing for concept drift streams,” *Neurocomputing*, vol. 416, pp. 340–351, 2020.
- [5] A. Bifet and R. Gavaldà, “Learning from time-changing data with adaptive windowing,” in *Proceedings of the 2007 SIAM International Conference on Data Mining*. SIAM, 2007, pp. 443–448.
- [6] R. Sebastião and J. M. Fernandes, “Supporting the Page-Hinkley test with empirical mode decomposition for change detection,” in *International Symposium on Methodologies for Intelligent Systems*. Springer, 2017, pp. 492–498.