ICN 2025

The Twenty-Fourth International Conference on Networks

ISBN: 978-1-68558-277-7

May 18 - 22, 2025

Nice, France

ICN 2025 Editors

Shintaro Mori, Fukuoka University, Japan

ICN 2025

Forward

The Twenty-Fourth International Conference on Networks (ICN 2025), held between May 18-22, 2025 in Nice, France, continued a series of events organized by and for academic, research and industrial partners.

We solicited both academic, research, and industrial contributions. We welcomed technical papers presenting research and practical results, position papers addressing the pros and cons of specific proposals, such as those being discussed in the standard fora or in industry consortia, survey papers addressing the key problems and solutions on any of the above topics short papers on work in progress, and panel proposals.

The conference had the following tracks:

- Communication
- Networking
- Advances in Software Defined Networking and Network Functions Virtualization
- Next generation networks (NGN) and network management
- Computation and networking
- Topics on Internet Censorship and Surveillance

We take here the opportunity to warmly thank all the members of the ICN 2025 technical program committee, as well as all the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to ICN 2025. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

We also thank the members of the ICN 2025 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that ICN 2025 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the field of networks. We also hope that Nice provided a pleasant environment during the conference and everyone saved some time to enjoy the historic charm of the city.

ICN 2025 Chairs

ICN Steering Committee
Pascal Lorenz, University of Haute Alsace, France
Eugen Borcoci, National University of Science and Technology Politehnica Bucharest, Romania
Muath Obaidat, City University of New York, USA
Shintaro Mori, Fukuoka University, Japan

ICN Publicity Chairs
José Miguel Jiménez, Universitat Politecnica de Valencia, Spain
Francisco Javier Díaz Blasco, Universitat Politècnica de València, Spain
Ali Ahmad, Universitat Politècnica de València, Spain

Laura Garcia, Universidad Politécnica de Cartagena, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

ICN 2025

Committee

ICN Steering Committee

Pascal Lorenz, University of Haute Alsace, France
Eugen Borcoci, National University of Science and Technology Politehnica Bucharest, Romania
Muath Obaidat, City University of New York, USA
Shintaro Mori, Fukuoka University, Japan

ICN 2025 Publicity Chairs

José Miguel Jiménez, Universitat Politecnica de Valencia, Spain
Francisco Javier Díaz Blasco, Universitat Politècnica de València, Spain
Ali Ahmad, Universitat Politècnica de València, Spain
Laura Garcia, Universidad Politécnica de Cartagena, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

ICN 2025 Technical Program Committee

Luis F. Abanto-Leon, Technische Universität Darmstadt, Germany
Qammer H. Abbasi, University of Glasgow, UK
Khelil Abdelmajid, Landshut University of Applied Sciences, Germany
Alireza Abdollahpouri, University of Kurdistan, Sanandaj, Iran
Ahmedin Mohammed Ahmed, FDRE Ministry of Innovation and Technology (MInT), Ethiopia
Francisco Airton Silva, Federal University of Piauí, Brazil
Sami Marzook Alesawi, King Abdulaziz University | Faculty of Computing and Information Technology at Rabigh, Saudi Arabia
Madyan Alsenwi, Kyung Hee University - Global Campus, South Korea
Reem Alshahrani, Kent State University, USA
Cristian Anghel, Politehnica University of Bucharest, Romania / Pentalog, France
Imran Shafique Ansari, University of Glasgow, Scotland, UK
Andrés Arcia-Moret, Xilinx, Cambridge, UK
Suayb S. Arslan, Boğaziçi University, Turkey
Mohammed A. Aseeri, King Abdulaziz City of Science and Technology (KACST), Kingdom of Saudi Arabia
Aishwarya Asesh, Adobe, USA
Michael Atighetchi, BBN Technologies, USA
Jocelyn Aubert, Luxembourg Institute of Science and Technology (LIST), Luxembourg
Marco Aurélio Spohn, Federal University of Fronteira Sul, Brazil
Omran Ayoub, Politecnico di Milano, Italy
Alvaro Barradas, University of Algarve, Portugal
Chemseddine Benkalfate, University of Tlemcen, Algeria
Luis Bernardo, NOVA University of Lisbon, Portugal
Robert Bestak, Czech Technical University in Prague, Czech Republic
Lucas Bondan, Research and Development Center in Information and Communication Technology (CTIC)

of the Brazilian National Research and Educational Network (RNP), Brazil
Eugen Borcoci, National University of Science and Technology POLITEHNICA Bucharest, Romania
Fernando Boronat Seguí, Universitat Politecnica de Valencia-Campus de Gandia, Spain
Radoslav Bortel, Czech Technical University in Prague, Czech Republic
Marilisa Botte, Federico II University of Naples, Italy
Christos Bouras, University of Patras, Greece
An Braeken, Vrije Universiteit Brussel, Belgium
Arslan Brömme, Vattenfall GmbH, Berlin, Germany
Claudia Canali, University of Modena and Reggio Emilia, Italy
Batyr Charyyev, Stevens Institute of Technology, USA
Aizaz Chaudhry, Carleton University, Canada
Hao Che, University of Texas at Arlington, USA
Marc Cheboldaeff, Cap Gemini, Germany
Adil Chekati, University of Tunis El Manar, Tunisia
Bo-Rong Chen, Google, USA
Jundong Chen, Dickinson State University, USA
Sixia Chen, Adelphi University, USA
Yitao Chen, Qualcomm, USA
Yuxuan Chen, Shandong University, China
Andrzej Chydzinski, Silesian University of Technology, Poland
Jorge Crichigno, College of Engineering and Computing | University of South Carolina, USA
Gregorio D'Agostino, ENEA, Italy
Monireh Dabaghchian, George Mason University, USA
Sofiane Dahmane, University of Laghouat, Algeria
Abdulhalim Dandoush, ESME-Sudria engineering school, France
Susumu Date, Cybermedia Center - Osaka University, Japan
Babu R. Dawadi, Tribhuvan University,Nepal
Glaucio H.S. de Carvalho, Brock University, Canada
Declan Delaney, University College Dublin, Ireland
Margot Deruyck, Ghent University - IMEC - WAVES, Belgium
Amir Djenna, University of Constantine, Algeria
Pengyuan Du, Facebook Inc., USA
Salahaldeen Duraibi, Jazan University, Saudi Arabia
Zakaria Abou El Houda, University of Montreal, Canada
Basem ElHalawany, Shenzhen University, China / Benha University, Egypt
Gledson Elias, Federal University of Paraíba (UFPB), Brazil
Levent Ertaul, California State University, East Bay, USA
Davide Ferraris, University of Malaga, Spain
Mário Ferreira, University of Aveiro, Portugal
Adriano Fiorese, Santa Catarina State University (UDESC), Brazil
Mathias Fischer, Universität Hamburg, Germany
Edelberto Franco Silva, Universidade Federal de Juiz de Fora, Brazil
Valerio Frascolla, Intel Deutschland GmbH, Neubiberg, Germany
Marco Furini, University of Modena and Reggio Emilia, Italy
Yu Gao, University of St. Thomas, USA
Yun Gao, Nanjing University of Posts and Telecommunications, China
Gourab Ghatak, IIIT-Delhi, India
Saptarshi Ghosh, London South Bank University, UK

Marco Giordani, University of Padova, Italy
Rita Girao-Silva, University of Coimbra & INESC Coimbra, Portugal
Srikrishna Gopu, Meta, USA
Florian Grießer, Technical University of Munich, Germany
Karl-Johan Grinnemo, Karlstad University, Sweden
Shay Gueron, University of Haifa / Amazon Web Services, Israel
Tina Gui, Anheuser-Busch InBev, Belgium
Brij B. Gupta, Asia University, Taiwan
Tibor Gyires, Illinois State University, USA
Muhammad Hanif, Hanyang University / Seoul National University of Science and Technology, South Korea
Luoyao Hao, Columbia University, USA
Esteve J. Hassan, Mohawk College of Applied Arts and Technology, Canada
William "Chris" Headley, Virginia Tech National Security Institute | Virginia Polytechnic Institute & State University, USA
Enrique Hernández Orallo, Universitat Politecnica de Valencia, Spain
M. Reza Hoseinyfarahabady, University of Sydney, Australia
Md Shafaeat Hossain, Southern Connecticut State University, USA
Chiu-Han Hsiao, Academia Sinica, Taiwan
Wen-Chen Hu, University of North Dakota, USA
Xinyue Hu, University of Minnesota, Twin Cities, USA
Fatima Hussain, Ryerson University / Royal Bank of Canada, Toronto, Canada
Dragos Ilie, Blekinge Institute of Technology (BTH), Sweden
Gal Itzhak, Technion - Israel Institute of Technology, Israel
Faouzi Jaidi, University of Carthage - Higher School of Communications of Tunis & National School of Engineers of Carthage, Tunisia
Yong Jin, Tokyo Institute of Technology, Japan
Omprakash Kaiwartya, Nottingham Trent University, UK
Faouzi Kamoun, ESPRIT School of Engineering, Tunisia
Kyungtae Kang, Hanyang University, Korea
Binayak Kar, National Taiwan University of Science and Technology, Taiwan
Erdem Karayer, Ege University, Turkey
Kallol Krishna Karmakar, University of Newcastle, Australia
Andrzej Kasprzak, Wrocław University of Science and Technology, Poland
Sokratis K. Katsikas, Norwegian University of Science and Technology, Norway
Roya Khanzadeh, Johannes Kepler University (JKU) Linz, Austria
Hakima Khelifi, Beijing Institute of Technology, China
Pinar Kirci, Bursa Uludag University, Turkey
Anne Josiane Kouam, TU Berlin, Germany
Sondes Ksibi, University of Carthage | Higher School of Communications of Tunis, Tunisia
Venkataramani Kumar, University of Puerto Rico, Mayaguez, USA
Rafael Kunst, University of Vale do Rio dos Sinos (UNISINOS), Brazil
Christo Kurisummoottil-Thomas, Eurecom, France
Mohammed Laroui, Djillali Liabes University, SBA, Algeria & Paris University, France
Vincent Latzko, Technische Universität Dresden, Germany
Riccardo Lazzeretti, Sapienza University of Rome, Italy
Piotr Lechowicz, Wroclaw University of Science and Technology, Poland
Chi-Han Lee, Academia Sinica, Taiwan

Agnieszka Piotrowska, Silesian University of Technology, Poland
Ravi Prakash, TU Delft, Netherlands
Cong Pu, Marshall University, USA
Abdellatif Rahmoun, Ecole Supérieure en Informatique, Sid Bel-Abbes, ESI-SBA, Algeria
Shankar Raman, Indian Institute of Technology Madras, India
Kurdman Rasol, Universitat Politècnica de Catalunya (UPC), Spain
Adib Rastegarnia, Purdue University, USA
Claudina Rattaro, Universidad de la República, Montevideo, Uruguay
Danda B. Rawat, Howard University, USA
Yenumula B. Reddy, Grambling State University, USA
Ghaya Rekaya, Telecom Paris, France
Eric Renault, IMT-TSP, France
Filip Rezabek, Technical University of Munich, Germany
Farhad Rezazadeh, Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Spain
Imad Rida, University of Technology of Compiegne, France
Elisa Rojas, University of Alcala, Madrid, Spain
Gerardo Rubino, INRIA, Rennes, France
Rukhsana Ruby, Shenzhen University, China
Marina Ruggieri, University of Roma Tor Vergata, Italy
Abdulhakim Sabur, Arizona State University, USA
Amit Samanta, IIT Kharagpur, India / Max Planck Institute for Software Systems, Germany
Rodrigo Sanches Miani, Universidade Federal de Uberlândia, Brazil
Masahiro Sasabe, Graduate School of Science and Technology - Nara Institute of Science and Technology, Japan
Jaimin Shah, University of Minnesota, Twin Cities, USA
Qi Shi, Liverpool John Moores University, UK
Yuankun Shi, Intel, China
Megumi Shibuya, The University of Electro-Communications, Japan
Yifei Song, Virginia Tech, Blacksburg, USA
Kostas Stamos, University of Patras, Greece
Cristian Lucian Stanciu, National University of Science and Technology POLITEHNICA Bucharest, Romania
Giovanni Stanco, University of Naples 'Federico II', Italy
Prasad Talasila, Aarhus University, Denmark
Sudeep Tanwar, Institute of Technology | Nirma University, Ahmedabad, India
Giorgio Terracina, Università della Calabria, Italy
Vasileios Theodorou, Intracom Telecom, Greece
Eirini Eleni Tsiropoulou, Arizona State University, USA
Abu Barkat Ullah, University of Canberra, Australia
Dalton C. G. Valadares, IFPE, Brazil
Rob van der Mei, Centre for Mathematics and Computer Science (CWI), Amsterdam, Netherlands
Costas Vassilakis, University of the Peloponnese, Greece
Quoc-Tuan Vien, Middlesex University, UK
César Viho, IRISA - ISTIC/Université Rennes 1, France
Calin Vladeanu, National University of Science and Technology POLITEHNICA Bucharest, Romania
Dmitriy Volkov, eQualit.ie, Canada
Xianzhi Wang, University of Technology Sydney, Australia
Bernd E. Wolfinger, University of Hamburg, Germany
Longfei Wu, Fayetteville State University, USA

Table of Contents

ECN Works Better for Selective Dropping in High Bandwidth-Delay Product Connections

Mahmoud Bahnasy and Ali Munir
Huawei Technologies Ltd.
Waterloo, Canada
e-mail: ali.munir@huawei.com

Junjie Niu, Zhibo Yan, and Peng Dong
Huawei Technologies Ltd.
Nanjing, China

Yashar Ganjali
University of Toronto.
Toronto, Canada
e-mail: ganjali7@cs.toronto.edu

***Abstract*—Modern Internet services and applications distribute data and compute on different geographically distributed data centers to improve performance and reliability. For example, content providers distribute their data centers among different regional areas and periodically replicate the content between data centers. Moreover, Geo-Distributed Machine Learning (Geo-DML) is emerged to satisfy the need to train large sophisticated models. Such a new model of training requires a very robust and reliable inter-data center transport layer. Such a design raises the need for high-speed reliable inter-data center transport. The conservative reaction of the Transport Control Protocol (TCP) in such long Round-Trip Time (RTT) connections cause huge degradation in the throughput. For example, TCP is able to utilize only 40% of a 10-Gbps link between two data centers with $RTT = 10ms$ and it gets even worse when packet loss occurs. We present Data Center Interconnect-Bridging (DCIB) as a method for recovering lost packets in such a scenario by allowing the router to drop pre-selected packets that can easily recovered by the edge routers without triggering TCP's reaction at the host. Several simulation experiments show that DCIB can increase link utilization by a factor of 6x in highly congested connections, and reduce flow completion time by up to 85%.**

Keywords-Inter-Data Centers Communication; Cross Data Centers Communications; Transport Protocol; Congestion Control.

I. INTRODUCTION

Data Center Interconnect (DCI) technology connects two or more data centers together over short, medium or long distances using high-speed leased lines or the Wide Area Network (WAN) (Figure 1). Data Centers (DC) often serve many geo-distributed applications requiring huge amounts of data transfers between the data centers. For example, in applications such as astronomy, storage nodes are often separated from compute nodes in geographically distributed data centers. Such an approach allows astronomers to collect data close to the origin where telescopes are located while transferring the data to remote data centers for low-cost processing. The performance of the inter-DC traffic relies on the quality of the network path and the underlying transport protocol.

DCI imposes significant challenges in designing the transport layer because of the distinct characteristics of the routers and switches along the datapath. First, there is no administrative control available at the switches and routers along the path. Therefore it is hard to provide any performance guarantees. Second, bursty behavior of Transmission Control Protocol (TCP) and synchronization between flows can cause packet loss with no congestion (i.e., false congestion signal) which triggers rate reduction that takes a few Round Trip Time (RTT)s to recover. Finally, there is no explicit congestion signal available. Hence, it is very difficult to know the location and reason for poor performance in the network.

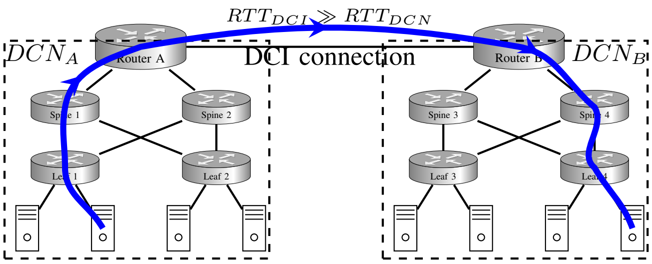


Figure 1. A typical DCI topology

Existing congestion control mechanisms leverage packet loss [1], Explicit Congestion Notification (ECN) [2][3] or delay [**timely**][4], or a combination of them [5] as congestion signals. However, congestion signals, such as ECN and RTT, get delayed because of the long latency of DCI connections. Moreover, traditional TCP reaction to packet loss significantly degrades the application performance. For example, TCP may react aggressively to packet loss, yet at the same time it probes for available bandwidth very carefully to minimize packet loss in the network. Because of the large Bandwidth-Delay Product (BDP), TCP reacts slowly to congestion, and it recovers slowly available bandwidth which hinder achieving full link utilization. For that reason, DCI links are usually underutilized and can only achieve less than 40% of its link capacity, on average, even with zero packet loss (Review Figure 4).

In this work, we provide a method, Data Center Interconnect-Bridging (DCIB), to minimize the impact of packet loss on TCP, to maintain high throughput, and to achieve higher link utilization. DCIB is not a new congestion control mechanism. In fact, it can be considered as a plugable that can be integrated into any TCP protocol. The key idea here is to handle packet loss and packet retransmission inside the network itself, without triggering rate reduction and packet retransmission at the host, which results in throughput degradation. To that end, DCIB adds two main features to the edge routers. First, it selectively marks the ECN bit of the outgoing packets. The routers along the DCI path use this ECN mark for selective packet drop instead of randomly dropping data packets. Although, we can't modify the programmability of routers across DCI path, DCIB only requires enabling a simple feature, such as ECN, which can be enabled as part of the SLA agreement. Second, the edge router is responsible for retransmitting lost packets. For this purpose, it stores the marked packets in a local buffer and then retransmits them upon detecting packet loss.

The main innovation is that DCIB presents a new way of using ECN, i.e., not to convey the congestion, but to inform routers to drop, in case of congestion, certain packets that can be recovered fast by the edge routers without triggering the conservative behavior of TCP at the end host. Hence, DCIB provides faster recovery from lost packets, while keeping the control loop short (between routers). This flow control mechanism aims to resolve transient congestion that is caused by flow synchronization and/or the bursty behavior of TCP. If edge routers can't recover packet loss either because the number of packet loss is high, or the congestion is long-lasting, normal TCP behavior is triggered at the hosts.

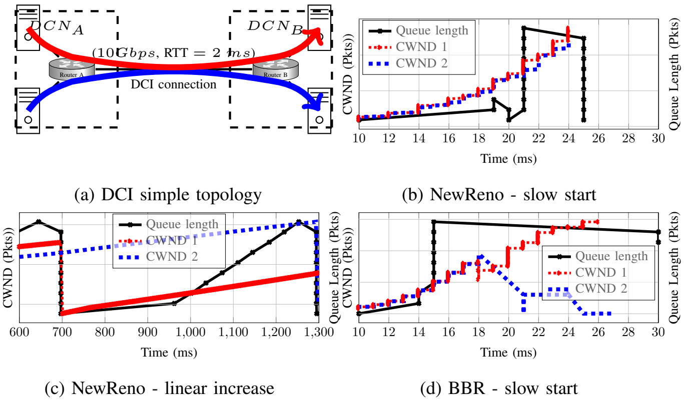
The rest of the paper is organized as follows; Section II presents the motivation behind this work. The design of our proposed protocol is discussed in Section III. Section IV includes the implementation details of our protocol and the experimental results. Several discussion points regarding the design of DCIB is discussed in Section V. We conclude the paper in Section VI.

II. Motivation

The challenge that congestion control mechanisms face in the large BDP scenario is the long control loop. Unfortunately, this issue is intrinsic to the physical property of the system. Hence, building a new congestion control based on traditional congestion signals, such as packet loss, delay, or ECN marking, is not going to help too much. For example in an ideal system where all network components participate in resolving congestion, when a switch detects congestion and starts signalling the congestion by ECN-marking data packets, assuming it knows exactly which flow is the culprit flow. Such a signal needs at least one RTT (i.e., a few ms) to convey the signal plus the reaction time the sender requires. One can conclude that by the time the host reacts to the received congestion signal, the congestion might have already been resolved. [6] shows experimentally that DCI average throughput drops by 18%-37% as buffer decreases. It also demonstrates that packet loss occurs in DCI reaches 5% for both TCP Cubic and Bottleneck Bandwidth and Round-trip propagation time (BBR) even with deep buffers (i.e., 50 MBytes).

A. Illustrative Example

To demonstrate the impact of unpredictable packet loss and delayed signal on TCP NewReno and BBR [5], we conduct a simulation using Ns-3 [7]. We consider two Data Center Network (DCN) networks connected using a dedicated link of capacity 10 $Gbps$ and latency 2 ms (Figure 2a). In this simulation, we start two long-lived flows from DCN_A towards DCN_B. Figure 2b shows the Congestion Window (CWND) and the queue length for the two protocols in this simple scenario. One can clearly see the slow start behavior of TCP NewReno where CWND is increased exponentially with every received ACK after one RTT (at $t = 10\ ms,\ 12\ ms, 14\ ms,\ \ldots\ etc$). Because RTT is larger than the time needed to transmit CWND worth of data, the two flows behave in an on-off manner. One can notice that although the queue reached the maximum capacity at $t = 21ms$ and a packet loss occurs, the two flows keep increasing their CWND with the arrival of every acknowledgment received in the next cycle and do not recognize the packet loss signal till $t > 25\ ms$.



(a) DCI simple topology

(b) NewReno - slow start

(c) NewReno - linear increase

(d) BBR - slow start

Figure 2. TCP reaction is slow and might unnecessarily reduce CWND even after congestion disappeared.

Similarly, Figure 2c depicts the same behavior when the two flows reach the end of the linear increase phase. It shows that Flow 1 (depicted in red) reacted to the first congestion at $t = 650\ ms$ after $\approx 50ms$ (i.e., 25 RTTs). The same occurs with Flow 2 at $t = 1250\ ms$ (depicted in blue).

Similar behavior was observed for the delay-based congestion control (BBR). Figure 2d demonstrates that BBR also reacts late to congestion signal and keeps increasing the CWND of the two flows even when the buffer size reaches the maximum allowed value at $t = 15ms$ causing higher packet loss rate. Figure 2d depicts that flow 1 and flow 2 do not start reducing their CWND till $t = 18\ ms$ and $t = 28\ ms$, respectively.

To understand the impact of large BDP on TCP delayed reaction we run the same experiment while increasing the DCI link latency (illustrated Figure 3 by RTT). One can notice that as the link latency increases, which increases BDP, the throughput of TCP protocol (i.e., NewReno, Cubic and BBR TCP) degrades drastically.

Furthermore, Figure 4 illustrate the impact of packet loss on TCP throughput. We vary the packet loss from 0‰ to 3‰ and plot the throughput of TCP NewReno, Cubic and BBR. We can observe that as the loss rate increases, the throughput drops exponentially. We conclude that TCP by itself can't maintain high throughput in such an environment because of the long control loop.

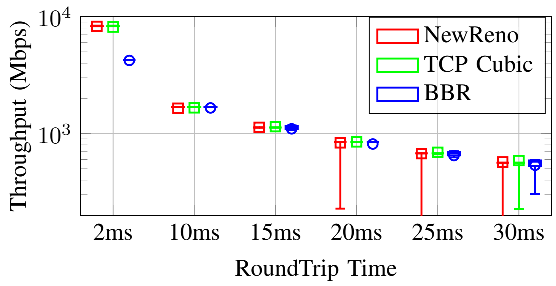


Figure 3. Large BDP greatly degrades the performance of TCP (The average is represented by circles).

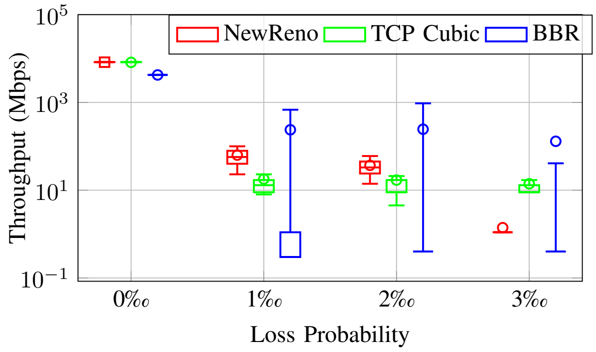


Figure 4. Packet loss on DCI link reduces throughput significantly (The average is represented by circles).

III. Design

We propose DCIB as a solution that helps mitigating the effect of transient congestion and packet loss while allowing TCP to react to long-lasting congestion.

The key idea behind DCIB is to handle packet loss and packet retransmission inside the network itself. To achieve this goal, DCIB leverages the existing feature in the routers and switches (i.e., ECN) for handling packet loss in the DCI, at the egress of the data center (at the border routers) for packet retransmission within the network. DCIB works as extension for any TCP to allow faster retransmission. Thus, hosts can maintain a high transmission rate while preventing rate reduction due to transient congestion. It also avoids waiting for three duplicate ACKs or the expiration of Retransmission Timeout (RTO) timer to trigger the retransmission which also accelerates the recovery process.

A. Selective Marking at the Source Edge Router

DCIB uses ECN for selective packet drop in case of congestion instead of congestion notification. DCIB allows edge routers to uniformly select a few packets that can be dropped in the network. The intuition here is that the edge routers can store and retransmit these packets if a packet drop is detected in the network. For this purpose, the ECN bit is set on all the outgoing packets except for the packets that are selected for drop. For flows going from router A to router B, router A selects or recommends the drop packets ratio within the range $[R - r, R]$ where R is the current transmission rate and r is the drop recommendation rate which is taken to be

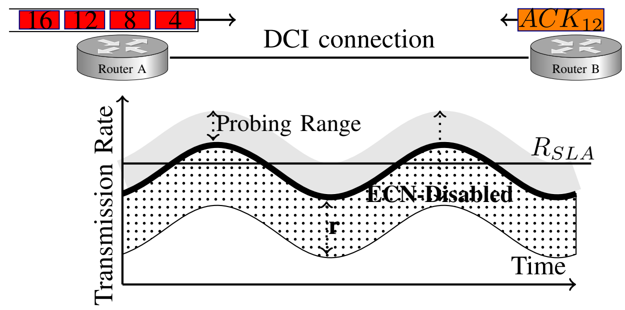


Figure 5. DCIB work mechanism

equal $0.1 \cdot R$ in our experiments (depicted by dotted area in Figure 5).

In addition, Router A clones these unmarked packets and stores them in a separate queue/buffer for retransmission in case of a packet loss. We call such a queue as stalled queue. Router A detect which packet was received/lost by comparing the received router-to-router ACK (explained later) with the head of the stalled queue. Therefore, Router A can retransmit packets, from the head of the stalled queue, for any packet for which an ACK is not received. We assume that there's one datapath between the two data centers. For multiple data paths, Router A can use one queue per path, however, we leave studying the effect of multiple data paths for future work. Figure 5 illustrate an example when Router A receives an ACK for packet 12, it indicates that all packets before packet 12 were lost, and must be retransmitted. Therefore, Router A starts retransmitting packets from the stalled queue till it reaches packet 12 which gets discarded (as it has been acknowledged) and the queue gets paused again until receiving another ACK. Such behavior can be carried out using PFC pause frames.

Moreover, Router A can probe for extra bandwidth by disabling ECN for few extra packets above the transmission rate (R) or the agreed-upon Service-Level Agreement (SLA) rate (R_{sla}) (depicted by gray area in Figure 5). These injected packets are used as probes which allows TCP to detect available bandwidth very quickly. In this paper, we did not consider changing TCP, hence, we leave that for future work.

B. Selective Packet Drop at Intermediate Routers

Intermediate routers between Router A and Router B perform the traditional Random Early Detection (RED) [8] process, or use Low Latency, Low Loss, and Scalable Throughput (L4S) [3] to drop ECN-disabled packets and mark for congestion ECN-enabled packets. Such a feature can be specified in the SLA agreement.

DCIB requires intermediate routers to drop selected packets in case of congestion which can be achieved by enabling RED [9] with ECN [10] or L4S [3].

We propose using such feature in a different way. We propose setting all packets to "ECN Capable" except selected packets that can be recovered at the edge router. Within the context of this paper, we also call the selected packets "ECN-Disabled" packets. In case of congestion (Average Qlen >

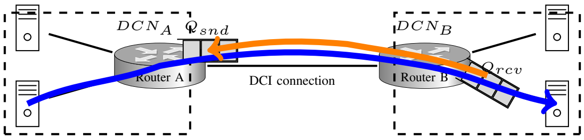


Figure 6. Queue Disciplines requirements

Threshold), The default behavior of RED is to mark "ECN Capable" packets for congestion and drop ECN-Disabled packets. By scarifying the selected packets that can be recovered fast at the edge router, we maintain stable throughput for TCP, hence, high resource utilization.

C. Router-to-Router ACK

At Router B, a Router-to-Router ACK is generated per ECN-disabled packet received. The intuition here is to act as a local receiver, and acknowledge a packet reception. Note that DCIB is designed to handle packet loss on the DCI links only, therefore we generate the ACKs at the destination edge router.

D. DCIB Internals

a) ***Maximum allowed Packet Drop Rate*** *(r):* The maximum packet drop can be calculated as $r \leq B/(\tau \times C)$, where B is dedicated stalled buffer capacity, τ is the inter-router round-trip time, and C is the link capacity. E.g., for a DCI link capacity of $C = 10\ Gbps$, $\tau = 10\ ms$ ($2,000Km$ distance) and edge routers' buffer sizes are $2\ MB$ each. We can calculate the maximum drop rate that DCIB mechanism can support as $r = (2e^6 \times 8bit)/(10e^{-3}s \times 10e^9 bps) = 0.16$. Hence, DCIB can recover up to 16% of packet loss without interrupting TCP protocol.

b) ***Queue Disciplines at the edge routers****:* Figure 6 shows the architecture of the Queue disciplines required at the edge routers. DCIB requires two types of queue disciplines; namely transmitting queue (Q_{snd}) and receiving queue (Q_{rcv}).

The algorithm of the sending process of Q_{snd} is illustrated in Figure 7. It calculates the selection rate (Line 1), modify ECN field in the IP header (Line 2), and clone the packet if selected (Lines 3-5). Figure 8 represents the receiving function at Q_{snd} (in Router A in our example). It reacts to received ACK from Router B by checking if it matches the head of the queue (Line 2). If a match is not found, it resumes transmission on the stalled queue until it reaches a packet that matches the received ACK (Lines 3-5). Otherwise, it drops the packet that matches the received ACK (Line 6). On the other hand, the process carried out at Router B, inside (Q_{rcv}), is depicted in Figure 9. Q_{rcv} verifies if the packet is selected for drop (ECN-disabled), it generates an ACK for each successfully received packet (Lines 1-3). Finally, it forwards the packet as normal towards its destination (Line 4).

IV. Evaluation

In this section, we illustrate the benefits of DCIB on transport performance using TCP Reno, TCP Cubic and BBR protocols.

A. DCIB with artificial loss:

In this experiment, we simulate two DCN network connected using a 10-Gbps DCI link. Link latency for all link inside each DCN is $5\mu s$. Link latency of the DCI link is $3\ ms$ which is equivalent to a 600-Km link between the two data centers. We start 15 long-lived flows from all hosts of DCN A towards DCN B. The simulation topology is depicted in Figure 10a. DCIB marks 10% of the traffic with ECN-disabled, and allows the rest to go through with ECN-enabled. In addition, we simulate congestion and packet loss in the WAN by artificially dropping 5% of the packets. We repeat the simulation twice, once with the assumption that WAN is not cooperative and they drop packets equally; i.e., 5% from both ECN-enabled and ECN-disabled traffic. In the second run, we imitate a cooperative WAN and drop 50% of the 10% ECN-disabled traffic only (i.e., 5%).

We demonstrate the performance by measuring the overall throughput at the DCI link. The results shown in Figure 10b and 10c illustrate the DCI throughput while using DCIB for both TCP Reno and TCP Cubic. It shows that DCIB enhances the performance by increasing the average throughput up to 6x for TCP Reno and up to 4.5x for TCP Cubic when packet drop takes place in ECN-disabled packets only. Moreover, when WAN drops packets regardless of the ECN marking, DCIB was able to enhance TCP throughput by 4x and 3.24x for TCP Reno and TCP Cubic, respectively. Figure 10d and 10e depict the same remarks by showing the Cumulative Distribution Function (CDF) of the DCI throughput.

B. DCIB with no artificial loss:

In this experiment we demonstrate the effect when packet loss only occurs in case of contention among high number flows on the limited buffer resources. To validate that, we repeat the previous simulation with the same number of long lived flows with 0% artificial drop probability. Figure 11a, 11b and 11c show that DCIB can greatly alleviate the effect of congestion-based packet loss for TCP Reno, Cubic and BBR, respectively. Even with a dedicated link for the DCI traffic, DCIB can enhance the average throughput up to 1.24x, 3.6x, and 2.4x for TCP Reno, Cubic and BBR, respectively. Although DCIB enhances the average and the median throughput for BBR by 1.85x and 2.4x, it did not enhance much the 99-

Data: Packet p
$r \leftarrow$ drop ratio;
Set $p.ECN$ on all packets except r percentage;
if $p.ECN != 0$ **then**
  `/* Packet not selected for drop */`
  Clone packet;
  Store at stalled queue;
end
Transmit packet p;

Figure 7. Packet processing at Q_{snd}

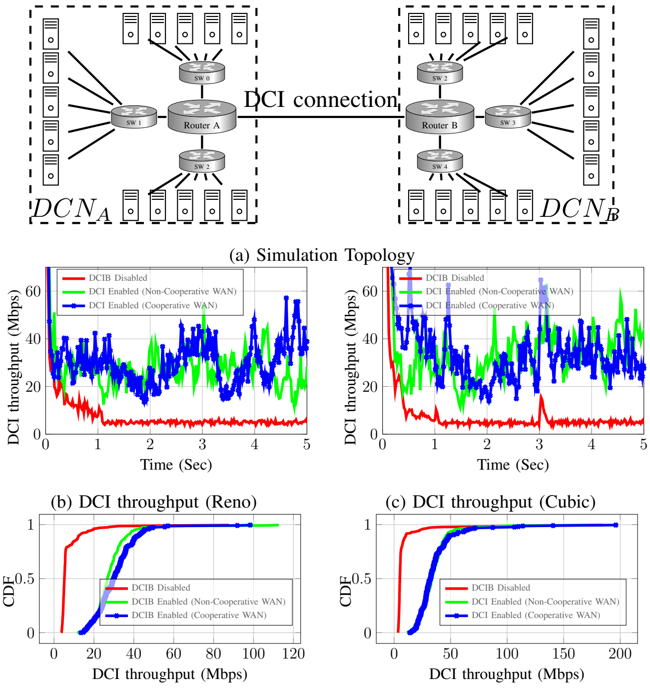
 ISBN: 978-1-68558-277-7

Data: Router-ot-Router ACK ack
Data: Stalled Queue $Q_{stalled}$
$head \leftarrow Q_{stalled}[0]$;
while $head.header! = ack.header$ **do**
    `/* Received ACK does not match head of the queue */`
    Transmit packet $head$;
    Wait for packet transmission;
    Drop $head$;
end
Drop $head$;
; `// When ACK matches the head of the queue; drop the head of the queue`
Return;

Figure 8. ACK processing at Q_{snd}

Data: Packet p
if $p.ECN == 0$ **then**
    `/* Packet marked for drop */`
    Generate Router-to-Router ACK ;
    Send Ack to Router A;
end
Transmit packet p;

Figure 9. Packet processing at Q_{rcv}



(a) Simulation Topology

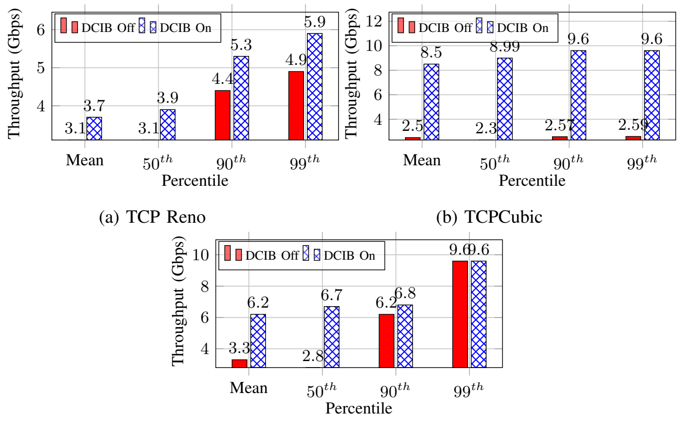
(b) DCI throughput (Reno)

(c) DCI throughput (Cubic)

(d) Throughput CDF (Reno)

(e) Throughput CDF (Cubic)

Figure 10. DCIB increases the average throughput by 4x and 6x when packet drop occurs in all packets or in selected packets only.

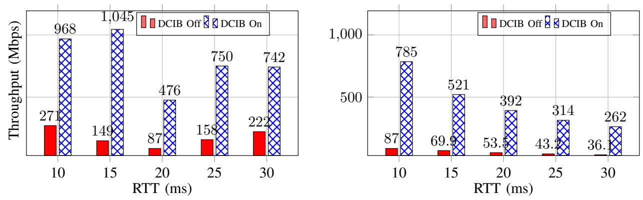


(a) TCP Reno

(b) TCPCubic

(c) TCPBBR

Figure 11. With no artificial loss, DCIB increases the average throughput by 24% in average.



(a) Cubic

(b) BBR

Figure 12. 5x and 7x higher throughput compared to TCP Cubic and BBR.

percentile. The main reason is BBR ignores to a certain extent packet loss to achieve high throughput.

C. DCIB with Limited Buffer Routers:

Because some edge router might have limited buffer capacity, we also explored using a low-priority buffer to store the selected packets while transmitting them when the main queue is idle. Hence, the buffer requirements are expected to be lower as long as the average transmission rate is lower than the link capacity. Moreover, we configure the end-to-end RTT to represent different sets of networks with different RTT values starting from $10ms$ up to $30ms$. Figure 12a and 12b demonstrate that DCIB enhances the average throughput of TCP Cubic and TCP BBR by 5x and 7x, respectively.

D. DCIB effect on FCT:

We also repeated the same experiment while generating 1000 flows using the characteristic of web search workload [2]. Flows start time is generated using Poisson distribution to generate an average load equal 80%. The base TCP variant used in this simulation is TCP Cubic. Figure 13 illustrates that DCIB reduces Flow Completion Time (FCT) by up to 88%.

V. Discussion

Performance-Enhancing Proxy (PEP) proposed terminating TCP connections at edge routers to alleviate the issue of DCI inter connectivity [11]. In such an approach, hosts do not need

 ISBN: 978-1-68558-277-7

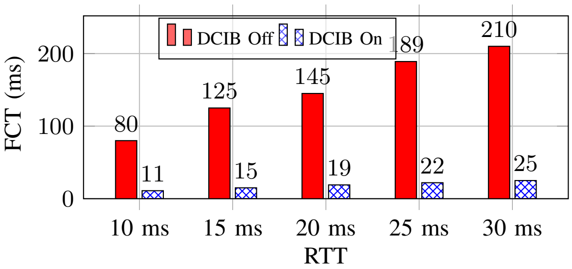


Figure 13. DCIB with stochastic retrans (FCT reduced by 88%)

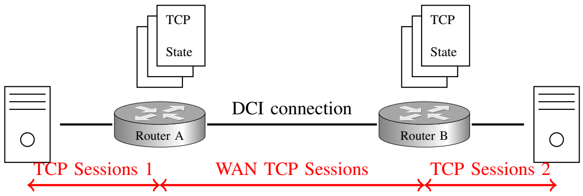


Figure 14. Router A terminates TCP connections and open new connections towards Router B; Router B reestablishes TCP connections towards clients

to wait for the end-to-end ACK to control transmission. However, edge routers need, not only to store all inflight packets but also to keep the state for each connection both to the hosts and to the other edge routers (Figure 14). In addition, managing these connections encounters more delay in processing the whole TCP stack twice (inbound and outbound at the Routers). It would get even more complex for retransmission and buffer management. On the other hand, DCIB does not need to keep any per-connection state, and the buffer requirement is very low and depends on the congestion rate of the communication channel, not the number of connections.

VI. Conclusion

In this paper, we present DCIB to enhance the performance of inter data center communication without changing the TCP protocol at the host. DCIB adds a new method of using ECN marking to selectively drop certain packets instead of sending congestion notification. This allows intermediate routers to drop recommended packets that can easily be recovered at the edge routers without interrupting TCP at the hosts. Our experimental results show that DCIB can increase the DCI throughput by a factor of 6x , 4x, and 7x for both TCP Reno, Cubic, and BBR, respectively while not changing the hosts. In addition, DCIB was able to reduce FCT up to 67%, 85%, and 88% for TCP Reno, Cubic, and BBR, respectively.

Defining an interaction between DCIB and the TCP should allow the hosts to increase their transmission rate even faster by allowing DCIB to select extra packets for drop as probes. Such behavior is left for future work. In addition, intercepting duplicate ACKs at the edge router and retransmitting packets that can be recovered at the edge router is also left for future work.

References

[1] S. Ha, I. Rhee, and L. Xu, "Cubic: A new tcp-friendly high-speed tcp variant," *SIGOPS Oper. Syst. Rev.*, vol. 42, no. 5, pp. 64–74, Jul. 2008, ISSN: 0163-5980. DOI: 10.1145/1400097.1400105. [Online]. Available: https://doi.org/10.1145/1400097.1400105.

[2] M. Alizadeh *et al.*, "Data center tcp (dctcp)," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 4, pp. 63–74, Aug. 2010, ISSN: 0146-4833. DOI: 10.1145/1851275.1851192. [Online]. Available: https://doi.org/10.1145/1851275.1851192.

[3] B. Briscoe, K. D. Schepper, M. Bagnulo, and G. White, *Low Latency, Low Loss, and Scalable Throughput (L4S) Internet Service: Architecture*, RFC 9330, Jan. 2023. DOI: 10.17487/RFC9330. [Online]. Available: https://www.rfc-editor.org/info/rfc9330.

[4] G. Kumar *et al.*, "Swift: Delay is simple and effective for congestion control in the datacenter," in *Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication*, ser. SIGCOMM '20, Virtual Event, USA: Association for Computing Machinery, 2020, pp. 514–528, ISBN: 9781450379557. DOI: 10.1145/3387514.3406591. [Online]. Available: https://doi.org/10.1145/3387514.3406591.

[5] N. Cardwell *et al.*, "Bbrv2: A model-based congestion control performance optimization," in *Proc. IETF 106th Meeting*, 2019, pp. 1–32.

[6] G. Zeng, J. Qiu, Y. Yuan, H. Liu, and K. Chen, "Flashpass: Proactive congestion control for shallow-buffered wan," in *2021 IEEE 29th International Conference on Network Protocols (ICNP)*, 2021, pp. 1–12. DOI: 10.1109/ICNP52444.2021.9651988.

[7] "Network simulations with the ns-3 simulator," 2008, [Online]. Available: https://www.nsnam.org/ (visited on 07/10/2023).

[8] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on networking*, vol. 1, no. 4, pp. 397–413, 1993. DOI: 10.1109/90.251892.

[9] A. Gurtov, T. Henderson, and S. Floyd, *The NewReno Modification to TCP's Fast Recovery Algorithm*, RFC 3782, Apr. 2004. DOI: 10.17487/RFC3782. [Online]. Available: https://www.rfc-editor.org/info/rfc3782.

[10] S. Floyd, D. K. K. Ramakrishnan, and D. L. Black, *The Addition of Explicit Congestion Notification (ECN) to IP*, RFC 3168, Sep. 2001. DOI: 10.17487/RFC3168. [Online]. Available: https://www.rfc-editor.org/info/rfc3168.

[11] J. Griner, J. Border, M. Kojo, Z. D. Shelby, and G. Montenegro, *Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations*, RFC 3135, Jun. 2001. DOI: 10.17487/RFC3135. [Online]. Available: https://www.rfc-editor.org/info/rfc3135.

Network-Performance Evaluation for Millimeter-Wave Information-Centric Wireless-Sensor-Network Ecosystem in Actual City

Shintaro Mori
Department of Electronics Engineering and Computer Science
Fukuoka University
8-19-1 Nanakuma, Jonan-ku, Fukuoka 814-0180, Japan
E-mail: smori@fukuoka-u.ac.jp

***Abstract*—This paper presents an evaluation of our research team previously proposed information-centric wireless-sensor-network-based ecosystem for smart-city applications, called mmICWSN. This ecosystem uses millimeter-wave communications for future broadband wireless networks. To demonstrate mmICWSN's feasibility in an actual city, the network performance of our ecosystem was evaluated, including throughput in the application and network layers. The experiment was conducted in Nogata City (Fukuoka, Japan), connecting two 1-km-distant locations. The results indicate that our ecosystem can be verified to communicate with a point-to-point environment.**

Keywords-information-centric wireless sensor network; millimeter-wave communications; smart-city ecosystem.

I. INTRODUCTION

A Wireless Sensor Network (WSN) is an essential foundational technology supporting smart-city application services, where Sensor Nodes (SNs) are heterogeneously interconnected to collect and distribute sensing data, such as real-time streaming and high-capacity data. The data include various demands and priorities, and their transmission method is designed on the basis of several protocols, all of which should be accommodated in the same ecosystem [1]. Millimeter-wave (mmWave) communications [2] and Information-Centric Networking (ICN) technology [3][4] have been gaining attention. MmWave communications have been recognized as a global frontier in future mobile technologies, enabling multi-gigabit data transfer over vast spectrums. ICN is an ideal candidate for a future network architecture that shifts the focus from host locations to data. Namely, the data are named instead of an address, enabling end-users to discover and obtain the data via names, resulting in network abstraction. The named data are handled separately by individual content units, i.e., they can be self-certified and encrypted by their producer, contributing to improved security, in-network caching schemes are also available, i.e., the data are copied and stored in cache memories on network nodes to facilitate further data retrieval.

By applying both mmWaves and ICN to WSNs, our research team previously developed and preliminary evaluated a mmWave Information-Centric WSN-based ecosystem, called mmICWSN that positively affects network performance by improving data delivery [5][6][7]. In one of these papers [7], toward a new sustainable smart-city platform, our mmICWSN can provide a high data rate and low latency with stable connectivity. In addition, we revealed that the ecosystem was ready for deployment in an actual city with prototype implementations. In this paper, an evaluation based on network performance was experimentally conducted to demonstrate the mmICWSN's feasibility in an actual city.

The remainder of this paper is organized as follows. Section II discusses related work. Section III provides a brief overview of the development of mmICWSN. Section IV presents the evaluation results and discussion. Finally, Section V concludes this paper with a summary and mention of future work.

II. RELATED WORK

Zhang et al. [8] identified the open challenges for transport- and network-layer protocols on the basis of a comprehensive simulation study, including congestion control, handover, connectivity, and packet-size control. Khorov et al. [9] investigated emerging wireless systems to provide high-speed data transmissions in mmWave and Terahertz. Note that the transport- and network-layer protocols are unsuitable for these frequency-band communications because of their specific features compared with commonly-used bands due to high signal attenuation and blockage. Kumar et al. [10] experimentally found throughput degradations (collapses) in the 60-GHz band (one of the key frequency bands for mmWaves), and Poorzare et al. [11] analyzed network performances related to this phenomenon under an urban deployment scenario. Yang et al. [12] investigated several congestion-control algorithms, and Vu et al. [13] proposed a state-of-the-art multi-path network protocol for vehicular networks. Netalkar et al. [14] proposed a cross-layer-design end-to-end protocol for fast data delivery in urban micro-cellular networks.

III. ECOSYSTEM DEVELOPMENT

Cellular and satellite telecommunications are the de-facto wireless communication systems for a smart-city ecosystem; however, they are costly to deploy. Although low-power wide-area networks are the primary solutions, they can only transmit a few small data packets within 100-Hz bandwidth below the 1-GHz band. Short-range personal area networks based on IEEE 802.15 standards are used for WSNs. They can conserve power and integrate a high-speed and energy-efficient protocol, but they can be used as a network inside a small regional area. Another global communication system, wireless local area networks based on IEEE 802.11 standards, known as Wi-Fi, can provide extensive and various connectivity for computers, tablets, smartphones, and Internet-of-Things devices. Similar to the idea of network-system selections [15], Wi-Fi-based networks are the optimal candidate for this purpose, because they have several advantages: low-cost wireless modules are readily available, they can be based on IP networks, and the multiple radio-frequency bands can include unlicensed bands, such as 920 MHz and 2.4, 5, 6, and 60 GHz, without regulations. Our mmICWSN uses the IEEE 802.11 ad/ay-compliant Terragraph (TG) communication system in mmWaves [16].

In our prototype mmICWSN [5][6], we used Cefore [17] for the middleware of the ICN platform. Note that Cefore is an open-source CCNx-based ICN platform available on Linux (Ubuntu). The prototype mmICWSN was implemented not limited to a specific application service but designed on the basis of a reliable and zero-touch design [7]. It was designed to be waterproof since it would be placed in outdoor environments.

IV. EVALUATIONS AND DEMONSTRATIONS

Millimeter waves have been used as an alternative to backhaul, both short-range and high-capacity indoor communications, or radar. Compared with the radio-frequency bands that are currently widely used, the extra attenuations for the link budget of mmWaves, such as rain, oxygen, and hydrophilic materials (e.g., trees, leaves, and humans) must be considered. Note that radio waves in the 60-GHz band are particularly affected by the rain and oxygen. Related studies include the applicability of mmWaves for outdoor applications to provide several hundred meters of coverage. To develop ecosystems in actual cities, it is necessary to conduct additional evaluations. To the best of our knowledge, there have been few experiments regarding mmWave long-distance data transmissions; hence, we believe that the contribution of this paper is valuable.

The node devices described in Section III were deployed at a community center and school in Nogata City (Fukuoka, Japan), as shown in Figure 1. The community center and school are three-story buildings, and the node devices were placed on their rooftops. In accordance with the three-dimensional map provided by the National Geographical Institute [18], their altitudes are respectively 7.5 and 16 m, and the straight-line distance between them is 1 km. Across the wireless link, there are a river, road, bridge, and car park, as shown in Figure 2, which might affect radio propagation. The river is the Onga River and the riverside area is well maintained and covered with grass and aquatic plants. During the experiment, the river surface was flat and calm, with no significant waves, i.e., factors affecting mmWave propagation were not observed. The Kanroku Bridge crosses the river and is connected to the main national road. Nogata City is an inter and suburban city between large cities (e.g., Fukuoka City and Kitakyushu City), but the amount of traffic is not dense. The riverside area in front of the community center is used as a car park, and several dozen cars were parked there.

Figure 3 shows the field view of the experimental site. Figures 3(a) and (b) and Figures 3(c) and (d) show the field

Figure 1. Location map of transmitter- and receiver-side nodes

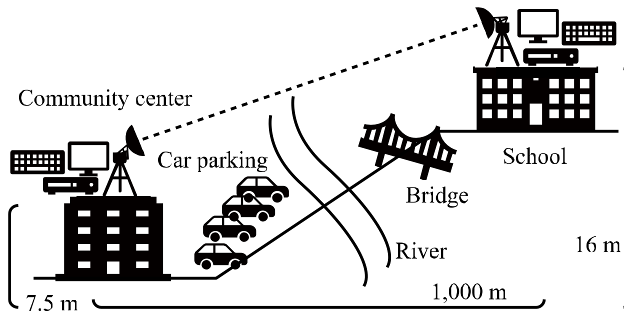


Figure 2. Outline of experimental environment

TABLE I. SPECIFICATION OF TG DEVICE

Terms	*Values*
Radio frequency	58.32 GHz (57–66 GHz) with 2.16 GHz
Transmission power	56 dBm (EIRP)
Antenna	Phased array antenna with dish Gain: 40 dBi Scan range: ±3°, Beam width: ±1°
Dimensions	355 × 355 × 315 mm
Weight	3 kg

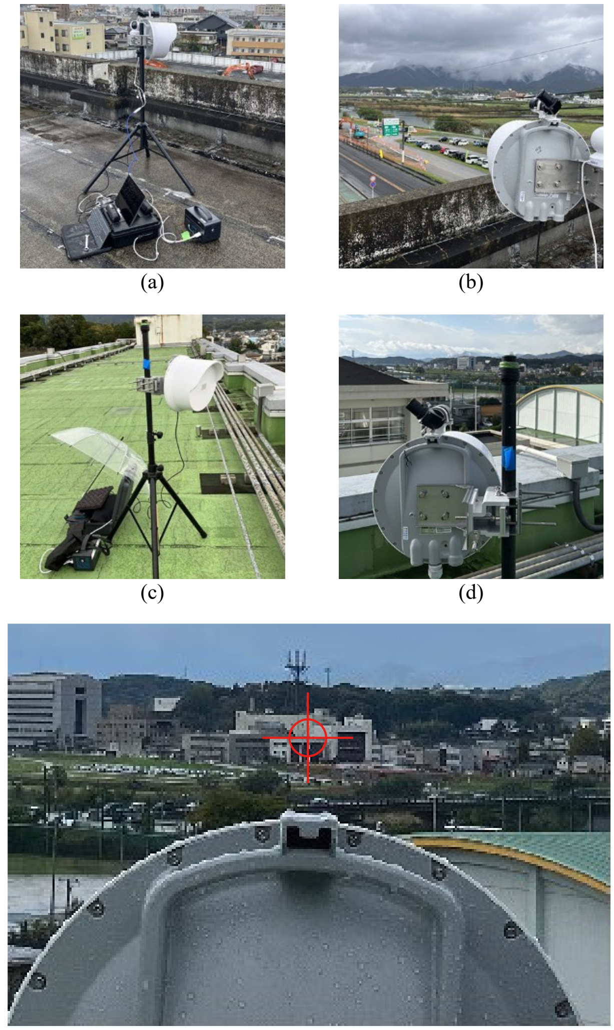
views of the rooftops of the community center and school, respectively. As shown in Figures 3(a) to (d), the mmICWSN node devices were connected to the BeMap MLTG-CN/LR [19] as the mmWave TG device, and the specifications of this device are listed in Table I. In the experiment, two MLTG-CN/LR devices communicated with each other, and the available communication distance was up to 1 km, according to the catalog specifications. Throughout the experiment, the Modulation and Coding Scheme (MCS) index was automatically set as 9. Table II shows the parameter settings regarding adaptive rate control in the IEEE 802.11 ay. Note that IEEE 802.11-compliant Wi-Fi systems achieve effective (high-throughput) data transmission to the control modulation scheme, code rate of error-correcting code, and repetition code based on wireless channel condition, and their combination is pre-defined as MCS settings. Figure 3(e) shows a photo taken behind the dish antenna on the school rooftop toward the community center. The community center is located at the red marking, where the opposite node was

TABLE II. MCS SETTINGS IN SINGLE CARRIER PHYSICAL MODE

Index	***Modulation method***	***Code rate***	***Repetition***	***Data rate (Mbit/s)***
1	BPSK	1/2	2	385
2	BPSK	1/2	1	770
3	BPSK	5/8	1	963
4	BPSK	3/4	1	1,155
5	BPSK	13/16	1	1,251
6	QPSK	1/2	1	1,540
7	QPSK	5/8	1	1,925
8	QPSK	3/4	1	2,310
9	QPSK	13/16	1	2,503
10	16-QAM	1/2	1	3,080
11	16-QAM	5/8	1	3,850
12	16-QAM	3/4	1	4,620

TABLE III. PHYSICAL-LAYER INFORMATION IN EXPERIMENT

Terms	***Antennas are matched***	***Antennas are mis-matched***
Radio channel	Ch 2 (60.48 GHz with 2.16 GHz)	
RSSI	-64 dBm	-62–63 dBm
MCS settings	8–9	6–9
Beam index	30 / 30	30 / 5



(e)

Figure 3. Field view of transmitter- and receiver-side nodes on rooftop of community center and school buildings

placed. As shown in Figure 3(e), the line of sight between the transmitter- and receiver-side nodes can be clearly maintained. The weather was cloudy during the experiment, i.e., the possibility of rainfall attenuation to degrade the mmWaves-band radio propagation could be ignored.

The experiment was conducted for two different scenarios: one in which both elevation and azimuth angles were appropriately adjusted (the antennas were matched), and the other when they were slightly offset (the antennas were mismatched). The status information of the physical layer for these scenarios is summarized in Table III. Note that, in the MLTG-CN/LR device, the antenna's front space is divided into a grid pattern in terms of elevation- and azimuth- angles, then each sub-region is assigned a beamforming index. The most central beam direction on the antenna surface is when the beamforming index is 30. Figure 4 shows the experimental results, which both (a) and (b), (c) and (d), and (e) and (f) are Transmission Control Protocol (TCP) with the CUBIC algorithm, User Datagram Protocol (UDP), and ICN performance, respectively. In Figures 4(a) to (d), iPerf3 [20], which is a well-known network-performance measurement tool, was used to measure TCP/UDP performance at every 1 s interval for 90 s. Note that the nodes in the community center and school were assigned as server and client nodes in the iPerf3 settings. Figures 4(e) and (f) show the results of retrieving the different data using Cefore [17].

As shown in Figure 4(a), the average TCP throughput was 941 and 94.4 Mbit/s for when the antennas were matched and mismatched, respectively. The TG antenna is a parabolic dish type; thus, even a few degrees of angle misalignment can cause significant TCP throughput degradation. For TCP congestion control, as shown in Figure 4(b), the average congestion-window size was 1.26 and 0.967 Mbytes; hence, there was a 39.3% difference. In the curve when the antennas were matched, several attempts were made to increase the congestion-window size.

As shown in Figure 4(c), the average UDP throughput was 902 and 93.3 Mbit/s for the two scenarios, respectively. In the curve when the antennas were matched, there were regions where UDP throughput temporarily decreased. The reason for this decrease is that automatic retransmission requests and forwarding-error-control mechanisms are omitted, resulting in these dramatic degradations. The results in Figure 4(a) indicate no degradation because the congestion-control mechanism in TCP is available and useful works. Figure 4(d) shows the packet-error probability for UDP transfer; the averages were 0.0294 and 0.903 for the matched and mismatched scenarios, respectively. When the antennas were mismatched, many packet losses occurred, which affected not only UDP throughput but also that of TCP, as shown in Figures 4(a) and (c).

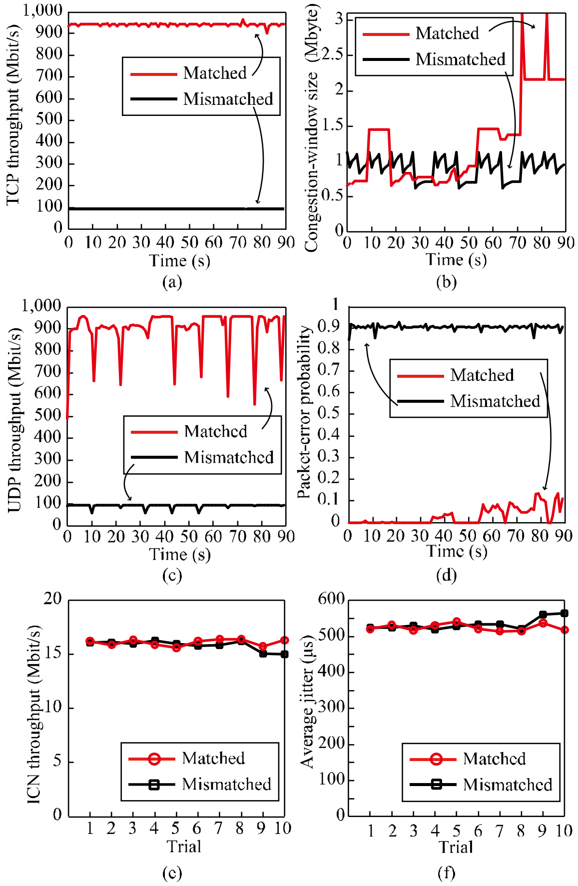


Figure 4. Experimental results

As shown in Figure 4(e), the average ICN throughput was 16.1 and 15.8 Mbit/s for the two scenarios, respectively. The ICN throughput was significantly smaller than that of TCP or UDP because Cefore has a bottleneck. In mmICWSN, the ICN layer was stacked on the TCP/UDP layers. Thus, due to the middleware implementation in Cefore, if the maximum data-transmission bandwidth is set as the maximum value, the failure probability of data registration, storage, and transfer becomes worse. As shown in Figure 4(f), the average jitter was 525 and 534 μs for the two scenarios, respectively. ICN throughput and jitter did not significantly differ between the two scenarios. In accordance with these results, we found that the performance of TCP/UDP/IP protocol stacks was not affected by that of the ICN-layer protocol. We also experimentally verified that we could obtain sufficient network performance for mmICWSN in an actual city.

V. CONCLUSION AND FUTURE WORK

Our research team evaluated the feasibility of the network performance in the TCP, UDP, and ICN protocols with mmICWSN. From the experimental results, we found that it was necessary to improve the ICN throughput by modifying the Cefore settings, and the antenna placement for mmWaves was sensitive to a few degrees of angle. Through the demonstration of the mmWaves experiment, the developed system could be applicable to long-distance wireless transmission in an actual city. For future work, we plan to deploy mmICWSN for practical smart-city applications, such as smart agriculture. In detail, we will develop a new ecosystem that supports an on-demand and real-time video and image forwarding platform for a common demand for smart-agriculture applications.

ACKNOWLEDGMENTS

This work was partly supported by JSPS KAKENHI Grant Number JP25K15104 and NICT Japan, Grant Number JPJ012368C05601. We are grateful to Dr. Kenji Kanai for their helpful discussions, and to Nogata City and Advantech Japan for their help with the experiments.

REFERENCES

[1] P. Mishra and G. Singh, "6G-IoT framework for sustainable smart city: Vision and challenges," *IEEE Consumer Electronics Mag.*, pp. 1–8, Sept. 2023.

[2] K. Aldubaikhy, W. Wu, N. Zhang, N. Cheng, and X. Shen, "MmWave IEEE 802.11ay for 5G fixed wireless access," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 88–95, Apr. 2020.

[3] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Mag.*, vol. 50, no. 7, pp. 26–36, July 2012.

[4] L. C. M. Hurali and A. P. Patil, "Application areas of information-centric networking: State-of-the-art and challenges," *IEEE Access*, vol. 10, pp. 122431–122446, Nov. 2022.

[5] S. Mori, "Zero-touch-design information-centric wireless sensor networking with availability assurance," *International J. Advances in Networks and Services*, vol. 16, no. 3&4, pp. 63–74, Dec. 2023.

[6] S. Mori, "Test-field development for ICWSNs and preliminary evaluation for mmWave-band wireless communications," *Proc. IEEE CCNC 2024*, Las Vegas, USA, Jan. 2024, pp. 1–2, doi: 10.1109/CCNC 51664.2024.10454799.

[7] S. Mori, "Development of UAV-aided information-centric wireless sensor network platform in mmWaves for smart-city deployment," *International J. Advances in Networks and Services*, vol. 17, no. 3&4, pp. 105–115, Dec. 2024.

[8] M. Zhang, et al., "Will TCP work in mmWave 5G cellular networks?," *IEEE Communications Mag.*, vol. 57, no. 1, pp. 65–71, Jan. 2019.

[9] E. Khorov, A. Krasilov, M. Susloparov, and L. Kong, "Boosting TCP & QUIC performance in mmWave, Terahertz, and Lightwave wireless networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 2862–2891, Aug. 2023.

[10] R. Kumar, et al., "TCP BBR for ultra-low latency networking: Challenges, analysis, and solutions," *Proc. IFIP Networking 2019*, Warsaw, Poland, May 2019, pp. 1–9, doi: 10.23919/IFIPNetworking. 2019.8816856.

[11] R. Poorzare and A. C. Augé, "How sufficient is TCP when deployed in 5G mmWave networks over the urban deployment?," *IEEE Access*, vol. 9, pp. 36342–36355, Mar. 2021.

[12] W. Yang, et al., "A measurement study of TCP performance over 60GHz mmWave hybrid networks," *Proc. WoWMoM 2022*, Belfast, UK, June 2022, pp. 300–305, doi: 10.1109/WoWMoM54355.2022.00057.

[13] N. T. T. Vu, H. D. Le, K. D. Dang, C. T. Nguyen, and A. T. Pham, "Performance analysis of multipath TCP over FSO/mmWave vehicular networks," *Proc. ATC 2023*, Da Nang, Vietnam, Oct. 2023, pp. 172–177, doi: 10.1109/ATC58710.2023.10318939.

[14] P. Netalkar, J. Chen, and D. Raychaudhuri, "mmCPTP: A cross-layer pull based transport protocol for 5G mmWave networks," *Proc. IEEE/IFIP NOMS 2023*, Miami, USA, May 2023, pp. 1–9, doi: 10.1109/NOMS56928.2023.10154364.

[15] A. U. H. Hashmi, et al., “Effects of IoT communication protocols for precision agriculture in outdoor environments,” *IEEE Access*, vol. 12, pp. 46410–46421, Mar. 2024.

[16] Terragraph, https://terragraph.com/ (retrieved: Jan. 2025).

[17] Cefore, https://cefore.net/ (retrieved: Jan. 2025).

[18] NGI, https://www.gsi.go.jp/ENGLISH/ (retrieved: Jan. 2025).

[19] BeMap, https://www.bemap.co.jp/ (retrieved: Jan. 2025).

[20] iPerf3, https://iperf.fr/ (retrieved: Jan. 2025).

 ISBN: 978-1-68558-277-7