RIDT: Reliability-State-Aware Performance Analytics for Open RAN Systems

- Integrating Reliability and Queueing Theory to Reveal Hidden Performance Degradation -

Kazuhira Okumoto
Sakura Software Solutions, LLC
Naperville, Illinois

Abstract

Open Radio Access Network (Open RAN) systems combine redundant hardware, virtualized network functions, and shared cloud infrastructure, creating operational conditions in which a system may remain structurally available while experiencing significant performance degradation. Conventional reliability analysis determines whether sufficient components remain operational, while conventional performance analysis generally evaluates traffic relative to an assumed available capacity. These approaches may fail to capture the performance consequences of a component failure that reduces capacity without causing immediate system unavailability.

This paper presents the Reliability-Integrated Digital Twin (RIDT), a framework that integrates reliability theory and queueing theory through reliability-state-dependent processing capacity. RIDT models component availability, k-of-n redundancy, joint operating-state probabilities, state-dependent capacity, utilization, queueing delay, and overload probability within a unified analytical framework. For each reliability state, the number of operating resources determines available processing capacity, which in turn determines utilization and M/M/1 queueing delay under the offered traffic.

An Open RAN reference analysis demonstrates the distinction between conventional utilization monitoring and reliability-state-aware delay analysis. Although the fully operational system may operate well below its nominal capacity limit, a software-function failure can transition the system to a degraded but structurally operational state with substantially reduced processing capacity. As traffic approaches the capacity boundary of such a degraded state, queueing delay increases nonlinearly. When combined with reliability-state probabilities, these effects can appear as localized delay signatures that are not readily apparent from the nominal utilization view.

The results demonstrate a quantitative mechanism by which a **reliability-state transition can produce an observable performance consequence before structural system unavailability occurs**. Such delay signatures do not, by themselves, prove a component failure, but they may indicate latent or persistent degradation that warrants further investigation. If recovery is

3S White Paper Series, WP-2026-003 | Version 1.0 | August 2026.

delayed, the degraded state can persist, and its performance impact may increase as traffic approaches the reduced capacity boundary, potentially progressing toward overload.

RIDT therefore provides an analytical framework for connecting component failure and recovery behavior with capacity, traffic, utilization, and delay. The framework can be calibrated and validated using System Integration Testing (SIT) and operational data, providing a foundation for predictive reliability–performance assessment and operational decision support in Open RAN systems.

Keywords

Open RAN; Reliability-Integrated Digital Twin (RIDT); Reliability; Queueing Theory; Reliability-State-Aware Performance; Availability; k-of-n Redundancy; Capacity; Utilization; Queueing Delay; Performance Degradation; Digital Twin

I. Introduction

Open Radio Access Network (Open RAN) architectures are transforming mobile-network deployment by disaggregating traditionally integrated radio-access functions into interoperable hardware, software, and cloud-native components. This architecture enables greater vendor flexibility, virtualization, scalability, and automation, but it also creates new challenges for evaluating end-to-end system reliability and performance [1]–[3].

This challenge is not unique to Open RAN. Modern complex software-driven systems increasingly combine distributed software functions, redundant computing resources, shared infrastructure, and dynamic workloads. In such systems, a software failure may not cause immediate system unavailability. Instead, redundancy or failover mechanisms may allow the system to remain operational while reducing the processing capacity available to serve the workload. The resulting reliability-state transition can therefore manifest as performance degradation before it appears as a conventional system failure [9].

An Open RAN system may include radio-frequency hardware, virtualized Distributed Units (vDU), virtualized Centralized Units (vCU), Kubernetes infrastructure, and other shared computing resources. These elements differ in failure behavior, redundancy structure, recovery characteristics, and processing capacity. As a result, end-to-end system behavior depends not only on whether individual components are operational, but also on **how much processing capacity remains available after failures or recovery events occur.**

Traditional reliability analysis and performance analysis generally address different questions.

Reliability analysis evaluates quantities such as component failure rate, availability, reliability, and k-of-n redundancy. It can determine whether sufficient components remain operational for a system to satisfy its structural requirements.

Performance analysis evaluates quantities such as traffic, capacity, utilization, throughput, and delay. It can determine whether available resources can support the offered workload.

For cloud-native redundant systems, however, these two perspectives are fundamentally connected.

Consider a redundant component pool in which one unit fails while the remaining units continue to satisfy the required k-of-n condition. From a structural reliability perspective, the system remains operational. From a performance perspective, however, the failure may have substantially reduced the available processing capacity.

Thus,

$$\boxed{\text{Structurally Operational} \not\Rightarrow \text{Nominal Performance}} \tag{1.1}$$

A system can therefore be **reliable enough to remain operational but insufficiently provisioned to maintain its expected performance under the current traffic load.**

This problem is particularly important in Open RAN because virtualized network functions operate on shared cloud infrastructure. The effective capacity of a vDU or vCU can depend not only on its own redundancy configuration but also on Kubernetes and other infrastructure constraints. The RIDT framework therefore treats Kubernetes as a shared capacity ceiling rather than merely another structural reliability element.

A second difficulty arises from conventional utilization monitoring.

For a fixed capacity C, utilization is

$$U = \frac{\lambda}{C}, \tag{1.2}$$

where λ represents offered traffic.

Utilization provides an intuitive measure of how close the system is to its assumed capacity. However, assuming capacity remains fixed can be misleading when a component failure moves the system into a degraded reliability state.

If the all-up system has capacity C_0, but a failure produces a degraded state s_d with

$$C(s_d) < C_0, \tag{1.3}$$

the same traffic level may represent moderate utilization in the nominal configuration but near-saturation behavior in the degraded configuration.

The operational question is therefore not simply:

How much of the nominal system capacity is currently being used?

It is also:

How would the same traffic affect performance if the system were operating in a degraded reliability state?

This paper addresses that question through the **Reliability-Integrated Digital Twin (RIDT)** framework. Digital-twin concepts are increasingly being investigated for RAN modeling, testing, monitoring, optimization, and operational decision support [4]–[6]. RIDT extends this direction by explicitly integrating reliability-state probabilities with state-dependent capacity and queueing performance.

RIDT integrates reliability-state probabilities, state-dependent processing capacity, utilization, and queueing performance within a common analytical model:

Reliability State → State-Dependent Capacity → Utilization → Queueing Delay → Overload Risk (1.4)

The key distinction is that capacity is not treated as a fixed system property. It is a function of the current reliability state.

This formulation creates a quantitative bridge between **reliability theory and queueing theory**. Reliability theory determines the probability that the system occupies a particular normal or degraded configuration. Queueing theory determines the performance consequence of the capacity available in that configuration.

The Open RAN reference analysis presented in this paper demonstrates an important implication of this integration. Degraded vCU and vDU configurations can approach their own capacity boundaries while nominal system utilization remains well below the all-up capacity limit. The resulting nonlinear queueing behavior produces localized delay signatures that are not readily apparent from a nominal-utilization view.

These signatures should not be interpreted as proof of a particular component failure. Rather, they show that reliability-state-aware performance analysis can reveal **potential latent or persistent degradation** that conventional utilization monitoring alone may not expose.

The principal contributions of this paper are therefore:

1. a reliability-state model combining component availability, k-of-n redundancy, and joint operating-state probabilities;
2. a state-dependent capacity model connecting component failures to changes in effective processing capacity;
3. an integrated queueing model that evaluates component and end-to-end delay for each operational reliability state;
4. a probability-weighted Delay vs. Traffic methodology for identifying reliability-induced performance degradation;
5. a comparison of Utilization vs. Traffic and Delay vs. Traffic showing the additional information produced by the integrated analysis; and
6. a practical validation framework using SIT and operational failure, recovery, traffic, capacity, and performance data.

The objective of RIDT is therefore broader than predicting whether a system will remain structurally available. It is to evaluate **how reliably the system can continue to perform as reliability state and traffic conditions change.**

II. Open RAN Reliability–Performance Architecture

A. System Architecture

The reference RIDT architecture represents an Open RAN system as an integrated set of hardware, virtualized software functions, and shared cloud infrastructure [1], [3].

The principal elements considered in the current model are:

- radio-frequency hardware;
- virtualized Distributed Unit (vDU);
- virtualized Centralized Unit (vCU); and
- Kubernetes-based shared infrastructure.

The vDU and vCU are modeled as redundant processing pools. Each pool may contain multiple units, of which a minimum number must remain operational to satisfy its structural reliability requirement.

The system can therefore be represented conceptually as

RF Hardware → vDU Pool → vCU Pool → End-to-End Service (2.1)

with Kubernetes providing the shared infrastructure required by the virtualized functions.

Figure 1 presents the functional Open RAN reference architecture used in this study. It is not intended to reproduce a specific release of the O-RAN ALLIANCE architecture [1]–[3]. Rather, it represents the principal radio, distributed/centralized processing, cloud infrastructure, and management functions relevant to the RIDT reliability–performance analysis. This abstraction applies to the continued evolution of Open RAN from current 5G deployments toward emerging 6G architectures.

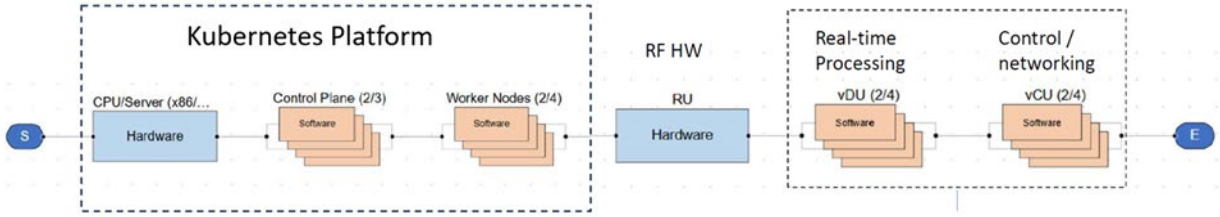

Figure 1. Reference Open RAN Architecture for RIDT Analysis

In the reference analysis developed in this paper, the principal degraded states result from failures of the virtualized software functions represented by the vDU and vCU processing pools. These software-function failures reduce the number of operating processing units and therefore reduce effective capacity, while redundancy allows the Open RAN system to remain structurally operational.

This architecture extends a conventional reliability block representation by associating each operational state with both a probability and a processing capacity.

B. Structural Reliability Layer

The first RIDT layer represents whether sufficient resources remain operational.

For a component pool containing *n* units and requiring at least *k* operating units [7], the pool remains structurally available whenever

$$N_{operating} \geq k. \tag{2.2}$$

A component failure therefore does not necessarily make the system unavailable.

For example,

$$n \rightarrow n-1 \tag{2.3}$$

may still satisfy Equation (2.2).

The resulting state is classified as **degraded but operational**.

This distinction is fundamental to RIDT because degraded states may satisfy structural redundancy requirements while providing substantially less processing capacity.

C. Reliability-State Representation

RIDT explicitly enumerates the relevant combinations of operating units.

A reliability state can be represented as

s = (*n*vDU,*n*vCU,*n*K8s), (2.4)

with an associated probability

P(*s*). (2.5)

The complete system is therefore represented not by a single nominal configuration, but by a collection of possible operating configurations:

$$s = (n_{vDU}, n_{vCU}, n_{K8s}), \tag{2.6}$$

Each state is classified as one of three operating conditions:

$$s \in \{\text{Stable}, \text{Overload}, \text{Unavailable}\}. \tag{2.7}$$

This state representation allows RIDT to preserve information that is lost when reliability is reduced to a single overall availability value.

D. Capacity Layer

For every operational reliability state, RIDT determines the processing capacity available from each major component.

Thus,

$$s \to \{C_{vDU}(s), C_{vCU}(s), C_{K8s}(s)\}. \tag{2.8}$$

The most constrained required component determines the effective end-to-end capacity:

$$C_{E2E}(s) = \min[C_{vDU}(s), C_{vCU}(s), C_{K8s}(s)]. \tag{2.9}$$

This means that a reliability-state transition can directly alter system capacity:

$$\boxed{s_0 \to s_d \quad \Rightarrow \quad C(s_0) \to C(s_d).} \tag{2.10}$$

For a degraded state,

$$C(s_d) < C(s_0) \tag{2.11}$$

when the failed resource contributes to effective processing capacity.

This is the principal bridge between the structural reliability layer and the performance layer.

E. Performance Layer

Given traffic λ, RIDT evaluates performance separately for each reliability state.

The first measure is utilization:

$$U(s) = \frac{\lambda}{C(s)}. \tag{2.12}$$

Utilization indicates proximity to the state-dependent capacity boundary.

RIDT then evaluates queueing delay. Under the current M/M/1 approximation,

$$W_q(s) = \frac{\lambda}{C(s)[C(s) - \lambda]}, \qquad \lambda < C(s). \tag{2.13}$$

The nonlinear behavior of Equation (2.13) is particularly important:

$$W_q(s) \to \infty \qquad \text{as} \qquad \lambda \to C(s). \tag{2.14}$$

Thus, a relatively small change in capacity caused by a reliability-state transition can produce a disproportionately large change in delay when traffic is sufficiently close to the degraded capacity boundary.

F. Reliability–Performance Coupling

The distinctive feature of RIDT is that the reliability and performance layers are not evaluated independently.

For each state,

$$\boxed{s \to P(s) \to C(s) \to U(\lambda, s) \to W_q(\lambda, s)} \tag{2.15}$$

and the state-specific performance results are subsequently combined using the corresponding reliability-state probabilities.

This produces a reliability-state-aware performance model:

$$\boxed{\text{Reliability Theory} + \text{Capacity Model} + \text{Queueing Theory}} \tag{2.16}$$

rather than separate reliability and performance calculations.

A component failure can therefore propagate analytically through the system as

Component Failure → Fewer Operating Units → Degraded Reliability State → Reduced Capacity → Higher Utilization → Nonlinear Delay Response. (2.17)

This propagation mechanism is the foundation of the degraded-state delay signatures analyzed later in the paper.

Figure 2 summarizes the RIDT analytical framework linking reliability-state transitions to state-dependent system performance.

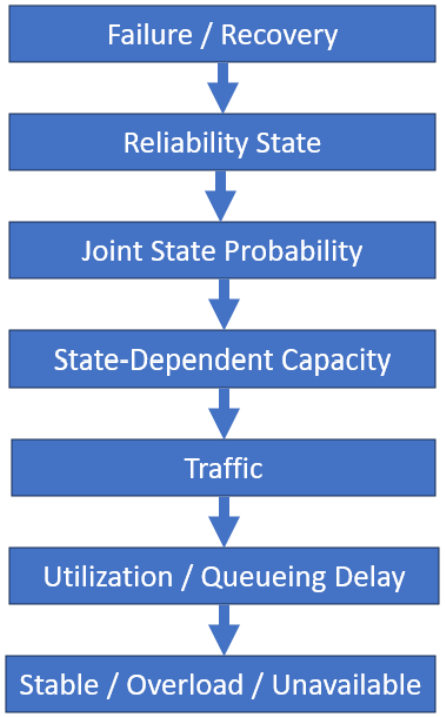

Figure 2. RIDT Reliability–Performance Integration Framework

G. Normal and Degraded Operating Envelopes

The architecture also introduces the concept of multiple performance envelopes.

A conventional capacity analysis typically evaluates the all-up system:

$$C_{normal}. \tag{2.18}$$

RIDT additionally evaluates

$$C_{degraded,1}, C_{degraded,2}, \ldots, C_{degraded,r}. \tag{2.19}$$

Therefore, the system does not have only one relevant capacity threshold. It has a set of reliability-state-dependent thresholds:

$$\mathcal{C} = \{C(s_1), C(s_2), \ldots, C(s_m)\}. \tag{2.20}$$

This distinction becomes increasingly important as traffic grows.

A traffic level may satisfy

$$\lambda < C_{normal} \tag{2.21}$$

while simultaneously satisfying

$$\lambda \geq C_{degraded}. \tag{2.22}$$

The system is therefore adequately provisioned **if it is fully operational** but unable to sustain the same workload in a particular degraded configuration.

This condition is central to reliability-aware capacity planning.

H. Role of Recovery

RIDT also incorporates recovery because a degraded state need not persist indefinitely.

A component failure creates a transition

$$s_0 \rightarrow s_d, \tag{2.23}$$

while successful recovery produces

$$s_d \rightarrow s_0. \tag{2.24}$$

Failure and recovery parameters determine the probability of occupying each state.

Consequently, recovery affects not only reliability but also the expected performance impact of degraded operation.

A short-degraded episode may produce only a transient performance effect. Delayed recovery increases the persistence of the degraded configuration and therefore the operational significance of its reduced capacity.

The complete conceptual loop is therefore

$$\text{Normal State} \rightarrow \text{Failure} \rightarrow \text{Degraded State} \rightarrow \text{Performance Impact} \rightarrow \begin{cases} \text{Recovery} \rightarrow \text{Normal State} \\ \text{Persistence} \rightarrow \text{Overload Risk} \end{cases} \tag{2.25}$$

I. RIDT Analytical Outputs

The architecture produces a coordinated set of reliability and performance outputs rather than a single system metric.

These include:

- component availability;
- k-of-n reliability/availability;
- individual and joint state probabilities;
- normal and degraded-state capacities;
- component and system utilization;
- component and system queueing delay;
- stable-state probability;
- overload probability;
- unavailable probability; and
- state-level diagnostic information.

The relationship among these outputs is important. Availability indicates the likelihood of the structural state; capacity describes what that state can process; utilization describes how heavily that capacity is loaded; and delay describes the nonlinear performance consequence.

Together, these measures provide the analytical foundation for the mathematical formulation in Section III.

III. Mathematical Model

A. Model Framework

The Reliability-Integrated Digital Twin (RIDT) framework integrates structural reliability, state-dependent processing capacity, and queueing performance within a common analytical model. The system consists of Kubernetes infrastructure, radio-frequency hardware, and virtualized software functions including vDU and vCU. vDU and vCU operate on Kubernetes and are therefore constrained by both their intrinsic processing capabilities and the available infrastructure capacity.

The central concept of RIDT is that system performance depends on the current reliability state. A component failure may reduce the number of operating units without immediately making the system structurally unavailable. Such a degraded state, however, has lower effective capacity and consequently different utilization and queueing-delay characteristics.

Accordingly, RIDT establishes the following analytical sequence:

$$\boxed{\text{Reliability State} \rightarrow \text{State-Dependent Capacity} \rightarrow \text{Utilization} \rightarrow \text{Queueing Delay} \rightarrow \text{Overload Risk}}$$

This formulation provides the mathematical link between reliability theory and queueing theory.

B. Reliability-State Model

For a repairable component with failure rate λ_f and repair rate μ_r, steady-state availability is

$$A = \frac{\mu_r}{\lambda_f + \mu_r} \tag{3.1}$$

or equivalently,

$$A = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}} \tag{3.2}$$

when constant failure and repair rates are assumed [7], [9].

For a pool containing *n* statistically independent identical units, the probability that exactly *j* units are operational is

$$P(N = j) = \binom{n}{j} A^j (1 - A)^{n-j} \tag{3.3}$$

and the probability that at least *k* units are operational is

$$A_{k|n} = \sum_{j=k}^{n} \binom{n}{j} A^j (1 - A)^{n-j}. \tag{3.4}$$

This k-of-n formulation [7] determines whether a component pool is structurally operational, consistent with the structural reliability model defined in Section II.

For the current Open RAN reference model, the reliability state can be represented as

$$s = (n_{vDU}, n_{vCU}, n_{K8s}), \tag{3.5}$$

where each term denotes the number of operating units in the corresponding subsystem.

Under the current independence assumption, the joint state probability is

$$P(s) = P(N_{vDU} = n_{vDU}) P(N_{vCU} = n_{vCU}) P(N_{K8s} = n_{K8s}). \tag{3.6}$$

A state may therefore be **fully operational**, **degraded but operational**, or **unavailable**, depending on whether the required k-of-n conditions are satisfied.

C. State-Dependent Capacity Model

Let the nominal processing capacity of one operating unit of component *i* be μ_i, and let oi denote its fractional processing overhead. For n_i operating units, the available component capacity is

$$C_i(s) = n_i \mu_i (1 - o_i). \tag{3.7}$$

For virtualized functions operating on shared Kubernetes infrastructure, the effective capacity is additionally constrained by the infrastructure capacity. Thus,

$$C_{i,\text{eff}}(s) = \min\left[C_i(s), C_{K8s}(s)\right]. \tag{3.8}$$

The most constrained component determines the end-to-end capacity of reliability state *s*:

$$C_{E2E}(s) = \min\left[C_{vDU,\text{eff}}(s), C_{vCU,\text{eff}}(s), C_{K8s}(s)\right]. \tag{3.9}$$

This extends the existing effective-capacity formulation, in which Kubernetes acts as a shared capacity ceiling for higher-layer functions.

Figure 3 illustrates how a component failure can preserve structural operation while reducing the system's effective end-to-end capacity.

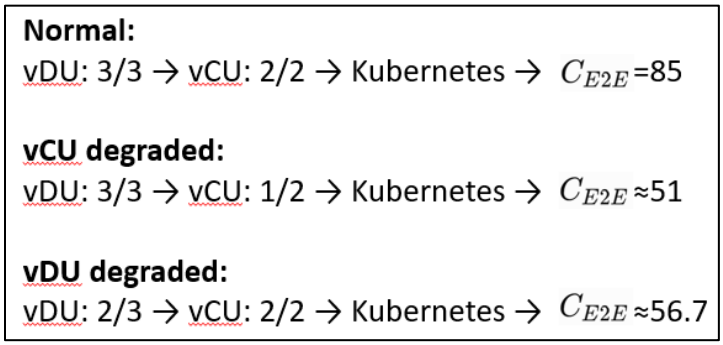

Figure 3. Reliability-State-Dependent Capacity in the RIDT Reference Model

Equation (3.9) provides the critical connection between reliability and performance. A component failure changes n_i, which changes $C_i(s)$, even when the remaining units continue to satisfy the structural k-of-n requirement.

Thus,

$$\boxed{\text{Failure} \rightarrow \text{Degraded State} \rightarrow \text{Reduced Effective Capacity}}$$

without necessarily causing immediate system unavailability.

D. Utilization Model

Let λ denote the offered traffic or processing demand. For component *i* in reliability state *s*, utilization [8] is

$$U_i(s) = \frac{\lambda}{C_{i,\text{eff}}(s)}. \tag{3.10}$$

The most heavily utilized required component determines system utilization:

$$U_{System}(s) = \max_i U_i(s). \tag{3.11}$$

Equivalently,

$$U_{System}(s) = \frac{\lambda}{C_{E2E}(s)}. \tag{3.12}$$

In the current RIDT model, λ is treated as a normalized processing-demand rate that can subsequently be calibrated to physical measures such as Gbps, session counts, or other workload units.

The utilization model provides a direct measure of proximity to capacity. However, utilization changes approximately linearly with traffic for a fixed capacity and therefore does not fully characterize the nonlinear performance consequences of approaching a capacity boundary.

E. Queueing-Delay Model

To capture this nonlinear behavior, RIDT introduces queueing delay as an additional performance measure.

For the current prototype, each processing component is approximated by an M/M/1 queue [7], [8]. For component *i* in reliability state *s*,

$$W_{q,i}(s) = \frac{\lambda}{C_{i,\text{eff}}(s)\,[C_{i,\text{eff}}(s) - \lambda]}, \qquad \lambda < C_{i,\text{eff}}(s). \tag{3.13}$$

When traffic and capacity are expressed per minute, W_q is expressed in minutes.

For a stable state, the end-to-end system queueing delay is approximated as the sum of the component delays:

$$W_{q,System}(s) = W_{q,vDU}(s) + W_{q,vCU}(s) + W_{q,K8s}(s). \tag{3.14}$$

Equation (3.14) reveals an important difference between utilization and delay. While utilization increases linearly with traffic for fixed capacity,

$$U \propto \lambda, \tag{3.15}$$

queueing delay increases nonlinearly and rises rapidly as traffic approaches available capacity:

$$W_q \to \infty \qquad \text{as} \qquad \lambda \to C^{-}. \tag{3.16}$$

Consequently, queueing delay can provide an early indication of performance stress that is much less apparent from utilization alone.

F. Reliability-State-Weighted Delay

Because each operational reliability state has its own probability and capacity, RIDT combines the reliability-state model with the queueing model.

Let $S(\lambda)$ denote the set of operational reliability states that remain below their respective capacity limits at traffic level λ:

$$S(\lambda) = \{s : s \text{ operational}, \lambda < C_{E2E}(s)\}. \tag{3.17}$$

The conditional expected system delay among stable states is

$$E[W_q \mid Stable] = \frac{\sum_{s \in S(\lambda)} P(s) W_{q,System}(s)}{\sum_{s \in S(\lambda)} P(s)}. \tag{3.18}$$

Similarly, component-specific expected delays are

$$E[W_{q,i} \mid Stable] = \frac{\sum_{s \in S(\lambda)} P(s) W_{q,i}(s)}{\sum_{s \in S(\lambda)} P(s)}. \tag{3.19}$$

This formulation is the central reliability–performance integration in RIDT. Reliability theory determines the probability of each operating configuration, while queueing theory determines the performance consequence of that configuration.

Thus,

$$\boxed{P(\text{Reliability State}) \times W_q(\text{State-Dependent Capacity})} \tag{3.20}$$

provides a quantitative connection between component failures and system-level performance behavior.

G. Overload Probability

For any operational state *s*, overload occurs when offered traffic equals or exceeds its state-dependent end-to-end capacity:

$$\lambda \geq C_{E2E}(s). \tag{3.21}$$

Because the M/M/1 queue has no finite steady-state delay under this condition, RIDT does not calculate W_q for an overloaded state. Instead, its probability contributes to the overload probability:

$$P_{OL}(\lambda) = \sum_{s} P(s) I\left[s \text{ operational} \wedge \lambda \geq C_{E2E}(s)\right], \tag{3.22}$$

where $I[\cdot]$ is an indicator function.

System unavailability is treated separately:

$$P_{UNAV} = \sum_{s} P(s) I[s \text{ structurally unavailable}]. \tag{3.23}$$

Therefore, the reliability-state probability mass is separated into three operating conditions:

$$P_{Stable} + P_{OL} + P_{UNAV} = 1. \tag{3.24}$$

This distinction matters because an overloaded system may still meet its structural redundancy requirements even if it cannot sustain the offered traffic.

H. Reliability-Induced Delay Signatures

The integrated formulation produces a behavior not evident from conventional utilization analysis alone.

Suppose a component failure moves the system from a fully operational state s_0 to a degraded but structurally operational state s_d. Then

$$C_{E2E}(s_d) < C_{E2E}(s_0). \tag{3.25}$$

As traffic approaches the degraded-state capacity,

$$\lambda \to C_{E2E}(s_d). \tag{3.26}$$

the delay associated with that degraded state rises rapidly. Its probability-weighted contribution can therefore appear as a localized increase, or **delay spike**, in the aggregate Delay vs. Traffic curve.

Once

$$\lambda \geq C_{E2E}(s_d), \tag{3.27}$$

the degraded state becomes overloaded, and its probability moves from the stable-state population to the overload population. Consequently, the finite-delay curve may fall after the spike because the overloaded degraded state is no longer included in Equation (3.19).

For the current reference configuration, the analysis identifies degraded-state capacity boundaries of approximately

$$C_{vCU,degraded} \approx 51 \tag{3.28}$$

and

$$C_{vDU,degraded} \approx 56.7.$$

These values are specific to the reference input parameters and are not universal thresholds.

The important result is the mechanism:

Component Failure → Reduced State Capacity → Nonlinear Delay Increase → Delay Signature → Recovery or Overload (3.29)

A transient spike does not by itself establish the presence of a silent failure. However, it can indicate reliability-induced performance degradation. If recovery is delayed or the degraded state persists, its operational impact may become increasingly significant.

IV. Reliability-State-Aware Delay Analysis

A. Reference Open RAN Configuration

The mathematical framework in Section III was applied to the reference Open RAN configuration consisting of vDU, vCU, and Kubernetes processing resources. The analysis evaluates system behavior as offered traffic increases while explicitly accounting for the probabilities of fully operational and degraded reliability states.

For the reference input parameters, the all-up effective capacities are approximately:

$$C_{vDU} = 85, \qquad C_{vCU} = 102, \qquad C_{K8s} = 360 \tag{4.1}$$

in the normalized traffic units used by the model. Consequently, the vDU is the nominal end-to-end bottleneck, giving

$$C_{E2E} = 85. \tag{4.2}$$

The reliability-state analysis also produces degraded configurations resulting from vCU and vDU software-function failures. Because redundancy preserves structural operation, these failures do

not immediately make the system unavailable; instead, they reduce the number of operating processing units and thereby lower the effective system capacity. Two degraded-state capacity boundaries are particularly important in the present case:

$$C_{vCU,degraded} \approx 51 \tag{4.3}$$

and

$$C_{vDU,degraded} \approx 56.7. \tag{4.4}$$

These degraded configurations remain structurally operational according to their redundancy requirements but have substantially less traffic-processing capacity than the all-up configuration.

The objective of the following analysis is to determine whether conventional utilization and reliability-state-aware queueing delay provide the same operational information.

B. Utilization vs. Traffic

Figure 4 shows component and system utilization as offered traffic increases.

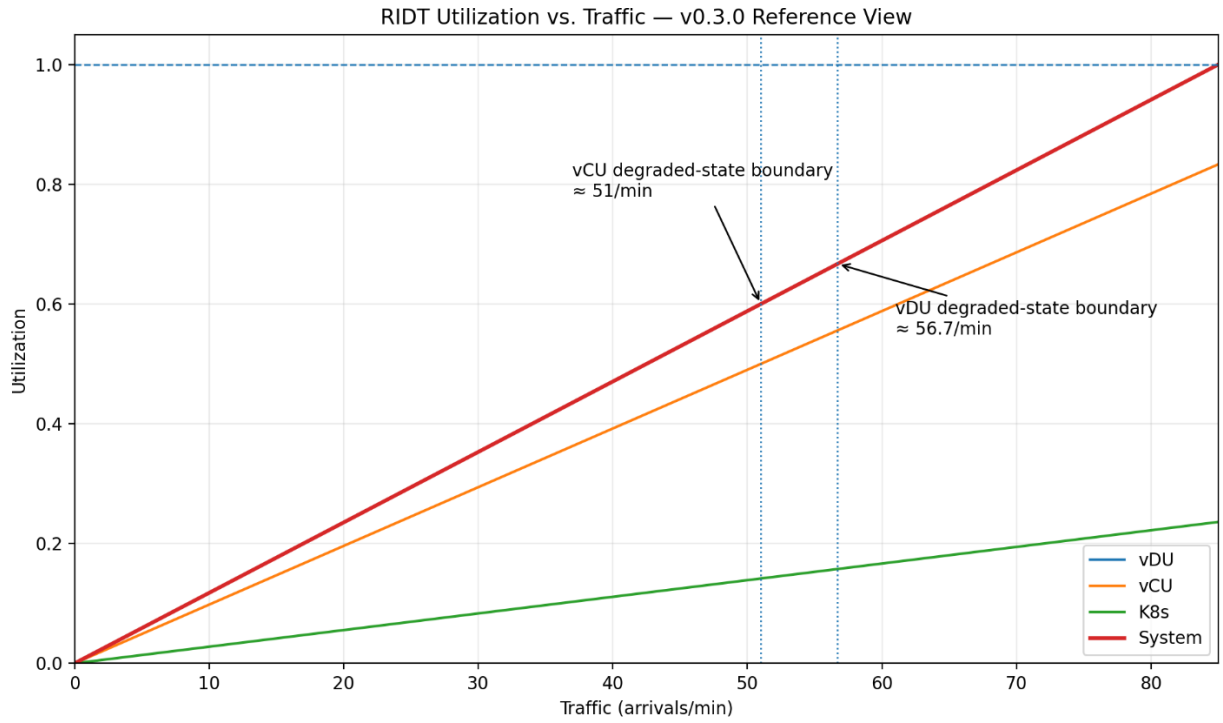

Figure 4. Utilization vs. Traffic

For a fixed operating configuration, utilization follows

$$U_i = \frac{\lambda}{C_i}. \tag{4.5}$$

Consequently, the utilization curves increase approximately linearly with traffic until they reach a capacity boundary.

The vDU dominates the nominal system behavior because its effective capacity of approximately 85 is lower than the corresponding vCU and Kubernetes capacities. As traffic approaches this limit,

$$U_{System} \to 1. \tag{4.6}$$

This view helps identify the nominal bottleneck and determine how close the system is operating to its available capacity.

However, the utilization plot provides limited visual evidence of the performance consequences associated with the lower-capacity degraded reliability states. In particular, the degraded-state boundaries near 51 and 56.7 occur well below the nominal system capacity of 85.

Thus, a conventional utilization view may suggest that substantial nominal capacity remains even though a component failure could place the system in a degraded state whose capacity is already close to the current traffic level.

This motivates examination of the same system through queueing delay.

C. Delay vs. Traffic

Figure 5 presents the probability-weighted component and system queueing delays calculated using the reliability-state-aware formulation in Section III.

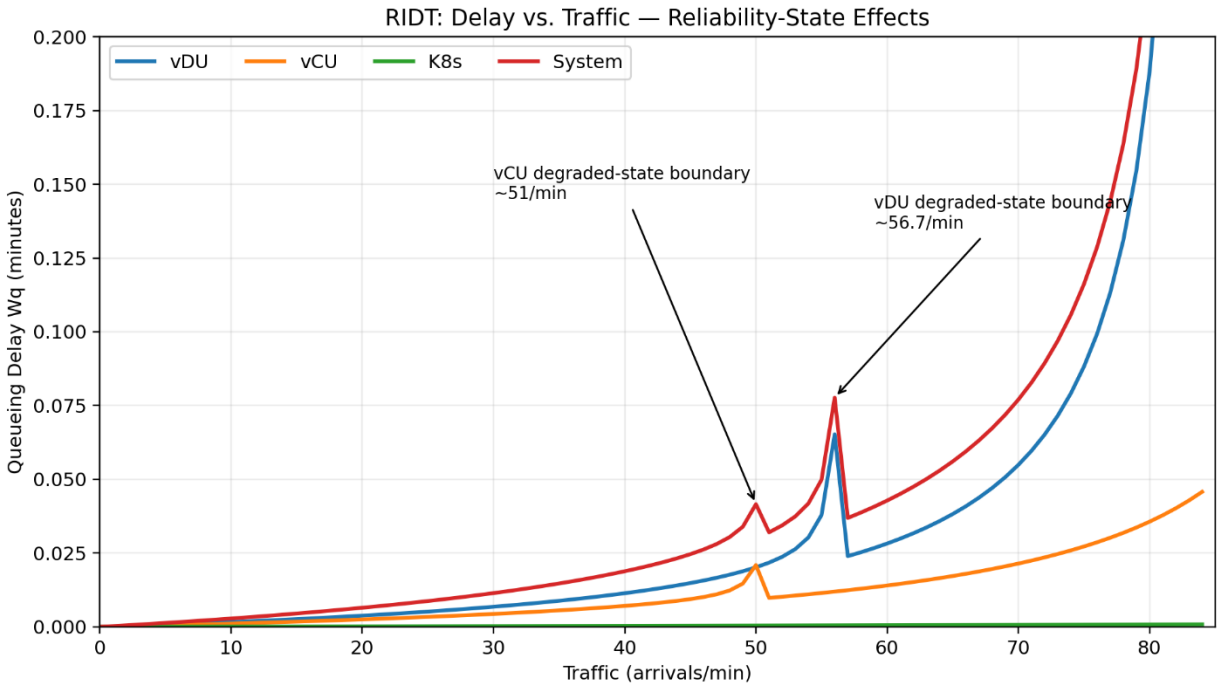

Figure 5. Reliability-State-Aware Delay vs. Traffic

At relatively low traffic levels, all operational reliability states have sufficient capacity, and the expected queueing delay remains small. As traffic increases, the nonlinear characteristic of the M/M/1 model becomes increasingly important:

$$W_q = \frac{\lambda}{\mu(\mu - \lambda)}. \tag{4.7}$$

Unlike utilization, delay increases rapidly as traffic approaches the capacity of a particular reliability state.

The resulting Delay vs. Traffic curve therefore contains information about both traffic loading and the underlying reliability-state structure.

Two localized delay increases are visible in the reference analysis. They occur immediately before traffic reaches approximately 51 and 56.7, corresponding to the capacities of the principal degraded vCU and vDU configurations.

These features are not numerical noise. They arise directly from the reliability-performance coupling defined in Section III.

D. Interpretation of the vCU Degraded-State Signature

Consider first the degraded vCU configuration with an effective end-to-end capacity of approximately

$$C_{vCU,degraded} \approx 51. \tag{4.8}$$

When traffic is below this value, the degraded configuration remains operational and stable. Its queueing delay is therefore included in the probability-weighted expected delay.

As

$\lambda \rightarrow 51$,

the queueing delay associated with this degraded state increases rapidly.

Consequently,

$$P(s_{vCU,degraded}) \times W_q(s_{vCU,degraded}) \tag{4.9}$$

becomes increasingly significant even though the probability of the degraded state itself is relatively small.

This produces the first localized delay signature.

When traffic reaches or exceeds the degraded-state capacity,

$\lambda \geq 51$,

that state is classified as overloaded. Its probability is transferred from the stable-state population to the overload probability, and no finite M/M/1 delay is assigned to it.

The aggregate finite-delay curve therefore returns toward the behavior of the remaining stable states.

The apparent rise and subsequent drop thus result from a **reliability-state transition**, not a decrease in the queueing delay of the degraded state itself.

E. Interpretation of the vDU Degraded-State Signature

A similar mechanism occurs for the degraded vDU configuration, whose effective capacity is approximately

$$C_{vDU,degraded} \approx 56.7. \tag{4.10}$$

As traffic approaches this boundary from below, queueing delay for the degraded vDU state rises nonlinearly and contributes more and more to the probability-weighted system delay.

At

$$\lambda \geq 56.7,$$

the degraded state enters overload and is removed from the finite-delay population.

The second delay signature therefore represents another reliability-state boundary.

Together, the two features demonstrate that the Delay vs. Traffic curve contains information about **which degraded reliability configurations are becoming operationally critical**, even though the nominal all-up system has not yet reached its capacity limit.

F. Why Utilization Alone May Not Reveal the Degradation

The comparison between Figures 4 and 5 is central to the RIDT methodology.

At a traffic level near 51, nominal system utilization based on the all-up capacity of 85 is only approximately

$$U_{nominal} = \frac{51}{85} \approx 0.60. \tag{4.11}$$

Near the second degraded-state boundary,

$$U_{nominal} = \frac{56.7}{85} \approx 0.67. \tag{4.12}$$

From the nominal utilization perspective, neither condition appears close to overload.

However, if a component failure places the system in the corresponding degraded reliability state, the relevant capacity is no longer 85. The system may instead be operating near

$$U_{degraded} \approx 1. \tag{4.13}$$

This difference explains why reliability-state-aware delay analysis provides information that a nominal utilization curve alone may not reveal.

The important comparison is therefore:

Nominal Utilization → How close is the all-up system to capacity?

versus

Reliability-State-Aware Delay →
What is the performance consequence if the system is degraded?

RIDT provides both views within the same analytical framework.

G. Recovery Time and Persistence of Degradation

The magnitude of a degraded state's contribution depends not only on its reduced capacity but also on its probability of occurrence. For repairable components, this probability depends on both failure and recovery behavior.

Under the steady-state availability model,

$$A = \frac{\mu_r}{\lambda_f + \mu_r}, \tag{4.14}$$

where μ_r represents the recovery rate.

The current reference calculation therefore reflects behavior based on the assumed average failure and recovery parameters.

If recovery takes longer than expected,

$$\mu_r \downarrow \quad \Rightarrow \quad A \downarrow, \tag{4.15}$$

and the probability of remaining in a degraded state increases.

Operationally, this means that a small transient delay signature under expected recovery conditions could become more persistent or more consequential when recovery is delayed.

The progression can be represented as

$$\text{Failure} \rightarrow \text{Degraded Capacity} \rightarrow \text{Delay Increase} \rightarrow \begin{cases} \text{Recovery} \rightarrow \text{Normal State} \\ \text{Persistence} \rightarrow \text{Increasing Risk} \\ \text{Traffic} \geq C_s \rightarrow \text{Overload} \end{cases} \tag{4.16}$$

This provides a potential mechanism for identifying degradation that may otherwise remain difficult to recognize from conventional monitoring.

The observed delay signature should not, by itself, be interpreted as proof of a silent component failure. Rather, it provides a **potential indicator of latent or persistent degradation** that can trigger further diagnosis.

H. RIDT as an Integrated Diagnostic View

The comparison demonstrates why reliability and queueing analysis should not be treated as independent engineering activities.

Traditional reliability analysis answers:

What is the probability that the required system components are operational?

Traditional performance analysis answers:

How does traffic affect utilization, delay, and capacity?

RIDT introduces a third question:

How does a change in reliability state alter system performance before the overall system becomes structurally unavailable?

The resulting analytical chain is

$$\text{Failure/Recovery Behavior} \rightarrow P(\text{Reliability State}) \rightarrow C(\text{Reliability State}) \rightarrow U(\lambda,s) \rightarrow W_q(\lambda,s) \rightarrow \text{Operational Risk} \quad (4.17)$$

The delay signatures observed in the reference case illustrate this integration. A component failure changes the available-unit configuration; the changed configuration reduces effective capacity; and queueing theory translates that reduced capacity into a nonlinear performance consequence.

This is the principal distinction between RIDT and an analysis based solely on structural reliability or nominal utilization.

I. Practical Significance

The analysis suggests a practical use of RIDT for Open RAN operation.

Traffic is directly observable and can also be forecast from historical trends. Reliability parameters characterize component failure and recovery behavior. RIDT combines these inputs to estimate how close both normal and degraded configurations are to performance limits.

The resulting Delay vs. Traffic view can therefore support:

- identification of reliability-induced performance degradation;
- recognition of degraded-state capacity boundaries;
- differentiation between nominal loading and degraded-state stress;
- early investigation of potentially persistent degradation;
- assessment of whether expected recovery is occurring; and
- prediction of progression toward overload as traffic increases.

The objective is not to replace conventional utilization monitoring. Rather, RIDT augments it with a reliability-state-aware performance view.

Thus, Figures 4 and 5 should be interpreted together:

Utilization tells us where the nominal capacity limit is;
RIDT delay analysis can reveal what happens when reliability changes the capacity itself.

V. Discussion

A. Integration of Reliability and Performance

The results in Section IV demonstrate that system reliability and system performance cannot always be evaluated independently in a cloud-native Open RAN environment. A system may remain structurally operational according to its k-of-n redundancy requirements while experiencing substantially different performance characteristics after one or more component failures.

Conventional reliability analysis primarily evaluates whether sufficient components remain operational. Conventional performance analysis, in contrast, generally evaluates traffic relative to an assumed available processing capacity. RIDT connects these two perspectives by making processing capacity explicitly dependent on the reliability state:

$$\text{Reliability State} \rightarrow \text{Available Capacity} \rightarrow \text{Performance}. \tag{5.1}$$

This relationship is particularly important in redundant systems. A component failure does not necessarily cause immediate system unavailability. Instead, the system may transition into a degraded but operational configuration with reduced capacity.

The performance consequences of that transition may therefore appear before conventional structural reliability measures indicate system failure.

B. Reliability-Induced Performance Degradation

The reference analysis illustrates this distinction.

Under the fully operational configuration, the nominal end-to-end capacity is approximately 85 traffic units per minute. However, degraded configurations associated with vCU and vDU failures have effective capacities of approximately 51 and 56.7 units per minute, respectively.

Consequently, the same offered traffic can represent very different operating conditions depending on the current reliability state:

$$U(\lambda, s) = \frac{\lambda}{C(s)}. \tag{5.2}$$

For example, at a traffic level of 51,

$$U_{normal} \approx \frac{51}{85} = 0.60, \tag{5.3}$$

while a degraded state with capacity near 51 is already approaching its stability boundary:

$$U_{degraded} \approx 1. \tag{5.4}$$

This distinction is fundamental. A traffic level that appears moderate relative to nominal system capacity can represent a critical operating condition after a component failure.

RIDT therefore introduces the concept of **reliability-induced performance degradation**: performance deterioration caused not primarily by increasing traffic, but by a reliability-state transition that reduces the capacity available to serve existing traffic.

C. Significance of the Delay Signatures

The localized delay increases observed in Section IV directly result from reliability-induced capacity reduction.

For a degraded state s_d,

$$C(s_d) < C(s_0), \tag{5.5}$$

where s_0 denotes the fully operational state. As traffic approaches $C(s_d)$, the M/M/1 queueing delay associated with the degraded state increases nonlinearly:

$$W_q(s_d) \to \infty \quad \text{as} \quad \lambda \to C(s_d). \tag{5.6}$$

Although the probability of the degraded state may be relatively small, its rapidly increasing delay can create a visible signature in the probability-weighted system delay.

This behavior matters because the corresponding nominal utilization curve may show no comparable anomaly. The utilization curve remains dominated by the all-up system capacity, whereas the delay calculation explicitly reflects the capacities and probabilities of degraded reliability states.

The delay signature therefore represents information produced by the **interaction between reliability and queueing behavior**, rather than by either analysis independently.

D. Potential Detection of Latent or Persistent Degradation

A practical implication of this result is the possibility of identifying degraded operating conditions that are not immediately evident from nominal utilization monitoring.

Under expected failure and recovery behavior, a degraded state may be transient. The resulting probability-weighted delay signature may therefore be relatively small.

However, component recovery is inherently stochastic. If recovery takes longer than expected, the system remains in the degraded configuration longer. The operational significance of the reduced-capacity state consequently increases.

Conceptually,

$$\text{Failure} \rightarrow \text{Degraded State} \rightarrow \begin{cases} \text{Expected Recovery} \\ \text{Delayed Recovery} \end{cases} \tag{5.7}$$

and delayed recovery can lead to

$$\text{Persistent Degradation} \rightarrow \text{Increasing Delay} \rightarrow \text{Overload Risk}. \tag{5.8}$$

This creates an important diagnostic opportunity. A delay anomaly occurring at a reliability-sensitive traffic level may warrant investigation even when conventional utilization indicators remain within nominal operating limits.

Such a signature should not be interpreted as definitive proof of a silent component failure. Multiple operational mechanisms can affect observed delay, and the current analytical model does not perform causal diagnosis from measured delay alone.

Rather, RIDT provides a **potential early indicator of latent or persistent degradation** that can be correlated with component status, recovery behavior, logs, alarms, and other operational information.

E. Utilization and Delay Provide Different Information

The results also clarify the respective roles of utilization and queueing delay.

Utilization is valuable because it is simple, intuitive, and directly related to available capacity:

$$U = \frac{\lambda}{C}. \tag{5.9}$$

For a fixed capacity, it clearly measures how heavily a resource is used.

Queueing delay provides a different perspective:

$$W_q = \frac{\lambda}{C(C-\lambda)}. \tag{5.10}$$

Its nonlinear behavior amplifies the effect of approaching a state-dependent capacity boundary.

Therefore, utilization and delay should not be viewed as competing metrics.

Instead,

Utilization identifies proximity to capacity

While

Delay reveals the performance consequence of that proximity.

RIDT's distinctive capability arises because the reliability state determines the capacity used in both calculations.

F. Why Utilization Remains Operationally Important

Although the delay analysis provides greater sensitivity near capacity boundaries, utilization remains especially useful for operational control.

Traffic and resource utilization can generally be measured continuously and with relatively low statistical complexity. End-to-end delay, by comparison, can vary substantially due to workload characteristics, routing, scheduling, network conditions, and other stochastic effects.

This suggests a practical control strategy in which RIDT uses the analytical relationship between traffic, utilization, reliability state, and expected delay to establish operating thresholds.

For example, if the queueing model indicates that delay begins increasing rapidly beyond a particular utilization region, operators may use utilization as the more stable operational control variable while retaining delay as a performance consequence and diagnostic measure.

Thus, RIDT can support the sequence

Delay Analysis → Risk Region Identification → Utilization Target → Operational Control. (5.11)

This is particularly useful when delay measurements are noisy, but utilization can be monitored reliably.

G. Predictive Use of Traffic

The framework also supports predictive analysis because traffic is not only measurable but often forecastable.

Let future traffic be represented as

$$\hat{\lambda}(t+\Delta t). \tag{5.12}$$

RIDT can evaluate the predicted traffic against both normal and degraded-state capacities:

$$\hat{\lambda}(t+\Delta t) \quad \text{versus} \quad C(s). \tag{5.13}$$

This allows the system to estimate whether expected traffic growth could place a degraded reliability state near a critical queueing or overload boundary.

The resulting analysis is therefore not limited to answering:

Is the system overloaded now?

It can also address:

If traffic develops as expected and a component becomes unavailable, which degraded configurations could become performance-critical?

This predictive capability is especially relevant for Open RAN environments, where traffic varies dynamically, and virtualized resources may be scaled or reconfigured.

H. Implications for Digital Twin Design

The findings also clarify RIDT's role as a digital twin.

Digital-twin approaches provide a framework for representing and analyzing network behavior using data and models that correspond to the physical system [4]–[6]. RIDT adds an explicit reliability-state-aware performance layer in which component degradation changes the capacity used in queueing analysis.

A conventional digital representation may reproduce the nominal architecture and current performance state. A reliability-integrated digital twin must additionally represent alternative system configurations created by component failure and recovery.

RIDT therefore maintains a set of possible reliability states,

$$S = \{s_1, s_2, \ldots, s_m\}, \tag{5.14}$$

with each state characterized by:

$$[P(s), C(s), U(\lambda, s), W_q(\lambda, s)]\,. \tag{5.15}$$

This creates a natural foundation for future decision support, including resource scaling, traffic redistribution, preventive maintenance, recovery prioritization, and other corrective actions.

I. Practical Value for Open RAN

The integrated approach is particularly relevant to Open RAN because system behavior depends on interactions among virtualized network functions, shared cloud infrastructure, redundancy configurations, traffic, and orchestration.

The RIDT architecture identifies Kubernetes as an important shared capacity constraint, not merely another reliability element. RIDT extends this concept by showing that component-level reliability transitions can also alter the system's effective performance envelope.

This provides operators and system integrators with a framework for addressing questions such as:

- Which component is the nominal bottleneck?
- Which degraded reliability state becomes critical first as traffic increases?
- Can the system satisfy its redundancy requirement but still experience unacceptable performance?
- Does an observed delay anomaly correspond to a predicted degraded-state boundary?
- How much operating margin exists if a vDU or vCU becomes unavailable?
- How quickly must recovery occur before increasing traffic creates overload risk?

These questions connect system design, reliability engineering, capacity planning, and operational control within a common quantitative framework.

J. Current Limitations and Validation Requirements

The current results should be interpreted as an analytical demonstration rather than as proof of field-level fault detection.

Several assumptions remain to be validated with operational data. The current prototype assumes statistically independent component failures, proportional capacity reduction as redundant units become unavailable, steady-state repairable-component availability, and M/M/1 queueing behavior.

Real Open RAN environments may exhibit correlated failures, common-cause events, non-exponential service distributions, multiple traffic classes, load balancing, autoscaling, and more complex queueing networks.

In addition, the degraded-state delay signatures identified in the reference analysis are predictions of the current model. Demonstrating that such signatures can reliably identify actual latent failures will require comparison with measured traffic, component-state, recovery, and delay data.

Accordingly, the present contribution is not a claim that a particular delay spike uniquely identifies a particular failure. The contribution is the analytical framework that establishes a quantitative path:

Reliability Change → Capacity Change → Queueing Response → Observable Performance Signature (5.16)

This relationship can then be tested and calibrated using operational data.

VI. Practical Implementation and Validation

A. Implementation Framework

The RIDT methodology is intended to operate as an analytical layer above existing Open RAN development, system integration, and operational monitoring environments [4], [6]. It does not require replacement of existing monitoring or test systems. Instead, it combines reliability, traffic, capacity, and recovery information to construct a reliability-state-aware representation of system behavior.

A practical implementation follows the sequence

SIT / Operational Data → Reliability Parameters → Reliability States → State-Dependent Capacity → Utilization and Delay → Risk Assessment (6.1)

The current Open RAN implementation uses vDU, vCU, and Kubernetes as the principal performance-processing components. The same methodology can be extended to additional components when their reliability and capacity characteristics are available.

B. Input Data Requirements

RIDT requires two primary categories of data: **reliability data** and **traffic/capacity data**.

1. Reliability Data

Reliability parameters can be estimated from System Integration Testing (SIT), operational records, or a combination of both [7], [9].

For each major hardware or software component, useful data include:

- number and timing of observed failures;
- operating or observation time;
- number of redundant units;
- minimum number of units required for operation;
- failure recovery or repair times; and
- identification of the affected component or subsystem.

These observations support estimation of failure and recovery parameters such as

$$\lambda_f = \frac{\text{Number of Failures}}{\text{Operating Exposure}} \tag{6.2}$$

and

$$MTTR = \frac{\text{Total Recovery Time}}{\text{Number of Recoveries}}. \tag{6.3}$$

The corresponding recovery rate is

$$\mu_r = \frac{1}{MTTR}. \tag{6.4}$$

For software components, the relevant exposure measure may be operational time or another appropriate workload-based exposure [9]. The specific exposure definition should be selected according to the system and available data.

2. Traffic and Capacity Data

Performance modeling requires the offered traffic and the processing capacity of each major component or processing stage.

The required information includes:

- observed traffic or workload;
- nominal processing capacity;
- number of operating processing units;
- virtualization or processing overhead, where applicable; and
- configuration information required to determine effective capacity.

Traffic and capacity must use a consistent basis. Depending on the deployment, the model may be calibrated using transactions per second, sessions per unit time, packets, normalized processing demand, or another workload measure.

The existing white-paper formulation intentionally allows normalized traffic to be calibrated to real-world workload units using measured traffic and capacity data.

Importantly, **measured utilization and delay are not required as fundamental RIDT model inputs** when they can be calculated from traffic, capacity, and reliability-state information. They are nevertheless valuable as validation measurements.

C. Role of System Integration Testing

For an Open RAN system integrator, SIT provides an especially useful source of reliability information.

During integration testing, the SIer observes the behavior of multiple vendor components as they operate together in the end-to-end system. Defects, failures, recovery events, configuration changes, and test exposure can therefore provide evidence about system-level reliability behavior without requiring each vendor to disclose proprietary internal development data.

This distinction is important.

RIDT does not require detailed proprietary defect histories from individual vendors. The primary requirement is the **system-level evidence observed during integration and operation**.

SIT can therefore provide information such as

Component + Failure Event + Exposure + Recovery Time + System Configuration. (6.5)

These data can be used to estimate reliability parameters and construct the state probabilities required by the RIDT model.

D. Operational Data

Operational data complement SIT by representing actual workload and recovery behavior in the deployed environment.

Useful operational inputs include:

- traffic history;
- component failure events;
- recovery times;
- active-unit configuration;
- scaling or resource-allocation changes; and
- relevant infrastructure capacity.

These observations allow RIDT to update or recalibrate the reliability and capacity model as operational experience accumulates.

The operational traffic series

$$\lambda(t) \tag{6.6}$$

can then be evaluated against the capacity of each possible reliability state:

$$\lambda(t) \quad \text{versus} \quad C(s). \tag{6.7}$$

This permits simultaneous evaluation of normal and degraded operating margins.

E. Model Calibration

The initial RIDT model can be constructed from engineering specifications and available SIT data. Its parameters can subsequently be refined as measured operational data become available.

A practical calibration process consists of four steps.

Step 1 — Reliability calibration

Estimate component failure and recovery parameters from SIT and operational event histories:

$$\{\lambda_{f,i}, \mu_{r,i}\}. \tag{6.8}$$

Step 2 — Capacity calibration

Determine or estimate effective processing capacities:

$$\{C_{vDU}, C_{vCU}, C_{K8s}\}. \tag{6.9}$$

Step 3 — State validation

Confirm that predicted degraded capacities are consistent with observed behavior when one or more units are unavailable.

Step 4 — Performance validation

Compare calculated utilization and queueing delay with measured operational values over selected traffic ranges.

The purpose is not to force measured behavior to match an assumed model. Instead, use discrepancies to refine capacity, redundancy, dependency, or queueing assumptions.

F. Validation of Utilization Analysis

Utilization provides a relatively direct first validation of the performance model.

For component *i*,

$$U_i(t) = \frac{\lambda(t)}{C_i(s_t)}. \tag{6.10}$$

Calculated utilization can therefore be compared with corresponding measured resource utilization when such telemetry is available.

Agreement between calculated and observed utilization provides evidence that the traffic-to-capacity mapping is reasonable.

Disagreement may indicate:

- incorrect capacity assumptions;
- unmodeled processing overhead;
- load-balancing effects;
- configuration changes;
- incorrect traffic normalization; or
- additional resource constraints.

This makes utilization an important calibration and troubleshooting metric even though it does not provide the full sensitivity of the queueing-delay analysis.

G. Validation of Delay vs. Traffic

The new RIDT delay capability requires a more detailed validation because measured delay is stochastic.

For a given reliability state *s*, the model predicts

$$W_q(\lambda, s). \tag{6.11}$$

Measured delay observations can be represented as

$$W_{obs}(t). \tag{6.12}$$

Rather than expecting individual observations to follow the theoretical curve exactly, validation should compare the model with statistically summarized measurements over comparable traffic and configuration conditions.

For example,

$$E[W_{obs} \mid \lambda, s] \quad \text{versus} \quad W_q(\lambda, s). \tag{6.13}$$

This distinction matters because observed delay can include variability from scheduling, routing, workload mix, network conditions, and other factors not captured in the current M/M/1 approximation [7], [8].

The objective of initial validation is therefore to determine whether the predicted **trend and critical regions** are consistent with observed behavior, rather than requiring exact point-by-point agreement.

H. Validation of Degraded-State Delay Signatures

The degraded-state delay signatures identified in Section IV provide a particularly important validation opportunity.

For the current reference model, predicted degraded-state boundaries occur near

$$\lambda \approx 51 \tag{6.14}$$

for the vCU-related degraded configuration and

$$\lambda \approx 56.7 \tag{6.15}$$

for the vDU-related degraded configuration.

A practical validation experiment can examine periods in which the corresponding component configuration is known to have changed.

For each event:

$$\text{Component State + Traffic + Recovery Time + Observed Delay} \tag{6.16}$$

should be aligned on a common timeline.

RIDT can then test whether:

1. the component entered the predicted degraded state;
2. effective capacity decreased as expected;
3. traffic approached the degraded-state capacity;
4. observed delay increased in the predicted region; and
5. delay returned toward normal following recovery.

This provides a direct empirical test of the reliability-performance mechanism proposed in Sections III and IV.

I. Recovery-Time Validation

Recovery behavior is especially important because the probability of a degraded state depends on the assumed recovery rate.

The model uses

$$\mu_r = \frac{1}{MTTR}. \tag{6.17}$$

If actual recovery is significantly slower than the historical or assumed MTTR, the system may remain degraded longer than expected.

RIDT can therefore compare an observed recovery duration,

$$T_{recovery,obs}, \tag{6.18}$$

with the expected recovery behavior.

A prolonged degraded condition accompanied by increasing delay would provide stronger evidence of persistent degradation than an isolated short-duration delay excursion.

This creates a potential diagnostic sequence:

Failure → Expected Recovery Window → Recovery Confirmed

or

Failure → Recovery Delayed → Persistent Degraded State → Increasing Performance Risk. (6.19)

J. Potential Silent-Degradation Investigation

The integrated analysis also provides a method for investigating conditions in which conventional monitoring does not clearly indicate the underlying reliability problem.

Suppose traffic and nominal utilization appear acceptable, but an unexpected delay pattern occurs near a predicted degraded-state capacity boundary.

RIDT can evaluate whether the observed behavior is consistent with an alternative reliability state:

$$W_{obs} \approx W_q(\lambda, s_d). \tag{6.20}$$

If so, the system can flag the condition for further investigation.

The appropriate interpretation is:

The observed performance behavior is consistent with a possible degraded reliability state.

It is not:

RIDT has proven that a silent failure occurred.

Confirmation should be sought from logs, alarms, component-state records, recovery information, or other independent evidence.

This distinction allows RIDT to serve as an analytical **early-warning and diagnostic-support mechanism** without overstating causal certainty.

K. Predictive Operational Use

Once calibrated, RIDT can move from retrospective analysis to predictive operation.

Given a forecast traffic trajectory,

$$\hat{\lambda}(t + \Delta t), \tag{6.21}$$

the model can evaluate that traffic against the capacities of normal and degraded reliability states.

For each relevant state s, RIDT can estimate:

$$U[\hat{\lambda}, s], \qquad W_q[\hat{\lambda}, s], \qquad P(s), \tag{6.22}$$

and determine whether the projected traffic approaches a critical performance region.

This enables questions such as:

If a vDU becomes unavailable during tomorrow's expected peak traffic, will the remaining configuration remain within an acceptable operating region?

or:

How much traffic margin remains if the system enters the most probable degraded state?

The combination of traffic forecasting and reliability-state analysis therefore provides a basis for proactive operational decision-making.

L. Validation Roadmap

A practical RIDT validation program can proceed incrementally:

Phase 1 — Historical SIT validation
Use existing SIT failure, recovery, configuration, and exposure records to estimate reliability parameters.

Phase 2 — Capacity and utilization validation
Compare calculated component capacities and utilization with measured system behavior.

Phase 3 — Delay validation
Compare predicted Delay vs. Traffic trends with measured delay under known operating configurations.

Phase 4 — Degraded-state validation
Examine known vDU/vCU failure or recovery events to determine whether predicted reliability-induced delay signatures are observable.

Phase 5 — Predictive validation
Use historical traffic to make forward predictions and compare those predictions with subsequently observed behavior.

Phase 6 — Operational decision support
Evaluate RIDT-generated warnings, operating margins, and potential corrective actions in a controlled operational setting.

This staged approach permits each layer of the model to be validated independently before relying on the integrated RIDT result.

M. Practical Deployment Perspective

RIDT is not intended to replace existing Open RAN observability, orchestration, or fault-management systems.

Instead, it adds an analytical layer that combines information these systems already produce:

Existing Data + Reliability Theory + Queueing Theory → Predictive Reliability–Performance Insight (6.23)

The practical value lies in converting conventional operational records into an integrated representation of how component reliability, recovery, traffic, capacity, utilization, and delay interact.

For an SIer, this provides a mechanism for using system-level SIT knowledge together with operational information to evaluate not only whether an integrated system is functioning, but also **how resilient its performance is to component degradation under changing traffic conditions**.

VII. Conclusion

This paper presented the **Reliability-Integrated Digital Twin (RIDT)** framework for analyzing the interaction among reliability, capacity, traffic, utilization, and queueing performance in Open RAN systems.

The central premise of RIDT is that structural reliability and system performance should not be evaluated independently. In a redundant system, a component failure may not immediately cause system unavailability. Instead, the system can transition to a **degraded but operational reliability state** with reduced processing capacity.

RIDT captures this relationship through the analytical sequence

Reliability State → State-Dependent Capacity → Utilization → Queueing Delay → Overload Risk (7.1)

The Open RAN reference analysis demonstrated the practical significance of this integration.

Under the fully operational configuration, the reference system has an effective end-to-end capacity of approximately 85 traffic units per minute. However, degraded configurations associated with vCU and vDU failures have substantially lower capacity boundaries of approximately 51 and 56.7 units per minute, respectively.

From a nominal utilization perspective, these traffic levels correspond to only approximately 60% and 67% of the all-up system capacity. They therefore do not appear to represent an immediate overload condition.

The reliability-state-aware queueing analysis reveals a different picture.

As traffic approaches the capacity of a degraded reliability state, the associated M/M/1 queueing delay increases nonlinearly. When weighted by the probability of the reliability state, this behavior can produce a localized **delay signature** in the system Delay vs. Traffic curve.

Thus, the analysis demonstrates that

a reliability change can create an observable performance effect

even while nominal utilization remains below the normal system capacity.

This represents an important distinction between conventional utilization monitoring and RIDT. Utilization identifies proximity to an assumed available capacity, whereas RIDT explicitly considers that **the available capacity itself can change when the reliability state changes**.

Accordingly,

Utilization identifies proximity to capacity;

RIDT delay analysis can reveal the consequence of reliability-induced capacity change. (7.2)

The resulting delay signature should not be interpreted as definitive evidence of a particular component failure. Rather, it provides a potential indicator of latent or persistent degradation

that can be investigated using component status, alarms, logs, recovery information, and other operational evidence.

Recovery behavior is particularly important. If a degraded component recovers within its expected recovery period, its performance effect may be transient. If recovery is delayed, the degraded state may persist, and the resulting performance impact can increase as traffic approaches the reduced-capacity boundary, potentially progressing toward overload.

The resulting operational progression is

$$\text{Component Failure} \rightarrow \text{Degraded State} \rightarrow \text{Reduced Capacity} \rightarrow \text{Delay Increase} \rightarrow \begin{cases} \text{Recovery} \\ \text{Persistent Degradation} \\ \text{Overload} \end{cases} \tag{7.3}$$

This relationship provides the foundation for using RIDT not only as a descriptive analytical model but ultimately as a **predictive operational decision-support framework**.

Traffic can be measured and forecast; reliability parameters can be estimated from SIT and operational failure/recovery histories; and state-dependent capacities can be calibrated against observed system behavior. RIDT can therefore evaluate future traffic against both normal and degraded configurations to estimate operating margin and identify conditions requiring closer investigation.

The proposed framework must now be validated using measured Open RAN data. In particular, future validation should determine whether the predicted degraded-state delay signatures are observable during known component failures and recovery events, and whether they provide useful diagnostic information beyond conventional utilization monitoring.

The principal contribution of RIDT can therefore be summarized as

$$\text{Reliability Theory} + \text{Queueing Theory} + \text{Operational Data} \rightarrow \text{Reliability-State-Aware Performance Insight} \tag{7.4}$$

By quantitatively linking component failure and recovery behavior to state-dependent capacity and nonlinear queueing performance, RIDT provides a framework for understanding **not only whether an Open RAN system is operational, but how reliably it can continue to perform as system conditions change.**

Acknowledgment

The author thanks Toshio Takahashi for valuable discussions and insights.

References

[1] ETSI, “O-RAN Architecture Description,” ETSI TS 103 982 V8.0.0, Jan. 2024.

[2] O-RAN Alliance, *O-RAN: Towards an Open and Smart RAN*, White Paper, Oct. 2018.

[3] M. Polese, L. Bonati, S. D’Oro, S. Basagni, and T. Melodia, “Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges,” *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, Second Quarter 2023, doi: 10.1109/COMST.2023.3239220.

[4] J. Mirzaei, I. Abualhaol, and G. Poitau, “Network Digital Twin for Open RAN: The Key Enablers, Standardization, and Use Cases,” arXiv:2308.02644, 2023.

[5] O-RAN Alliance, *Digital Twin RAN: Key Enablers*, O-RAN NGRG Contributed Research Report RR-2024-09, 2024.

[6] Y. Deshpande, E. Sulkaj, and W. Kellerer, “TwinRAN: Twinning the 5G RAN in Azure Cloud,” in *Proc. IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2025, pp. 1–10, doi: 10.1109/NOMS57970.2025.11073616.

[7] K. S. Trivedi, *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*, 2nd ed. New York, NY, USA: Wiley, 2002.

[8] L. Kleinrock, *Queueing Systems, Volume 1: Theory*. New York, NY, USA: Wiley, 1975.

[9] J. D. Musa, A. Iannino, and K. Okumoto, *Software Reliability: Measurement, Prediction, Application*. New York, NY, USA: McGraw-Hill, 1987.