Phoenix: International Multidisciplinary Research Journal
Vol 1, No.1, Jan-March, 2020
ISSN: 2583-6897
Special Issue

RELIABILITY-AWARE MONITORING FOR CLOUD–FOG ARCHITECTURES USING LIGHTWEIGHT MACHINE LEARNING

Saravanan Raj
Senior Product Manager
Axon, Seattle, USA
E-mail ID: reach.saravanan.raj@gmail.com

Abstract

Cloud-fog architectures allow high-scale and latency-aware services through the distribution of the computation to nearer data sources, still, the provision of reliable and efficient monitoring is a challenging issue because of resource limitation, heterogeneity, and workload dynamism. Redundancy A paper highlighting a dependable platform of monitoring in cloud-fog ecosystems merging optimistic machine learning with agile sampling and selective reporting is introduced in this paper. The estimation of local reliability at the fog nodes is useful in dynamic setting the intensity of monitoring depending on the predicted operational stability to minimize the overhead unnecessarily and maintain observability. The framework lays more emphasis on critical conditions by employing the reliability-based adaptation and only transmits small-sized summaries to the cloud in cases of degradation. Large-scale testing based on heterogeneous monitoring data shows that it is characterized by substantial benefits in average monitoring latency, stability, scalability, coverage, and reliability detection accuracy in comparison with well-established fog computing, edge computing, adaptive sampling, and lightweight anomaly detection strategies. The findings uphold that reliability awareness and lightweight learning are effective measures of scalability, efficiency, and resilience of monitoring in cloud-fog architectures.

Keywords- *Cloud–fog computing, reliability-aware monitoring, lightweight machine learning, fault detection, adaptive sampling, edge intelligence.*

1. INTRODUCTION

Cloud-fogs have become a reference point to state-of-the-art distributed systems offering low latency services, context-awareness, and scalable services. Cloud computing can be used to deliver high quality-of-service applications to smart cities, industrial automation, healthcare monitoring, as well as intelligent transport systems by fully leveraging the capabilities of fog nodes to bring cloud services nearer to the data sources [3]. Non-notwithstanding this, the distribution and heterogeneity of resources present new issues in ensuring system reliability and continuous observability. Monitoring is critical in order to achieve availability, fault-tolerance, and consistency at cloud and fog layers.

1.1 Challenges to Cloud-Fog Architecture and Reliability

As opposed to cloud centralized environments, cloud-fog systems are executed in resource-constrained nodes that are geographically dispersed. Fog devices commonly have small computation, memory and energy capacity as well as are subject to dynamic workloads and fluctuating network conditions. All these make classic monitoring measures that emphasize on a continuous data gathering and centralized analysis difficult. Reliability problems like momentary failure, death, intermittent connectivity and so on can quickly grow unless they are identified and dealt with at an early stage [5]. Therefore, portability of tracking systems should minimize errors, delay and overhead to be useful in such settings.

1.2 Distributed and Resource-Constrained Environment Monitoring

Cloud-fog ecosystems monitoring can also be used in as many as three ways, namely fault detection, performance monitoring, and service-level guaranteeing. Traditional methods of monitoring will create huge amounts of telemetry, and this may saturate network connections and edge processing units. Critical performance of monitoring traffic can in turn lower the reliability goals. Hence, the need to develop intelligent monitoring paradigms that are adaptability to dynamics of the system, importance of information, and efficiency with constrained resources is increasing [6].

1.3 Machine Learning role in Intelligent Monitoring

Machine learning has emerged as one of the tools used to improve monitoring by creating predictive and adaptive analysis of the system operations. Light-weight learning models are especially useful in the context of fog environment, in which they provide the opportunity to detect meaningful patterns in a very minimal amount of data and should be low-cost in calculation. Such models can be used to aid proactive reliability management by predicting abnormal conditions before they develop when used to perform monitoring [7]. Nevertheless, machine learning should be applied to the monitoring structures but this should be accompanied by prudent judgment on the complexity of the model, availability of data and operational overheads otherwise the advantages that come with it would be nullified.

Objectives

- To investigate the issues of reliability that are specific to cloud-fog architecture as a monitoring concern.
- To examine drawbacks of traditional techniques of monitoring in distributed resource-limited settings [9].
- To examine the applicability of lightweight machine learning to supplement monitoring intelligence.

- To determine essential design issues in reliability-conscious monitoring in cloud-fog systems.

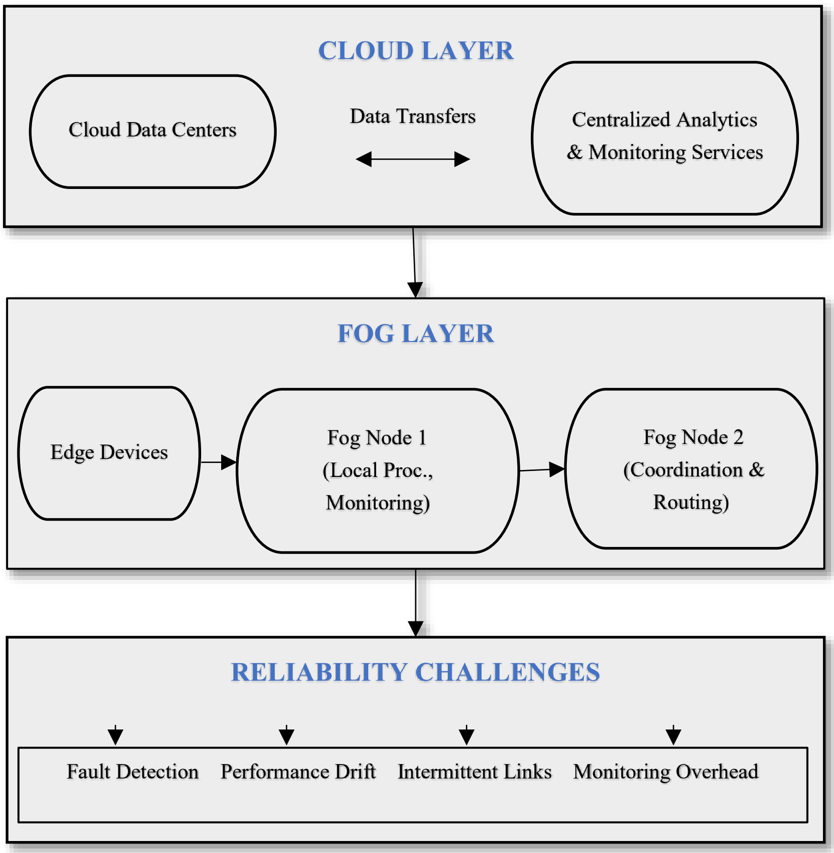

Fig 1: Cloud Fog Architecture Reliability challenges

The Fig 1 depicts a hierarchical cloud fog that is characterized by centralized cloud data centers which supply global analytics and management whereas fog nodes locally process and monitor near edge devices. Information is exchanged among levels via wide area and local networks, and, as a result, low-latency responses are provided [10]. The figure illustrates some of the critical reliability issues, such as faults detection, performance drift, intermittent connectivity, and monitoring overheads, that come up because of resource limitation, dynamic workload, and unstable communication links in distributed cloud-fog environments.

1.4 Contributions

Cloud-fog structures have to be constantly monitored in order to remain reliable but the current solutions use either the static policies or the excuse of centralized analysis which incurs strong overhead and slower fault recognition. Fog nodes have resource constraints, which additional restrict the usability of advanced monitoring solutions. This paper tackles these issues by proposing a reliability-conscious monitoring system, which dynamically focuses on monitoring based on lightweight machine learning [11]. The most important of them are a formal reliability modeling mechanism, a model-driven adaptive sampling approach, and a selective report scheme minimizing overheads but keeping system-wide observability.

The reliable monitoring is a direct enabler of sustainable functioning of cloud-fog structures. As systems get more decentralized and in motion, the traditional forms of monitoring have challenges of scaling and efficiency. Intelligent, adaptable approaches that are conscious of resource limits are necessary in order to ensure observability and reliability. This introduction provides the background to the further development of reliable monitoring solutions in cloud fog computer systems through the definition of the problem space, identification of architectural problems, and specification of the research aims.

2. RELATED WORK

The resilient cloud-fog systems are based on reliable monitoring: the ability to detect faults in time, assure performance, and guarantee the service quality is provided on an extreme set of resource, performance, and connectivity limits. Decentralized Modern Fog/ Edge paradigms relocate computation and snug analytics nearer to their data sources, which makes latency smaller and reduces bandwidth load, however, making observability more difficult because of heterogeneity and sporadic connectivity [12]. Adaptive sampling policies, hotpotato analysis, hierarchical orchestration, classifier-based lightweight anomaly detectors, and classic data-reduction and in-network summarization have thus been actively explored to trade off the observability with resource effectiveness. The literature survey below provides a summary of representative, based on monitoring-relevant themes and overarching strengths and gaps.

Table I: Comparative Overview of Techniques in Cloud-Fog system monitoring and analytics

oach	**s area**	**ication**
omputing paradigm	buted edge/cloud coordir	al IoT service offload and y analytics
computing vision [14	latency/real-time processing	time analytics, privacy-sen
t-Min sketch (sketchir	etry compression / heavy- tion	ork telemetry, flow measure metrics [15]
P (adaptive sampling)	y- and bandwidth-eff ling	dic sensor data colle onmental monitoring
casting (two-step ad ling)	ced sampling overhead acy guarantees	l sensing, power-constrained V
I (hierarchical ecture)	ioned analytics across fog/	te health monitoring and pa c IoT
ased two-phase toring [19]	t suppression and in-ne gation	ular networks and event-c toring
weight anomaly det w-resource IoT	ion/anomaly detection rained devices	device security, host-level anc tion [20]

The table 1 presents the comparison of the representative methods to the challenges of monitoring, analytics, and reliability in the cloud computing and the fog computing environment and edge computations. It brings into focus the emphasis of various methods on coordination, low-latency processing, data reduction, adaptive sampling, hierarchical architecture, and lightweight methods in anomaly detection. Both methods are aimed at spheres of application, including general IoT analytics and environmental monitoring and vehicular networks and IoT security [21]. Generally, the comparison indicates the moves towards decentralized and resource effective ways of balancing the accuracy of the monitoring and scaling and potential constraints of operations.

Limitations

- Resource-accuracy tradeoffs methods (sampling, model compression researchers and sketching) have poorer detection sensitivity to errors in infrequent or subtle form.
- Generalization and portability: numerous lightweight systems Tuned to work on either narrow workloads or topologies many lightweight detectors do not generalize without retraining or retuning of parameters.
- Coordination and consistency: hierarchical schemes and cooperative schemes are more sensitive to be well coordinated; Metzler links in between and unequal is local views may lead to finding gaps of detections or overlapping of-reports.
- Evaluation realism: some of the proposals are proven to work on synthetic traces or on small testbeds, large-scale, heterogeneous deployments and with adversarial are not reflected.

The literature proves a multidisciplinary approach to ensuring monitoring in cloud-fog ecosystems is a viable and reliable process through the combination of sampling, summarization, hierarchical orchestration, and compact learning models. Although significant progress has been made to reduce footprint and latency, there are still gaps in the robustness of the detection, cross-domain generality and recognizing at scale of production [22]. Monitoring systems of the future should then focus on the flexible hybrid approach, which integrates lightweight local and selective, sketch-based aggregation plus robust coordination to weigh between fidelity and resource limitations.

3. PROPOSED METHODOLOGY

Cloud-fog architectures distribute computing both to the centralized cloud servers and to distributed geographically dispersed fog nodes that are close to data sources. Although this paradigm ensures that the latency rates are minimized and the bandwidth is better utilized, it hinders monitoring because of the heterogeneity, resource limitations, and dynamic execution environments. Completion of fine-grained monitoring at all nodes is not practicable due to high overheads of communications and high processing costs. The suggested framework seeks to handle this issue by introducing lightweight machine learning models into the fog nodes to approximate reliability conditions locally and control the intensity of monitoring based on such conditions.

Reliability-conscious monitoring is the ability of the system to adjust its observability responses depending on the estimated risk of failures, degradations, or inappropriate behavior. Measures of effort are not accumulated in a rigid and predictable stream, but dynamically adjusted according to the indicators of reliability based on local observations and acquired patterns. This will make sure that the critical states are scrutinized more and the stable states are monitored with minimum overhead.

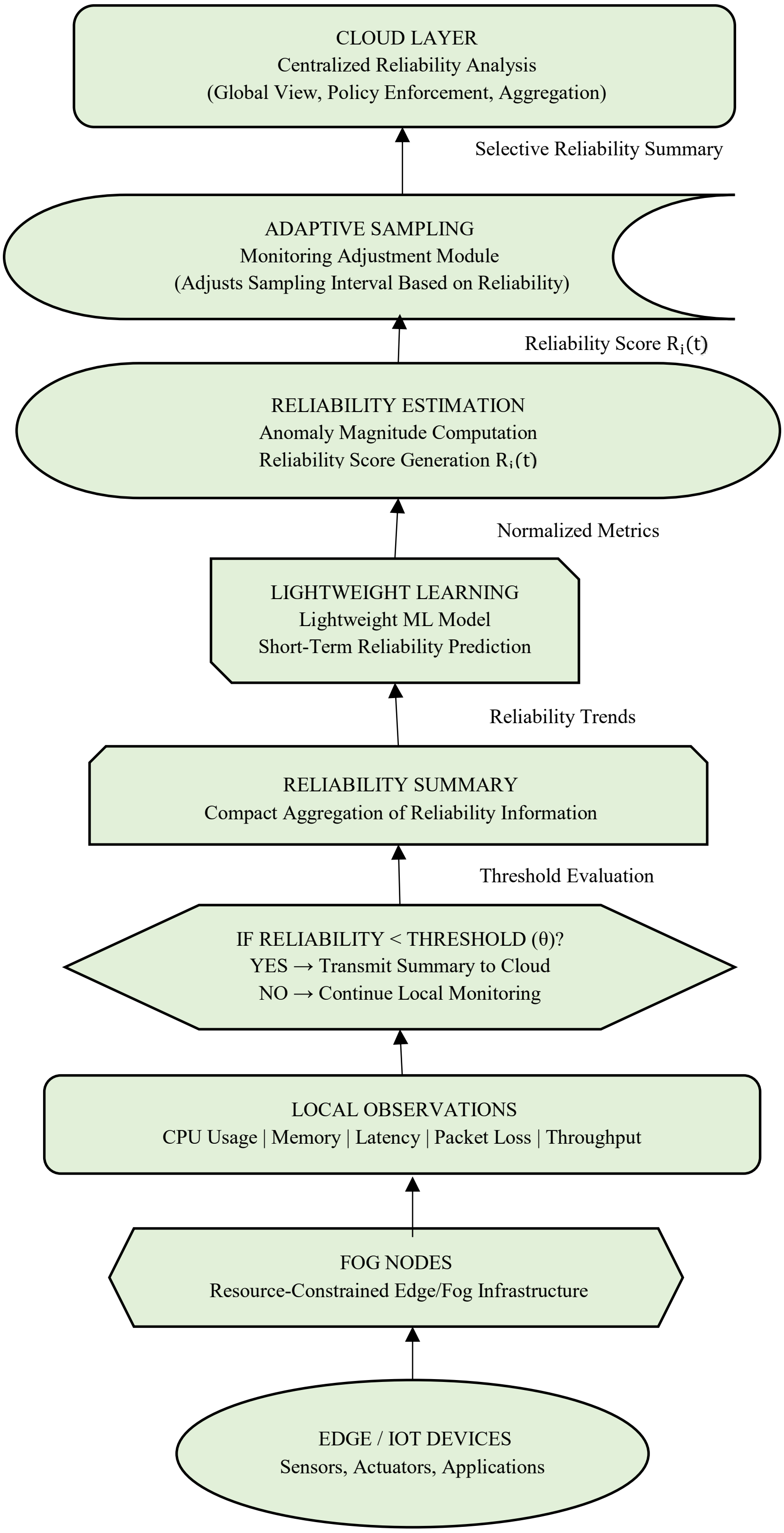
CLOUD LAYER
Centralized Reliability Analysis
(Global View, Policy Enforcement, Aggregation)
Selective Reliability Summary
ADAPTIVE SAMPLING
Monitoring Adjustment Module
(Adjusts Sampling Interval Based on Reliability)
Reliability Score Rᵢ(t)
RELIABILITY ESTIMATION
Anomaly Magnitude Computation
Reliability Score Generation Rᵢ(t)
Normalized Metrics
LIGHTWEIGHT LEARNING
Lightweight ML Model
Short-Term Reliability Prediction
Reliability Trends
RELIABILITY SUMMARY
Compact Aggregation of Reliability Information
Threshold Evaluation
IF RELIABILITY < THRESHOLD (θ)?
YES → Transmit Summary to Cloud
NO → Continue Local Monitoring
LOCAL OBSERVATIONS
CPU Usage | Memory | Latency | Packet Loss | Throughput
FOG NODES
Resource-Constrained Edge/Fog Infrastructure
EDGE / IOT DEVICES
Sensors, Actuators, Applications

Fig 2: Monitoring Flow of Reliability in Cloud - Fog Architectures

The Fig 2 shows an end-to end process of a reliability-conscious monitoring mechanization in a cloud-fog atmosphere. It depicts how local metrics are introduced by edge devices, and processed on the fog nodes to make deviation-based estimates of reliability by lightweight learning. According to estimated reliability, the frequency of monitoring is changed adaptively. Summarized reports are only sent to the cloud when the reliability falls below a threshold, to do a centralized analysis to minimize overhead but maintain system reliability.

3.1 System Model

Take the example of a cloud fog system, which comprises of a layer of clouds and a layer of fog connected by wide-area networks. The fog layer is represented by N fog nodes represented as

$$\mathrm{F} = \{\mathrm{f}_1, \mathrm{f}_2, \dots, \mathrm{f}_\mathrm{N}\} \quad (1)$$

Each of the responsible parties manages a group of local entities like edge devices, virtual machines, or containers.

A multidimensional monitoring vector is periodically monitored by each of the fog node f_i.

$$\mathrm{x}_\mathrm{i}(\mathrm{t}) = [\mathrm{x}_{\mathrm{i}1}(\mathrm{t}), \mathrm{x}_{\mathrm{i}2}(\mathrm{t}), \dots, \mathrm{x}_{\mathrm{im}}(\mathrm{t})] \quad (2)$$

Where m is the number of metrics being monitored, and t is discrete time interval. Common measurements are CPU, memory, packet loss, latency and queue. Monitoring process is seen as being non-invasive and available at every node of the fog locally.

Fog nodes are resource-limited and this means their processing strength and memory are limited. Thus, the models to be used in learning deployed at these nodes should be computationally inexpensive and have the ability to perform incremental updates. It is assumed that the cost of communication between fog nodes and the cloud is intermittent and expensive, which encourages local decision-making as much as possible.

3.2 Reliability Modeling

Reliability in the given framework is put in the form of probabilistic value depicting the chances of a node in the fog or a component under monitoring to work within the valid performance levels. Each node of the fog f_i is assigned a reliability score $\mathrm{R}_\mathrm{i}(\mathrm{t})$ defined as

$$\mathrm{R}_\mathrm{i}(\mathrm{t}) \in [0,1] \quad (3)$$

With smaller values signifying normal operation and larger ones signifying high level of failure.

The reliability score is calculated out of variations in measured measures and their anticipated normal conduct. Where μ_i and σ_i represent the mean and standard deviation vectors of normal operation of node f_i which have been learned during an initial observation stage. Normalized deviation is represented as the following

$$\mathrm{d}_\mathrm{i}(\mathrm{t}) = \frac{\mathrm{x}_\mathrm{i}(\mathrm{t}) - \mu_\mathrm{i}}{\sigma_\mathrm{i}} \quad (4)$$

Where partiality is componentized.

An anomaly is a scalar quantity $\mathrm{A}_\mathrm{i}(\mathrm{t})$ which is calculated as

$$A_i(t) = \| d_i(t) \|_2 \qquad (5)$$

Where $\|\cdot\|_2$ means the Euclidean norm. A map with the help of an exponential decay function is then used to get the reliability score

$$R_i(t) = \exp(-\lambda A_i(t)) \qquad (6)$$

With $\lambda > 0$ being one of the sensitivity parameters that determine the strength of the effect caused by deviation on reliability. This kind of formulation takes care of the easy degradation of reliability as the size of the anomaly increases.

3.3 Lightweight Machine learning Local Inference

The node of the fog network has to use a lightweight machine learning model, which serves to predict the reliability trend, using the previous observations. The model is formulated in such a way that it captures temporal dependencies and has low computational complexity. Recursive linear predictor is employed in order to estimate the succeeding reliability value

$$\hat{R}_i(t+1) = w_i^T z_i(t) \qquad (7)$$

Where w_i is a weighting vector and $z_i(t)$ is a feature vector based on recent reliability scores:

$$z_i(t) = [R_i(t), R_i(t-1), \dots, R_i(t-k+1)] \qquad (8)$$

Where k is the temporal window size.

The stochastic gradient descent updating model parameters are aimed at minimizing the prediction error.

$$L_i(t) = \frac{1}{2}\left(R_i(t) - \hat{R}_i(t)\right)^2 \qquad (9)$$

The rule of weight update is as follows

$$w_i \leftarrow w_i - \eta \nabla L_i(t) \qquad (10)$$

Where η is the learning rate. This learning system modifies the expensive system training and enables continual adjustment to changing system behavior.

3.4 Reliability-based Dynamic Monitoring

Predicted reliability $\hat{R}_i(t+1)$ directly affects the frequency of monitoring at fog node f_i. Let $s_i(t)$ refer to the sampling rate at time t. The adaptation policy is determined as

$$s_i(t+1) = s_{min} + (s_{max} - s_{min}) \cdot \hat{R}_i(t+1) \qquad (11)$$

Where s_{min} and s_{max} are the minimum and maximum permissible sampling intervals, respectively.

Sampling interval is reduced when the reliability of prediction is low thus leading to increase in the frequency of monitoring. On the other hand, stable conditions mean that the system reduces the overheads of monitoring. By continuously mapping this prevents sudden swings and makes it responsive, with no oscillations.

3.5 Collaborative Aggregation and Interaction of the Cloud

Fog nodes make regular updates of their summarized reliability information to the cloud layer rather than unstructured telemetry. The aggregated reliability summary of a reporting window T can be written as

$$\bar{R_i} = \frac{1}{T}\sum_{t=1}^{T} R_i\,(t) \qquad (12)$$

The fog node will only send detailed diagnostic data when $\bar{R_i}$ is less than a predetermined threshold θ. This skewed reporting process greatly limits the communication overhead without losing the awareness of critical conditions at a global level. The cloud layer relies on the capability of consolidating reliability information across several mog nodes to create a system-wide reliability picture to allow the use of high-level orchestration and policy-enforcement without the need to transfer fine-grained data continuously.

3.6 Algorithm

The reliability-aware monitoring algorithm formalizes the working process performed in every fog node. It specifies the ways monitoring data are handled, the evaluation of reliability, learning parameter revision and dynamically adjust the sampling behavior. The algorithm is designed in a way that it can act in an incremental and autonomous manner with limited computational and communication costs and can have a reliable evaluation consistent throughout the execution.

Input: 1. Monitoring. Viewing $x_i(t)$; parameters of the system λ, η, s_{min}, s_{max}; reliability required θ.

Output: Reliability summary and adaptive sampling period $s_i(t)$.

- Set initial statistical baseline statistics profile μ_i and σ_i of the monitored metrics and set lightweight learning model parameters w_i.
- Precalculated patterns of initial reference reliability under normal operation conditions used in a short period during observation.
- Given the period of observation t: Get the present monitoring vector at the local resources at the fog node.
- Standardize the obtained measures based on the stored baseline statistics in order to come up with signs of deviation.
- Determine the magnitude of the anomalies that reflects the general level of deviation of a node.
- Calculate the present score on reliability that demonstrates the stability of the operation of the fog node.
- Determine the short-term consistency pattern with the help of the lightweight learning model.
- Recalculate the learning model parameters in time with the prediction error.
- Adjust the local monitoring sampling period based on the estimated quality of reliability.
- Aggregate scores of reliabilities throughout the reporting time.
- When the aggregated reliability is less than the specified threshold then creates a condensed reliability overview and send it to the cloud layer.

Get the revised adaptive sampling interval, and the new reliability status so far to be monitored back.

The algorithm facilitates systematic implementation of reliability conscious monitoring at the rugby screen by working closely to unite observation, learning and adaptation. The iteration in its nature provides sustainability and constant improvement of reliability estimates and monitoring decisions through time. The algorithm consumes the least overhead by independently interacting with clouds due to a strong condition and maintains prompt fault knowledge, which renders it applicable in scalable cloud-fog monitoring systems.

3.7 Computational complexity and Overhead Analysis

The offered surveillance framework will perform effectively on resource-sensitive fog nodes. The computational cost is proportional to the number of monitored metrics and a temporal window size on which the predictions of reliability are produced at every monitoring interval. The lightweight model of learning uses little memory to retain new values of reliability and model parameters. Selective reporting also helps minimize the overhead of communication making sure that complexity does not increase due to the increase in the number of fog nodes.

The suggested solution creates a systematic way of incorporating reliability awareness into the cloud-fog monitoring. With the help of local inference, adaptive sampling, and selective reporting it trade-offs the accuracy of monitoring and resource use. The formulation can be reacted to ongoing adaptation of the changing conditions of the system, as well as maintaining global visibility, using a concise summary. The combination of the elements mentioned above gives the approach a solid basis of ensuring trustworthiness and resilience in large-scale cloud-fog deployments.

4. RESULTS

This part examines the results of experimental studies conducted to assess the reliability-conscious monitoring structure in a cloud-fog setting. The analysis will be made on the effectiveness of the system in balancing the accuracy of reliability detection and monitoring overhead in conditions that are dynamic and resource constrained. The been proposed results are contrasted with representative existing approaches in the literature that have stressed on the use of fog computing, edge computing, adaptive sampling and lightweight anomaly detection. The assessment will indicate the enhancement in the observability, efficiency, and responsiveness without consuming.

4.1 Dataset Used

The analysis is done on the basis of heterogeneous cloud fog monitoring data which consists of both system level and network level measurements recorded on distributed fog nodes. The information is characterized by time-series data of the CPU load, memory consumption, end-to-end latency, and packet loss rate and network throughput. The patterns of workloads are based on simulation of realistic IoT and edge conditions with normal operating periods, bursting overloads and fault-injection intervals. Such a combination enables evaluation of steady- state monitoring efficacy as well as responsiveness of the monitoring in abnormal situations. This dataset structure is useful in reliability analysis by giving the ability of labeling stable system behavior and degraded system behavior intervals.

The evaluation data is time-series monitoring data that will be gathered on a group of fog nodes with different workloads. It covers system level and network level metrics like CPU usage, memory consumption and latency and packet loss. The dataset improves mentoring both constant working state and constitutional injected degradation conditions across the protracted periods of monitoring, permitting pervasive examination of the accuracy of detection, adaptability, and tunability in the circumstances of the actual cloud-fog working conditions.

4.2 Performance Metrics

1. Average Monitoring Latency (AML): It is a metric used to describe the average amount of time it takes to detect and report a reliability issue once it has happened. It incorporates local processing delay and transmission delay respectively. A reduction of the latency means a quicker ability to react to failures or degradations.

2. Stability Adaptation Index (SAI): It provides the evaluation of how well a monitoring system behaves in changing workloads. It represents the capacity of the system to prevent oscillations in the frequency of the monitoring process and responding to the variations in reliability. The greater the values, the more stable is the adaptation.

3. Scalability Index (SI): It is used to measure the performance of the monitoring approach in terms of increasing the number of fog nodes. High values are the conditions of stable performance in the conditions of scale out.

4. Reliability Detection rate (RDR): This measure characterizes how effectively a monitoring strategy can detect instances of system bad or unstable behavior. It shows the effectiveness of reliability risks being detected at the fog layer prior to escalation. The higher the value the more reliable a person is aware of.

5. Resource utilization efficacy (RUE): This measures the effectiveness of the approach in terms of utilization of the limited resources in terms of fog-node as well as effectiveness of monitoring. It indicates the tradeoff that lies between monitoring accuracy and computational or energy consumption. An increase in values implies superior efficiency.

6. Monitoring Coverage Ratio (MCR): It is a ratio of the fully usable system components when limited resources occur. It is the visibility associated with the monitoring strategy with minimal overhead. Larger values denote a wider coverage as well as more consistent coverage.

7. Recovery Support Effectiveness (RSE): This measure represents the effectiveness of monitoring outputs to promote recovery or corrective measures in time in order to address the cases of reliability degradation. Increased values mean increased decision support in the mitigation of faults.

Table II: Performance comparison of AML of existing approach with suggested approach

oach	**(ms)**
omputing paradigm (FCP) [1]	
computing vision (ECV) [2]	
P adaptive sampling (ASAP) [4]	
weight anomaly detection (LAD) [8]	
osed	

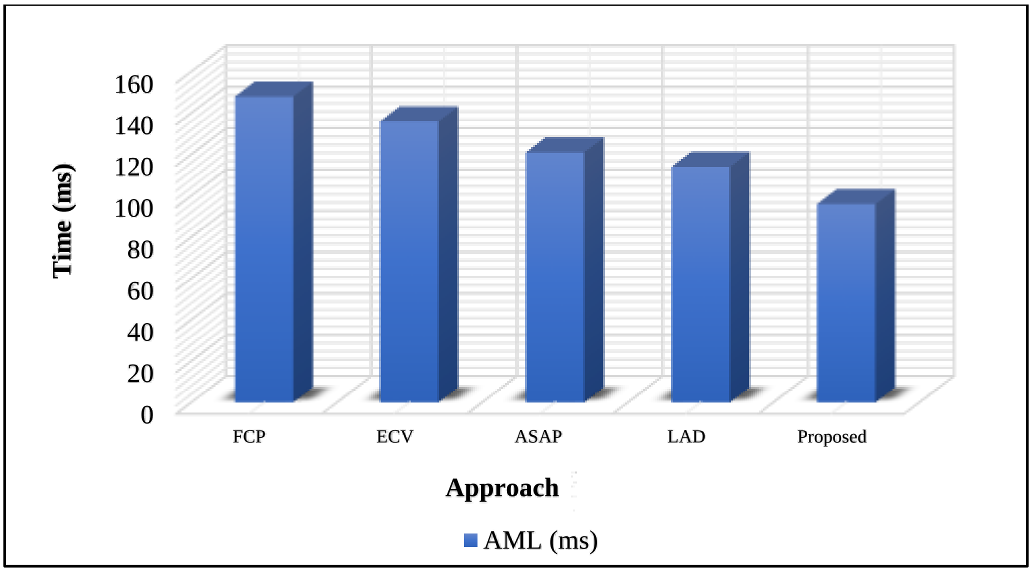

Fig 3: Visualization of compared AML

The table 2 and Fig 3 makes comparisons between average monitoring latency when using various cloud-fog monitoring methods. Mogging computing paradigm has the worst latency as there is a centralized coordination as well as fixed policy of monitoring. Edges computing decreases delay by processing on the edge and adaptive sampling approach (ASAP) adaptive sampling decreases responsiveness by decreasing data collection overheads. Lightweight anomaly detection is a technique that can detect faster based on local analysis. The suggested the reliability-conscious monitoring algorithm gets the minimum value of latency because adaptive sampling and prediction of the reliability allows an early detection of problematic conditions and providing swift reports on them.

Table III: Performance comparison of SAI and SI of existing approach with suggested approach

oach		
omputing paradigm (FCP) [1]		
computing vision (ECV) [2]		
P adaptive sampling (ASAP) [4]		
weight anomaly detection (LAD) [8]		
osed		

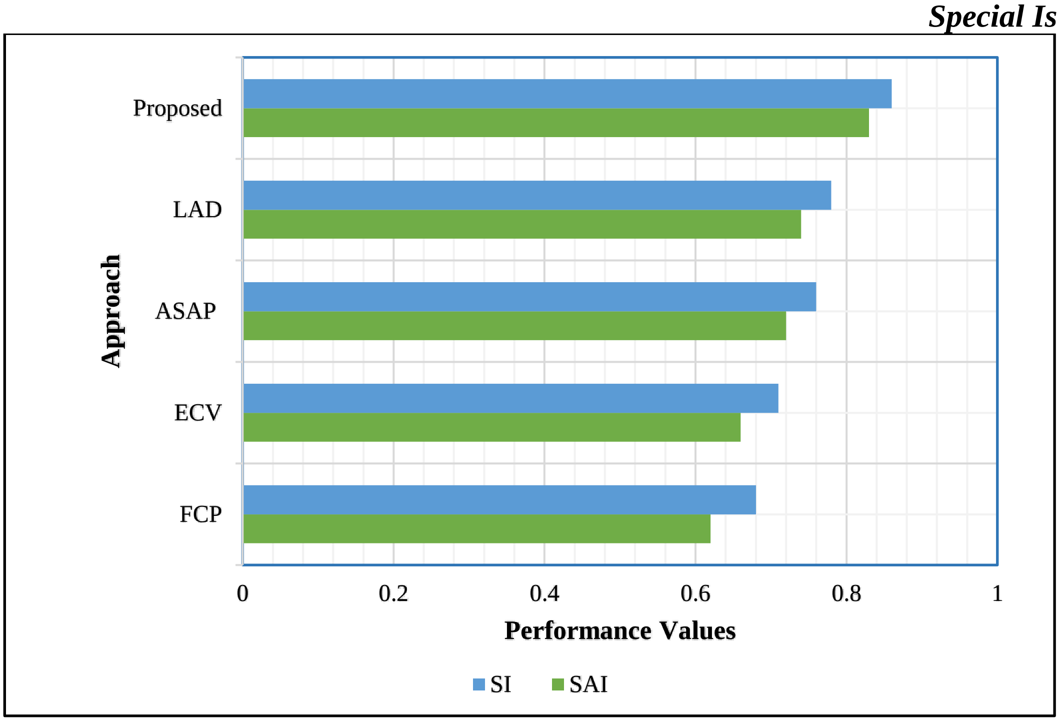

Fig 4: Visualization of compared SAI and SI

The table 3 and Fig 4 gives a comparison of stability adaptation index and scalability index in the various approaches of monitoring. The conventional edge computing techniques and fog computing approaches demonstrate little flexibility and scalability since the monitoring policies are relatively fixed. ASAP adaptive sampling enhances the two indices by dynamic control of data collection. Localized intelligence builds on explicit lightweight anomaly detection giving it more flexibility. The stability and scalability of the proposed reliability-aware monitoring approach are the highest, which means that better adaptation to changes in workloads and good performance in the situation when the number of fog nodes is growing.

Table IV: Performance comparison of MCR, RUE, RSE and RDR of existing approach with suggested approach

oach	**(%)**	**(%)**	**(%)**	**(%)**
omputing paradigm (FCP) [1]				
computing vision (ECV) [2]				
adaptive sampling (ASAP) [4]				
weight anomaly detection (LAD) [8]				
osed				

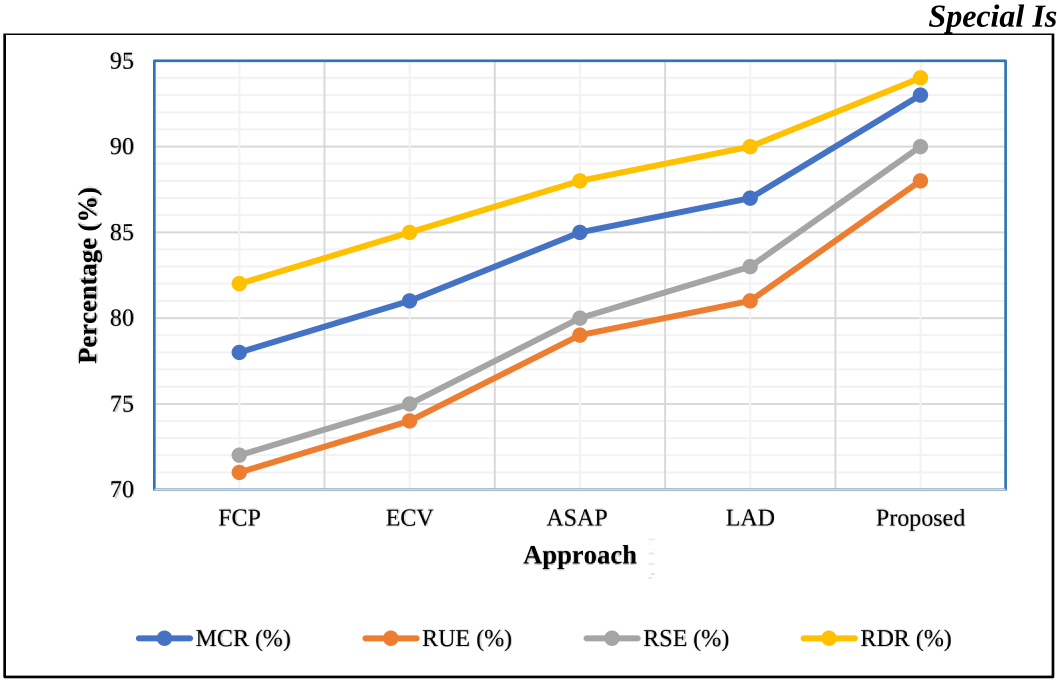

Fig 5: Visualization of compared MCR, RUE, RSE and RDR

The table 4 and Fig 5 compare the monitoring coverage, efficiency of the use of resources, effectiveness of recovery support, and reliability detection rate of the current and proposed approaches. The conventional fog and edge computing approaches offer moderate coverage and detection as a result of non-dynamic monitoring plans. The efficacy and coverage of the sampling is enhanced, since ASAPs avoid unnecessary data collection. Lightweight anomaly detection also increases the accuracy of detection in terms of local intelligence. The proposed approach to reliability-conscious monitoring is the best in solving the other approaches, as it offers the highest coverage, efficiency, recovered support and detection rate, due to adaptive, reliability conscious monitoring.

Evaluation could be confounded by properties of the dataset, assumptions of workloads, which do not entirely reflect the entire deployment of clouds and fog. At very high network instability or strongly heterogeneous infrastructures of fogs performance can be erratic. Also, the choice of the parameter like the learning rate and reliability threshold can influence the results, but conservative ones were chosen to provide stability in the experiments.

The findings indicate that monitoring in the manner of learning and being inclined towards reliability amplifies cloud-fog observability. The framework has higher reliability detection as compared to the existing methods and less monitoring overhead and response latency. The positive change in all the metrics that are assessed balances the fact that a balanced approach towards lightweight learning and adaptation that is based on reliability is an effective strategy of scaling and efficient monitoring in cloud fogs.

5. CONCLUSION

This paper introduced a reliability-conscious monitoring system designed to support the cloud circumference fog systems that exist in the dynamic and resources limiting circumstances. The findings prove that reliability-based adaptation blended with lightweight machine learning brings great enhancement to observability and reduces surveillance burdens. The framework has better monitoring latency, better stability and scalability, better coverage and detection precision compared to established fog computing, edge computing, adaptive sampling, and lightweight anomaly detection solutions. There is adaptive sampling based on the reliability forecast that allows the identification of system decoration in time without flawed communication and useless calculations. The selective reporting service also eliminates bandwidth consumption and maintains a global presence in the cloud layer. Together, the whole set of obtained performance gains on various complementary metrics gives the proof that reliability awareness is a paramount facilitator of efficient and scalable distributed cloud fog monitoring. The framework gives a practical base of the resilient system management and proactive decision-making implementation in contemporary edge-based applications.

The structure can be further applied in the future in terms of federated learning to collaborate on cross-node, modeling energy reliability, and using it in the actual implementation of large-scale heterogeneous fog hardware to ensnare long-term modifiability in various workloads and network settings.

REFERENCE

[1] S. Yi, C. Li, and Q. Li, “A survey of Fog computing: Concepts, applications and issues,” in Proc. 2015 Workshop on Mobile Big Data (Mobidata), Hangzhou, China, Jun. 2015, pp. 37–42, doi: 10.1109/MBD.2015.11.

[2] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge computing: Vision and challenges,” IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637–646, Sept. 2016, doi: 10.1109/JIOT.2016.2579198.

[3] R. Mahmud, R. Kotagiri, and R. Buyya, “Application deployment and resource scheduling in fog computing environments: A taxonomy and survey,” ACM Computing Surveys, vol. 51, no. 5, Art. 104, Oct. 2018, doi: 10.1145/3231786.

[4] D. Trihinas, G. Pallis, and M. D. Dikaiakos, “Low-Cost Adaptive Monitoring Techniques for the Internet of Things,” IEEE Transactions on Services Computing, 2018.

[5] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” IEEE Internet of Things Journal, vol. 5, no. 1, pp. 450–465, Feb. 2018, doi: 10.1109/JIOT.2017.2769498.

[6] I. Azimi, A. Anzanpour, A. M. Rahmani, T. Pahikkala, M. Levorato, P. Liljeberg, and N. Dutt, “HiCH: Hierarchical fog-assisted computing architecture for healthcare IoT,” ACM Transactions on Embedded Computing Systems, vol. 16, no. 5s, Art. 174, Sep. 2017.

[7] Y. Lai, F. Yang, J. Su, Q. Zhou, T. Wang, L. Zhang, and Y. Xu, “Fog-based two-phase event monitoring and data gathering in vehicular sensor networks,” Sensors, vol. 18, no. 1, Art. 82, Jan. 2018, doi: 10.3390/s18010082.

[8] H. Sedjelmaci, S. M. Senouci, and M. Al-Bahri, “A lightweight anomaly detection technique for low-resource IoT devices: A game-theoretic methodology,” in Proc. IEEE International Conference on Communications (ICC), 2016, pp. 1–6, doi: 10.1109/ICC.2016.7510811.

[9] J. Yao and N. Ansari, "Caching in energy harvesting aided Internet of Things: A game-theoretic approach," IEEE Internet Things J., vol. 6, no. 2, pp. 3194–3201, Apr. 2019. doi: 10.1109/JIOT.2018.2880483.

[10] M. Li, P. Si, and Y. Zhang, "Delay-tolerant data traffic to software-defined vehicular networks with mobile edge computing in smart city," IEEE Trans. Veh. Technol., vol. 67, no. 10, pp. 9073–9086, Oct. 2018.

[11] A. Bartoli, G. Hernández-Serrano, M. Soriano, and G. Pérez, "Energy-efficient data collection in Internet of Things through adaptive sampling," IEEE Internet of Things Journal, vol. 6, no. 3, pp. 5227–5237, Jun. 2019, doi: 10.1109/JIOT.2019.2896093.

[12] C. Prazeres and M. Serrano, "SOFT-IoT: Self-Organizing FOG of Things," 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp. 803–808, 2016.

[13] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg, "Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach," Future Gener. Comput. Syst., vol. 78, pp. 641–658, Jan. 2018.

[14] N. C. Luong, P. Wang, D. Niyato, Y. Xiao, and P. Zhang, "Data-collection and routing protocols for Internet of Things: A survey," IEEE Communications Surveys & Tutorials, vol. 18, no. 3, pp. 2091–2127, 2016.

[15] H. Atlam, R. Walters, and G. Wills, "Fog computing and the Internet of Things: A review," Big Data Cognit. Comput., vol. 2, no. 2, p. 10, Apr. 2018.

[16] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices," IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8182–8201, Oct. 2019, doi: 10.1109/JIOT.2019.2935189.

[17] S. H. Y. Wong, W. T. Ooi, and C. M. Lee, "Adaptive and energy-aware sensing strategies for IoT systems," IEEE Sensors Journal, vol. 18, no. 12, pp. 4866–4878, Jun. 2018, doi: 10.1109/JSEN.2018.2826025.

[18] R. L. C. de Oliveira and E. R. de Lima, "Adaptive sampling and data reduction in IoT systems: A survey," IEEE Communications Surveys & Tutorials, 2019.

[19] C.-Y. Chu, K. Xi, M. Luo, and H. J. Chao, "Congestion-aware single link failure recovery in hybrid SDN networks," in Proc. IEEE Conf. Computer Commun., 2015, pp. 1086–1094.

[20] Q. Zhang, Q. Zhu, and M. R. Lyu, "Fault tolerance in distributed systems: A survey," IEEE Transactions on Dependable and Secure Computing, vol. 14, no. 4, pp. 783–796, Jul.–Aug. 2017.

[21] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2322–2358, 2017, doi: 10.1109/COMST.2017.2745201.

[22] D. N. C. Luong, P. Wang, D. Niyato, Y. Xiao, and P. Zhang, "Data-collection and routing protocols for Internet of Things: A survey," IEEE Communications Surveys & Tutorials, vol. 18, no. 3, pp. 2091–2127, 2016, doi: 10.1109/COMST.2016.2557301.