ECN Parameter Tuning for RoCEv2 Networks: A Digital Twin-based Approach

Fuming Chu[1,2], Lizhuang Tan[1,2], Chengxiao Yu[3], Yanyan Tan[4]
[1]*Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Ji'nan, China*
[2]*Shandong Provincial Key Laboratory of Computing Power Internet and Service Computing, Shandong Fundamental Research Center for Computer Science, Ji'nan, China*
[3]*Department of New Networks, Pengcheng Laboratory, Shenzhen, China*
[4]*School of Computer Science and Artificial Intelligence, Shandong Normal University, Jinan, China*
Email: 10431250144@stu.qlu.edu.cn, tanlzh@sdas.org, yuchx@pcl.ac.cn, yytan928@sdnu.edu.cn

***Abstract*—RDMA over Converged Ethernet version 2 (RoCEv2) has been widely deployed in data centers for artificial intelligence training, distributed storage, and high-performance computing. Its performance, however, is highly sensitive to Explicit Congestion Notification (ECN) parameter settings. Static ECN configurations are difficult to maintain under dynamic and bursty traffic, while distributed tuning may suffer from local-policy coupling and centralized online search can introduce considerable telemetry, computation, and probing overhead. This paper presents DT-ECN, a digital twin-based ECN parameter tuning method for RoCEv2 networks. Instead of repeatedly evaluating candidate configurations in the production network, DT-ECN constructs a calibrated digital twin from In-band Network Telemetry (INT), generates traffic-aware ECN candidates, evaluates them through fluid-model-based parallel simulation, and selects the best configuration using a multi-objective utility function that jointly considers throughput, queue length, and Priority-based Flow Control (PFC) risk. Prototype experiments under representative Incast scenarios show that DT-ECN reduces the candidate evaluation cost by 94.4% compared with ByteTuning when reaching the same target utility. Compared with the default configuration, DT-ECN improves average throughput by 10.3% and up to 21.6%, while reducing P99 flow completion time (FCT) and PFC pause frames by 58.2% and 85.9% on average, respectively.**

***Index Terms*—RDMA Over Converged Ethernet Version 2, Explicit Congestion Notification Parameter Tuning, Digital Twin, In-band Network Telemetry**

I. Introduction

RDMA over Converged Ethernet version 2 (RoCEv2) [1] carries Remote Direct Memory Access (RDMA) traffic over standard Ethernet and provides low delay, high throughput, and low CPU overhead while maintaining Ethernet compatibility. It has been widely adopted in data center scenarios such as artificial intelligence training, distributed storage, and high-performance computing [2]. However, the lossless transmission requirement of RDMA [3] places strict demands on queue control and congestion management.

RoCEv2 networks typically combine Priority-based Flow Control (PFC) and Explicit Congestion Notification (ECN) for congestion management [4]. PFC reduces packet loss risk by pausing upstream senders, while ECN marks packets before queue buildup reaches the PFC threshold and guides end-host congestion control algorithms [5], such as Data Center Quantized Congestion Notification (DCQCN) [6], to adjust sending rates. Therefore, switch-side ECN parameters, including K_{min}, K_{max}, and P_{max}, directly affect congestion feedback timing [7], queue length, link utilization [8], PFC risk, and flow completion time [9]. Since data center traffic is highly dynamic, fixed ECN parameters are difficult to maintain across different workloads, and ECN tuning must dynamically balance throughput, queue length, and PFC risk.

Existing ECN tuning approaches mainly include static empirical configuration, distributed local tuning, and centralized feedback search. Static configuration is simple but cannot adapt to changing traffic patterns. Distributed methods are vulnerable to local observations and policy coupling among switches. Centralized feedback search provides a global view, but it often requires repeatedly applying candidate configurations in the real network, causing additional control overhead and service disturbance. Therefore, RoCEv2 networks require an ECN tuning method that simultaneously provides global coordination, efficient search, and low online disturbance.

To address these issues, this paper proposes DT-ECN, a digital twin-based ECN parameter tuning method for RoCEv2 networks. We focus on the automatic configuration of switch-side ECN parameters K_{min}, K_{max}, and P_{max}, while treating PFC thresholds as safety constraints for candidate filtering and configuration deployment rather than primary tuning targets. DT-ECN migrates candidate ECN parameter generation, evaluation, and selection to the digital domain. The physical domain collects telemetry measurements through INT [10], while the digital domain performs traffic identification, model calibration, candidate generation, fluid-model-based parallel simulation, and multi-objective decision making. A new con-

This work was supported by the research projects funded by the National Key R&D Program of China under Grant No.2024YFB2907000, the Shandong Provincial Natural Science Foundation under Grant No.ZR2024LZH006, the National Natural Science Foundation of China under Grant No.62402257, the Major Key Project of PCL under Grant No.PCL2025A08, the Key Laboratory of Computing Power Network and Information Security, Ministry of Education under Grant No.2024PY009.

figuration is deployed only when it satisfies safety constraints and provides a utility improvement above a predefined threshold, thereby reducing online probing and tuning disturbance in production networks.

The main contributions of this paper are summarized as follows:

- We propose DT-ECN, a digital twin-based ECN parameter tuning framework for RoCEv2 networks. By combining physical-domain telemetry with digital-domain candidate evaluation, DT-ECN shifts most candidate exploration from the production network to the digital domain, reducing online probing cost and tuning disturbance.
- We design a traffic-aware and safety-constrained ECN tuning mechanism that integrates biased Monte Carlo candidate generation, fluid-model-based parallel simulation, and multi-objective utility optimization. This mechanism efficiently identifies ECN configurations that jointly balance throughput, queue buildup, and PFC risk.
- We implement a DT-ECN prototype and evaluate it under representative Incast scenarios. Experimental results show that DT-ECN reduces the candidate evaluation cost by 94.4% compared with ByteTuning, improves average throughput by 10.3% and up to 21.6%, and reduces P99 FCT and PFC pause frames by 58.2% and 85.9% on average, respectively.

II. Related Work and Motivation

ECN parameter configuration directly affects congestion feedback timing, queue buildup, link utilization, and PFC risk in RoCEv2 networks. Due to the burstiness and dynamics of data center traffic, automatically adjusting ECN parameters according to the current traffic state has become an important problem in RoCEv2 network management [11]. Existing approaches can be broadly categorized into static empirical configuration, distributed local tuning, and centralized feedback search.

Static empirical configuration typically sets ECN/PFC thresholds according to link speed, round-trip time, switch buffer size, and vendor recommendations. This type of method is easy to deploy and stable in operation, but its parameters are usually determined offline and are difficult to adapt to Incast, mixed small-flow and large-flow traffic, and dynamic workload changes.

Distributed tuning methods, represented by ACC approaches [12], deploy agents or local control logic on switches and adjust ECN thresholds according to local queue length, throughput, and congestion feedback [13]. These methods have good scalability, but without global coordination, parameter updates across switches may interfere with each other, leading to policy coupling, unstable convergence, and local optima.

Centralized feedback search methods, represented by ByteTuning [14], use global runtime states to tune congestion-control-related parameters. They can alleviate local-observation and policy-coupling issues in distributed approaches. However, they usually rely on online probing in the physical network, requiring repeated configuration deployment, feedback observation, and search-direction updates. This can introduce additional control overhead and traffic disturbance.

In summary, existing approaches are difficult to simultaneously achieve global coordination, efficient search, and low online disturbance. This paper therefore migrates candidate parameter generation, evaluation, and selection to the digital domain through DT-ECN [15], preserving a global state view while reducing online configuration trials in the physical network.

III. DT-ECN System Design

This section presents DT-ECN, which moves ECN parameter exploration from the production network to a calibrated digital domain. We then describe its architecture, state modeling, candidate generation, and simulation-based decision making.

A. System Architecture

DT-ECN targets automated ECN parameter tuning for RoCEv2 networks. Given the current telemetry state, switch-supported configuration range, and safety constraints, DT-ECN aims to automatically select an ECN configuration $x = (K_{min}, K_{max}, P_{max})$ that balances throughput, queue length, and PFC risk. The overall architecture is illustrated in Fig. 1. The system consists of a physical domain and a digital domain. The physical domain includes RoCEv2 switches, servers equipped with RNICs, network telemetry channels, and configuration interfaces. It carries real traffic, collects runtime states, and receives configuration updates. The digital domain includes traffic-state identification, model calibration, candidate parameter generation, fluid-model-based parallel simulation, multi-objective optimization, and configuration deployment modules. It generates candidate ECN parameter sets, evaluates their throughput, queue, and PFC risk, and selects the best configuration from the current candidate set.

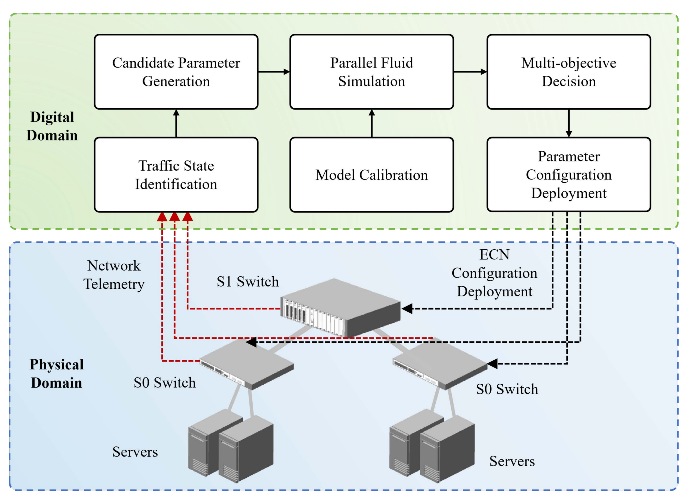


Fig. 1: System architecture of DT-ECN.

B. State Sensing and Model Calibration

The physical-domain state sensing layer maps the runtime state of the real RoCEv2 network to the digital domain. Switches report flow identifiers, port identifiers, timestamps, queue lengths, ECN marking states, and byte counters through packets or telemetry reports. Based on this information, the digital domain reconstructs active flow paths, estimates end-to-end throughput, and identifies bottleneck ports and congestion levels [16].

To reduce telemetry overhead, we use sampled telemetry [17] to estimate throughput. Let Δt be the observation period, N_{report} be the number of telemetry reports received during the period, L_i be the actual payload length in bytes of the i-th sampled packet, and ρ be the telemetry sampling rate. The throughput is estimated as

$$T \approx \frac{\sum_{i=1}^{N_{report}} L_i}{\rho \Delta t}. \tag{1}$$

Since L_i is measured in bytes, the estimated throughput is first obtained in bytes per second and is converted into bit/s or Gbps when reporting the experimental results. Compared with estimation based on a fixed maximum transmission unit (MTU), this method uses the actual payload length contained in telemetry reports and is better suited to mixed small-flow and large-flow scenarios, reducing the overestimation of small-flow throughput.

The digital twin model must be dynamically calibrated according to the physical network state. Let $q_e^{real}(t)$ and $q_e^{sim}(t)$ denote the real and simulated queue lengths of port e, respectively. Let $T_g^{real}(t)$ and $T_g^{sim}(t)$ denote the real and simulated throughput of flow group g, respectively. The queue error $E_q(t)$ and throughput error $E_T(t)$ are defined as

$$E_q(t) = \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \frac{|q_e^{real}(t) - q_e^{sim}(t)|}{q_e^{real}(t) + \epsilon_0}, \tag{2}$$

$$E_T(t) = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \frac{|T_g^{real}(t) - T_g^{sim}(t)|}{T_g^{real}(t) + \epsilon_0}. \tag{3}$$

When either $E_q(t)$ or $E_T(t)$ exceeds its threshold for multiple consecutive periods, the system enters the model calibration state. Since network devices cannot directly obtain the internal state of heterogeneous RNICs [18], we infer equivalent macroscopic parameters from observation feedback, including feedback delay τ, rate reduction coefficient η, and additive increase step δ. The calibration objective over a recent window is

$$\min_{\tau,\eta,\delta} \sum_{t \in \mathcal{W}} \left(E_q(t) + E_T(t)\right). \tag{4}$$

In our prototype, calibration is triggered when either the queue error or the throughput error exceeds a predefined threshold for three consecutive monitoring windows. The equivalent macroscopic parameters are fitted by a lightweight grid search over recent telemetry windows, which avoids introducing high online computation overhead while keeping the digital model close to the physical network state.

Through this calibration, the fluid model in the digital domain can dynamically approximate the queue and throughput dynamics of the physical network. When the model error is large or calibration is not complete, DT-ECN suspends aggressive parameter deployment and only keeps the current configuration or falls back to a conservative configuration, reducing the tuning risk caused by prediction bias.

C. Candidate Parameter Generation

The candidate generation layer generates a set of ECN candidate configurations according to the current traffic state. The i-th candidate configuration is denoted as $x_i = (K_{min}^i, K_{max}^i, P_{max}^i)$, where K_{min}^i, K_{max}^i, and P_{max}^i represent the ECN marking threshold, the threshold where the marking probability reaches the maximum, and the maximum marking probability, respectively. The objective is to generate configurations that match the current traffic state while satisfying the switch-supported configuration range and PFC safety constraints, thereby providing an effective search space for fluid simulation and multi-objective decision making.

Since the ECN parameter space is continuous, uniform random search over the entire space may produce many candidates that do not match the current traffic state. We therefore adopt a traffic-aware biased Monte Carlo generation method. It adjusts the sampling distribution according to the current traffic state and concentrates candidates in more promising parameter regions. Lightweight traffic-state statistics are used as prior information to guide the sampling center and reduce online probing in the physical network.

The candidate generation process is illustrated in Fig. 2. It consists of three steps. First, the system identifies elephant flows, potential elephant flows, and mice flows based on telemetry statistics within a window, and determines the sampling bias factor. Second, it generates candidate (K_{min}, K_{max}) values around the biased baseline ECN thresholds and combines them with P_{max} to form candidate vectors. Finally, it filters high-risk configurations using PFC safety constraints to obtain a valid candidate set.

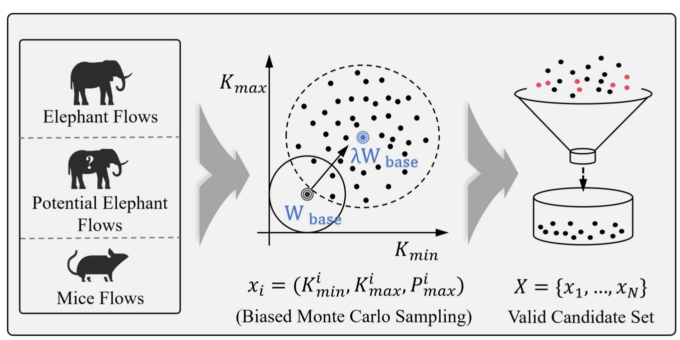


Fig. 2: Illustration of ECN candidate parameter generation in DT-ECN.

Let $b_f(i)$ denote the bytes reported by flow f in the i-th monitoring period, and let W be the sliding window length.

The accumulated bytes of flow f within the window are

$$B_f(t) = \sum_{i=t-W+1}^{t} b_f(i). \tag{5}$$

Let τ_b be the threshold for elephant-flow detection. If $B_f(t) \geq \tau_b$, flow f is identified as an elephant flow. If $B_f(t) < \tau_b$ but remains active within the window, it is identified as a potential elephant flow. Otherwise, it is treated as a mice flow. The system determines the bias factor λ according to the proportions of these three traffic types. When elephant flows dominate, $\lambda > 1$ shifts candidates to higher threshold regions to maintain throughput. When mice flows dominate, $\lambda < 1$ shifts candidates to lower threshold regions to reduce queue length. When potential elephant flows or mixed traffic dominate, λ is set close to or slightly larger than 1 to balance throughput and queue control. In implementation, λ can be selected from a predefined discrete set to avoid additional continuous optimization overhead.

Let the baseline ECN thresholds be $W_{base} = (K_{min}^{base}, K_{max}^{base})$. We construct biased candidate distributions for K_{min} and K_{max} around λW_{base}. For the i-th candidate configuration, K_{min}^i and K_{max}^i are sampled from truncated normal distributions as

$$K_{min}^i \sim \mathcal{N}_{trunc}(\lambda K_{min}^{base}, \sigma_{min}^2), \tag{6}$$

$$K_{max}^i \sim \mathcal{N}_{trunc}(\lambda K_{max}^{base}, \sigma_{max}^2), \tag{7}$$

where σ_{min} and σ_{max} control the candidate search range. A smaller variance enables faster search around the current biased center, while a larger variance preserves random exploration. The maximum marking probability P_{max}^i is uniformly sampled from the discrete set $\mathcal{P}$ supported by the switch. The resulting candidate configuration is $x_i = (K_{min}^i, K_{max}^i, P_{max}^i)$.

To reduce the risk of triggering PFC, let K_{PFC} be the PFC threshold and Δ_{safe} be the safety margin. Each candidate must satisfy $0 < K_{min}^i < K_{max}^i$ and $K_{max}^i + \Delta_{safe} < K_{PFC}$. If a sampled candidate violates these constraints, the system either resamples it or projects it back to the legal range. Through biased sampling and safety filtering, the candidate generation module dynamically adjusts the ECN search region, reduces invalid candidates, and provides effective inputs for subsequent fluid-based parallel simulation and multi-objective decision making. The concrete parameter settings used in the experiments are reported in Section IV-A.

D. Parallel Simulation and Multi-objective Decision Making

The digital-domain parallel simulation and decision layer evaluates candidate ECN configurations and selects the best one. The input of the digital simulation is a candidate ECN configuration and the calibrated traffic state, while the outputs are predicted throughput, average queue length, and PFC risk over the simulation window [19]. For each candidate $x_i = (K_{min}^i, K_{max}^i, P_{max}^i)$, DT-ECN injects it into the digital-domain fluid model, simulates the evolution of port queues, ECN marking feedback, and source sending rates [20], and then calculates the corresponding throughput, queue length, and PFC risk. Since candidate configurations are independent, the candidate set $X = \{x_1, x_2, \ldots, x_N\}$ can be evaluated in parallel in the digital domain.

Candidate ECN parameters enter the fluid model through the port-level ECN marking mechanism. The digital model computes marking feedback according to the current queue length and candidate ECN parameters, and further drives source-rate adjustment and queue evolution. For port e, the queue update is expressed as

$$q_e^i(t+\Delta t) = \left[q_e^i(t) + \Delta t \left(\sum_{g \in \mathcal{G}_e} R_g^i(t) - C_e \right) \right]^+, \tag{8}$$

where $\mathcal{G}_e$ denotes the set of flow groups traversing port e, $R_g^i(t)$ denotes the sending rate of flow group g under candidate x_i, C_e denotes the link capacity of port e, and $[\cdot]^+$ denotes the non-negative constraint. The source sending rate is updated by a macroscopic fluid response model driven by ECN marking feedback. Its feedback delay, rate reduction coefficient, and additive increase step are set to the equivalent macroscopic parameters calibrated in Section III-B.

After simulation, the system computes three evaluation metrics according to the rate and queue trajectories generated by each candidate. $\widehat{T}(x_i)$ denotes the normalized throughput, obtained from the sending rates of all flow groups in the simulation window. $\widehat{Q}(x_i)$ denotes the normalized queue length, computed from the average ratio of port queue trajectories to PFC thresholds. $\widehat{L}(x_i)$ denotes the normalized PFC risk and is computed as the frequency with which queues exceed PFC thresholds

$$\widehat{L}(x_i) = \frac{1}{|\mathcal{T}||\mathcal{E}|} \sum_{t \in \mathcal{T}} \sum_{e \in \mathcal{E}} \mathbb{I}\left(q_e^i(t) > K_{PFC}^e \right), \tag{9}$$

where $\mathcal{T}$ denotes the simulation window, $\mathcal{E}$ denotes the set of ports, and K_{PFC}^e denotes the PFC triggering threshold of port e.

The multi-objective utility function used by DT-ECN to evaluate candidate configurations is

$$U(x_i) = \omega_T \widehat{T}(x_i) - \omega_Q \widehat{Q}(x_i) - \omega_L \widehat{L}(x_i), \tag{10}$$

where ω_T, ω_Q, and ω_L are the weights of throughput, queue length, and PFC risk, respectively. The selected candidate with the maximum utility is

$$x^* = \arg\max_{x_i \in X} U(x_i). \tag{11}$$

To avoid network oscillation caused by frequent updates, DT-ECN deploys x^* only when its utility improvement over the current configuration x_{cur} exceeds a threshold ϵ, and when it simultaneously satisfies PFC safety constraints and model-error constraints. Otherwise, the current configuration remains unchanged.

IV. Implementation and Experimental Analysis

We build a prototype of DT-ECN that integrates telemetry collection, traffic-state identification, model calibration, candidate generation, parallel fluid simulation, multi-objective decision making, and configuration deployment. The prototype collects INT or sampled telemetry reports from switches, updates the digital-domain model with runtime feedback, evaluates candidate ECN configurations in batches, and deploys a new configuration only when it improves the utility while satisfying safety constraints.

A. Experimental Setup

We implement DT-ECN on a real two-tier Clos RoCEv2 testbed. The testbed consists of four Inspur CN9500 200G spine switches, eight Inspur CN9300 100G leaf switches, and 32 servers. Each server is equipped with two Yunbao 100 Gbps RDMA NICs. The spine–leaf and leaf–server links operate at 200 Gbps and 100 Gbps, respectively. RoCEv2 with Data Center Quantized Congestion Notification (DCQCN) is enabled at the end hosts, while ECN and Priority-based Flow Control (PFC) are enabled on the switches.

We compare DT-ECN with four representative baselines: Default, Expert, ACC, and ByteTuning. Default uses the vendor-recommended ECN configuration. Expert uses a static configuration derived from link capacity, round-trip time (RTT), and empirical operating rules. ACC represents distributed local ECN tuning based on local queue and throughput observations, while ByteTuning represents centralized online search based on simulated annealing. DT-ECN differs from these baselines by evaluating candidate ECN configurations in the calibrated digital domain before deployment.

Experiments are conducted under 2-to-1, 8-to-1, and 16-to-1 Incast workloads to evaluate performance under different congestion intensities. We report throughput, average queue length, P99 flow completion time (FCT), PFC pause frames, and tuning efficiency. Unless otherwise specified, DT-ECN generates $N = 64$ candidates in each tuning round. The traffic-state bias factor is selected from $\lambda \in \{0.8, 1.0, 1.2\}$, and the maximum marking probability is sampled from $\mathcal{P} = \{1\%, 10\%, 20\%, 50\%, 80\%, 100\%\}$. The standard deviations of the truncated normal distributions are set to $\sigma_{min} = 0.15K_{min}^{base}$ and $\sigma_{max} = 0.15K_{max}^{base}$, and the safety margin is set to $\Delta_{safe} = 0.1K_{PFC}$.

B. Tuning Efficiency

The tuning efficiency experiment evaluates whether DT-ECN can find better ECN parameters with fewer candidate evaluations. We use multiple Incast scenarios and compare ByteTuning, ACC, and DT-ECN in terms of utility values under different numbers of candidate evaluations. Since DT-ECN biases the sampling range according to the traffic state and evaluates candidates in parallel in the digital domain, it can reduce invalid candidate search and improve tuning efficiency.

Fig. 3 compares the normalized utility under different numbers of candidate evaluations. DT-ECN reaches a utility of 0.80 after 50 evaluations and converges to 0.85, while ByteTuning requires 900 evaluations to reach 0.80 and ACC reaches only 0.59 after 1000 evaluations. Taking 0.80 as the target utility, DT-ECN reduces the candidate evaluation cost by 94.4% compared with ByteTuning. It also outperforms the expert and default configurations, whose utilities are 0.55 and 0.42, respectively.

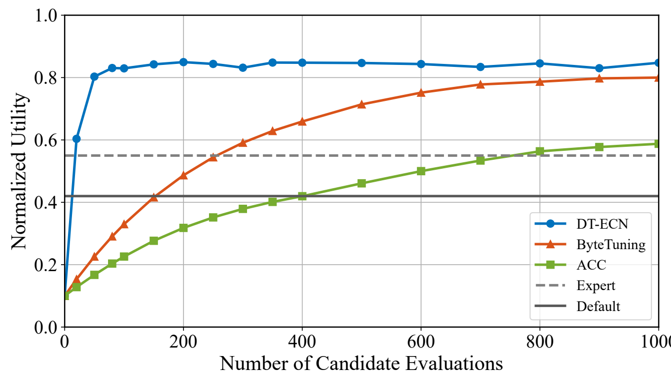


Fig. 3: Utility comparison of different methods under varying numbers of candidate evaluations.

C. Network Performance

We evaluate DT-ECN under 2-to-1, 8-to-1, and 16-to-1 Incast scenarios. Fig. 4 reports throughput, average queue length, P99 FCT, and PFC pause frames to characterize utilization, queueing, tail latency, and PFC risk.

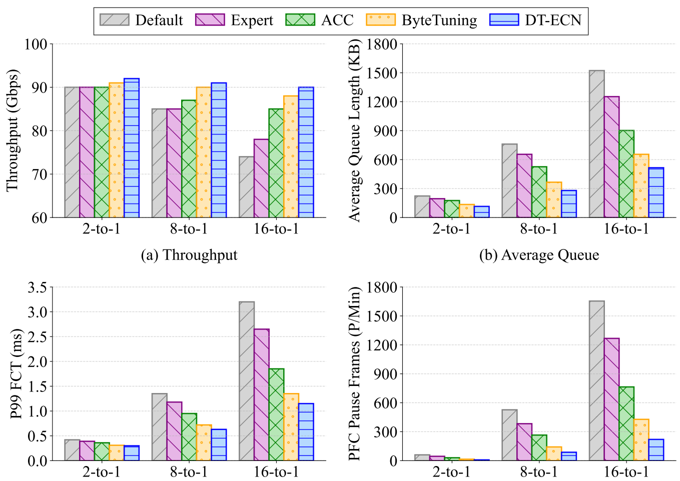
Fig. 4 (a) shows the throughput comparison. Under light Incast traffic, all methods achieve relatively high throughput. As the fan-in degree increases, the throughput of the default and expert configurations drops more significantly because static ECN parameters cannot adapt well to stronger burst contention. DT-ECN maintains the highest throughput in all three scenarios. Compared with the default configuration, DT-ECN improves throughput by 2.2%, 7.1%, and 21.6% under 2-to-1, 8-to-1, and 16-to-1 Incast, respectively. The average throughput improvement is 10.3%, indicating that DT-ECN can better preserve link utilization under severe Incast congestion.

Fig. 4 (b) compares the average queue length. The queue length increases with the Incast fan-in degree for all methods, but DT-ECN consistently maintains the lowest queue buildup. Under 16-to-1 Incast, the average queue length is reduced from 1523 KB under the default configuration to 516 KB under DT-ECN. This result shows that the ECN parameters selected by DT-ECN can trigger congestion feedback earlier and prevent excessive queue accumulation.

Fig. 4 (c) presents the P99 flow completion time (FCT). Since tail latency is closely related to queue buildup and congestion feedback delay, the default and expert configurations suffer from much higher P99 FCT under high fan-in traffic. DT-ECN achieves the lowest P99 FCT across all scenarios. In the 16-to-1 Incast case, DT-ECN reduces P99 FCT from 3.20 ms to 1.15 ms compared with the default configuration.

On average, DT-ECN reduces P99 FCT by 58.2%, demonstrating its effectiveness in improving tail latency.

Fig. 4 (d) shows the number of PFC pause frames. PFC pause frames increase rapidly when queues approach or exceed the PFC threshold, especially under high fan-in Incast. DT-ECN generates the fewest PFC pause frames among all methods. Under 16-to-1 Incast, DT-ECN reduces PFC pause frames from 1654 P/Min to 220 P/Min compared with the default configuration. The average reduction reaches 85.9%, indicating that DT-ECN can effectively keep queues away from the PFC threshold and reduce the risk of PFC-triggered performance degradation.



(a) Throughput
(b) Average Queue
(c) P99 FCT
(d) PFC Pause Frames

Fig. 4: Network performance comparison under static Incast scenarios.

V. Conclusion

This paper proposed DT-ECN, a digital twin-based ECN parameter tuning method for RoCEv2 networks. To address the trade-off among search efficiency, runtime stability, and system cost in existing automatic parameter tuning methods, DT-ECN migrates most candidate search and performance evaluation processes to the digital domain. The physical domain uses INT to collect per-flow states, while the digital domain generates biased ECN candidates according to the traffic state and evaluates throughput, queue length, and PFC risk through fluid-model-based parallel simulation. A multi-objective utility function then selects the best configuration, which is deployed only when safety constraints are satisfied.

Compared with static configuration, local distributed tuning, and centralized online search methods, DT-ECN provides low online disturbance, high parallel evaluation capability, and explainable decisions. Future work will further validate the method in larger-scale real RoCEv2 clusters and explore joint tuning of ECN parameters, PFC thresholds, and DCQCN end-host parameters for more comprehensive automated RoCEv2 network operation.

References

[1] C. Guo, H. Wu, Z. Deng, G. Soni, J. Ye, J. Padhye, and M. Lipshteyn, "RDMA over Commodity Ethernet at Scale," in *SIGCOMM'16*, Florianopolis, Brazil, Aug 2016, pp. 202–215.

[2] A. Gangidi, R. Miao, S. Zheng, S. J. Bondu, G. Goes, H. Morsy, R. Puri, M. Riftadi, A. J. Shetty, J. Yang, S. Zhang, M. J. Fernandez, S. Gandham, and H. Zeng, "RDMA over Ethernet for Distributed AI Training at Meta Scale," in *SIGCOMM'24*, Sydney, NSW, Australia, Aug 2024, pp. 57–70.

[3] R. Mittal, A. Shpiner, A. Panda, E. Zahavi, A. Krishnamurthy, S. Ratnasamy, and S. Shenker, "Revisiting Network Support for RDMA," in *SIGCOMM'18*, Budapest, Hungary, Aug 2018, pp. 313–326.

[4] Y. Zhang, Q. Meng, C. Hu, and F. Ren, "Revisiting Congestion Control for Lossless Ethernet," in *NSDI'24*, Santa Clara, CA, USA, Apr 2024, pp. 131–148.

[5] Z. Wan, J. J. Zhang, H. Wei, Z. Jiang, X. Zhong, W. Wu, H. Zhou, T. Pan, and T. Huang, "RECC: Joint Congestion Control Based on RTT and ECN for High-Speed RDMA Networks," *Proceedings of the ACM on Networking*, vol. 2, no. CoNEXT4, pp. 1–18, Dec 2024.

[6] Y. Zhu, H. Eran, D. Firestone, C. Guo, M. Lipshteyn, Y. Liron, J. Padhye, S. Raindel, M. H. Yahia, and M. Zhang, "Congestion Control for Large-Scale RDMA Deployments," in *SIGCOMM'15*, London, United Kingdom, Aug 2015, pp. 523–536.

[7] R. Mittal, V. T. Lam, N. Dukkipati, E. Blem, H. M. G. Wassel, M. Ghobadi, A. Vahdat, Y. Wang, D. Wetherall, and D. Zats, "TIMELY: RTT-Based Congestion Control for the Datacenter," in *SIGCOMM'15*, London, United Kingdom, Aug 2015, pp. 537–550.

[8] Y. Li, R. Miao, H. H. Liu, Y. Zhuang, F. Feng, L. Tang, Z. Cao, M. Zhang, F. Kelly, M. Alizadeh, and M. Yu, "HPCC: High Precision Congestion Control," in *SIGCOMM'19*, Beijing, China, Aug 2019, pp. 44–58.

[9] G. Kumar, N. Dukkipati, K. Jang, H. M. G. Wassel, X. Wu, B. Montazeri, Y. Wang, K. Springborn, C. Alfeld, M. Ryan, D. Wetherall, and A. Vahdat, "Swift: Delay Is Simple and Effective for Congestion Control in the Datacenter," in *SIGCOMM'20*, Virtual Event, USA, Aug 2020, pp. 514–528.

[10] L. Tan, W. Su, Z. Zhang, J. Miao, X. Liu, and N. Li, "In-Band Network Telemetry: A Survey," *Computer Networks*, vol. 186, p. 107763, Feb 2021.

[11] Z. Chen, M. Zhang, G. Li, J. Cao, Y. Jing, M. Xu, R. Xie, H. Liu, F. Jiao, and X. Hu, "Paraleon: Automatic and Adaptive Tuning for DCQCN Parameters in RDMA Networks," in *ICNP'24*, Charleroi, Belgium, Oct 2024, pp. 267–278.

[12] S. Yan, X. Wang, X. Zheng, Y. Xia, D. Liu, and W. Deng, "ACC: Automatic ECN Tuning for High-Speed Datacenter Networks," in *SIGCOMM'21*, Virtual Event, USA, Aug 2021, pp. 384–397.

[13] X. Zhong, J. Zhang, Y. Zhang, Z. Guan, and Z. Wan, "PACC: Proactive and Accurate Congestion Feedback for RDMA Congestion Control," in *INFOCOM'22*, London, United Kingdom, May 2022, pp. 2228–2237.

[14] L. Tan, Z. Jiang, K. Liu, H. Wei, P. Huo, H. Shi, W. Zhang, and W. Su, "ByteTuning: Watermark Tuning for RoCEv2," *IEEE Transactions on Cloud Computing*, vol. 13, no. 1, pp. 303–320, 2025.

[15] L. Hui, M. Wang, L. Zhang, L. Lu, and Y. Cui, "Digital Twin for Networking: A Data-Driven Performance Modeling Perspective," *IEEE Network*, vol. 37, no. 3, pp. 202–209, May–Jun 2023.

[16] Y. Zhang, Y. Liu, Q. Meng, and F. Ren, "Congestion Detection in Lossless Networks," in *SIGCOMM'21*, Virtual Event, USA, Aug 2021, pp. 370–383.

[17] R. Ben-Basat, S. Ramanathan, Y. Li, G. Antichi, M. Yu, and M. Mitzenmacher, "PINT: Probabilistic In-Band Network Telemetry," in *SIGCOMM'20*, Virtual Event, USA, Aug 2020, pp. 662–680.

[18] K. Liu, J. J. Zhang, Z. Jiang, H. Wei, X. Zhong, L. Tan, T. Pan, and T. Huang, "Diagnosing End-Host Network Bottlenecks in RDMA Servers," *IEEE/ACM Transactions on Networking*, vol. 32, no. 5, pp. 4302–4316, 2024.

[19] Q. Zhang, K. K. W. Ng, C. W. Kazer, S. Yan, J. Sedoc, and V. Liu, "MimicNet: Fast Performance Estimates for Data Center Networks with Machine Learning," in *SIGCOMM'21*, Virtual Event, USA, Aug 2021, pp. 287–304.

[20] Z. Wang, X. Wan, L. Li, Y. Sun, P. Xie, X. Wei, Q. Ning, J. Zhang, and K. Chen, "Fast, Scalable, and Accurate Rate Limiter for RDMA NICs," in *SIGCOMM'24*, Sydney, NSW, Australia, Aug 2024, pp. 568–580.