OPEN ACCESS
IOP Publishing | ISECAEP
Journal of Reliability Science and Engineering
J. Reliab. Sci. Eng. **2** (2026) 035401 (19pp)
https://doi.org/10.1088/3050-2454/ae67a1

Delay-sensitive cascading failure modeling and analysis of port Internet of Things

Xiuwen Fu[1], **Xindi Wang**[1] **and Liudong Xing**[2,*]

[1] Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, People's Republic of China
[2] Electrical and Computer Engineering, University of Massachusetts Dartmouth, North Dartmouth, MA, 02747-2300, United States of America

E-mail: xwfu@shmtu.edu.cn

Received 2 February 2026, revised 7 April 2026
Accepted for publication 2 May 2026
Published 15 May 2026

Abstract

This paper addresses the limitations of existing Internet of Things (IoT) cascading failure models in capturing the unique characteristics of port IoT (PIoT) systems, particularly delay sensitivity, heterogeneous data flows, and priority-aware transmission. To this end, a delay-sensitive cascading failure model is proposed by integrating sensing–control data flows and priority-based packet scheduling, while incorporating two alternative caching mechanisms for performance comparison. The model characterizes the interactions among data transmission, node load dynamics, and failure propagation under time-constrained conditions. Simulation results demonstrate that priority-aware transmission improves packet survival and mitigates congestion propagation. In addition, in-transit caching outperforms end-of-line caching in preserving data effectiveness. Furthermore, the packet survival rate increases with the number of routing nodes, while expansion factors enhance system robustness only up to critical thresholds. In contrast, higher congestion coefficients significantly degrade system performance.

Keywords: cascading failures, port internet of things, load distribution, network reliability

1. Introduction

The efficient operation of smart ports relies on comprehensive perception of port status, ubiquitous connectivity, and intelligent decision-making. As the key infrastructure supporting the acquisition and transmission of port operation information, the Internet of Things (IoT) provides the digital foundation for the operation of smart ports [1–3].

In the port IoT (PIoT), deployed IoT nodes need to forward large volumes of data, but their bandwidth and cache resources are often limited. These nodes must receive both sensing data (i.e. uplink data) and control data (i.e. downlink data). During data transmission, downlink data are prioritized. If the link capacity is insufficient for data transmission, any data exceeding the link capacity must be returned to the original node. This easily leads to data accumulation, which causes node congestion and further impairs the nodes' processing capacity. If the load in a node's cache exceeds its capacity, the node can fail due to overflow, and the data will be rerouted. In this data rerouting process, other nodes may become overloaded. This process, in which failures propagate due to load redistribution, is referred to as cascading failure [4, 5].

Due to the significant threat cascading failures pose to IoT operations, the issue of cascading failures has garnered

* Author to whom any correspondence should be addressed.

Original content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

© 2026 The Author(s). Published by IOP Publishing Ltd on behalf of the Institute of Systems Engineering, China Academy of Engineering Physics

widespread attention. Currently, research on cascading failures in IoT domains such as wireless sensor networks (WSNs) [6–8], industrial IoT [9, 10], smart home [11, 12], and edge assisted IoT [13, 14] has made encouraging progress. However, due to the following characteristics, existing IoT cascading failure models are not applicable to real-world PIoT:

(1) In PIoT, nodes not only collect sensing data but also receive control instructions from gateways or edge nodes. The interaction between these two types of data significantly affects load distribution and failure propagation.
(2) PIoT applications involve time-critical operations (e.g. unmanned container trucks and quay cranes), making the system highly sensitive to transmission delay. Therefore, priority-aware data transmission plays a critical role in system performance.
(3) Unlike traditional models that rely on static topological metrics, node load in PIoT is determined by real-time data flows within cache spaces, resulting in dynamic and time-varying load characteristics.

Therefore, this study proposes a delay-sensitive cascading failure model for the PIoT. The key contributions of this paper are as follows:

(1) A delay-sensitive cascading failure model for the PIoT is constructed, in which node load is characterized using dynamic data flows (including sensing data and control data), and data transmission is restricted by the data processing capacity of nodes and the bandwidth of links.
(2) Based on the delay requirements of real-world port operations, different types of data packets are assigned distinct priority levels. In addition, two data caching mechanisms are proposed: the in-transit caching mechanisms and end-of-line caching mechanisms—to reflect the impact of data cache management on the network's cascading failure process.
(3) Through extensive experiments, we systematically evaluate the impacts of node and link failure triggering factors in the PIoT, and analyze the effects of modeling parameters on network performance under various failure mechanisms. Furthermore, comparative experiments without priority settings are conducted to demonstrate the role of priority in enhancing network reliability.

The rest of this paper is organized as follows: section 2 reviews related work; section 3 provides a preliminary analysis of the PIoT; section 4 describes the PIoT cascading failure model; section 5 presents simulation experiments; and section 6 concludes the paper.

2. Related work

Building on the above mechanism, cascading failures pose a significant threat to IoT systems, as local disruptions may propagate through network connections and cause large-scale performance degradation. As a result, cascading failures have become a critical factor affecting the reliability of IoT systems and have attracted increasing attention in recent studies [15, 16].

The study of cascading failures in the IoT and WSNs has evolved from basic topological analysis to complex, context-aware modeling. A significant portion of the literature focuses on the relationship between network structure and resilience. For instance, Fu *et al* [17] and Ye *et al* [18] utilized the probability generating function method to analyze scale-free networks, discovering that the scale of cascading failures is closely linked to the network's power-law exponent. Similarly, research by Hu *et al* [19] highlighted that Newman–Watts (NW) small-world topologies generally outperform Watts–Strogatz configurations in terms of robustness. Building on these structural insights, Mo *et al* [20] revealed the impact of community structures, while the pivotal role of increasing the proportion of cluster heads in limiting failure propagation within clustered IoT networks was demonstrated in [21]. Furthermore, Yin *et al* [22] integrated Markov mobility models with scale-free topologies to study how node movement affects cascading risks under overload attacks. Beyond pure topology, researchers have developed diverse mathematical frameworks to model failure mechanisms. Biabani *et al* [23] introduced a directed betweenness-based model to propose load-balancing and energy-efficient routing strategies. In the specialized domain of smart homes, Xing *et al* [24] and Zhao *et al* [25] employed combinatorial analysis to define cascading failures as competitive behaviors among devices, focusing on the isolation effects and functional dependencies that arise within these environments. To capture a more comprehensive range of stressors, a model based on cellular automata [26] was proposed to account for hardware and software malfunctions, energy depletion, and connectivity impairments simultaneously, providing a holistic view of network degradation. More recent research focuses on functional interdependencies and specific constraints of modern network environments like edge computing and industrial systems. Wang *et al* [27] explored 'weak interdependence' in cyber-physical networks, where failures in sensor nodes propagate to physical layers with a specific probability. In the realm of edge-assisted IoT, a model was developed [28] to represent node load through real-time data packets, accounting for link congestion and the data-compression capabilities of edge nodes. Finally, in the context of IIoT, cascading failures were examined from the perspective of network service coupling [29], emphasizing the dual roles of nodes as both data forwarders and service providers. Collectively, these studies demonstrate a shift toward more granular models that distinguish between sensing and relay loads [20] and consider the real-time operational demands of the network.

Although research on cascading failures in the IoT has made notable progress, there are still limitations in describing the characteristics of cascading failures in PIoT systems, including: (1) In existing cascading failure models, nodes are sensor nodes that only handle data collection and transmission. In contrast, nodes in PIoT systems possess

Table 1. Comparison between general IoT models and the proposed PIoT model.

Aspect	Conventional IoT models	Proposed PIoT Model
Model assumptions	Homogeneous sensing nodes; single data flow; no delay or priority	Heterogeneous nodes; sensing and control data; delay and priority considered
Node load characterization	Static topology-based load	Dynamic load based on real-time cache
Failure propagation mechanism	Topology-driven load redistribution	Data flow, priority, and congestion-driven
Data flow	Sensing data only	Coupled sensing and control data
Data priority	No priority	Priority-aware
Delay sensitivity	Not considered	Directly affecting the cascading failure process
Caching	Ignored or simplified	Comparative analysis of different caching mechanisms

sensing, communication, and automated operational capabilities. Both the sensing data they upload and the control data they receive jointly influence the network's load distribution, thereby affecting the network's cascading failure process; (2) Existing cascading failure models usually do not distinguish between data priorities. However, in the delay-sensitive PIoT, different types of data have varying transmission priorities, and these differences can affect node load—ultimately influencing the cascading failure process; (3) Existing models usually rely on static topological parameters to define node load, which cannot adequately capture the time-varying load characteristics of nodes in PIoT systems. As shown in table 1, the proposed model exhibits essential differences from conventional IoT cascading failure models in terms of model assumptions, node load characterization, and failure propagation mechanisms.

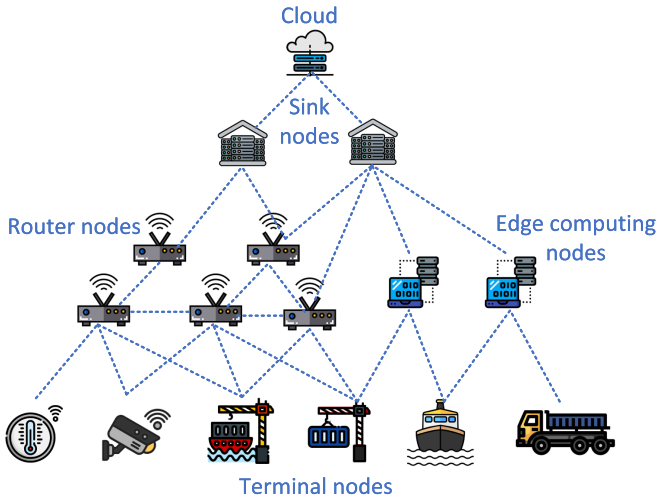


Figure 1. System architecture of PIoT.

3. Preliminary analysis of PIoT

3.1. System architecture

The PIoT is a distributed network system comprising nodes and communication links, which adopts a hierarchical network architecture. In the PIoT, nodes are divided into four categories according to their functions: terminal nodes, router nodes, sink nodes, and edge computing nodes. Terminal nodes are devices for port data collection, with functions including: (1) Collecting status data from port operation equipment (container trucks, quay cranes, yard cranes) and receiving control data; (2) Monitoring and collecting port status data such as temperature, humidity, static electricity, and cargo position. A terminal node can establish communication links with multiple router nodes. Router nodes are relay devices in the PIoT, responsible for data forwarding. A router node can establish communication links with multiple other nodes within its communicationradius. Sink nodes are network facilities (e.g. base stations) used to receive sensing data and generate control data, which is then delivered to terminal nodes via router nodes. In the PIoT, the massive volume of data makes it impossible for sink nodes alone to handle all data processing. Edge computing nodes possess powerful data processing capabilities [30]; therefore, in the PIoT, edge computing nodes serve as edge gateways and generate corresponding control data. The network architecture of the PIoT is shown in figure 1.

In the PIoT architecture, edge nodes and sink nodes are interconnected either directly or through multi-hop communication links, forming a hierarchical yet partially distributed structure. The sink node functions as a global coordinator that maintains a system-level view and issues high-level control instructions, while edge nodes operate as local decision-makers with autonomous processing capabilities. In terms of control relationships, edge nodes do not strictly depend on real-time instructions from the sink node. Instead, they can make local decisions based on current network states, while following global policies or guidelines issued by the sink node. Therefore, the relationship between sink and edge nodes can be characterized as a semi-hierarchical structure combining centralized coordination and distributed decision-making. Control data can be generated by both sink and edge nodes. Sink-generated control data are typically global and strategic, whereas edge-generated control data are local and adaptive, enabling faster responses to dynamic network conditions.

3.2. Characteristic analysis

Unlike general-purpose IoT, the PIoT has the following unique characteristics:

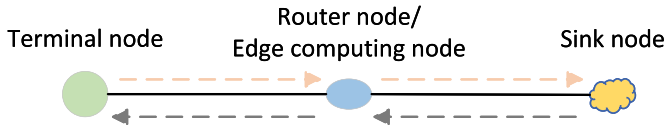


Figure 2. Data flow of the network in PIoT.

(1) Sensing-control data: in the PIoT, data is typically classified into sensing data and control data. Sensing data refers to data collected by terminal nodes, which is transmitted to sink nodes via router nodes and edge computing nodes. This type of data is also referred to as uplink data. Control data refers to control signals generated by sink nodes and edge computing nodes; these signals are transmitted as downlink data to terminal nodes to schedule and control equipment. For example, when two automated guided vehicles (AGVs) are about to meet, the terminal nodes on the AGVs generate sensing data and transmit it to an edge computing node. After the edge computing node processes the data in real time, it generates control data and immediately transmits it to the corresponding AGVs, instructing them to detour or adjust their routes to avoid collisions. The transmission direction of sensing-control data is shown in figure 2.

In the PIoT architecture, edge nodes and sink nodes are interconnected either directly or through multi-hop communication links, forming a hierarchical yet partially distributed structure. The sink node functions as a global coordinator that maintains a system-level view and issues high-level control instructions, while edge nodes operate as local decision-makers with autonomous processing capabilities. In terms of control relationships, edge nodes do not strictly depend on real-time instructions from the sink node. Instead, they can make local decisions based on current network states, while following global policies or guidelines issued by the sink node. Therefore, the relationship between sink and edge nodes can be characterized as a semi-hierarchical structure combining centralized coordination and distributed decision-making. Control data can be generated by both sink and edge nodes. Sink-generated control data are typically global and strategic, whereas edge-generated control data are local and adaptive, enabling faster responses to dynamic network conditions.

(2) Priority of delay-sensitive data: during port operations, different data types have distinct priorities. Compared to sensing data, control data directly affects the normal operation of port equipment and thus requires priority processing. The specific priority order of data transmission in real-world PIoT systems is shown in table 1.

4. PIoT cascading failure model

4.1. System model

The PIoT is represented as $G=\{V,E\}$, where $V=\{c_i|i=1,2,\ldots,N\}$ represents the set of N network nodes, and $E=[a_{i,j}]_{N\times N}$ is the set of communication edges between network nodes. If there is data exchange between node i and node j, then $a_{i,j}=a_{j,i}=1$; otherwise, $a_{i,j}=a_{j,i}=0$. The life cycle of a data packet is divided into three stages: generation, transmission, and deletion. The initial network has no load, and the data generated by nodes are based on the actual volume of port data. The next hop of each data packet is determined by the shortest transmission path. After reaching a router node, data packets are queued according to their priority order. Different time-to-live (TTL) values are assigned to different types of data packets. Table 2 summarizes all the parameters in the proposed cascading model. Without loss of generality, we make the following assumptions:

(1) Only terminal nodes can generate sensing data packets, while sink nodes and edge computing nodes generate control data packets;
(2) All routing nodes are assumed to have identical hardware configurations, their functional processing capacity C_i is assigned proportionally to their initial load $L_i(0)$ according to equation (7).
(3) The link capacity is limited. If the number of data packets being transmitted exceeds the link capacity, any excess low-priority data packets will be returned to the original node based on priority order;
(4) The node capacity is limited. If the number of real-time data packets cached in the node exceeds a certain threshold (below its maximum capacity), the node will enter a congested state and its data processing capability will reduce. If the number of real-time data packets further exceeds the node's maximum capacity, the node will fail due to overflow.

4.2. Node load

4.2.1. Terminal nodes. The real-time load $L_i^T(t)$ of terminal node i is related to the number of data packets collected by the node at time t and its transmission capability. The load $L_i^T(t)$ of terminal node i is described as

$$L_i^T(t)=I_i^T(t)-R_i^T(t), \tag{1}$$

$$I_i^T(t)=L_i^T(t-1)+S_i^T(t), \tag{2}$$

$$R_i^T(t)=\begin{cases} I_i^T(t) & I_i^T(t)<D_i^T(t) \\ D_i^T(t) & I_i^T(t)\geqslant D_i^T(t) \end{cases}, \tag{3}$$

where $I_i^T(t)$ denotes the number of data packets present in terminal node i at time t; $R_i^T(t)$ denotes the number of data packets processed by terminal node i at time t; and $S_i^T(t)$ denotes the number of data packets generated by terminal node i at time t .According to 2, the data packets present in terminal node i have two sources: the residual data packets from the previous time step and the newly generated data packets from the node itself. The data processing capability of terminal node i at time t is denoted as $D_i^T(t)$, which characterizes the maximum load

Table 2. Parameters of the cascading failure model.

Items	Descriptions
$L_i^T(t)$	Load of terminal node i at time t
$I_i^T(t)$	Number of data packets in terminal node i at time t
$R_i^T(t)$	Number of packets processed by terminal node i at time t
$D_i^T(t)$	Processing capability of terminal node i at time t
$S_i^T(t)$	Number of packets generated by terminal node i at time t
$L_i^C(t)$	Load of router node i at time t
$I_i^C(t)$	Number of data packets in router node i at time t
$R_i^C(t)$	Number of packets processed by router node i at time t
$D_i^C(t)$	Processing capability of router node i at time t
$X_{i,j}(t)$	Number of packets sent from neighbor j to node i at time t
$E_{i,j}(t)$	Indicator of the existence of a link between nodes i and j
$U_i^B(t)$	Number of data packets returned to node i at time t
$f_i^C(t)$	Congestion level of node i at time t
$L_i^E(t)$	Load of edge computing node i at time t
$I_i^E(t)$	Number of data packets in edge computing node i at time t
$R_i^E(t)$	Number of packets processed by edge computing node i at time t
$\langle k \rangle$	Average degree of the initial router network
W_i	Capacity of node i
$W_{i,j}$	Link capacity between nodes i and j
$A_{i,j}(t)$	Number of packets that node i attempts to transmit to node j at time t
α	Node overload-tolerance coefficient
λ	Link overload-tolerance coefficient
β	Congestion-tolerance coefficient
N^R	Number of router nodes in the network
$v_i(t)$	Processing speed of node i at time t
$T_{p,i}(t)$	Processing time of packet p at node i
B_p	Resource requirement of packet p
δ_p	Complexity of data packet p
$T_p^H(t)$	Remaining time-to-live of packet p in transit
μ_p	Time-to-live of data packet p
$\rho(t)$	Packet survival rate
$T_{\text{delay}}(t)$	Network-wide average transmission latency of all valid packets at time t

the terminal node can process. When the total number of data packets in node i exceeds its processing capability $D_i^T(t)$, the node can only transmit $D_i^T(t)$ data packets; the remaining data packets will be cached in the node and processed for transmission in the next time step.

4.2.2. Router nodes. The real-time load $L_i^C(t)$ of router node i at time t is related to the number of data packets transmitted to it by neighboring nodes and the number of data packets unsent at time $t-1$ due to link bandwidth limitations. Therefore, the load $L_i^C(t)$ of router node i at time t is described as

$$L_i^C(t) = I_i^C(t) - R_i^C(t), \tag{4}$$

$$I_i^C(t) = L_i^C(t-1) + \sum_{j=1}^{\Omega(t)} X_{i,j}(t) E_{i,j}(t) + U_i^B(t), \tag{5}$$

$$R_i^C(t) = \begin{cases} I_i^C(t) & I_i^C(t) < D_i^C(t) \\ D_i^C(t) & I_i^C(t) \geqslant D_i^C(t) \end{cases}, \tag{6}$$

$$D_i^C(t) = \left[1 - f_i^C(t)\right] \frac{\Omega^C(t)}{\langle k \rangle}, \tag{7}$$

where $I_i^C(t)$ denotes the number of data packets present in router node i at time t; $R_i^C(t)$ denotes the number of data packets processed by router node i at time t; $X_{i,j}(t)$ denotes the number of data packets transmitted from neighboring node (sink node, router node, or terminal node) j to node i; $E_{i,j}(t)$ denotes the presence of a link between node i and node j at time t (if a link exists, $E_{i,j}(t) = 1$; otherwise, $E_{i,j}(t) = 0$); and $U_i^B(t)$ denotes the number of data packets that fail to be sent due to bandwidth limitations at time t and thus return to node i. Details of $U_i^B(t)$ are provided in the section 4.3. The data processing capability of router node i at time t is denoted as $D_i^C(t)$, which characterizes the maximum load the router node can process. When the total number of data packets in node i exceeds its processing capability $D_i^C(t)$, the node can only transmit $D_i^C(t)$ packets, and the remaining packets will remain in the node's cache. Otherwise, the node can transmit all data packets in its cache, and its real-time load is fully released

(i.e. $L_i^C(t) = 0$). In the router network of the PIoT, the processing capability of a router node is positively correlated with its degree (i.e. the number of neighboring nodes). The data processing capability $D_i^C(t)$ in equation (7) is normalized by the average degree $\langle k \rangle$ of the initial router network to reflect the inherent relationship between topological connectivity and resource requirements. In the PIoT infrastructure, the average degree $\langle k \rangle$ functions as a quantitative baseline for the connectivity density of the routing layer. By defining the processing capability as a function of the ratio between the local degree $\Omega_i^C(t)$ and this baseline, the model implements a proportional resource allocation strategy: nodes with higher connectivity are allocated greater processing capacity to accommodate the aggregated sensing and control data flows converging through multiple links. This mathematical structure ensures that the processing resources are distributed according to the expected traffic load, thereby optimizing the load-balancing performance of the routing infrastructure and preventing localized congestion in high-connectivity regions of the network. $f_i^C(t)$ represents the congestion degree of node i, which indicates the negative impact of congestion on the router node's processing capability. Details of $f_i^C(t)$ are provided in the section 4.4.

Sensing and control flows exhibit complex load coupling characteristics at router nodes. On one hand, sensing data as uplink traffic features high concurrency, which is prone to forming initial backlogs at critical router nodes, thereby reducing the nodes' residual processing capacity. On the other hand, according to priority policies, high-priority control data preempts bandwidth resources, forcing sensing data already on the verge of congestion to further remain in the cache. This not only results in a decrease in the survival rate of sensing data but also indirectly delays the generation of subsequent control instructions by extending the round-trip latency of sensing data. Such a cross-data-type delay positive feedback loop serves as a critical incentive for PIoT cascading failures to evolve from localized congestion into network-wide paralysis.

4.2.3. Edge computing nodes. The real-time load $L_i^E(t)$ of edge computing node i at time t is defined as

$$L_i^E(t) = I_i^E(t) - R_i^E(t), \tag{8}$$

$$I_i^E(t) = I_i^E(t-1) + \sum_{j=1}^{\Omega^T(t)} X_{i,j}^T E_{i,j}(t) + \sum_{j=1}^{\Omega^S(t)} X_{i,j}^S E_{i,j}(t), \tag{9}$$

where $R_i^E(t)$ denotes the number of data packets processed by edge computing node i at time t, and $I_i^E(t)$ denotes the total number of data packets in the cache space of edge computing node i at time t. $L_i^E(t-1)$ represents the number of residual data packets from the previous time step; $X_{i,j}^T$ represents the number of data packets received by the edge computing node from terminal node j ; $\Omega^T(t)$ represents the set of terminal nodes connected to edge computing node i, thus $\sum_{j=1}^{\Omega^T(t)} X_{i,j}^T(t)$ denotes the total number of data packets received by edge computing node i from its connected terminal nodes; $X_{i,j}^S$ represents the number of data packets received by edge computing node i from sink node j; and $\sum_{j=1}^{\Omega^S(t)} X_{i,j}^S(t)$ denotes the total number of data packets received by edge computing node i from its connected sink nodes.

4.3. Capacity and transmission

In a real-world PIoT, the hardware configuration of network node devices determines the system's overall data transmission and storage capabilities. When a node experiences high load, additional bandwidth and cache resources are typically allocated to enhance its load-handling capacity. Therefore, it is reasonable to assume that a network node's capacity is positively correlated with its actual load. For terminal nodes, router nodes, and edge computing nodes, their capacities can thus be expressed as

$$W_i = (1+\alpha) L_i(0), \tag{10}$$

where α denotes the overload tolerance coefficient, and $L_i(0)$ is defined as the steady-state nominal load of node i, representing the baseline traffic demand.

Link capacity refers to the maximum number of data packets that can be transmitted between nodes i and j; thus, it can be expressed as

$$W_{i,j} = (1+\lambda) L_{i,j}(0), \tag{11}$$

where λ is the link capacity limit coefficient, and $L_{i,j}(0)$ denotes the steady-state nominal load of the communication link (i,j). Therefore, the number of data packets returned to node i in the transmission from node i to node j due to bandwidth constraints equals the number of data packets node i attempts to send to node j via the link minus the link capacity, which is expressed as:

$$U_{\mathrm{i}}^B(t) = \sum_{j=1}^{\Omega(t)} \max\left(0, A_{i,j}(t) - W_{i,j}\right), \tag{12}$$

where $A_{i,j}(t)$ represents the number of data packets that node i attempts to transmit to node j at time t.

4.4. Node congestion state

In most of the existing cascading models [31–33], nodes have only two states: normal and overloaded. Under their assumptions, if a node's load exceeds its capacity, it is permanently removed from the network. This assumption is reasonable for power grids; however, it does not hold for the PIoT. Unlike power overload in power grids, packet overload in the PIoT does not cause physical damage to packet-processing nodes (i.e. router nodes). The primary negative impacts include packet expiration, congestion-induced failure, and packet loss. When the number of generated packets is excessively large, the real-time traffic load on certain key nodes will exceed

their data processing capacity. This will further cause some packets to fail to be transmitted in a timely manner, requiring them to remain temporarily in the node's cache. Following multiple rounds of data accumulation, packet backlogs will lead to severe network traffic congestion, with the congestion degree worsening over time. Therefore, in our model, in addition to 'normal' and 'overloaded', there are two other states for router nodes, namely 'congested' and 'isolated'. The congestion function $f_i^C(t)$ is defined as

$$f_i^C(t) = \begin{cases} 0 & L_i^C(t) < (1-\beta) W_i \\ \dfrac{L_i^C(t) - (1-\beta) W_i}{\beta W_i} & (1-\beta) W_i \leqslant L_i^C(t) \leqslant W_i \\ 1 & L_i^C(t) > W_i \end{cases}, \tag{13}$$

where β is the congestion tolerance coefficient, which aims to adjust the threshold between the normal state and the congested state. Its value is determined by the resources of the router node.

When $L_i^C(t) < (1-\beta)W_i$, the node is in a normal state, its processing capability remains unchanged, and it functions normally. When $(1-\beta)W_i \leqslant L_i^C(t) \leqslant W_i$, the node is in a congested state, its processing speed decreases, but it can still receive and transmit data packets. When $L_i^C(t) > W_i$, the node is in an overloaded state, its processing speed is 0, and it cannot operate temporarily. If all neighboring nodes of a router node are overloaded, even if its load is within the capacity range, the node will still be unable to send data packets due to interrupted transmission paths, and such nodes will transition to the 'isolated' state. When some neighboring nodes recover from the overloaded state and resume normal communication with the sink node, isolated nodes can revert to a normal state. If a node is damaged due to its inability to receive control data, its state is termed the 'interdependent failure' state. An Overloaded state is considered a temporary, recoverable condition; a component is permitted to rejoin the network functional set as soon as its real-time load $L(t)$ returns to a level below its capacity. However, Isolated and Interdependent Failure states are non-recoverable within the simulation timeframe. Specifically, the operational status of a terminal node depends on the Control TTL: if the node does not receive a valid control packet before the expiration of its control TTL, it is deemed to have lost control consistency. Consequently, the node enters a permanent shutdown state to ensure system safety, representing a physical failure or emergency stop of the port machinery.

4.5. Caching mechanism

In the PIoT, there are two types of caching mechanisms: the in-transit caching mechanism and the end-of-line caching mechanism. We set TTL μ_p for data packets p. In the end-of-line caching mechanism, control packets are discarded once their μ_p is exceeded during transmission. In contrast, sensing packets are not discarded upon exceeding their μ_p; instead, they are retained and continue to be transmitted until they reach the sink node. In the in-transit caching mechanism, if a data packet p exceeds its μ_p during transmission, it is immediately cleaned. If the sensing data exceed μ_p when reaching the sink node, the sink node will not generate the corresponding control data. If a control data packet exceed its μ_p when reaching the terminal node, the terminal node will fail. The capability of node i to process data packets can be defined as

$$v_i(t) = D_i^C(t). \tag{14}$$

To further analyze the transmission characteristics of different types of data packets and their impacts on network performance, it is necessary to comprehensively consider the relationship between the data packet priority mechanism and the processing capabilities of nodes. In the PIoT, setting data packet priorities can effectively improve the transmission efficiency of key tasks, but it also introduces potential impacts on the processing of data packets from other task types. Therefore, we incorporate parameters such as data packet complexity, resource capacity, and node processing capability to quantify the waiting time and processing time of data packets at nodes. We define the complexity of data packet p as δ_p and the resource capacity of each data packet p as B_p. Thus, the processing time of data packet p at node i is defined as

$$T_{p,i}(t) = \frac{B_p \cdot \delta_p}{v_i(t)}. \tag{15}$$

4.5.1. In-transit caching mechanism. According to the data priority policy, if data packet p has the highest priority at node i, its total time at the node equals its processing time. If there are q data packets with higher priority than data packet p at node i, the total time of data packet p at node i is the sum of the waiting time and its own processing time. The waiting time refers to the total time of all higher-priority data packets at node i.

$$T_{p,i}(t) = \sum_{q=1}^{n_q(t)} T_{q,i}(t) + \frac{B_p \cdot \delta_p}{v_i(t)}, \tag{16}$$

where $n_q(t)$ denotes the number of data packets q with higher priority and $T_{q,i}(t)$ denotes the total time of a single higher-priority data packet at node i. Thus, the remaining effective transmission time of data packet p is

$$T_p^H(t) = \mu_p - \sum_{i=1}^{m_i(p)} T_{p,i}(t), \tag{17}$$

where $m_i(p)$ denotes the number of nodes that data packet p has passed through. Therefore, in this mechanism, if $T_p^H(t) \leqslant 0$, this data packet is evicted at this node; if $T_p^H(t) > 0$, the data packet is processed after waiting and transmitted to the next node until it is transmitted to the last node.

4.5.2. End-of-line caching mechanism. The total time that data packet p spends at all nodes it passes through is defined as

$$T_p(t) = \sum_{i=1}^{m_i(p)} T_{p,i}(t), \tag{18}$$

when the data packet is transmitted to the last node, $T_p(t)$ is compared with the set TTL μ_p based on the packet's type. If $T_p(t) > \mu_p$, the data packet is deemed invalid due to timeout.

4.6. Reliability and performance metrics

To quantitatively evaluate the impact of cascading failures on the PIoT, two core metrics are defined. First, the data packet survival rate $\rho(t)$ measures the reliability:

$$\rho(t) = \frac{Q_S(t) + Q_T(t)}{Q(t)} \tag{19}$$

where $Q_S(t)$ and $Q_T(t)$ denote the cumulative numbers of valid sensing and control data packets, respectively, and $Q(t)$ denotes the total number of generated data packets up to time t.

The end-to-end delay $T_{\text{delay}}(t)$ is defined as the network-wide average transmission latency of all valid packets at time t. In high-bandwidth smart port environments (e.g. 5G private networks), the physical link transmission delay is assumed negligible compared to nodal queuing and processing delays.

$$T_{\text{delay}}(t) = \frac{1}{Q_S(t) + Q_T(t)} \sum_{p=1}^{Q_S(t)+Q_T(t)} \left(\sum_{i=1}^{m_i(p)} T_{p,i}(t) \right) \tag{20}$$

where $Q_S(t) + Q_T(t)$ denotes the total number of packets processed or delivered during time step t, and $T_{p,i}(t)$ represents the real-time delay experienced by packet p at node i along its path of length $m_i(p)$.

5. Experimental results

5.1. Experimental setup

An experimental PIoT platform is constructed (figure 3) comprising 485 concentrators, 4G LTE DTUs, RS-232 interfaces, and environmental sensors to simulate port equipment and generate realistic data flows. Based on these flows, we simulate a network of 250 fixed, 50 mobile, and 30 edge nodes (average degree $k = 4$). The topology is updated each 10 ms step: mobile nodes follow the Random Waypoint model, while stationary nodes maintain a disk-based communication range R. Global shortest paths are recomputed via Dijkstra's algorithm to capture these dynamics. All nodes comply with IEEE 802.11ax. To account for gateway overheads, a 1.2 scaling factor maps empirical latency to packet TTL. Traffic includes high-frequency sensing (1.0 pkt/step) and intermittent control packets (0.1 pkt/step from edge/sink nodes). Finally, we evaluate the impact of modeling parameters and failure mechanisms on network reliability.

In the PIoT, the presence of mobile terminal nodes such as AGVs and container trucks means the network topology is not static. The network topology at different time steps is depicted in figure 4. Furthermore, the changing locations of mobile terminal nodes over different time steps also contribute to dynamic load distribution. Figure 5 shows the load distribution at different time steps.

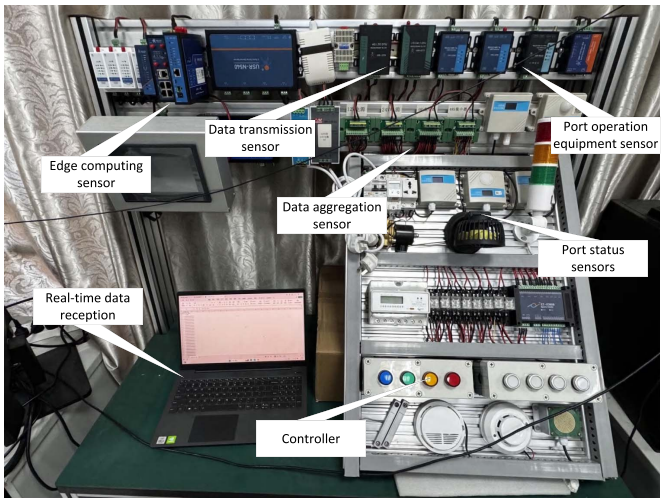


Figure 3. Experimental platform of PIoT.

In the experiments, to comprehensively evaluate the data packet survival rate of the PIoT under cascading failures, we introduce both node (single-node and multi-node) and link failure triggering mechanisms. Under single-node failure triggering mechanisms, we evaluate the impact of key modeling parameters on PIoT performance and compare the two caching mechanisms. For multi-node failure triggering mechanisms, we introduce four node failure mechanisms depending on the node type: terminal failure triggering (TFT), router failure triggering (RFT), sink failure triggering (SFT), and edge failure triggering (EFT). These mechanisms are designed to observe how triggering failures in different node types affects the network's data packet survival rate. For link failure triggering, we introduce two mechanisms depending on the node type: sensing link failure triggering (SLFT) and sensing-control link failure triggering (SCLFT), to observe their distinct impacts on the network when control data or sensing data is lost. Specifically, the failure selection for TFT and SLFT is based on random selection from the corresponding component sets. In contrast, RFT, EFT, SFT, and SCLFT are implemented as targeted attacks, where components are prioritized for removal based on their topological importance, within their respective functional tiers. To ensure a fair comparison between the in-transit caching mechanism and the end-of-line caching mechanism, we employ fixed random seeds for each failure scenario to maintain consistent initial network states and node mobility trajectories . Furthermore, each simulation is conducted for 50 independent runs, and the results are reported as average values to minimize stochastic bias and ensure statistical accuracy.

5.2. Impact of Key parameters under single-node failure

5.2.1. Impacts of the number of router nodes N^R. Figure 6 shows that as the number of router nodes N^R increases, the damage caused by cascading failures tends to decrease. As the number of router nodes increases from 100 to 300, the data packet survival rate rises by approximately 20 percent under both caching mechanisms. This improvement is attributed to the increased number of router nodes, which helps

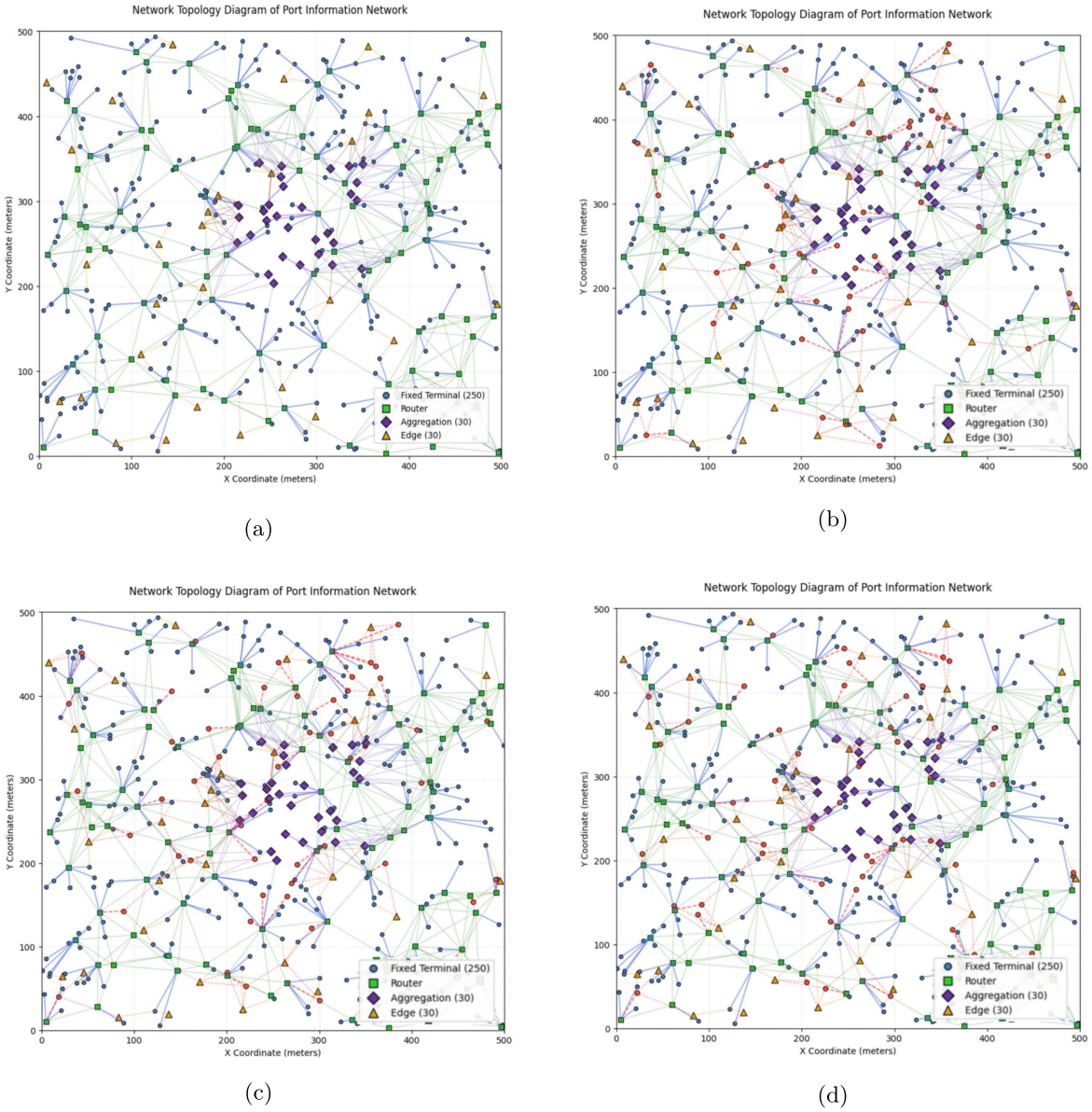


Figure 4. Topological of the PIoT at different time steps. (a) $t = 0$, (b) $t = 100$, (c) $t = 200$, (d) $t = 300$.

reduce traffic load and minimize losses caused by cascading failures.

5.2.2. Impact of node capacity expansion coefficient alpha. Figure 7 shows that as the expansion coefficient α increases, the data packet survival rate after cascading failures improves under both mechanisms. In our model, α denotes the node capacity expansion coefficient: a higher α value provides more buffer space to handle traffic load, thereby reducing the risk of data packet loss caused by node failures or congestion. However, increasing α indefinitely does not lead to infinite improvements in the data packet survival rate; instead, a critical threshold α^* exists. When α reaches or exceeds α^*, the data packet survival rate saturates.

5.2.3. Impact of link capacity expansion coefficient lambda. Figure 8 illustrates that as the expansion coefficient λ increases, the data packet survival rate increases under both mechanisms. In our model, λ denotes the link capacity expansion coefficient: increasing λ enhances link capacity, thereby improving transmission capability and enabling links to handle larger volumes of data traffic per unit time. With improved link capacity, both queuing delay and congestion probability during data packet transmission are reduced, leading to fewer backlogged data packets awaiting processing. This significantly mitigates cascading failures triggered by insufficient link capacity. Furthermore, a critical threshold λ^* is observable: when λ reaches or exceeds λ^*, the system's data packet survival rate saturates.

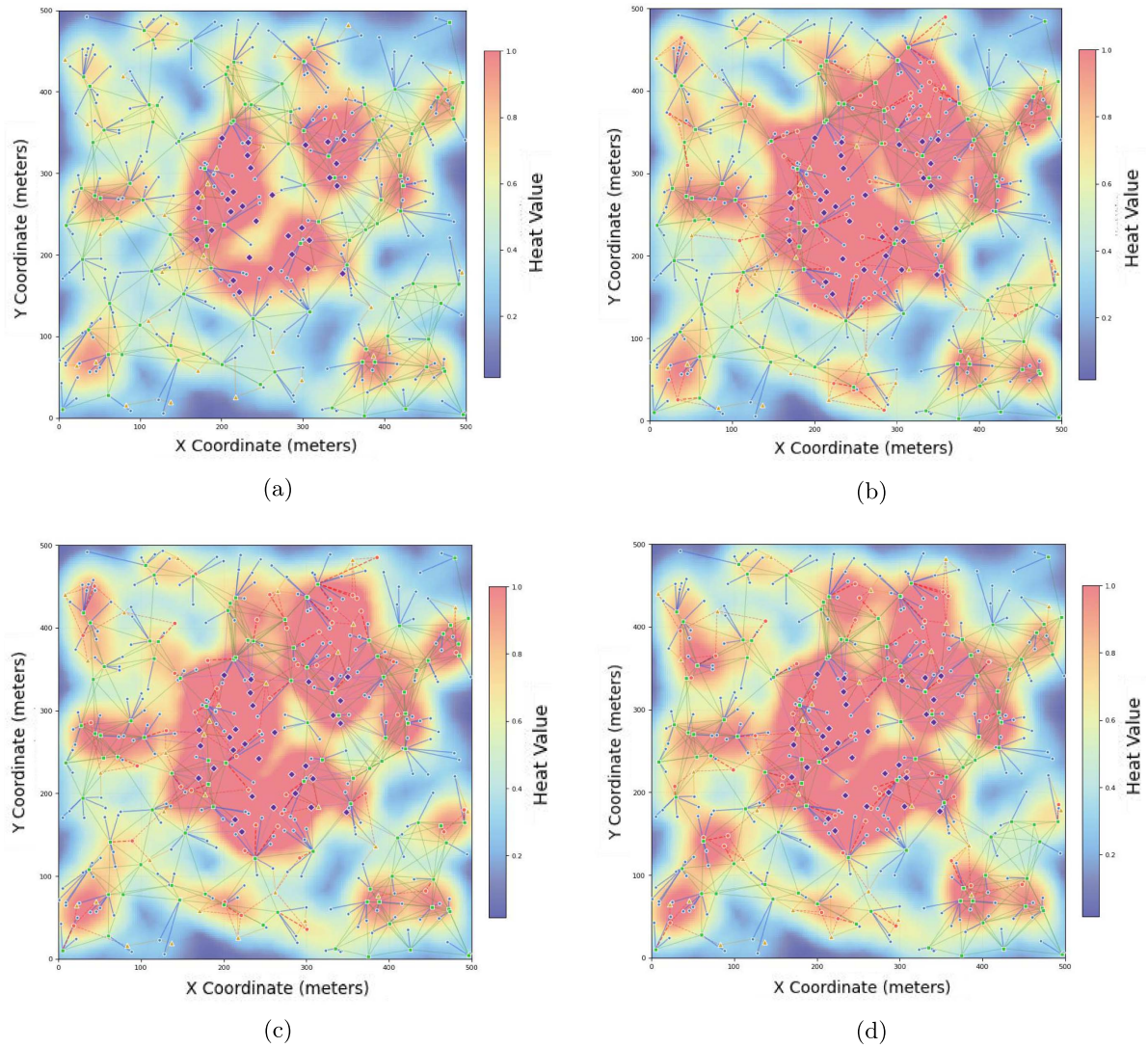


Figure 5. Heatmap of load distribution in PIoT at different time steps. (a) $t = 0$, (b) $t = 100$, (c) $t = 200$, (d) $t = 300$.

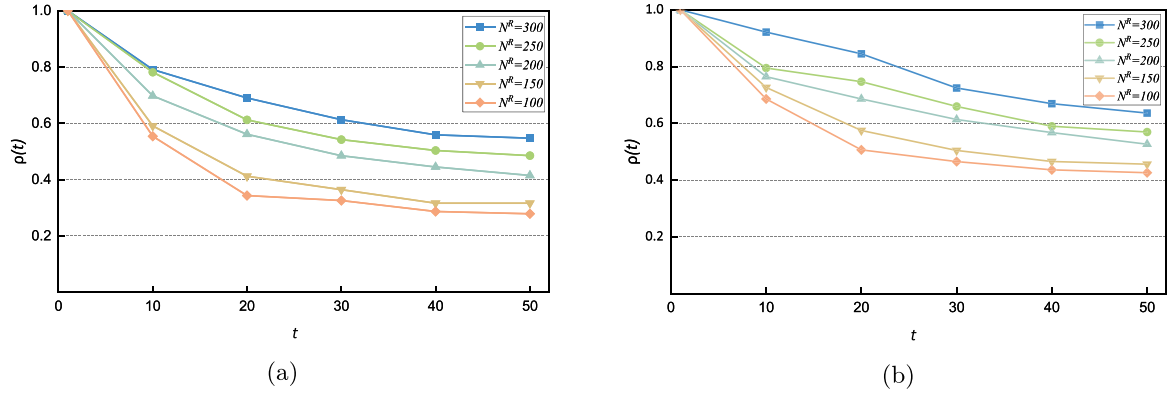


Figure 6. Variation of data packet survival rate with N^R ($\alpha = \lambda = 0.6, \beta = 0.2$). (a) End-of-line caching mechanism, (b) in-transit caching mechanism.

5.2.4. Impact of congestion coefficient beta. Figure 9 shows that as β increases, the data packet survival rate decreases. In our model, β denotes the node congestion tolerance coefficient, which adjusts the threshold for distinguishing between a node's normal and congested states. A higher β value makes router nodes more likely to enter a congested state, thereby increasing data packet delivery latency. Furthermore, an increase in β raises the risk of node

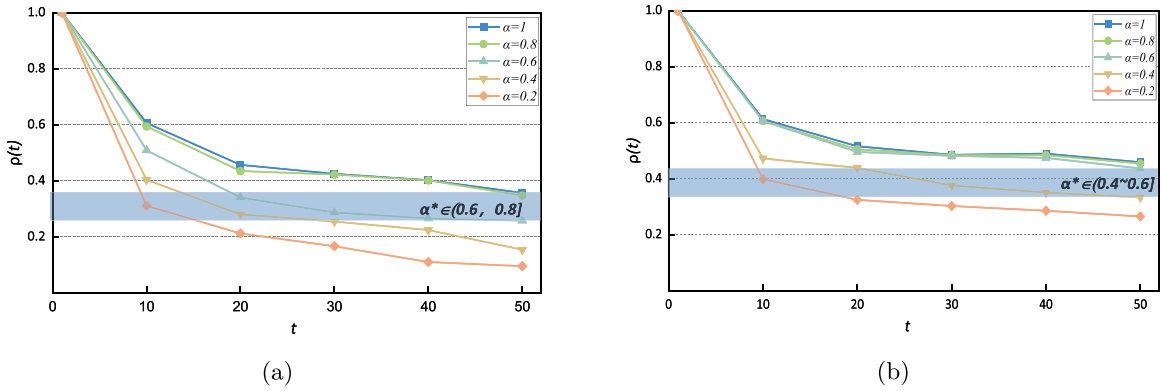


Figure 7. Variation of data packet survival rate with α ($\lambda = 0.6, \beta = 0.2, N^R = 300$). (a) End-of-line caching mechanism, (b) in-transit caching mechanism.

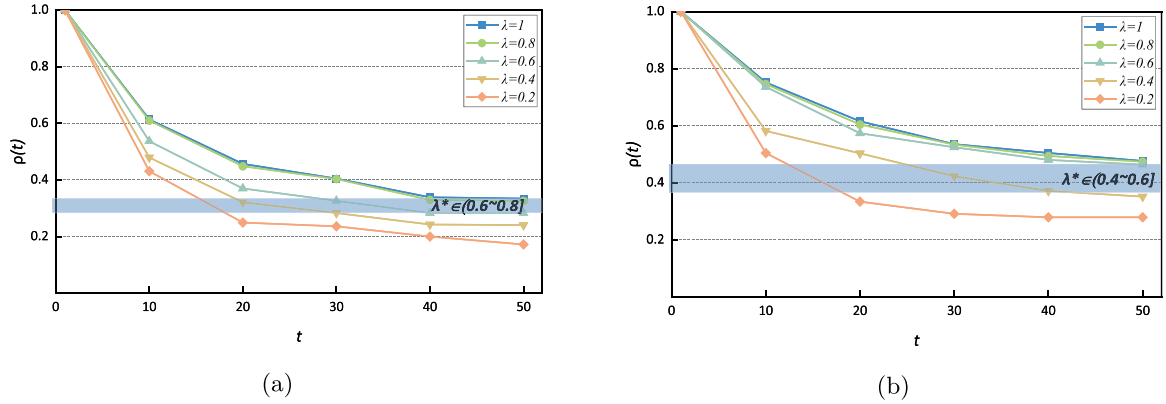


Figure 8. Variation of data packet survival rate with λ ($\alpha = 0.6, \beta = 0.2, N^R = 300$). (a) End-of-line caching mechanism, (b) in-transit caching mechanism.

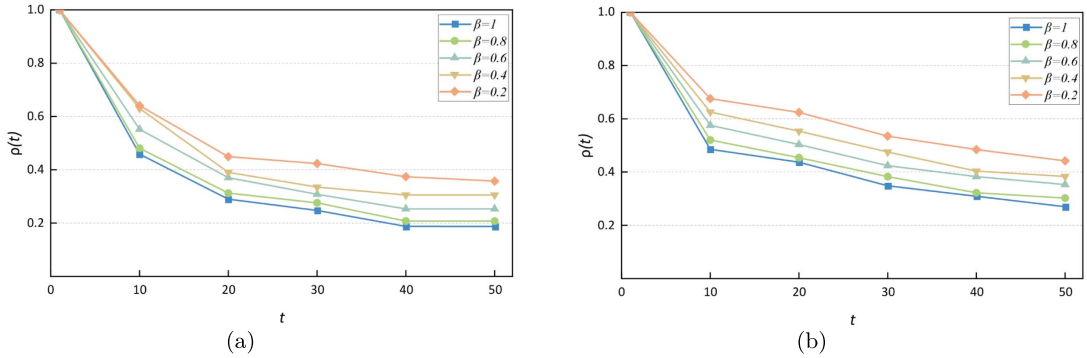


Figure 9. Variation of data packet survival rate with β ($\alpha = \lambda = 0.6, N^R = 300$). (a) End-of-line caching mechanism, (b) in-transit caching mechanism.

overload due to data packet backlog, making data packet delivery more likely to be blocked.

5.3. Comparison of two caching mechanisms

Figure 10 shows that the in-transit caching mechanism achieves a higher data packet survival rate during cascading failures when compared to the end-of-line caching mechanism. This advantage stems from the fact that the in-transit caching mechanism immediately removes expired data packets during transmission, preventing them from occupying the node buffer space. In contrast, the end-of-line caching mechanism only assesses data packet validity upon arrival at the last node. This can lead to the accumulation of expired data packets during transmission. This accumulation consequently blocks the forwarding of valid data packets, causing them to exceed their TTL. Although the in-transit caching mechanism demonstrates superior performance in terms of data packet

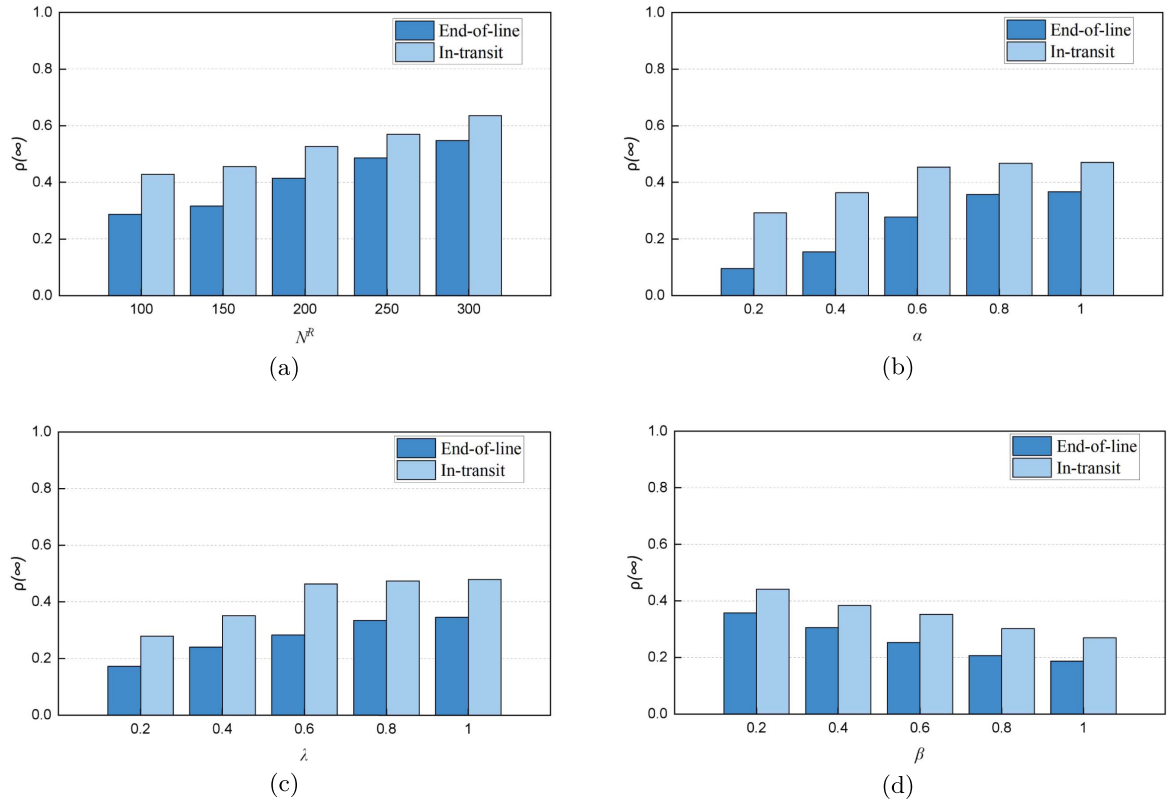


Figure 10. Comparison of packet survival rates between two caching mechanisms under different parameters. (a) Different N^R values, (b) different α values, (c) different λ values, (d) different β values.

survival rate, the end-of-line caching mechanism is preferable in certain scenarios. This is because the end-of-line caching mechanism ensures the complete delivery of data packets to both sink nodes and terminal nodes, thereby preserving data integrity. Figure 11 shows the average end-to-end delay of the two caching mechanisms under different parameter settings. As a direct measure of transmission latency, this metric reflects the combined effects of routing path length, queuing delay, and node congestion during cascading failures. It can also be observed that the in-transit caching mechanism outperforms the end-of-line caching mechanism in reducing delay and improving data transmission efficiency.

Figure 12 presents heatmaps illustrating the data packet survival rates of both mechanisms across the [α, λ] parameter space under single-node failure triggering. The results further validate the existence of critical thresholds α^* and λ^*, which demarcate the boundary beyond which the system's data packet survival rate reaches saturation. From a physical perspective, the threshold boundaries [α, λ] represent the minimum capacity conditions required for the system to maintain a stable transmission state under cascading disturbances. The region below the boundary corresponds to a failure-prone regime, where insufficient node or link capacity leads to rapid congestion accumulation and cascading propagation. In contrast, the region above the boundary indicates a relatively stable regime, where enhanced capacity effectively suppresses load redistribution and prevents large-scale failures. Moreover, the difference between the two heatmaps highlights the distinct resource utilization mechanisms of the caching mechanism. The in-transit caching mechanism achieves comparable survival performance under lower α and λ values, as it dynamically buffers packets along transmission paths and alleviates local congestion. In contrast, the end-of-line caching mechanism relies more on node-side storage, requiring higher capacity expansion to reach the same stability region. This demonstrates that in-transit caching is more efficient in utilizing distributednetwork resources, while end-of-line caching demands stronger local buffering capability.

The in-transit caching mechanism generally achieves a higher data packet survival rate $\rho(t)$ during cascading failures compared to the end-of-line caching mechanism. This performance advantage stems from the in-transit caching mechanism's ability to immediately remove expired data packets during the transmission process, which prevents them from occupying critical buffer space within the network nodes. In contrast, the end-of-line caching mechanism only assesses data packet validity upon its arrival at the final destination node. This delay in validation often leads to the accumulation of invalid data packets, thereby obstructing the forwarding of valid, time-sensitive information and causing further packet timeouts.

To further investigate the impact of these mechanisms on system reliability, we evaluate their performance under the failure of a single edge computing node. Unlike standard

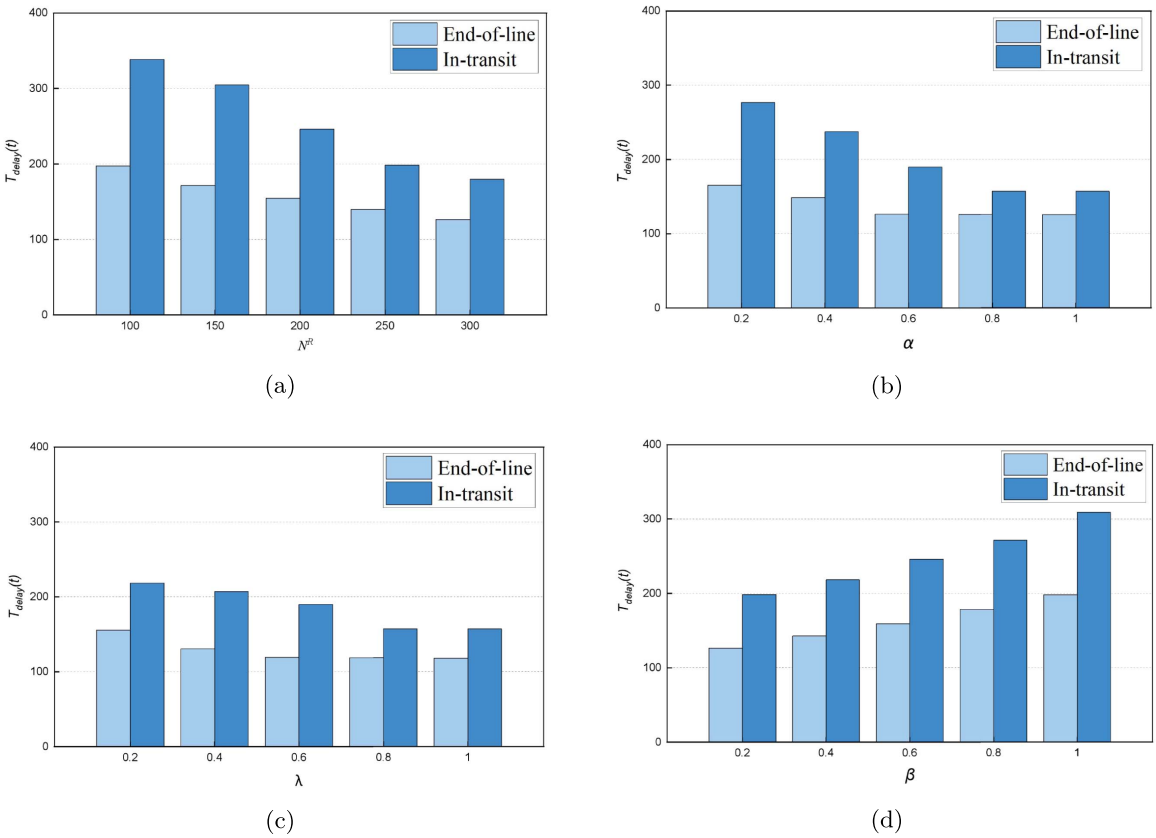


Figure 11. Comparison of average end-to-end delay between two caching mechanisms under different parameters. (a) Different N^R values, (b) different α values, (c) different λ values, (d) different β values.

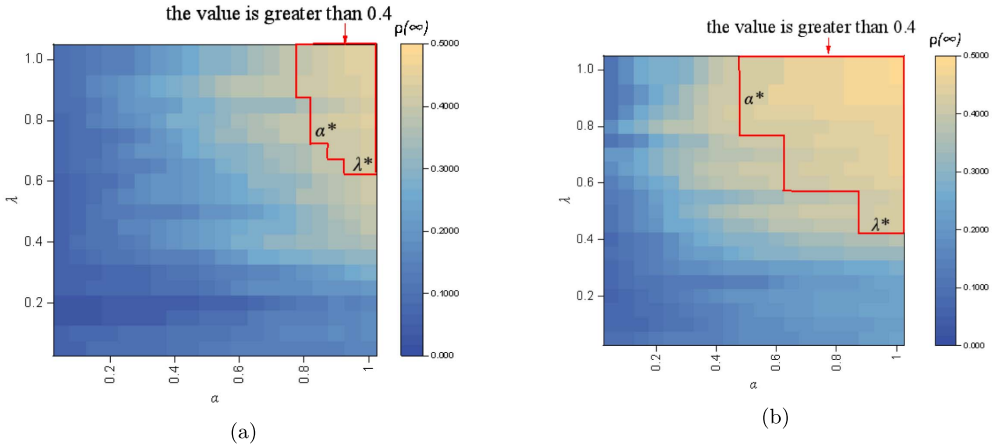


Figure 12. Data packet survival rate in the parameter space [α, λ] under two caching mechanisms. (a) End-of-line caching mechanism, (b) in-transit caching mechanism.

router nodes, edge computing nodes in the PIoT serve as critical hubs for processing sensing data and generating necessary control instructions. For these simulations, the key modeling parameters are fixed at a node overload-tolerance coefficient $\alpha = 0.6$, a link overload-tolerance coefficient $\lambda = 0.6$, a congestion-tolerance coefficient $\beta = 0.2$, and the number of router nodes $N_R = 300$. The experimental results are summarized in table 3.

5.4. Impact of multi-node failures

The experimental results indicate that the in-transit caching mechanism outperforms the end-of-line caching mechanism in terms of data packet survival rate. Therefore, to mitigate the impact of data congestion on subsequent experiments, the in-transit caching mechanism is adopted in the experiments involving multi-node failure triggering and link

Table 3. Performance comparison of caching mechanisms under single edge node failure ($\alpha = \lambda = 0.6$, $\beta = 0.2$, $N^R = 300$).

Mechanism	$\rho(t)$	$T_{\text{delay}}(t)$ (ms)
In-transit caching	0.57	142
End-of-line caching	0.39	198

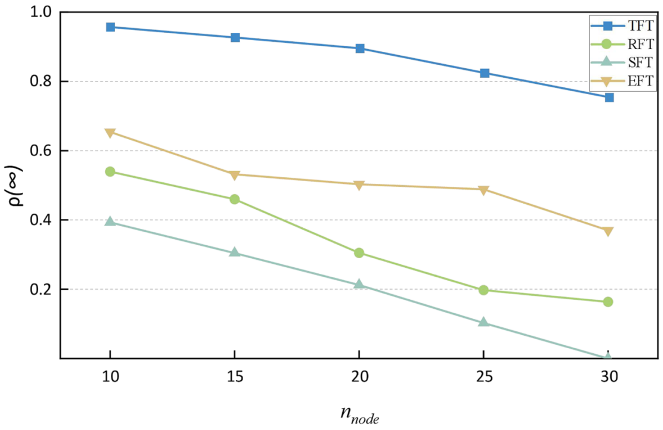


Figure 13. Data packet survival rate under different failure triggering mechanisms ($\alpha = \lambda = 0.6$, $\beta = 0.2$,$N^R = 300$).

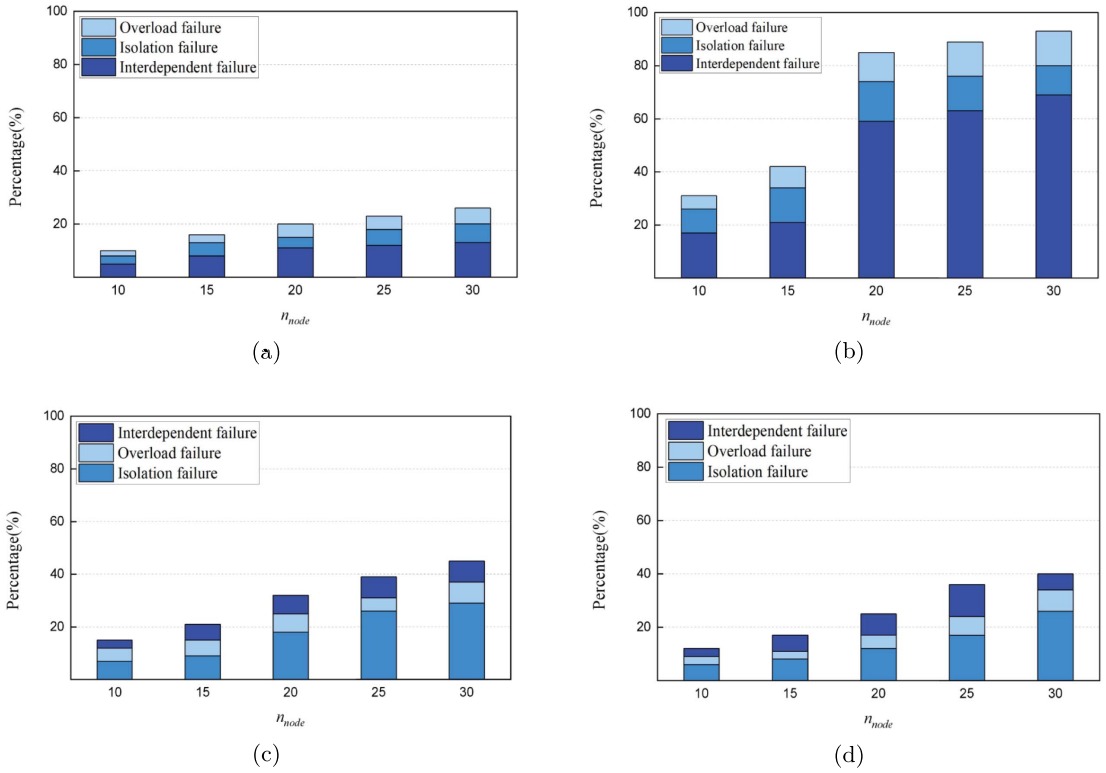


Figure 14. Proportion of node states under different failure triggering mechanisms ($\alpha = \lambda = 0.6$, $\beta = 0.2$,$N^R = 300$). (a) TFT, (b) SFT, (c) RFT, (d) EFT.

failure triggering. Figure 13 shows that the four failure triggering mechanisms exert significantly different levels of damage on the PIoT. Under TFT, even when 30 terminal nodes are compromised, the remaining nodes can still retain functionality. In contrast, both RFT and EFT lead to a progressive decline in the data packet survival rate as the number of affected nodes increases—this drives an increasing number of node failures due to cascading effects. The most severe impact on data packet survival rate occurs under SFT, where the degradation is particularly pronounced. Figure 14 shows that networks

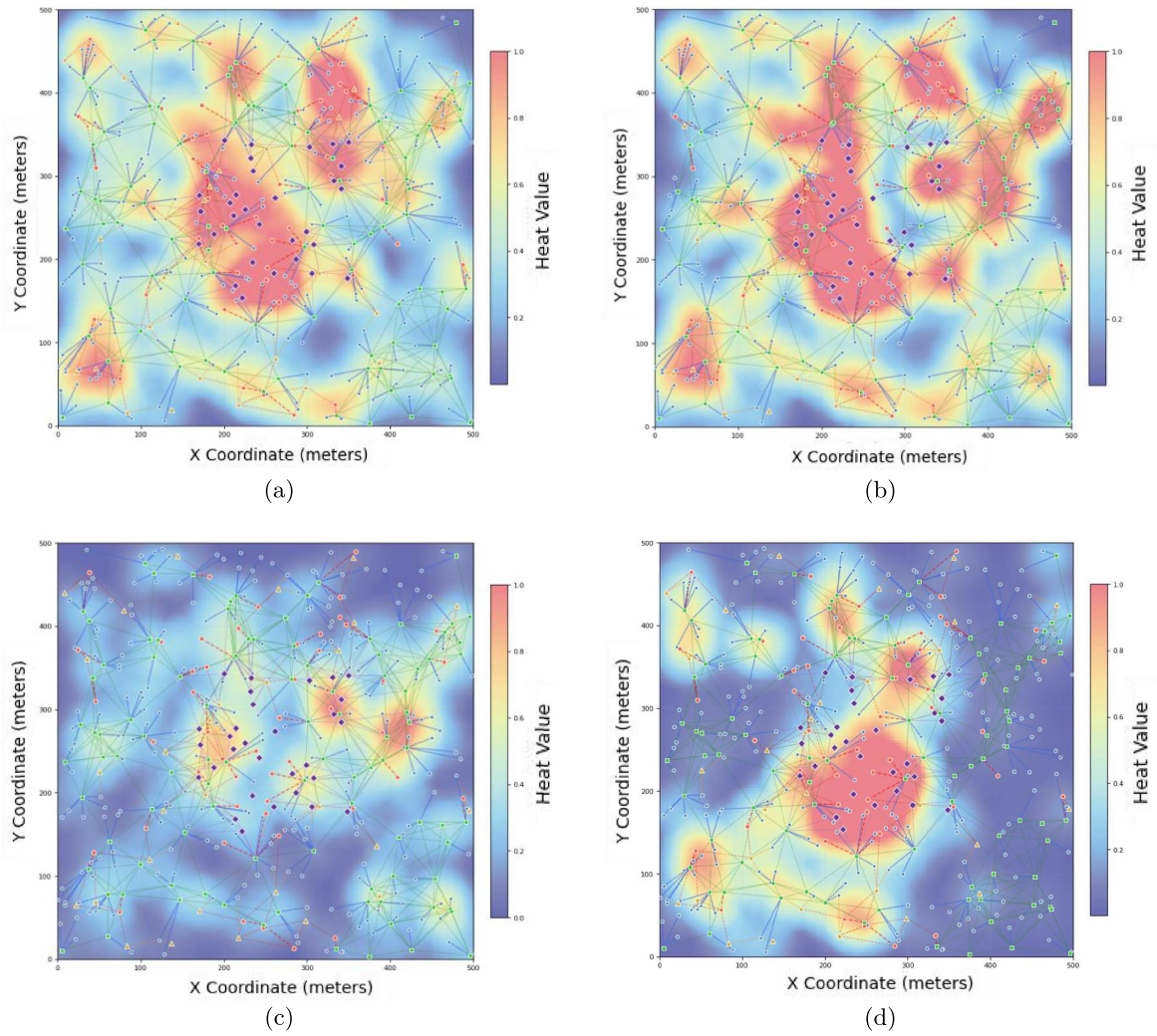


Figure 15. Heatmap of load distribution under different failure triggering mechanisms at time t ($\alpha = \lambda = 0.6$, $\beta = 0.2$,$N^R = 300$). (a) TFT, (b) SFT, (c) RFT, (d) EFT.

under RFT and EFT have a higher proportion of nodes in interdependent failure states. This is because when a router node or edge computing node fails, the subsequent loss of control data can cause numerous terminal nodes to enter interdependent failure states. In contrast, TFT leads to a relatively lower proportion of failed nodes. This is primarily because terminal node failures have limited impact on other network components, resulting in smaller-scale cascading failures. SFT, meanwhile, results in the highest proportion of overloaded nodes: when a sink node is compromised, the router nodes connected to it lose the ability to transmit data to their destinations, leading to complete paralysis of the associated data transmission paths.

Figure 15 shows that among the four node failure triggering mechanisms, SFT exerts the most severe impact on the PIoT. This demonstrates that sink nodes play a critical role in the network and should be prioritized as key protection targets in port environments.

5.5. Impact of link failures

Our prior research has demonstrated that both RFT and EFT can cause partial failures of terminal nodes connected to the affected router or edge nodes. Therefore, we further investigate cascading failures triggered by different types of link failures within the network. Specifically, we trigger failures in sensing links and sensing-control links separately to examine their respective impacts on cascading failure.

Figure 16 shows that for the same scale of failure triggering, the data packet survival rate under SCLFT is lower than that under SLFT. Even when only a small number of links are compromised under SCLFT, the data packet survival rate is significantly impacted. This indicates that sensing-control links play a more critical role in maintaining the normal operation of the PIoT.

Figure 17 shows that compared to SLFT, the network under SCLFT exhibits a larger proportion of nodes in interdependent

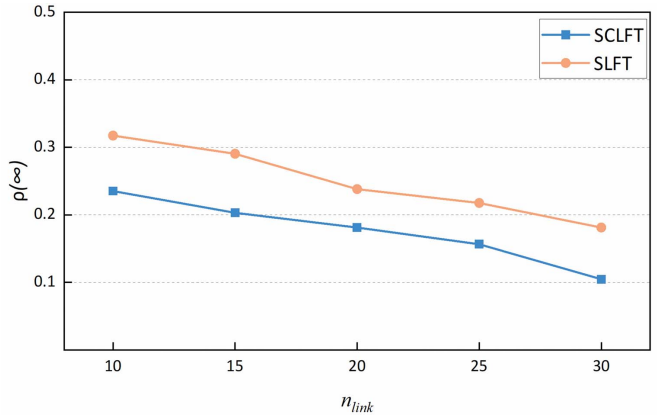


Figure 16. Data packet survival rate under SCLFT and SLFT($\alpha = \lambda = 0.6$, $\beta = 0.2$,$N^R = 300$).

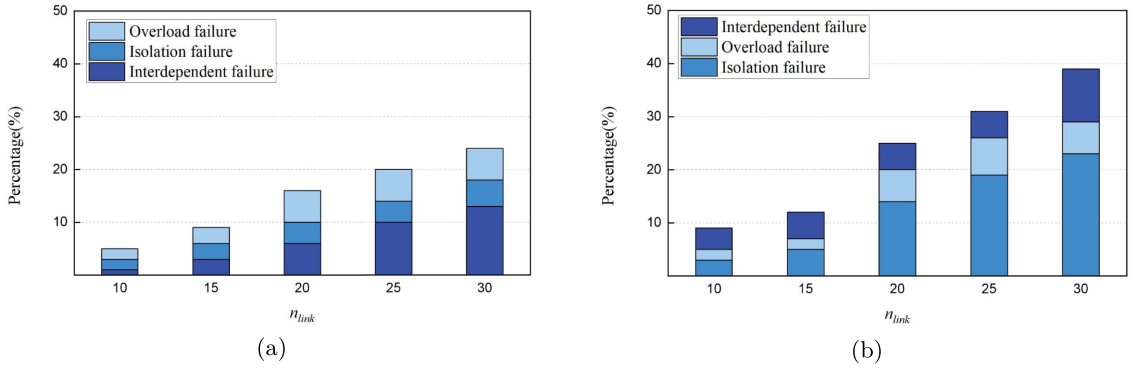


Figure 17. Proportion of nodes under two failure triggering mechanisms ($\alpha = \lambda = 0.6$, $\beta = 0.2$,$N^R = 300$). (a) SLFT, (b) SCLFT.

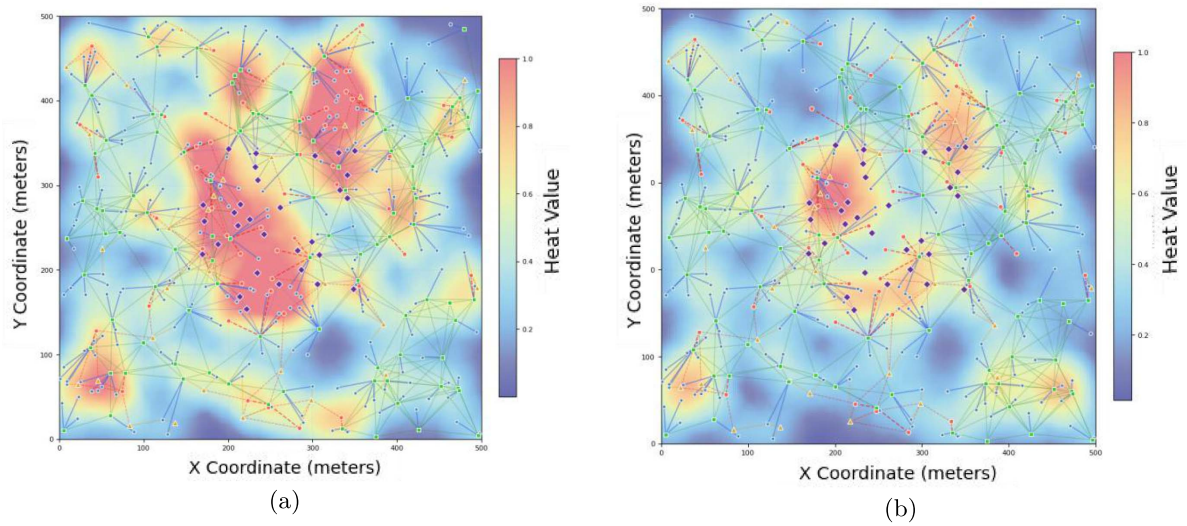


Figure 18. Heatmap of load distribution in PIoT under different failure triggering mechanisms at time t ($\alpha = \lambda = 0.6$, $\beta = 0.2$,$N^R = 300$). (a) SLFT, (b) SCLFT.

failure states. This is because when sensing-control links are compromised, terminal nodes that depend on these links may become inoperable due to losing control data support. This is also the primary reason for the lower data packet survival rate observed under SCLFT. It can be seen from figure 18 that SCLFT has a greater impact on network load than SLFT. The load distribution differences depicted highlight the critical role of sensing-control links. In the SCLFT scenario (figure 18(b)), the network exhibits significantly lower load density than in SLFT (figure 18(a)). This is primarily because SCLFT disrupts the transmission of control data, which directly governs the normal operation of port equipment. The resulting loss of

Table 4. Comparison of failure mechanisms under different priority settings.

Failure Mechanism	Non-priority $\rho(t)$	Priority $\rho(t)$	Non-priority $T_{\text{delay}}(t)$ (ms)	Priority $T_{\text{delay}}(t)$ (ms)
TFT	0.67	0.81	150	105
RFT	0.08	0.19	185	130
EFT	0.16	0.38	190	135
SFT	0.06	0.12	230	160
SLFT	0.12	0.23	175	120
SCLFT	0.07	0.17	210	145

control support triggers large-scale interdependent failures of terminal nodes. Consequently, data packets accumulate rapidly at upstream router nodes due to the inability to complete delivery, leading to severe network-wide congestion. In contrast, SLFT only hinders sensing data collection, which has a lower priority and a more localized impact on the overall network load.

Between the two link failure triggering mechanisms, targeting SCLFT exerts a more substantial impact on the network. Therefore, in the PIoT, prioritizing the protection of sensing-control links is critical to mitigating damage caused by deliberate attacks.

5.6. Effectiveness of the data priority mechanism

To further validate the necessity of the proposed data priority mechanism, this section compares the performance of our model against a non-priority baseline (FIFO-based processing) under typical multi-node failure scenarios. For all comparative simulations in this section, the key network parameters are fixed at $\alpha = 0.6$, $\lambda = 0.6$, $\beta = 0.2$, and $N_R = 300$ to provide a consistent baseline for evaluation. In the non-priority model, sensing data and control data are treated with equal processing weight. As summarized in table 4, the comparative results across multiple metrics—including the final survival rate $\rho(\infty)$, average end-to-end delay, and control packet deadline miss rate—demonstrate a significant performance gap.

As illustrated in the table, the absence of a priority mechanism leads to a rapid decline in the system survival rate, particularly under EFT and SFT scenarios. In the non-priority baseline, the high-volume sensing traffic competes for limited buffer resources with time-critical control packets. This undifferentiated competition results in a 'congestion collapse' where control packets experience prolonged queuing delays. Consequently, the average end-to-end delay for control streams escalates sharply, leading to a substantial increase in the deadline miss rate.

6. Conclusion and future work

In this paper, we construct a delay-sensitive cascading failure model tailored to the PIoT, leveraging the unique characteristics of the PIoT and characterizing node loads using dynamic data flows. We also account for node classification based on their distinct functions in real-world port environments, as well as the flow directions of sensing and control data between nodes. Additionally, we designed two caching mechanisms: the in-transit caching mechanism and the end-of-line caching mechanism. We conducted extensive experiments to investigate how key parameters affect the system's cascading failures, and examined the impacts of different failure triggering mechanisms on the PIoT via multi-node and link failure scenarios. The research findings can offer theoretical guidance for building reliable PIoT.

(1) Identified the critical thresholds for the node overload-tolerance coefficient α (0.6–0.8) and link overload-tolerance coefficient λ (0.4–0.8) within the PIoT. This study demonstrates that exceeding these saturation points yields diminishing returns for the data packet survival rate $\rho(t)$, thereby establishing the minimum resource requirements necessary to effectively mitigate cascading failures.
(2) Compared with attacking sensing links, attacking sensing-control links may cause much greater damage to the PIoT through cascading failures. Nodes subject to interdependent failures account for the largest proportion in the network, and most of them are terminal nodes damaged due to the loss of control data. Therefore, priority should be given to enhancing the robustness of such nodes, for example, by allocating higher capacity, deploying redundancy, or applying differentiated protection strategies.
(3) Compared with attacking nodes that only transmit sensing data, attacking nodes that generate or transmit control data may cause much greater damage to the PIoT through cascading failures. When attacking nodes that only transmit sensing data, failed nodes experiencing overload account for the largest proportion in the network, and most of these nodes are router nodes that become overloaded due to load redistribution.
(4) In terms of data packet survival performance, the in-transit cache mechanism consistently outperforms the end-of-line cache mechanism. This finding suggests that in-transit caching is more suitable for delay-sensitive PIoT applications. However, in scenarios where data integrity is more critical, the end-of-line caching mechanism may still be preferable. Therefore, caching strategies should be selected based on application requirements.
(5) A larger number of router nodes in a PIoT results in enhanced connectivity between nodes, which in turn provide greater backup and redundant resources across the network. Consequently, such systems generally exhibit better performance in mitigating cascading failures.

In this work, we employ the packet delivery ratio and average end-to-end delay to evaluate the reliability of the PIoT system. While these reliability metrics intuitively reflect the extent of damage caused by cascading failures to the network, it still has certain limitations. In practical PIoT environments, factors such as the system's topology, configuration, and the structural location of faulty IoT devices can significantly impact overall functionality and reliability. Therefore, in future work, we plan to extend the cascading reliability metrics from a quality of service perspective, for example, by incorporating communication efficiency and network coverage. Additionally, we will focus on strategies to enhance the cascading reliability of PIoT systems. We believe two feasible approaches hold promise: (1) Optimizing Cascading Parameters: This study concludes that cascading parameters (such as α and β) significantly influence network reliability. This finding provides a theoretical basis for optimizing these parameters to improve cascading reliability. (2) Implementing Network Optimization Strategies: Many studies [34–37] have demonstrated that network optimization strategies, such as reinforcement learning [38–41], can effectively help networks resist cascading failures.

References

[1] Yang Y, Zhong M, Yao H, Yu F, Fu X and Postolache O 2018 Internet of things for smart ports: technologies and challenges *IEEE Instrum. Meas. Mag.* **21** 34–41

[2] Li K X, Li M, Zhu Y, Yuen K F, Tong H and Zhou H Q 2023 Smart port: a bibliometric review and future research directions *Transp. Res. E* **174** 103098

[3] Bouhlal R, Aitabdelouahid R and Marzak A 2022 The internet of things for smart ports *Proc. Comput. Sci.* **203** 819–24

[4] Wang J, Pambudi S, Wang W and Song M 2019 Resilience of IoT systems against edge-induced cascade-of-failures: a networking perspective *IEEE Internet Things J.* **6** 6952–63

[5] Rossi M, Centenaro M, Ba A, Eleuch S, Erseghe T and Zorzi M 2020 Distributed learning algorithms for optimal data routing in IoT networks *IEEE Trans. Signal Inf. Process. Netw.* **6** 179–95

[6] Fu X and Li W F 2020 Cascading failures of wireless sensor networks *Proc. 11th IEEE Int. Conf. on Networking, Sensing and Control* pp 1–6

[7] Fu X, Li W, Yang Y and Postolache O 2021 Cascading failures analysis of wireless sensor networks with varying routing schemes *IEEE Sens. J.* **21** 10193–203

[8] Yang L 2025 An evidence-based likelihood approach for the reliability of a complex system with overlapped failure data *Comput. Ind. Eng.* **201** 110893

[9] Boyes H, Hallaq B, Cunningham J and Watson T 2018 The industrial internet of things (IIoT): an analysis framework *Comput. Ind.* **101** 1–12

[10] Sisinni E, Saifullah A, Han S, Jennehag U and Gidlund M 2018 Industrial internet of things: challenges, opportunities and directions *IEEE Trans. Ind. Inform.* **14** 4724–34

[11] Rajesh P, Shajin F H and Kannayeram G 2022 A novel intelligent technique for energy management in smart home using internet of things *Appl. Soft Comput.* **128** 109442

[12] Zielonka A, Sikora A, Woźniak M, Wei W, Ke Q and Bai Z 2020 Intelligent internet of things system for smart home optimal convection *IEEE Trans. Ind. Inform.* **17** 4308–17

[13] Fortino G, Fotia L, Messina F, Rosaci D and Sarnè G M L 2023 A social edge-based IoT framework using reputation-based clustering for enhancing competitiveness *IEEE Trans. Comput. Social Syst.* **10** 2051–63

[14] Rupanetti D and Kaabouch N 2024 Combining edge computing-assisted internet of things security with artificial intelligence: applications, challenges and opportunities *Appl. Sci.* **14** 7104

[15] Zhong J, Zhang F, Yang S and Li D 2019 Restoration of interdependent network against cascading overload failure *Physica A* **514** 884–91

[16] Xing L 2020 Reliability in internet of things: current status and future perspectives *IEEE Internet Things J.* **7** 6704–21

[17] Fu X, Yao H and Yang Y 2020 Modeling and optimizing the cascading robustness of multisink wireless sensor networks *IEEE Trans. Reliability* **70** 121–33

[18] Ye Z *et al* 2016 Fault-tolerant scheme for cascading failure of scale-free wireless sensor networks *Proc. IEEE Int. Conf. Information and Automation (ICIA)* pp 1–6

[19] Hu X, Li W and Fu X 2015 Analysis of cascading failure based on wireless sensor networks *Proc. IEEE Int. Conf. Systems, Man and Cybernetics* pp 1279–84

[20] Mo Y, Xing L, Guo W, Cai S, Zhang Z and Jiang J 2020 Reliability analysis of IoT networks with community structures *IEEE Trans. Netw. Sci. Eng.* **7** 304–15

[21] Fu X, Yang Y and Postolache O 2019 Invulnerability of clustering wireless sensor networks against cascading failures *IEEE Syst. J.* **13** 1431–42

[22] Yin R, Zhang K, Ma X, Wang Y and Li L 2023 Analysis of cascading failures caused by mobile overload attacks in scale-free networks *Physica A* **615** 128598

[23] Biabani M, Yazdani N and Fotouhi H 2021 REFIT: robustness enhancement against cascading failure in IoT networks *IEEE Access* **9** 40768–82

[24] Xing L, Zhao G and Xiang Y 2020 *Phased-Mission Modelling of Physical Layer Reliability for Smart Homes* (*Stochastic Models in Reliability Engineering*) (CRC Press) pp 317–30

[25] Zhao G and Xing L 2020 Reliability analysis of IoT systems with competitions from cascading probabilistic function dependence *Reliab. Eng. Syst. Saf.* **198** 106812

[26] Fu X and Yang Y 2021 Analysis on invulnerability of wireless sensor networks based on cellular automata *Reliab. Eng. Syst. Saf.* **212** 107616

[27] Wang S, Gu X, Chen J, Chen C and Huang X 2023 Robustness improvement strategy of cyber-physical systems with weak interdependency *Reliab. Eng. Syst. Saf.* **229** 108837

[28] Fu X, Wang Y, Yang Y and Postolache O 2022 Analysis on cascading reliability of edge-assisted internet of things *Reliab. Eng. Syst. Saf.* **223** 108463

[29] Fu X, Li Q and Li W 2023 Modeling and analysis of industrial IoT reliability to cascade failures: an information-service coupling perspective *Reliab. Eng. Syst. Saf.* **239** 109517

[30] Xue T, Zhang Y, Wang Y, Wang W, Li S and Zhang H 2025 Edge computing for IoT: novel insights from a comparative analysis of access control models *Comput. Netw.* **270** 111468

[31] Fu X, Li W and Yang Y 2021 Exploring the impact of node mobility on cascading failures in spatial networks *Inf. Sci.* **576** 140–56

[32] Fu X, Yao H and Yang Y 2019 Modeling and analyzing cascading dynamics of the clustered wireless sensor network *Reliab. Eng. Syst. Saf.* **186** 1–10

[33] Zhao G and Xing L 2023 Reliability analysis of body sensor networks with correlated isolation groups *Reliab. Eng. Syst. Saf.* **236** 109305

[34] Wei F, Ma X, Chen Y, Qiu Q and Yang L 2024 A reliability evaluation and optimization policy for two-phase industrial plants considering time-varying stochastic environmental effect *Proc. 6th Int. Conf. System Reliability and Safety Engineering* (*Beijing, China*) pp 1–6

[35] Zhou Y, Hu L, Peng R and Li J 2025 Reliability analysis and optimization of multi-product manufacturing systems based on two types of maintenance mechanisms *Reliab. Eng. Syst. Saf.* **256** 110676
[36] Yang L, Zhang X, Zhu F, Wang Z and Zhang X 2026 Semi-supervised cross-domain fault diagnosis via contrastive pre-training and annotation-efficient alignment strategy *J. Ind. Inform. Integr.* **50** 101076
[37] Yang L, Wang J and Xie M 2025 Graph-based reliability evaluation of a reconfigurable multi-stage system using sequential unconnected path sets *Reliab. Eng. Syst. Saf.* **261** 111093
[38] Chen W, Qiu X, Cai T, Dai H-N, Zheng Z and Zhang Y 2021 Deep reinforcement learning for internet of things: a comprehensive survey *IEEE Commun. Surv. Tutorials* **23** 1659–92
[39] De Oliveira H, Kaneko M and Boukhatem L 2024 Federated multiagent deep reinforcement learning for intelligent IoT wireless communications: overview and challenges *IEEE Veh. Technol. Mag.* **19** 67–77
[40] Fu X, Deng C and Guerrieri A 2025 Low-AoI data collection in integrated UAV-UGV-assisted IoT systems based on deep reinforcement learning *Comput. Netw.* **259** 111044
[41] Zhu X, Zhang T, Zhang J, Zhao B, Zhang S and Wu C 2023 Deep reinforcement learning-based edge computing offloading algorithm for software-defined IoT *Comput. Netw.* **235** 110006



Xiuwen Fu received his PhD in Mechatronics in 2016 from Wuhan University of Technology, China. He was a visiting researcher at the University of Calabria. He is currently an associate professor at Shanghai Maritime University, China. He has authored more than 80 papers in international journals and conferences. His research interests include the Internet of Things and unmanned systems. He is a member of IEEE.



Xindi Wang received a BS degree in detection, guidance and control technology from Shenyang Aerospace University, Shenyang, China, in 2023. She is currently pursuing an MS degree in control science and engineering with the Shanghai Maritime University, Shanghai, China. Her current research interests include reliability modeling and optimization.



Liudong Xing received her PhD degree in Electrical Engineering from the University of Virginia, Charlottesville, USA in 2002. She is a Commonwealth Professor and College of Engineering Director of Research at the University of Massachusetts (UMass) Dartmouth, USA. Her current research interests include reliability and resilience modeling, analysis and optimization of complex systems and networks. She has received multiple awards for teaching and research and has co-authored eleven award-winning papers. Prof. Xing was an IEEE ComSoc Distinguished Lecturer (2024–2025). In 2025, she was recognized as a Star in Networking and Communications by N2Women and received the IEEE ComSoc IoT-AHSN Technical Achievement and Recognition Award. She serves on the editorial boards of several leading journals. She is a Fellow of the International Society of Engineering Asset Management.