INTERNET 2023

The Fifteenth International Conference on Evolving Internet

ISBN: 978-1-68558-062-9

March 13th - 17th, 2023

Barcelona, Spain

INTERNET 2023 Editors

Bedir Tekinerdogan, Wageningen University & Research, Wageningen, The Netherlands

Alexey Cheptsov, TIK/University of Stuttgart, Germany

INTERNET 2023

Forward

The Fifteenth International Conference on Evolving Internet (INTERNET 2023), held between March 13th and March 17th, 2023, continued a series of events focusing on the challenges raised by the evolving Internet, making use of the progress in different advanced mechanisms and theoretical foundations.

Originally designed in the spirit of interchange between scientists, the Internet reached a status where large-scale technical limitations impose rethinking its fundamentals. This refers to design aspects (flexibility, scalability, etc.), technical aspects (networking, routing, traffic, address limitation, etc.), as well as economics (new business models, cost sharing, ownership, etc.). Evolving Internet poses architectural, design, and deployment challenges in terms of performance prediction, monitoring and control, admission control, extendibility, stability, resilience, delay-tolerance, and interworking with the existing infrastructures or with specialized networks.

We take here the opportunity to warmly thank all the members of the INTERNET 2023 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 INTERNET 2023. 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 INTERNET 2023 organizing committee for their help in handling the logistics of this event.

We hope that INTERNET 2023 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of the evolving Internet.

INTERNET 2023 Chairs

INTERNET 2023 Steering Committee

Renwei (Richard) Li, Future Networks, Futurewei, USA
Eugen Borcoci, University "Politehnica" of Bucharest (UPB), Romania
Terje Jensen, Telenor, Norway
Przemyslaw (Przemek) Pochec, University of New Brunswick, Canada
Parimala Thulasiraman, University of Manitoba – Winnipeg, Canada
Dirceu Cavendish, Kyushu Institute of Technology, Japan

INTERNET 2023 Publicity Chairs

José Miguel Jiménez, Universitat Politecnica de Valencia, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

INTERNET 2023
Committee

INTERNET 2023 Steering Committee

Renwei (Richard) Li, Future Networks, Futurewei, USA
Eugen Borcoci, University "Politehnica" of Bucharest (UPB), Romania
Terje Jensen, Telenor, Norway
Przemyslaw (Przemek) Pochec, University of New Brunswick, Canada
Parimala Thulasiraman, University of Manitoba – Winnipeg, Canada
Dirceu Cavendish, Kyushu Institute of Technology, Japan

INTERNET 2023 Publicity Chairs

José Miguel Jiménez, Universitat Politecnica de Valencia, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

INTERNET 2023 Technical Program Committee

Renwei (Richard) Li, Future Networks, Futurewei, USA
Majed Alowaidi, Majmaah University, Saudi Arabia
Mohammad Alsulami, University of Connecticut, USA
Mário Antunes, Polytechnic of Leiria & INESC-TEC, Portugal
Andrés Arcia-Moret, Xilinx, Cambridge, UK
Marcin Bajer, ABB Corporate Research Center Krakow, Poland
Michail J. Beliatis, Research Centre for Digital Business Development | Aarhus University, Denmark
Driss Benhaddou, University of Houston, USA
Nik Bessis, Edge Hill University, UK
Maumita Bhattacharya, Charles Sturt University, Australia
Filippo Bianchini, Studio Legale Bianchini, Perugia, Italy
Eugen Borcoci, University "Politehnica" of Bucharest (UPB), Romania
Fernando Boronat Seguí, Universidad Politécnica De Valencia-Campus De Gandia, Spain
Christos Bouras, University of Patras, Greece
Matthew Butler, Bournemouth University, UK
Alina Buzachis, University of Messina, Italy
Lianjie Cao, Hewlett Packard Labs, USA
Dirceu Cavendish, Kyushu Institute of Technology, Japan
Hao Che, University of Texas at Arlington, USA
Albert M. K. Cheng, University of Houston, USA
Hongmei Chi, Florida A&M University, USA
Andrzej Chydzinski, Silesian University of Technology, Poland
Franco Cicirelli, ICAR-CNR, Italy
Victor Cionca, Munster Technical University, Ireland
Monireh Dabaghchian, Morgan State University, USA
Noel De Palma, University Grenoble Alpes, France
Angel P. del Pobil, Jaume I University, Spain
Jun Duan, IBM T. J. Watson Research Center, USA

Table of Contents

A Review of Next Generation Internet Architecture Open-Source Software Projects 1
Mirko Presser and Rita Madsen

Transfer of Session State Between Satellites in a Space Information Network 6
Anders Fongen

On the Study of Internet Ossification and Solution 10
Lin Han and Richard Li

BBR Performance over Variable Delay Paths on Multipath TCP Video Streaming 18
Masayoshi Kondo, Dirceu Cavendish, Daiki Nobayashi, and Takeshi Ikenaga

A Review of Next Generation Internet Architecture Open-Source Software Projects

Towards a Human Centric Internet

Mirko Presser
Department of Business Development and Technology, Aarhus University
Herning, Denmark
e-mail: mirko.presser@btech.au.dk

Rita Madsen
Department of Business Development and Technology, Aarhus University
Herning, Denmark
e-mail: rita.madsen@btech.au.dk

***Abstract*—Internet and web technologies have evolved over the past 50 years to a patchwork of architectures, frameworks, applications/services and protocol stacks, with many actors contributing solutions as well as novel uses of the technology emerging rapidly. A drive from the European Commission (EC) called Next Generation Internet (NGI) supports developers, researchers and entrepreneurs towards building a human-centric Internet, focusing on privacy and trust as central themes and building contributions based on Open-Source Software (OSS). This paper reviews 32 projects supported by the EC in terms of their OSS community contributions and the planned sustainability of their solutions and organisation.**

***Keywords* — Internet architecture, network, transport, applications, services.**

I. INTRODUCTION

This article presents work in progress of an analysis of Next Generation Internet projects regarding their financial sustainability and overall contribution to the evolution of Internet Architecture. It includes projects covering all layers of the Internet and its core applications and new communication paradigms on decentralisation.

In 2017, a new European initiative called the Next Generation Internet was created to support European values in the future development of the Internet and related technologies. Robert Viola states in his speech [1]:

"When policy-makers, researchers and society reflect on the future evolution of the Internet, we should take a fresh look at all these issues. The Internet should offer more to the people and to our society, providing better services and greater involvement and participation. It should be designed for humans, so that it can meet its full potential for society and economy and reflect the social and ethical values that we enjoy in our societies."

The rest of the paper is structured as follows. Section II introduces the background of the data, followed by the categorisation of the projects in Section III. The most significant cases are described in Section IV. Section V proposes future research and conclusion. The article is finalised with acknowledgement for the projects which allow data collection and research work within open-source projects.

II. BACKGROUND

Data for this article originates from the NGI Pointer [2] project. NGI Pointer supported a total of 36 OSS projects with grants ranging from 50,000EUR to 200,000EUR to develop topics around Next Generation Internet Architecture. A total of 250 applicants were evaluated. An independent committee of experts selected the projects based on a set of common criteria. The project addressed topics included: Privacy-by-design; Internet at the Edge; Network Optimisation; Virtualisation and isolation; Limitations in the TCP/IP protocol suite; Autonomous Network operations and control; Energy efficiency; Industrial Internet Security; Trust for New Internet/Web Users.

32 out of the 36 projects are studied in this research article. Three projects were excluded due to the limited information the projects provided publicly. One project (Nyxt) received two separate grants from NGI Pointer and is treated as one project for this research work.

All data collected in this research is publicly available on the various websites of the projects, organisations and code repositories (See Tables 1-3). The data was collected over a period of three years, starting in January 2020, with the project to conclude in December 2022. Several interviews, dialogues and a podcast series helped enrich the information and clarified ambiguous data. The podcasts are available publicly [3].

III. CATEGORISATION OF PROJECTS

The Internet architecture can be broken down into six layers [4]. In this article, the six layers are bundled into:

- Applications and Services:
  - Specific applications and services
  - Application protocols
- Network and Transport Layers
  - Transport protocols
  - Network protocols
- Other
  - Data-link protocols
  - Physical layer protocols

The "other" category also includes projects that do not fall strictly into any other layers, such as hardware projects related to chip design or chip design tools and communication systems spanning several layers.

IV. REVIEW OF NEXT GENERATION INTERNET CASES

Tables 1-3 provide a complete list of the studied projects. Furthermore, a few of them must be highlighted as particularly interesting projects.

 ISBN: ISBNFILL

A. Case Examples

SCION is an initiative that has been ongoing for over a decade, with over 50 individuals, mostly from research, participating in the development of SCION [5]. The contribution from NGI Pointer is a public grant to develop a specific mechanism called VerfiedRouting for SCION. SCION, as such, is then a publicly funded open research project. However, alongside the development as an open research project, the founder and founding university created a spin-off called Anapaya that also monetises SCION through corporate distribution and offers consulting-style services around the technology to customers, mostly financial institutions and Internet Service Providers.

WireGuard is a straightforward yet fast and modern VPN that utilises state-of-the-art cryptography [6]. It was initially released for the Linux kernel but has since been ported to all major operating systems. The project is a large open community project that has largely been supported by public funds, however, the key persons behind the technology also operate consultancies offering consulting services in cyber security, amongst other topics. In addition, Wireguard is now provided via Mozilla as a corporate distribution. Finally, WireGuard also accepts donations to the project via its website.

Ltt.rs is a user-friendly and encrypted by default e-Mail client for Android based on modern standards like JMAP (RFC8621) [7] and Autocrypt [8]. Using JMAP instead of IMAP makes the app more maintainable and reliable on current mobile operating systems due to the built-in push capabilities. Ltt.rs is sponsored via public funding but simultaneously offers a small-scale corporate distribution of the application as a freemium/premium application.

Nyxt is a keyboard-driven web browser designed for power users. Inspired by Emacs [9] and Vim [10], it has familiar key bindings and is infinitely extensible in Lisp. For example, users can switch between tabs by topic or URL, search all URLs on a page by name or target, search through all of the bookmarks with compound queries, etc. Nyxt focuses on corporate distribution as a freemium/premium application, where the basic version of the browser is free to use, and tools, as well as add-ons, become part of a subscription-based service at a premium. Additionally, the team behind Nyxt is offering consulting services.

The PANAPI [11] project designs a sophisticated host-based network-path selection engine on top of the SCION [5] network architecture and provides it as an open-source implementation of the abstract next-generation transport service API, currently being drafted in the IETF TAPS Working Group. The project is an OSS-sponsored project and predominantly an open research project with contributions to standards.

V. Conclusion and Future Work

This article presents a first look at 32 projects under the umbrella of the NGI. The aim is to analyse the projects from a financial sustainability perspective and gain a more general understanding of how to support and grow future Internet initiatives through public, private and community support. More projects need to be studied to find more evident trends and develop a model of modern Business Model Patterns for open-source software and hardware projects driven by European Values under the NGI.

A suggested framework for such analysis is based on [4], but other theoretical and practical studies should be considered.

Acknowledgement

NGI POINTER (NGI Program for Open Internet Renovation) has received funding from the European Commission, as part of the Horizon 2020 Research and Innovation Programme, under Grant Agreement No.871528.

In addition, the work was also supported by the NGI Search project, which also received funding from the European Commission as part of the Horizon Europe Research and Innovation Programme under Grant Agreement No. 101069364.

References

[1] R. Viola, ʻA human centric Next Generation Internet shall reflect the openness, diversity and the inclusion that are at the core of European values,ʼ *Technology at the European Commission*. Publication 30 June 2017 https://digital-strategy.ec.europa.eu/en/library/Internet-humans-how-we-would-Internet-future-be

[2] NGI Pointer: https://pointer.ngi.eu/ (4/11/22).

[3] NGI Architects: https://shows.acast.com/spotlight-on-ngi-architects (4/11/22)

[4] R. T. Watson, M. C. Boudreau, P. T. York, M. E. Greiner, and D. Wynn Jr, ʻThe business of open source,ʼ *Communications of the ACM*, vol. *5*1 no. 4, pp. 41-46, 2008.

[5] L. Chuat, et al. ʻThe Complete Guide to SCION-From Design Principles to Formal Verification,ʼ *Information Security and Cryptography*, 2022.

[6] J. A. Donenfeld, ʻWireguard: next generation kernel network tunnel,ʼ in NDSS, pp. 1-12, 2017, February.

[7] N. Jenkins and C. Newman, *The JSON Meta Application Protocol (JMAP) for Mail*, No. rfc8621, 2019.

[8] H. Krekel, K. McKelvey, and E. Lefherz, ʻHow to fix e-mail: making communication encrypted and decentralised with autocrypt,ʼ *XRDS: Crossroads*, *The ACM Magazine for Students*, vol. *2*4, no. 4, pp. 37-39, 2018.

[9] R. M. Stallman, ʻEMACS the extensible, customisable self-documenting display editor,ʼ *Proceedings of the ACM SIGPLAN SIGOA symposium on Text manipulation*, pp. 147-156, 1981, April.

[10] Vim: https://www.vim.org/ (4/11/22).

[11] T. Krüger and D. Hausheer, ʻTowards an API for the Path-Aware Internet,ʼ *Proceedings of the ACM SIGCOMM 2021 Workshop on Network-Application Integration*, pp. 68-72, 2021.

[12] J. Morton, P. Heist, and R. Grimes, ʻThe Some Congestion Experienced ECN Codepoint,ʼ I*nternet Engineering Task Force*, Internet-Draft draftmorton-taht-sce-00, 2019.

[13] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, ʻ*TCP extensions for multipath operation with multiple addresses*,ʼ no. rfc6824, 2013.

[14] D. Pujol-Perich, et al., ʻIGNNITION: Bridging the Gap between Graph Neural Networks and Networking Systems,ʼ IEEE Network, vol. 35, no. 6, pp. 171–177, 2021.

[15] C. A. Esterhuyse and H. D. A. Hiep, ʻReowolf: Synchronous Multi-party Communication over the Internet,ʼ *International*

Conference on Formal Aspects of Component Software, pp. 235-242. Springer, Cham, 2019, October.

[16] M. Coudron, 'Passive analysis for multipath TCP,' *Proceedings of the Asian Internet Engineering Conference*, pp. 25-32, 2019, August.

[17] Q. De Coninck, et al., 'Pluginizing quic,' *Proceedings of the ACM Special Interest Group on Data Communication*, pp. 59-74, 2019.

[18] C. Diaz, H. Halpin, and A. Kiayias, 'The Nym Network,' The Next Generation of Privacy Infrastructure, 2021.

[19] A. M. Piotrowska, 'Studying the anonymity trilemma with a discrete-event mix network simulator,' *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*, pp. 39-44, 2021, November.

[20] K. Djemame, 'Energy Efficiency in Edge Environments: a Serverless Computing Approach,' *International Conference on the Economics of Grids, Clouds, Systems, and Services*, pp. 181-184. Springer, Cham, 2021.

[21] M. Bagnulo and A. García-Martínez, 'An experimental evaluation of LEDBAT++,' *Computer Networks*, Elsevier, 2022.

[22] M. Bagnulo and A. García-Martínez, 'When less is more: Bbr versus ledbat++,' *Submitted for publication at SSRN*, 2022.

[23] R. Bailey and P. Misra, Interoperability of Social Media: An appraisal of the regulatory and technical ecosystem. Available at SSRN use: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4095312, 2022.

[24] A. Madhavapeddy and D. J. Scott, 'Unikernels: Rise of the Virtual Library Operating System: What if all the software layers in a virtual appliance were compiled within the same safe, high-level language framework?' *Queue*, vol. 11, no. 11, pp. 30-44, 2013.

[25] A. Hemel, 'Using ELF symbols extracted from dynamically linked ELF binaries for fingerprinting,' 2021.

[26] A. Hemel, K. T. Kalleberg, R. Vermaas, and E. Dolstra, 'Finding Software License Violations Through Binary Code Clone Detection-A Retrospective,' *ACM SIGSOFT Software Engineering Notes*, vol. *46, no.* 3, pp. 24-25, 2021.

[27] W. Termont, W. Janssens, R. Verborgh, and T. Haegemans, 'A Decentralised Registry to Support the Dynamic Binding of Semantic Components in Solid Data Browsers,' *Digital Research Report DGT_2*, pp. 1-1, 2021.

[28] L. Mustafa and E. Černčič, 'Low-cost NIR measurement device,' *2015 4th International Workshop on Optical Wireless Communications (IWOW)*, pp. 127-131. IEEE, 2015.

TABLE 1. OTHER PROJECTS

Name	Description	Contributors
Mirage OS [24]	MirageOS is a library operating system that constructs unikernels for secure, high-performance network applications across various cloud computing and mobile platforms. Developers can write code on traditional operating systems such as Linux or macOS. They can then compile their code into a fully-standalone, specialised unikernel that runs under the Xen or KVM hypervisors and lightweight hypervisors like FreeBSD's BHyve, and OpenBSD's VMM. These unikernels can deploy on public clouds, like Amazon's Elastic Compute Cloud and Google Compute Engine, or private deployments.	Medium 10-100 https://github.com/mirage/mirage
BANG [25] and [26] The official project title is European Firmware and App Library	BANG is a framework for processing binary files (like firmware). It consists of an unpacker that recursively unpacks and classifies/labels files and separate analysis programs that work on the results of the unpacker.	Small <10 https://github.com/armijnhemel/binaryanalysis-ng
Solid Shape - Reshaping Linked Data on the Fly [27]	The project implemented an open-source registry of linked data shapes and forms. A registry of linked data shapes and forms is an essential part of the Solid ecosystem, and the execution of this project will accelerate Solid's adoption.	Small <10 https://github.com/digita-ai/semcom/
LKRT Linux kernel regression tracking	Build and integrate mechanisms into the Linux kernel development processes to track all regressions reported by humans or CI systems. Together with the existing "no regression" rule, this will help ensure new releases with improved security techniques work as well as their predecessors.	Small <10 https://gitlab.com/knurd42/regzbot https://linux-regtracking.leemhuis.info/post/regzbot-approach/
KORUZA: Wireless Optical Communication System for Urban Environments [28]	KORUZA is a wireless optical communication system based on the FSO technology for urban environments, designed for last-mile, 5G and IoT applications. It uses an eye-safe collimated beam of IR light for point-to-point data transmission through the air. The solution avoids digging up roads, allowing distances up to 150m with fibre-like speeds of 1-10 Gbps.	Small <10 https://github.com/IRNAS/koruza-v2-pro
Libre EDA	LibrEDA is a libre software framework for the physical design of silicon chips. A strong motivation is democratising silicon technology by making ASIC toolchains accessible for research, education and hobbyists.	Medium 10-100 https://codeberg.org/LibrEDA
Libre SOC	The LibreSOC Project brings an ethically developed privacy respectful, power-efficient SoC to the world. Full source to the bedrock. No spying backdoor co-processors. No leaked firmware keys. Fully transparently developed.	Small <10 https://git.libre-soc.org/

TABLE 2. APPLICATIONS AND SERVICES PROJECTS

Name	Description	Contributors
Ltt.rs - Open Source E-mail client	Ltt.rs is a user-friendly and encrypted by default e-Mail client for Android based on modern standards like JMAP (RFC8621) [7] and Autocrypt [8]. Using JMAP instead of IMAP makes the app more maintainable and more reliable on current mobile operating systems due to the built-in push capabilities.	Medium 10-100 https://github.com/iNPUTmice/lttrs-android
DREAM: Replicable extensible agency machine	The project enables the convergence of distributed peer-to-peer (P2P) networks and linked data models within a social solidary economy and organisation among trusted groups.	Small <10 https://gitlab.com/public.dream
Nyxt Browser	Nyxt is a keyboard-driven web browser designed for power users. Inspired by Emacs [9] and Vim [10], it has familiar key bindings and is infinitely extensible in Lisp. For example, users can switch between tabs by topic or URL, search all URLs on a page by name or target, Search through all bookmarks with compound queries, etc.	Large 100-1000 https://github.com/atlas-engineer/nyxt Note: The Nyxt Browser project was split into two parts during NGI Pointer.
Lightmeter	Lightmeter makes it easy to run e-mail servers by visualising, monitoring and notifying users of problems and opportunities for improved performance and security. People regain control of sensitive communications either directly by running their own mail servers or indirectly via the increased diversity and trustworthiness of mail hosting services.	Medium 10-100 https://gitlab.com/lightmeter/controlcenter Note: since the completion of the project, Lightmeter has received venture funding via Y-Combinator
Scuttlebutt - The Gossip Protocol	Scuttlebutt (SSB) is an edge computing, peer-to-peer communications protocol.	Medium 10-100 https://github.com/dominictarr/scuttlebutt
P2Panda	p2panda + Bamboo + OpenMLS p2panda is a user-friendly peer-to-peer communications protocol with browser support, local deletion, fork detection, efficient partial replication, large-scale group messaging encryption and future-proof schema migrations to build secure and user-friendly applications. p2panda has been in development since 2019, focusing on prototyping and research.	Small <10 https://github.com/p2panda
Advanced privacy-preserving protocols for **GNU Taler**	GNU Taler is a project that offers a free software infrastructure and protocols for privacy-friendly online payments. The software is continuously extended with added features.	Medium 10-100 https://git.taler.net/
Garage: Geo-distributed data store compatible with the S3 API	Garage is a geo-distributed data store notably compatible with the S3 API. Garage makes it easy to distribute the storage layer of digital services, supporting multi-cloud and on-premise deployments and even allowing household computers to join the cluster without hassle.	Small <10 https://git.deuxfleurs.fr/Deuxfleurs/garage
Peergos [23]	Peergos is building the next web - the private web, where end users are in control. For example, web apps are secure by default and unable to track individuals, and individuals control precisely what personal data each web app can see.	Small <10 https://github.com/peergos/peergos
DT4DW: Developer Tools for Decentralised Web	The essential technologies to build a decentralised web are available today, but the broader ecosystem and uptake are still in its infancy. One reason for this is a lack of developer tooling. Moving from traditional client/server architectures to building decentralised applications requires developers to replace many day-to-day debugging & testing tools with manual logging, custom scripts and guesswork. This tooling gap contributes to the significant difficulties of decentralised development today. By extending HTTP Toolkit and Mockttp to support IPFS, WebRTC and Ethereum, this project gives developers the tooling to debug and test next-generation decentralised web applications.	Small <10 https://github.com/httptoolkit

TABLE 3. NETWORK AND TRANSPORT LAYER PROJECTS

Name	Description	Contributors
SCE - Some Congestion Experienced [12]	SCE is a high-fidelity congestion control signalling protocol beyond the edge and into core networks and aggregation points.	Very large >1000 https://github.com/chromi/sce
WireGuard [5]	WireGuard is a straightforward yet fast and modern VPN that utilises **state-of-the-art cryptography**. Initially released for the Linux kernel, it is cross-platform (Windows, macOS, BSD, iOS, Android) and widely deployable.	Large 100-1000 for the original linux version https://www.wireguard.com/repositories/ https://www.wireguard.com/
MTCP5G Multipath TCP for 5G networks [13]	MPTCP is an ongoing effort of the IETF Multipath TCP working group that aims at allowing a TCP connection to use multiple paths to maximise resource usage and increase redundancy. In other words, MPTCP will allow one TCP session to be conveyed on several paths over different access networks.	Very large >1000 https://github.com/multipath-tcp/mptcp
IGNNITION Fast prototyping of complex GNN for network optimisation [14]	IGNNITION is a framework for the fast prototyping of Graph Neural Networks (GNN). This framework allows users to design and run their own GNNs without specialised knowledge, such as TensorFlow or PyTorch.	Small <10 https://github.com/BNN-UPC/ignnition
SCION Scalability, Control, and Isolation on Next-generation Networks [5]	SCION is the first clean-slate Internet architecture designed to provide route control, failure isolation, and explicit trust information for end-to-end communication.	Medium 10-100 https://github.com/scionproto/scion
REOWOLF [15]	Reowolf replaces sockets with connectors which support high-level verification, compilation, and optimisation techniques.	Small <10 https://gitlab.com/nl-cwi-csy/reowolf
MPTCP analyser [16]	The MPTCP analyser is a tool to help analyse the performance of a multipath protocol and the software to auto-configure the system depending on the application objective and network conditions.	Small <10 https://github.com/ngi-mptcp/mptcpanalyzer
DataHop - Incentivised Content Dissemination at the Network Edge	DataHop is a mobile content distribution infrastructure based on smartphone device-to-device (D2D) communications. Content is pushed to source selected mobile users, hops from device to device, and spreads in the network to destination nodes.	Small <10 https://github.com/datahop
EDGESec	Edgesec defines a new architecture and toolset for edge-based routers addressing fundamental security weaknesses that impact current IP and IoT router implementations.	Small <10 https://github.com/nqminds/EDGESec
TCPLS [17]	TCPLS is an extension to Transport Layer Security (TLS) 1.3 that closely couples TLS with one of the most important Internet protocols: TCP. This allows greater extensibility for TCP by overcoming the limits of TCP Options and limiting middlebox interference.	Small <10 https://github.com/p-quic
SPHINX [18] and [19]	The Sphinx packet format provides an essential potentially standardised component for privacy-enhanced networking.	Small <10 https://github.com/nymtech/sphinx
PANAPI: Path Aware Networking Application Programming Interface [10]	The PANAPI project designs a sophisticated host-based network-path selection engine on top of the SCION [5] network architecture and provides it as an open-source implementation of the abstract next-generation transport service API currently being drafted in the IETF TAPS Working Group.	Small <10 https://github.com/netsys-lab/panapi
EDGNSS: Energy Efficiency, Edge and Serverless Computing [20]	The project revisits the Internet Architecture by leveraging Software Defined Networks (SDN) with Network Function Virtualisation (NFV) technologies to allow efficient and on-demand placement of Virtual Network Functions (VNF) on a serverless platform for energy-aware function provisioning in edge environments.	Small <10 https://github.com/EDGNSS
P4EDGE	The project develops an open-source software stack to enable the creation of accessible P4-switches based on open hardware (e.g., RaspPI, x86/ARM-based router boards) that have low cost and low power consumption and are accessible for a wide range of edge users, where moderate performance (100Mbps to few Gbps) is needed.	Small <10 https://github.com/P4EDGE
RIM: Receiver-driven Incoming-traffic Management [21] and [22]	The project evolves the Internet's resource management approach to enable receivers to execute congestion control functions, taking an active role in determining the capacity of incoming traffic.	Small <10 https://github.com/net-research/rledbat_module

Transfer of Session State Between Satellites in a Space Information Network

Anders Fongen
Norwegian Defence University College (FHS)
Lillehammer, Norway
email: anders@fongen.no

***Abstract*—In a Space Information Network (SIN), there will be frequent handovers of service connections between ground-based clients and orbiting satellites above. In the case where the state of the client sessions are represented in the satellite computers, the handover operation becomes considerably more complicated and expensive, and several methods for the transfer of client session state are investigated in this article.**

***Keywords*—LEO satellites; space information networks; session state; state transfer; mobile computing.**

I. Introduction

The term *Space Information Network* (SIN) describes a set of satellites that cooperatively offer services for information processing and sharing, as well as traditional communication services. SIN is regarded as a natural evolution of satellite services, coming from radio mirrors in geostationary orbit and Low Earth Orbit (LEO) constellation for communication services (e.g., Iridium) [1], [2].

In a series of previous publications, different aspects of SIN operation (architecture [3], security [4], cache management [5] and routing [6]) have been addressed. This article will focus on the management of *session state* as a client engages the SIN in information processing tasks. During handover, the existing state of the session/dialog needs to be made accessible to the next satellite. This can be formulated as a *process migration* problem, well known in the field of Distributed Computing.

While the general problem of stateful process migration has no practical solution, a SIN offers properties which, when taken into account, can reduce the general problem into something for which a practical solution can be found. The two most important properties are:

1) The predictability of satellites' relative position at any time.
2) The uneven density distribution of the world's population.

The perspective of the presented analysis is that from Distributed Computing. Technical and physical properties of satellites, related to energy management, beamforming, modulation, coding, jamming resistance etc., are not taken into account.

The remainder of the paper is organized as follows: Section II provides a discussion on the organization and representation of session state. Section III discusses how Docker Swarms and Kubernetes can provide a framework for service code management. Section IV explains briefly the software simulation model used for the experiment. Section V analyses how session state representation can be handled during a handover operation. Section VI shows the paging arrangement evaluated in the experiment, while Section VII discusses some remaining issues in the results. Section VIII sums up the text and identifies remaining research problems.

II. Representation of Session State

During a dialog between a service provider and a client, there always exists a context within which the next transaction will be interpreted. The representation of this context is called the *session state*, which is a collection of several data elements in the service provider and the client (CPU register content, user and system space memory values). Together, these elements allow the dialogue to form a coherent chain of transactions, without relying on the CPU instruction pointer (i.e., the transactions can all start at a common execution entry point).

In the well understood field of Web Services, the state is most often represented by the content of the Application Protocol Data Unit (A-PDU) (i.e., the content of the URL, HTML page and cookies), as well as a *session object* in the server storage. Since the HTTP protocol is stateless, every transaction has the same starting point, which reduces the problem of state maintenance considerably. Besides the ability for the web browser to maintain dialog state, many programming environments offer the server to maintain a session object which is preserved across service invocations, sometimes even across server incarnations ("restarts").

This model can be ported to a SIN environment, where all satellites of the constellation offer the same service, and the session object is made accessible to the new service provider during the handover operation. The session object can be stored in one dedicated satellite or on the ground to which there exist a route, distributed across a small set of satellites, or moved to the new serving satellite in its entirety.

Although these methods seems viable, they should be evaluated in terms of their communication requirements. A SIN will always be under-provisioned with regard to the Earth's population, so a close look at scalability properties is essential to the analysis.

III. Replication of Service Code

Following a handover operation, the client expects to find the same set of services in the newly connected satellite. The service code must be deployed and maintained in every satellite, which is not expected to consume much communication resources. The handover operation resembles what would be called *Process Migration* in the Distributed Systems literature.

 ISBN: ISBNFILL

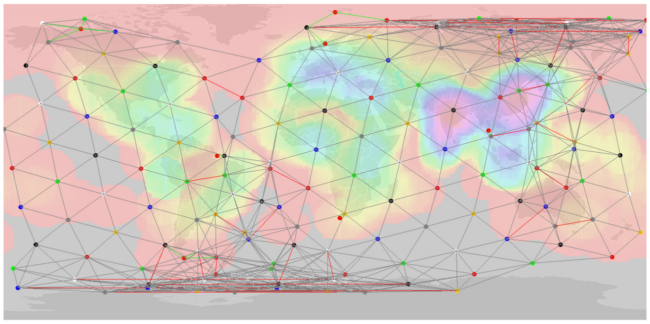


Figure 1. Screenshot from the satellite constellation model.

The Docker architecture has been studied for this purpose. Even though Docker was not designed for a SIN environment, the *Docker Swarm* [7] platform has appealing properties with regard to SIN operation.

In a Docker Swarm, every participant is an endpoint for a given service. All participants cooperate through an *overlay network* and distribute incoming service requests for the purpose of load balancing and failure recovery. An arrangement of a SIN as a Docker Swarm will allow clients all over the world to compete for the same computing resource pool, regardless of their location. Idle resources in satellites momentarily flying over inhabited areas may be employed to offload busy satellites. The price for this is the traffic volume in the overlay network, which will compete for the communication capacity in the constellation. Please observe that a Docker Swarm only supports *stateless* services.

IV. The Software Model

The results presented in this article are based on a software simulation of a satellite constellation. A screenshot from the model is shown in Figure 1. The constellation consists of 150 satellites at 500 km altitude.

The colored backdrop in the figure indicates the population density inside the satellite footprint at a given location, based on gridded population data from NASA [8]. This data set has also been used to calculate the graph in Figure 2, discussed in Section V-D.

V. Strategies for Session State Maintenance

The maintenance and transport of session state between satellites during a handover operation can be divided into three different techniques:

1) Keep the session object separate from the service component, e.g. in a computer on the Earth's surface, while the service client provides a reference to its existence through, e.g., a cookie value.
2) Move the entire session object proactively from the outgoing satellite to the oncoming satellite during the handover operation.
3) Move elements of the session object from former satellites to current satellite *on demand*, so that the session object is effectively distributed across a trail of past connected satellites.

These alternatives will now be discussed in more detail.

A. Separate and stable session object

Keeping the session object in the same location contradicts the entire idea of using satellites as units for information processing and sharing, and reduces the constellation to an ordinary communication network for information routing in a multi-hop infrastructure. This alternative is therefore abandoned for not meeting the required objectives.

B. Proactively moving entire session object

From a functional perspective, the proactive moving of the entire session object will meet the requirement to offer access to the session object elements at any time, with the shortest possible access latency. From a non-functional perspective, this alternative will consume unnecessary many resources and will not scale well. The reason for this is twofold: (1) The people of the Earth are concentrated in small regions, and a group of clients are likely to employ a small fraction of the satellite constellation, which will have to carry a majority of the session objects while the rest of the satellites are idling. (2) The links between satellites in these densely populated areas will be disproportionately loaded with traffic related to session object traffic, while other links in the constellation will be left unemployed.

These two observations both cause the SIN to scale poorly, due to the ineffective workload distribution of both storage capacity and link capacity. This alternative is taken into account in the traffic simulation results shown later, mostly for the sake of being a baseline for comparison.

C. Move session object elements on demand

The third alternative up for discussion is to defer the movement of session object elements and fetch them from past satellite connections on demand. Access operations to the session object elements are not expected to be uniformly distributed across the elements, but follow a *Scale Free Distribution* (SFD) [9], also known as Zipf's law. According to SFD, the frequency (f) of accesses to an element is inversely proportional to the *rank* (r) of the element. Applied to this use case, SFD predicts that the most frequently used element will be accessed twice as often as the second most frequently used element, three times more often than the third most frequently access element, and so on. Mathematically, this may be expressed as

$$f = \frac{a}{r} \tag{1}$$

where a is given a value so that

$$\sum \frac{a}{r} = 1 \tag{2}$$

Assuming SFD, the elements of a session object are then expected to form a "wake" along the row of past connected satellites, where the least frequently used elements are stored in the earliest connected satellite. Frequently accessed elements may be accessed faster because their access path is shorter. This arrangement is quite similar to the concept of Virtual

 ISBN: ISBNFILL

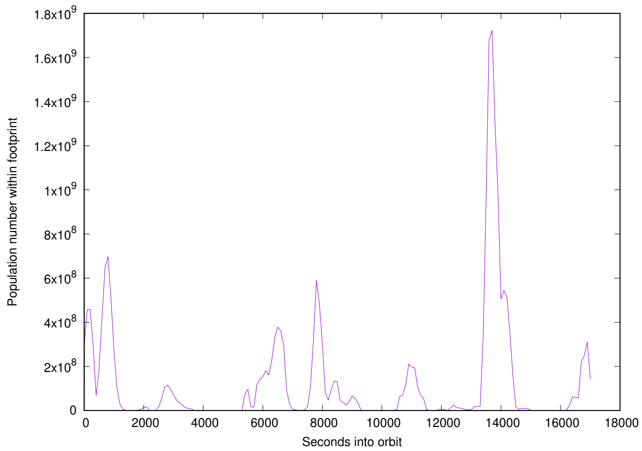


Figure 2. The population number inside the footprint of a satellite during three subsequent orbits.

Memory in operative system design, where the placement of a data element in a *storage hierarchy* is decided by its access frequency. The implementation of this arrangement is inspired by the OS design of Virtual Memory.

D. Employment of idle satellites

The distribution of the Earth's population density is highly uneven. The graph in Figure 2 shows the population size inside the satellite footprint over three consecutive orbits. The population size represents an estimate for the satellite's workload at that location. As shown in the figure, the "rush hour peaks" are so brief that most busy satellites will probably have an idle neighbor, or at least a much less busy one. This observation suggests that a distributed representation of the session object will utilise idle resources better than a session object kept in only one busy satellite. A simulation based evaluation of this scheme will be presented later in the article.

VI. Session state management based on Paging

Memory Paging in Distributed Systems, as it is known from Operating System design, is an on-demand replication technique. It is well understood under the name *Distributed Virtual Memory* [10]. For the purpose of on-demand replication of session state, the technique will be investigated with the following requirements in mind:

- Consistency and convergence
- Latency of page fault handling
- Volume of generated traffic

In this proposed solution, each service in a satellite will maintain their own *Page Table* (PT) for the storage used to keep the session state. Each entry in the PT will contain a reference to a *Page* stored in a *Page Frame* (PF) in one satellite. Once a page is to be retrieved, the PT is inspected to see if it is stored locally. If it is, the content is returned to the caller. Otherwise, the page is located in a distant PF and copied to a local PF (by the data routing service using the inter-satellite links), then removed from the distant PF. The PT is then updated and the page content returned to the caller. The resulting distribution of pages based on this arrangement is shown in Figure 3.

TABLE I
Distribution of pages across satellites before each handover.

satellite total	0	1	2	3	4	5
336	336					
499	172	327				
608	118	168	322			
704	91	113	157	343		
763	67	94	108	164	330	
806	54	73	86	110	173	310

If the page in question does not exist, it is created in a local PF and referred to in the PT. During handover, the new satellite will inherit the PT, but not the content of the PFs. Subsequent page requests will therefore cause the frequently used pages to be moved first, while other pages may never be moved, and will be deleted by the end of the session.

The distribution of pages has been simulated with these parameters:

- 5 handover operations, involving 6 satellites
- 1000 accesses to the session state object to each satellite
- PT has 1000 entries (for page references)
- Pages are accessed according to SFD

The resulting distribution is shown in Figure 4. The sum of all numbers (806) shows that far from every page in the session state object were ever accessed, and existing pages remaining in PF of previous satellites indicate that they were never accessed since that satellite's time of service.

A. Scalability properties

The number of hops for pages being moved through a path of satellites is chosen as an indicator of the scalability properties of the chosen session state management arrangement. Therefore, the number of single hop movements of pages will be analysed under the scenario described in the previous section. The resulting number from the proactive and the on-demand replication will be compared and reported.

The distribution of the session state pages across the current and past satellites was measured just before a handover operation, after 1000 access operations. The numbers are shown in Table I. From these numbers, it is possible to calculate the total number of page movements across inter-satellite links during the scenario of 5 handover operations with 1000 access operation between each.

For the proactive page movement method, the total number of page movements is 2910 (the sum of the 5 first numbers in the "total" column). For the on-demand method, the total number is 1162. This means that the on-demand method consumes only 40 % of the communication capacity required by the proactive method. These numbers are also shown in Figure 5.

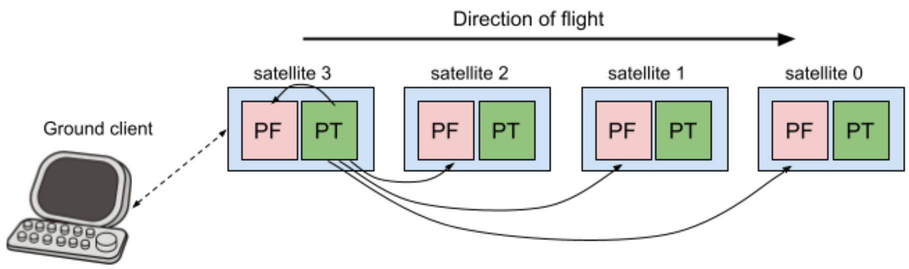


Figure 3. Pages are distributed in PFs along the trail of satellites, and referenced from a single PT.

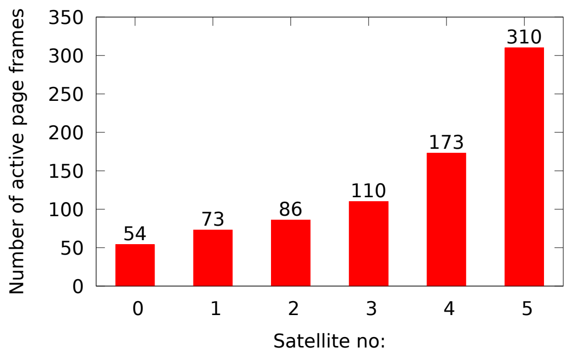


Figure 4. The distribution of session state pages after 5 handover operations.

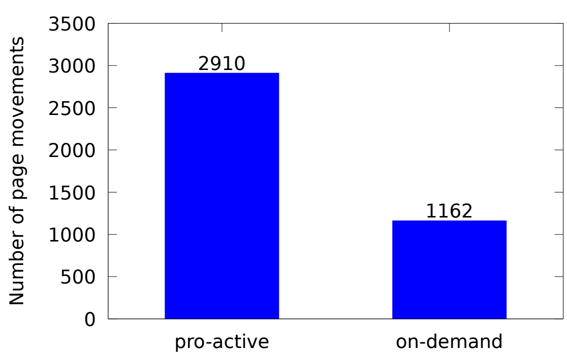


Figure 5. The number of session state page movements for pro-active and on-demand method

VII. Discussion

The results from the simulation experiment support the assumption that an on-demand paging arrangement saves a lot of communication resources, even though page retrievals through several hops may increase the access latency.

A. Inclusion of population density

The population density should be taken into account for our scalability analysis, as pointed out in Section V-D. The population density data was included in the early stage of the experiment and showed the expected results. On the other hand, the results are highly variable depending on the chosen location on the Earth, so an average value will not provide useful information, and are not included in the article.

B. Session state as a file/memory system

The data elements of a session state object have varying sizes and do not fit well with the arrangement of pages with fixed size. Two possible improvements on this matter are:

- Provide access to the session state pages through a file API. On a Linux computer, this requires the construction of a *Block Device Driver*, which involves kernel module programming. This option is left for future investigation.
- Provide access as a virtual memory block. This solution requires modification to the virtual memory management as well as access to the PT and related structures inside the CPU, and involves modifications of the OS kernel itself. This option is therefore considered infeasible.

VIII. Conclusion

In the presented article, the problem related to stateful service components in a cloud of satellite-based services providers has been analysed. The proposed solution has been evaluated and found to be sound and effective. Remaining problems include byte-level access to session state pages, and solution to possible satellite faults.

References

[1] S. Briatore, N. Garzaniti, and A. Golkar, “Towards the internet for space: Bringing cloud computing to space systems,” in *36th International Communications Satellite Systems Conference (ICSSC 2018)*, 2018, pp. 1–5.

[2] L. Bai, T. de Cola, Q. Yu, and W. Zhang, “Space information networks,” *IEEE Wireless Communications*, vol. 26, no. 2, pp. 8–9, 2019.

[3] A. Fongen, “Application services in space information networks,” in *CYBER 2021*. Barcelona, Spain: IARIA, Oct 2021, pp. 113–117.

[4] A. Fongen, “Trust management in space information networks,” in *SECURWARE 2021*. Athens, Greece: IARIA, Nov 2021, pp. 14–18.

[5] A. Fongen, “Cooperative caching in space information networks,” in *INTERNET 2022*. Vienna, Italy: IARIA, May 20212, pp. 1–5.

[6] A. Fongen, “Population-based routing in leo satellite networks,” in *MOBILITY 2022*. Porto, Portugal: IARIA, June 2022, pp. 1–4.

[7] A. Modak, S. D. Chaudhary, P. S. Paygude, and S. R. Ldate, “Techniques to secure data on cloud: Docker swarm or kubernetes?” in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018, pp. 7–12.

[8] “Gridded population of the world v.4.11,” [Online; retrieved 09-Nov-2022]. [Online]. Available: https://sedac.ciesin.columbia.edu/data/collection/gpw-v4/sets/browse

[9] A.-L. Barabasi, *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*. Plume Books, April 2003.

[10] C. Morin and I. Puaut, “A survey of recoverable distributed shared virtual memory systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 8, no. 9, pp. 959–969, 1997.

On the Study of Internet Ossification and Solution

Lin Han, Richard Li
Futurewei Technologies, Inc.
Santa Clara, California, U.S.A
email: {lin.han, richard.li}@futurewei.com

***Abstract*— The current Internet is based on IPv4 and IPv6. It has been in service for many years and is very successful. However, it is facing challenges in protocol ossification, security, and service quality. Recently, the geographical tension, trading confrontation, the regulation for data protection and localization have raised decentralization requirements for the Internet. This paper analyses the factors for the Internet ossification and proposes a new architecture that is distributed based on region or country. It can maintain the support of the current IPv4/IPv6, and provide more flexibility for the protocol, thus mitigating the ossification of the Internet.**

Keywords- Future Internet; Ossification; Decentralization; Distributed; Internet Protocol.

I. INTRODUCTION

The Internet has penetrated everywhere in our life and has provided tremendous momentum to the development and progress in communication, technology, culture, and economy. The current Internet is based on IPv4 [1] and IPv6 [2] protocols, and consists of many other protocols for different areas, such as address assignment, domain name service, routing and switching, security, transport. All these protocols are governed by the Internet Engineering Task Force (IETF). In the document thereafter, the name IP represents both IPv4 and IPv6.

However, the Internet's deficiency and ossification are also noticed. This includes slow evolution, protocol ossification, resource allocation unfairness, security and privacy concerns.

The paper briefs our research on a new architecture for the Internet and associated protocol structures. It can provide extra flexibility for the Internet while maintaining the current IP based technologies and services. Internet ossification can be mitigated by a new architecture including distributed Internet resource management and domain name service, free choice of address type, and heterogeneous communications.

The rest of the paper is structured as follows. In Section II, we present an overview of the Internet architecture and protocols. Section III discusses the Internet ossification and analyzes the root causes. The technical factors are analyzed in Section IV. Our new network protocol is proposed in Section V. Section VI presents the detailed architecture with the new network protocol. The compatibility issues are discussed in Section VII. Section VIII summarizes the benefits of the new proposal. Section IX concludes the paper and gives further research directions.

II. OVERVIEW OF THE INTERNET

The Internet is the global system of interconnected computer networks that uses the Internet protocol suite to communicate between networks and devices [3]. Recently, with the growth of 5G [4], Internet of Things (IOT) [5], Non-Terrestrial-Network (NTN) integration [6], the Internet has become the communication infrastructure that almost every person, every device and everything can be connected to. The Internet scope is very broad and has a couple of key fundamental blocks:

- The definition of IP address, the mechanism to allocate and assign the IP addresses. There are two types of IP addresses, one in IPv4 and another is IPv6. Currently, IPv4 is in the process of becoming obsolete from the perspective of IETF, and IPv6 is the only supported address. The IP address (except the local address and non-routed address) is globally significant and unique in the world. It is allocated by the Internet Assigned Numbers Authority (IANA) [7] to each region and country. There are five Regional Internet Registries (RIRs). Each RIR has a couple of Local Internet Registries (LIRs) or National Internet Registries (NIRs). They are responsible for the allocation of the IP addresses block on their authorized areas. Figure 1 and Figure 2 show the hierarchical architecture of IANA.

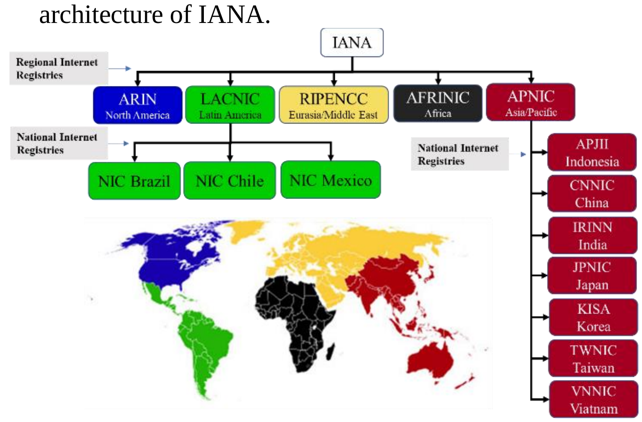


Figure 1. The hierachy of IANA architecture.

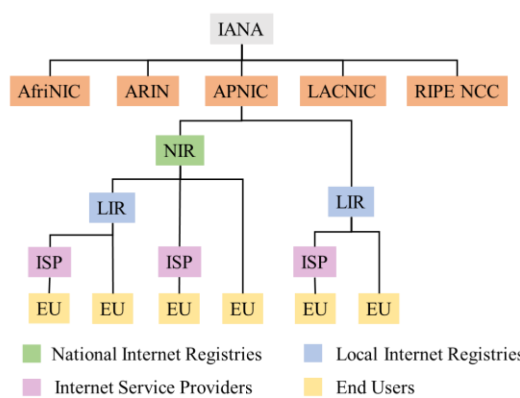


Figure 2. Understanding address management hierarchy [25].

- The definition of Asynchronous System Number (ASN) [8], and the mechanism to assign ASN. ASN is used for Border Gateway Protocol (BGP) [9] to represent autonomous systems across the Internet. Similar to the IP address, the public ASN is also globally significant. It is managed by IANA. ASN is key to BGP, which is the critical protocol for the inter-connection and inter-working of different networks distributed globally. BGP will exchange the global IP address of different networks, thus making every global IP address reachable from anywhere around the world.
- The definition of Domain Name, the mechanism to manage Domain Name Servers and provide the Domain Name System (DNS) [10] Service. Similar to the IP address, the Domain Name is also globally significant. The DNS root zone management [11] and DNS root servers [12] are managed by IANA as well. Domain Name and Domain Name Servers are distributed globally. There are thirteen DNS root servers located in the U.S.A. Different leaf servers belonging to different regions and countries are deployed globally. In addition to this, some countries may have mirror root servers in their own region to back up the root server and speed up the DNS services.
- The protocols to control the Internet. The fundamental protocols are IPv4, IPv6 and many other protocols on top of IPv4 and IPv6. Excluding protocols on L2 that are controlled by the Institute of Electrical and Electronics Engineers (IEEE) and the International Telecommunication Union (ITU), the protocols for Internet include layers from L3 to L7 that are controlled by IETF. There are thousands of protocols related standards that are called RFC (Request for Comments) documents, e.g., more than 500 RFC for IPv6 has been published. Below just lists a very small portion of RFCs and very typical protocols:
    1. Host configuration related protocols (ND [13], DHCPv6 [14], etc.)
    2. L3 or routing protocols (BGP, IS-IS [15], OSPF [16], etc.),
    3. Traffic Engineering (MPLS [17], RSVP-TE [18], SRv6 [19], etc.)
    4. L4 or transport protocols (TCP [20], UDP [21], etc.),
    5. Upper layer protocols (QUIC [22], TLS [23], HTTP [24], etc.),

III. INTERNET OSSIFICATION

The Internet was essentially designed with simplicity and scalability. [26] has detailed an analysis of how this is achieved and lists the important timeline for Internet evolution. After the Internet has become available to the public in the 1990s, it experienced more than 40 years of development of technology. Gradually, the evolution of the Internet became slower and slower. There are less and less new technologies and services coming up for the Internet, especially for the parts of infrastructure and fundamentals. The structure of the Internet has become more rigid and difficult to change over time, and this sometime is called Internet ossification. For example, IPv6 was designed to replace IPv4, but this has not been accomplished since the first IPv6 standard RFC 2460 [27] was introduced in 1998. Even right now, there are still arguments that IPv4 should not be obsoleted [28], and the adoption of IPv6 at the level of Service Providers is still slow.

There are some research works that proposed a new or enhanced architecture for the Internet, such as RINA [29], SCION [30], New IP [31], IPv10 [32], and Extensible Internet (EI) [33][34]. Detailed analysis and comparison of the proposals of RINA, SCION and New IP can be found in [35]. IPv10 is to allow the communication between IPv6 and IPv4. EI introduces Layer 3.5 between L3 and L4 to provide services that were not available in the current Internet architecture.

Two categories of factors associated with management and technical solutions can contribute to Internet ossification:

- Consensus challenges:
  The Internet is a huge global network. Many technical definitions, solutions, and changes are globally significant. Any decisions or changes about its development, operation and deployment involve a wide range of stakeholders, including governments, organizations, operators, and individual users. Reaching consensus on changes can be very difficult and slow, especially when there are competing interests or different priorities. As a comparison in the standardization in wireless area, 3GPP has finished the 5G (the fifth generations of wireless technology) in almost the same period that IETF has not completed the IPv4 to IPv6 transition.
- Technical solutions:
  Due to the vast number of users, devices and applications, the Internet has accumulated many technical feedbacks and problem reports. Completely fixing those problems or enhancing the existing solutions are always slow. Some quick fixes are implemented in the short term but may need to be addressed or replaced later on. The slow global consensus on any problem fixing, new enhancements or features, can make it more difficult to change any piece of the Internet's infrastructure. The Internet is a complex system that involves many different networks, technologies, and standards. How to drive the Internet moving forward but maintain the previous investment is not only a business objective but also a technical challenge. Ensuring compatibility between these different elements can be difficult, and changes to one part of the system may have unintended consequences elsewhere. Due to this reason, people are always conservative and hesitate to adopt new technologies.

Overall, Internet ossification has made it more difficult to adopt new technologies and hinders the Internet's ability to continue evolving and progressing.

IV. DESIGN FACTORS FOR INTERNET OSSIFICATION

Even there are many factors, technical or non-technical, contributing to the Internet ossification. We think some short-term design of the Internet has made the Internet less flexible at the beginning, thus is one of the most important factors we need to consider when thinking about the future architecture.

The following are some technical perspectives that contribute to the Internet ossification.

- The Internet resources (IP address, ASN and Domain Name) assignment and management are essentially a centralized hierarchical architecture. The problem of this centralized architecture is that (1) IANA and Regional Internet Registries are both non-profit organizations that do not have any jurisdiction. (2) The Internet resources are hardly allocated fairly, for example, the IPv4 address block is not enough in some countries but more than required in other countries. (3) Address preference is not the same in different regions, countries, operators, users, and applications. For example, IPv4 is still preferred by many service providers. That is one reason why IPv6 deployment is so slow. (4) A centralized architecture makes the Internet fragile when the geopolitical tensions are high. In the recent events of war and trading confrontation, some voices to stop the Internet service to a specific area have been heard and have put a threat on the integration of the Internet.
- Since IP address is globally significant, it requires that all end-user devices and network devices use IP as unique format for the data packet header. All L3 devices should follow the same principle to process IP packets and provide the services to the upper layer. This design is called "narrow waist". Obviously, it has benefits in simplicity and scalability, but it becomes one factor contributing to the Internet ossification, since any changes in the IP header will have a global impact and it will be hard to get consensus in IETF.
- From the IP packet forwarding perspective, the IP based Internet is flat. All Internet packets are forwarded based on IP address lookup; thus, all globally reachable IP addresses must be stored in every network device (even in MPLS network, the Provider Edge Routers also must store all reachable IP prefixes). This can result in two problems: (1) a huge amount of IP addresses or prefixes storage leads to huge lookup table size. (2) BGP, the only protocol to exchange the global IP reachability between different networks in different regions or countries, must process huge numbers of global IP prefixes.

V. CONSIDERATION OF NETWORK LAYER

From the analysis in Section IV, we can see one of the major factors for Internet ossification is the IP design is too rigid. Such rigid design was partially because the hardware or semiconductor performance was limited in the 80s and 90s in the last century. To achieve the line rate of packet processing, it is hard to give too many flexibilities in the address and functions in the packet header, e.g., the address type and size, the extensions, and options. After many years' development, the semiconductor industry has progressed a lot. Recently, high-performance chips with programmability have been commercialized. It is time to think about what we can do from a technical perspective that can mitigate the Internet ossification.

As a global data communication network, the Internet is supposed to be only responsible for the inter-connection between different networks in the world. The networks could be for enterprises, Internet Service Providers (ISPs), a country or a region. Let us compare the similar situation in the phone network and the mail system. For those two global communication systems, there is no restriction on how to define a local phone number, and local address format. The international community only needs to get consensus on the country code for international calls, or the country names for global mail delivery. Each country will manage and design its own structure of phone numbers, mail addressing system and delivery infrastructure. Similarly, we can have a high level of design principles for the Internet in the future:

- The Internet should have more democracy and freedom, less restrictions and centralization.
- The Internet should be distributed globally based on region or country. All regions are equal and there is no central control. No region can impact other's decision in address selection, peering and service.
- Small countries can decide to form a region if the countries do not want to be independent in Internet resource and DNS management due to economic or other constraints.
- Each region has the freedom and authorization to manage the Internet resources used locally, such as address selection, address allocation, ASN allocation, domain name registration, DNS root server, etc.
- The Internet should support heterogeneous address types and communications.

By using such a principle, we can design a new network protocol packet header for the Internet, as shown in Figure 3. The following packet format is preliminary and only for illustration. A final design will decide the detailed coding. This new packet would be on top of Layer 2, thus, a new EtherType assignment from IANA is required.

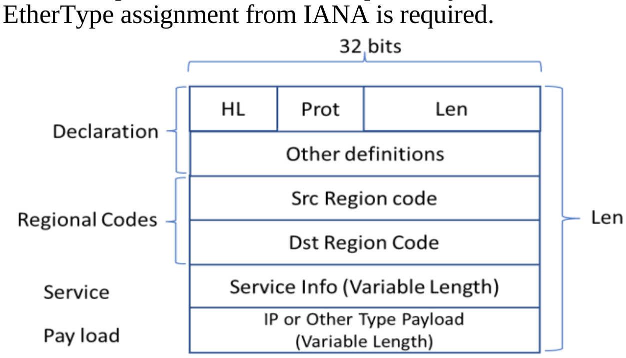


Figure 3. New Internet protocol packet header.

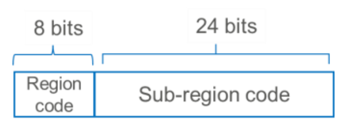


Figure 4. The Region Code Example.

- Declaration: This field defines the basic information about the packet, it may contain the following essential information:
  1. HL: Hop limit. This value is decremented by one at each forwarding node and the packet is discarded if it becomes 0 (except on the last node).
  2. Prot: The protocol number for payload. It could be a protocol number defined currently by IANA, e.g., IPv4 or IPv6, TCP or UDP, or a new protocol number defined in the future.
  3. Len: Total length of the packet including the Pay Load. The unit can be defined in standardization.

 ISBN: ISBNFILL

4. Other definitions: other definitions for the packet header will be defined later.

- Regional codes: This field may contain the "Src (Source) Region Code" and "Dst (Destination) Region Code" for source and destination. The size, code structure and detailed coding should be standardized by an international organization. It could contain a region or country code that was defined by ITU E.164 [36] and has its own hierarchy, e.g., region, sub-region, and more granular definitions. See Figure 4 as an example. Only the 8-bit "Region Code" needs to be standardized by an international organization. The "Sub-region code" will be managed locally in the region.
- Service: This field contains information about the service and is to be defined. Its length is variable.
- Payload: This part contains the payload whose type is specified by the protocol number defined in Declaration. The Payload could be IP type or any other types for L2 to L4.

VI. ARCHITECTURE FOR INTERNET BASED ON NEW PROTOCOL

A. Internet Resource Management

The Internet resources will include region code, IP address space or other type of address space, ASN, and protocol number. The management of those resource is based on the following rules:

International organization managed items:

- The Region code structure and Region code assignment are the responsibility of an international organization, ITU or IANA.
- For the protocols that the interconnection between different regions or countries are supported, e.g., the new protocol defined by this paper (new EtherType), IPv4, IPv6, Ethernet, MPLS, etc., the protocol numbers are still managed by the international organization IANA.

Regional authority managed items:

- Each region or country will be responsible for the sub-region code assignment and management.
- Each region or country will be responsible for the IPv4/IPv6 address and ASN number allocation and management for its own jurisdiction area. Different regions or countries may have different policies and schemes to manage the resource.
- Each region or country can use the whole IPv4/IPv6 address and ASN space. All addresses only have local significance in the region or country, thus different regions or countries may have the same address.
- Each region or country can define new protocol numbers that are only used locally within the region or country.

B. Scope of New Protocol

The new protocol applies to the Internet connection between different regions and countries, as shown in Figure 5. It does not restrict communication within the region or country. The current IPv4 and IPv6 can still work. A region or country can define and run a new version of IP without any interruption or interference to the whole Internet. For example, IPv10 to support communication between IPv4 and IPv6 was proposed in IETF, but was not accepted. With the new protocol, one region only needs to get consensus on IPv10 in its own sovereignty and then use it within the region.

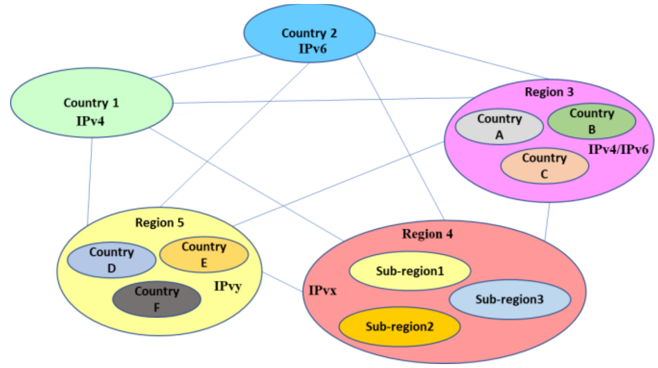


Figure 5. Internet based on new network protocol.

It is important to note that a region can also use the new region-based protocol for communication within its own territory (see the communications between sub-regions in Region 4 in Figure 5).

C. Domain Name Service

The Domain Name Service architecture is similar to the current DNS hierarchy architecture. Figure 6 illustrates the new DNS architecture and Figure 7 demonstrates a DNS request and response crossing different regions or countries. The major difference with the current architecture is that the current centralized DNS root zone and root servers are removed, thus is a distributed architecture. Following are the details:

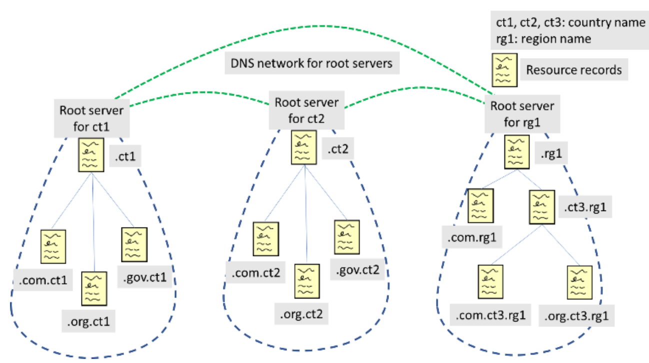


Figure 6. Domain Name System architecture.

- Each region or country will have its own DNS root server and different root servers from different regions or countries are fully equal and there is no central control, thus the current DNS root zone and root servers are not needed.
- All DNS root servers are connected virtually to form a DNS network. The network may run a protocol to exchange DNS information for all root servers. This

network will overlay on top of either existing IP or the new network protocol proposed in this paper.

- The connection between all DNS root servers is fully meshed virtually. Any connection between two servers is voluntary and only managed by the two servers' regions or countries. When a new root server for a region or country joins the network, it should have agreement and then connection with existing root servers.
- The ".region" or ".country" domain is the only Top Level Domain (TLD) for the region or country. All other domain names are lower-level domains.
- The ".region" or ".country" suffix is needed when the DNS requester and real domain name are in a different region or country. The suffix can only be omitted when the DNS requester and the real domain name are in the same region or country.
- A domain name with a ".region" or ".country" suffix is always associated with an address physically located within the region or country.

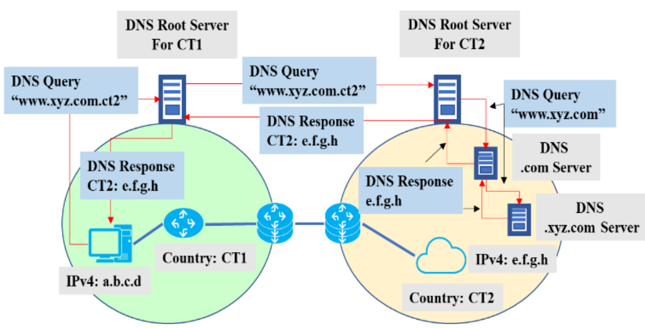


Figure 7. DNS service crossing different regions or countries.

The DNS service will have some corresponding implementation changes with the new architecture. Also, there are some regulation or legal issues involved, e.g., a company name in a "domain name" in a different region must be approved by the local authority.

Here is an example: An international company xyz has the headquarter in the country named "ct1". Then, the domain name "www.xyz.com.ct1" always points to an address assigned by the DNS authorization in the country ct1. In another country ct2, if there is a branch or service from the company xyz, the DNS request of "www.xyz.com" from ct2 will return an address information found in the name server ".com" in the country ct2. If there is no registration for the company in ct2, the DNS request of "www.xyz.com" from ct2 will return null.

Due to the bonding of a name and IP address in every region physically, the new DNS mechanism will make the Internet service localization more transparent and easier to be compliant with the local regulation or laws.

D. *Communication Between Regions or Countries*

To provide interconnection between different regions or countries using the new network protocol, proper control plane and data plane must be defined.

1) *Control Plane*

- The border devices connecting different regions need to support the new control protocol.
- The new control protocol will exchange information about the interconnected border devices, the associated links, the region code, the reachable end-user's address details, etc.
- The new control protocol could be a link-state routing protocol like IGP, or a path-vector protocol like BGP.
- The new control protocol must also be running within a region or a country to populate the information learnt from border devices about the outside interconnected networks of other regions or countries, e.g., the links that can reach other regions or countries, the associated remote reginal code, the remote reachable address associated with the regional code, etc.

2) *Data Plane*

- For the egress region, where the traffic is originated from, the data packet forwarding is based on the lookup of "Region/Country code" at all network devices. See the country CT1 in Figure 8.
- For the ingress region, where the traffic is destinated to, the data packet forwarding is based on the lookup of "the address of payload" at all network devices. See the country CT2 in Figure 8. In the example, the "address of payload" is IPv6 address.
- For the transit region, there are two approaches: one is Transparent Mode, another is Tunnel Mode.

  1. For Transparent Mode, the data packet forwarding is based on the lookup of "Region/Country code" at all network devices in a transit region. See the country CT3 in Figure 8.

  2. For Tunnel Mode, the data packet forwarding is based on the lookup of "Region/Country code" at edge network devices in a transit region. Proper packet encapsulation (at ingress router) or decapsulation (at egress router) are needed. See country CT4 in Figure 8. In the example, the IPv4 tunnel is used and IPv4 address lookup for the tunnel is done on every network device within the region.

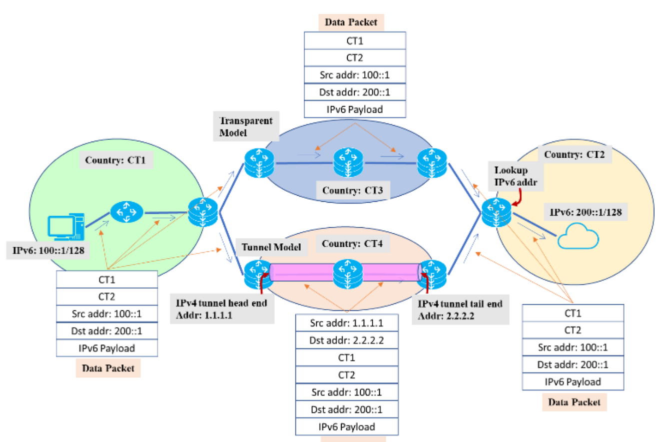


Figure 8. Homogeneous communication: Transparent Mode and Tunnel Mode (only the essential parts of packet header are shown).

- For all scenarios, a very small table is needed to store all "Region/Country codes" for the communication crossing regions. The table lookup will use "exact match". These two behaviors are different than the IP prefix lookup, which needs a huge amount of table to store global IP prefixes, and the lookup is the Longest Prefix Match using TCAM (Ternary Content-Addressable Memory).

3) *Heterogeneous Communication between Regions or Countries*

The above discussions are about the homogeneous communication between regions or countries, or the address type are the same for all end users.

The new network protocol and architecture can support heterogeneous communication worldwide. Heterogeneous communications are communications with different types of addresses. This is very useful to many applications in security, privacy, Internet of Things (IoT), etc. Below are some supported address combinations for heterogeneous communication:

- Different length of IP for source and destination, e.g., IPv10 or other type of IP that the address length is not 32-bit and 128-bit.
- Different type of address for source and destination, e.g., between Ethernet and IP.
- No source address, the source address is hidden in the application data.
- Variable length public key as address.

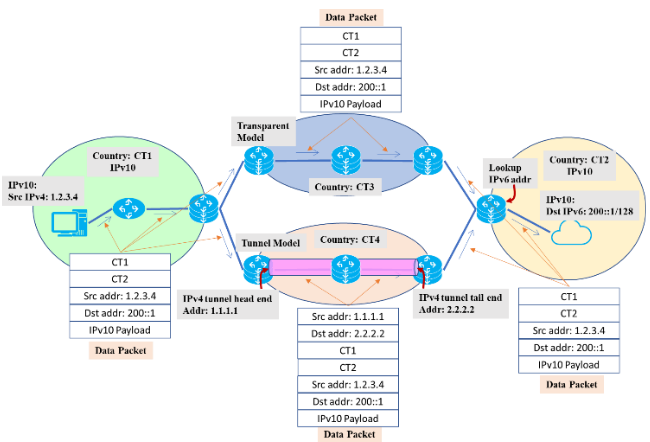


Figure 9. Heterogeneous communication: Transparent Mode and Tunnel Mode (only the essential parts of packet header are shown).

Figure 9 illustrates the data plane for a case where IPv10 is supported in country CT1 and CT2, and how an IPv4 host in CT1 sends data to IPv6 host in CT2. For the IPv10 case, both IPv4 and IPv6 addresses are supported, thus the lookup of IPv6 in CT2 is obviously supported. We can see that, to support IPv10, only communication participants (CT1 and CT2) need to have an agreement to support it. This is much easier to have a global consensus to support IPv10.

VII. Compatibility Issues

The major changes of the Internet based on the proposed new network protocol are the Internet resource management, the DNS architecture, and the use of the new network protocol.

For the communication or IP service within the same region or country, the current IP based Internet service can still be used, and there is no compatibility issue. The new Internet resource management and new DNS architecture have very little impact on the end-user applications and network operation, i.e., some provisioning (to the DNS server and domain name management) may need to be changed.

For the communication or IP service crossing different regions or countries, the new network protocol needs to be used, and it is not compatible with the existing IP, but we can maximize the current Internet investment through the detailed design of a new network protocol header.

It is easy to notice that the new network protocol packet header is very similar to the IPv4. This is intended to make the future design easier to be implemented in IPv4 capable hardware. We have two options in the final design of the packet header encoding: (1) re-use the IPv4 packet header for the new network protocol, or (2) only re-use the 32-bit IPv4 address space for the region code and redesign other fields in the packet header. Since the current IPv4 header has design flaws in some areas, such as: (a) The protocol is not extensible due to the limited IPv4 option size, (b) The header checksum is not required, (c) Fragmentation is not a good design, then, we prefer option (2): define the 32-bit source and destination region codes and redesign other fields in the packet header.

With the above design considerations and coupled with the redesigned protocol running between regions, by minimal re-programming, the existing hardware can be easily re-used for the future Internet.

VIII. Benefits and Advantages of New Network Protocol

A. *Benefits*

The proposed new network protocol is only for the interconnection between regions and countries. The Internet based on the new protocol will have the following benefits:

- Much less restriction at the protocol for interconnection: The new network protocol only defines the regional interconnection mechanism that is based on regional codes but does not limit the communication address and communication mechanism within a region or a country, thus reducing the restriction caused by globally uniformed IPv6 header for global network. Heterogeneous communication support will be easier to achieve between interested parties.
- Minimized changes on the current Internet architecture: The current IPv4 and IPv6 protocols and data forwarding can still work in a region or country. DNS changes very slightly. The architecture of the IP based Internet is kept, and the investment is not wasted.
  The control protocol and data forwarding for the interconnection between regions and countries can be realized based on an extension of existing IP routing protocols and IP packet forwarding. It needs minimal investment.
  Existing and future IP based applications within a region can still run without any feeling that the underlayer networking is changed for the interconnection between

regions. The application to reach outside of a region just needs minor modifications for the address format to include the regional codes.

The routing table size will be dramatically reduced due to the fact that routers in a region will only keep the prefix defined in the region. All addresses to outside of a region can be summarized as regional codes.

- Independent technology evolution:
  With the new network protocol, the Internet technology can evolve in different regions or countries independently. It is expected to be much easier and faster than the current situation when a global consensus is needed, thus will mitigate the Internet ossification a lot.
- Distributed Internet resource management and DNS:
  The new Internet resource management and DNS are distributed and based on sovereignty and jurisdiction, thus have no legal obstacles to making the regional Internet technologies adaptive to local laws or regulations. It will make any security, privacy changes or enforcement much easier and faster to implement.
  The new Internet resource management and DNS root servers are distributed and fully controlled by a region or country. The Internet service of any country will not be impacted by other countries. It makes the Internet more robust and resilient to any disasters and geopolitical interruption.
  The new distributed Internet resource management also makes each region or country able to use the whole IP address space and ASN space. This will not only eliminate the unfairness issues in IP address allocation, but also expand the IP address resource for all countries.
  The new architecture and network protocol gives each region or country full control and freedom on what type of addresses and communications are used for the Internet service within the region. This will eliminate the IPv4 to IPv6 migration mandates if IPv4 is preferred in a region or country. Also, other new types of addresses can be invented and adopted locally.
- Internet integrity is maintained:
  Internet fragmentation [37] is always a concern for new technology proposals. From a technical perspective, the new proposal does not impede the ability of systems to fully interoperate and exchange data packets. The Internet functions are consistent as before at all end points. Internet interoperability, universal accessibility, the reusability of capabilities, and permissionless innovation are all not impacted. While the data protection and localization from many regional regulations can be naturally satisfied by the architecture, more freedom in addressing can provide more possibilities for new technologies in security and privacy.

B. *Advantages*

Comparing with the existing proposals, RINA, SCION, New IP, IPv10 and EI, the new proposal has the following advantages:

- Unlike RINA and SCION, the new proposal is not a clean slate solution. It can keep the current IP based Internet service in a region or a country unchanged. It only impacts the interconnection between regions and countries. Considering most of Internet traffic is local and international traffic crossing borders of countries is relatively small, the impact to the current Internet service is limited. Additionally, for the impacted interconnections between regions, a proper migration strategy can be developed to upgrade inter-links individually to the new protocol and minimize the service interruption.
- The new protocol is orthogonal to other variations of IP, like New IP, IPv10 and EI. It can make those technologies easier to be adopted locally without global consensus and impacts.

IX. Conclusions And Future Work

This paper has proposed a new network protocol and architecture that can provide more flexibility and mitigate Internet ossification. The new architecture is distributed without any central control, thus making the Internet more robust and resilient to geopolitical interruption. It can also expand the usable Internet resources for each region and country. Meanwhile, the new proposal can keep the current IP based Internet in regions, thus it can minimize the impact on the Internet and maximize the old investments.

Further work is needed for detailed solutions in every area where new technologies or protocol redesigns are required, such as a protocol for distributed DNS, the control protocols and forwarding engine for interconnection between regions, upgrading and migration approaches, etc.

It must be noted that the purpose of this paper is to analyze the Internet ossification and possible solutions for future Internet. It is expected that any solution including the proposal in this paper will face a lot of questioning, challenges, and objections. But it is believed that doing something will be better than doing nothing. As the most important invention of human beings, the Internet can only be pushed forward after all interested parties join the work and contribute the ideas.

References

[1] "Internet Protocol," RFC 791, Internet Engineering Task Force, Sept. 1981.

[2] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 8200, Internet Engineering Task Force, July, 2017.

[3] "Internet," Merriam-Webster.com Dictionary, Merriam-Webster, [Online]. Available: https://www.merriam-webster.com/dictionary/Internet. [Accessed on Mar 7, 2023].

[4] 3GPP, "5G System Overview," [Online]. Available: https://www.3gpp.org/technologies/5g-system-overview. [Accessed on Mar. 7, 2023].

[5] "Internet of Things," Encyclopedia Britannica, Encyclopedia Britannica, Inc., [Online]. Available: https://www.britannica.com/science/Internet-of-Things. [Accessed on Mar 7, 2023].

[6] 3GPP, "Solutions for NR to support Non-Terrestrial Networks (NTN)," TS 38.821. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.821/38821-g10.zip.

[7] "Internet Assigned Numbers Authority (IANA)," [Online]. Available: https://www.iana.org/. [Accessed on Mar 7, 2023].

[8] "Autonomous System (AS) Numbers," Internet Assigned Numbers Authority. [Online]. Available:

https://www.iana.org/assignments/as-numbers/as-numbers.xhtml. [Accessed on Mar 7, 2023].

[9] Y. Rekhter, T. Li, and S. Hares, “A Border Gateway Protocol 4 (BGP-4),” RFC 4271, Internet Engineering Task Force, Jan., 2006.

[10] “Domain Name Services,” Internet Assigned Numbers Authority. [Online]. Available: https://www.iana.org/domains. [Accessed on Mar 7, 2023].

[11] “Root_Zone Management,” Internet Assigned Numbers Authority. [Online]. Available: https://www.iana.org/domains/root. [Accessed on Mar 7, 2023].

[12] “Root Servers,” Internet Assigned Numbers Authority. [Online]. Available: https://www.iana.org/domains/root/servers [Accessed on Mar 7, 2023].

[13] T. Narten, E. Nordmark, and W. Simpson, “Neighbor Discovery for IP Version 6 (IPv6),” RFC 4861, Internet Engineering Task Force, Sept., 2007.

[14] T. Mrugalski et al., “Dynamic Host Configuration Protocol for IPv6 (DHCPv6),” RFC 8415, Internet Engineering Task Force, Nov., 2018.

[15] C. Hopps, “Routing IPv6 with IS-IS,” RFC 5308, Internet Engineering Task Force, Oct., 2008.

[16] R. Coltun, D. Ferguson, J. Moy, and A. Lindem, “OSPF for IPv6,” RFC 5340, Internet Engineering Task Force, July 2008.

[17] E. Rosen, A. Viswanathan and R. Callon, “Multiprotocol Label Switching Architecture,” RFC 3031, Internet Engineering Task Force, Jan., 2001.

[18] Daniel O. Awduche et al., “RSVP-TE: Extensions to RSVP for LSP Tunnels,” RFC 3209, Internet Engineering Task Force, Dec., 2001

[19] L. Ginsberg, B. Decraene, S. Litkowski, and R. Shakir, “Segment Routing Architecture,”. RFC 8402, Internet Engineering Task Force, July 2018.

[20] J. Postel, “Transmission Control Protocol,” RFC 793, Internet Engineering Task Force, Sept. 1981.

[21] J. Postel, “User Datagram Protocol,” RFC 768, Internet Engineering Task Force, Aug. 1980.

[22] J. Iyengar and M. Thomson, “QUIC: A UDP-Based Multiplexed and Secure Transport,” RFC 9000, Internet Engineering Task Force, May 2021.

[23] E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.3”, RFC 8446, Internet Engineering Task Force, August 2018,

[24] E. Rescorla and A. Schiffman, “The Secure HyperText Transfer Protocol", RFC 2660, Internet Engineering Task Force, August 1999.

[25] APNIC, “Understanding address management hierarchy,” [Online]. Available: https://www.apnic.net/manage-ip/manage-resources/address-management-objectives-2/address-management-objectives/ [Accessed on Mar 7, 2023].

[26] J. Mccauley, S. Shenker, and G. Varghese, “Extracting the Essential Simplicity of the Internet,” Communications of the ACM, vol. 66, no. 2, pp. 64-74, February 2023.

[27] S. Deering, R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification,” RFC 2460, Internet Engineering Task Force, Dec., 1998.

[28] S.D. Schoen, J. Gilmore, and D. Täht, “IETF Will Continue Maintaining IPv4, ” draft-schoen-intarea-ietf-maintaining-ipv4, Internet Engineering Task Force, Sept. 2022.

[29] J. Day, Patterns in Network Architecture: A Return to Fundamentals, Prentice Hall, 2008.

[30] X. Zhang et al., "SCION: Scalability, Control, and Isolation on Next-Generation Networks," *2011 IEEE Symposium on Security and Privacy*, Oakland, CA, USA, 2011, pp. 212-227, doi: 10.1109/SP.2011.45.

[31] S. Jiang, S. Yan, L. Geng, C. Cao, and H. Xu, “New IP, Shaping Future Network: Propose to initiate the discussion of strategy transformation for ITU-T”, TSAG C-83, Sept. 2019.

[32] K. Omar, “Internet Protocol version 10 (IPv10) Specification,” draft-omar-ipv10, Internet Engineering Task Force, Sept. 2017.

[33] H. Balakrishnan et al., “Revitalizing the public internet by making it extensible,” ACM SIGCOMM Computer Communication Review, vol. 51, no. 2, pp. 18–24, April 2021.

[34] International Computer Science Institute, University of California, Berkeley, “An Extensible Internet for Science Applications and Beyond,” [Online]. Available: https://www.icsi.berkeley.edu/icsi/projects/extensible-internet/science-applications

[35] T. Eckert, L. Han, C. Westphal, and R. Li, “An Overview of Technical Developments and Advancements for the Future of Networking,” ITU Journal on Future and Evolving Technologies, vol. 3, no. 3, December 2022.

[36] “E.164 : The international public telecommunication numbering plan,” International Telecommunication Union. [Online]. Available: https://www.itu.int/rec/T-REC-E.164/. [Accessed on Mar 7, 2023].

[37] W. J. Drake, V. G. Cerf, and W. Kleinwächter, “Internet Fragmentation: An Overview,” [Online]. Available: https://www3.weforum.org/docs/WEF_FII_Internet_Fragmentation_An_Overview_2016.pdf. [Accessed on Mar 7, 2023].

BBR Performance over Variable Delay Paths on Multipath TCP Video Streaming

Masayoshi Kondo*, Dirceu Cavendish**, Daiki Nobayashi**, Takeshi Ikenaga**
*Graduate School of Engineering, **Faculty of Engineering
Kyushu Institute of Technology
Fukuoka, Japan
e-mail: kondo.masayoshi146@mail.kyutech.jp {nova@ecs, ike@ecs}.kyutech.ac.jp

***Abstract*—Video streaming makes most of Internet traffic nowadays, being transported over Hypertext Transfer Protocol/Transmission Control Protocol (HTTP/TCP). Being the predominant transport protocol, TCP stack performance in transporting video streams has become paramount, specially with regard to MultiPath Transport Control Protocol (MPTCP) innovation and multiple client device interfaces currently available. An important component of MPTCP is the packet scheduler, which selects on a packet basis the transport path to inject each packet. In this paper, we provide an extensive analysis of the Bottleneck Bandwidth and Round-trip propagation time (BBR) TCP variant when transporting video streams over Long Term Evolution (LTE) and Wi-Fi access networks, comparing its performance to other available congestion control schemes and various path schedulers. We use network performance level, as well as video quality level metrics to characterize multiple path schedulers and the resulting network and application layers. We show that BBR video streaming performance degrades for challenging path delay variation scenarios.**

Keywords—Video streaming; TCP congestion control; Multipath TCP; TCP BBR; Packet Scheduler.

I. INTRODUCTION

Reliable data transmission over the Internet relies on transport protocols to regulate data injection so as to control network congestion and avoid uncontrolled data losses along the communication path. In particular, TCP has become the defacto transport protocol of the Internet, supporting reliable data delivery for most applications. Regarding streaming applications, the most dominant type of application in data volume over the Internet, stream quality is related to two factors: the amount of data discarded at the client end point due to excessive transport delay/jitter and data rendering stalls due to lack of timely playout data. Transport delays and data starvation depend heavily on how TCP handles retransmissions upon packet losses during flow and congestion control.

Regarding multipath data delivery, the evolution of portable devices, in particular equipped with multiple high bandwidth interfaces, has motivated the development of the MultiPath Transport Control Protocol (MPTCP), allowing video streaming over multiple IP interfaces and diverse network paths to become reality. Multipath video streaming is attractive because it not only increases aggregated device downloading bandwidth capacity, but also improves transport session reliability during transient radio link impairments in handoff situations. An important function of multipath transport is the selection of a path among various active networking paths (sub-flows), which can be done on a packet by packet basis. However, a path packet scheduler should be designed so as to prevent head-of-line blocking across various networking paths, potentially with diverse loss and delay characteristics. Head-of-line blocking occurs when data already delivered at the receiver has to wait for additional packets that are blocked at another sub-flow, potentially causing incomplete or late frames to be discarded at the receiver, as well as stream rendering stalls. As TCP variants greatly impact streaming quality, we propose to analyze video performance vis-a-vis widely deployed TCP variants, with attention to Bottleneck Bandwidth and Round-trip propagation time (BBR) [1].

The paper is organized as follows. Related work is included in Section II. Section III describes video streaming transport over TCP, with focus on BBR and CUBIC TCP variants. Section VI characterizes video streaming performance over Wi-Fi and Long Term Evolution (LTE) paths via network emulation. We compare the application and network performance of BBR against CUBIC, using a default (shortest delay) path scheduler. Our goal is to uncover unfavorable network scenarios that may lead to the design of new path schedulers. Section VII summarizes our studies and addresses directions we are pursuing as follow up to this work.

II. RELATED WORK

Several multipath transport studies have appeared in the literature, mostly focusing on throughput performance of data transfers over mobile networks (see [2] and related work).

Meanwhile, the BBR TCP variant [1] has gained popularity, becoming Linux variant of choice. Only recent research work has focused on the performance evaluation of BBR in multipath transport. Austria et al. [3] carry an analysis of MPTCP in asymmetric latency subflows, focusing on throughput performance of data transfers. They compare favorably BBR against other TCP variants, such as BIC and CUBIC, on latency asymmetric subpath network scenarios. Our contribution is similar to theirs, but focusing on video streaming applications. In Mahmud et al. [4], a coupling of BBR with MPTCP packet scheduler is proposed with the goal of maintaining high throughput performance while achieving fairness among different subflows. They use emulation of different subflow scenarios to compare their coupled MPTCP throughput and fairness performance against other TCP variants, including Linked Increase Algorithm (LIA) [5], Opportunistic Linked Increase Algorithm (OLIA) [6], Balanced Linked Adaptation algorithm (BALIA) [7]. In contrast, this work evaluates the video streaming performance of BBR and other TCP variants, using various schedulers.

Little research work has focused on video streaming performance over multiple paths. In Matsufuji et al. [8], we evaluate the performance of several TCP variants and path schedulers in transporting video streams over multipath, quantifying frame discards and play stalls. Morawski et al. [10] conduct Linux based experiments of multipath video streaming over Digital Subscriber Line (DSL) path scenarios using LIA, and OLIA, as well as Reno, CUBIC, and BBR TCP variants. They show head-of-line blocking as a major concern. Unfortunately, they do not provide application level performance measures, to evaluate video quality impact. Similarly, Amend et al. [11] evaluate throughput of multipath video streaming DSL multipath scenarios, without providing video level performance measures. Although they also propose a cost optimized scheduler, the lack of video quality performance measures limits conclusions about impact of such scheduler to video quality. Along the same lines, Imaduddin et al. [12] provide a performance evaluation of MPTCP using CUBIC and Vegas TCP variants, as well as minimum Round Trip Time (RTT), round-robin and coupled BALIA schedulers. Focusing on throughput performance, they conclude CUBIC to deliver best performance, regardless of the scheduler. Finally, Xing et al. [13] propose a new MPTCP scheduler which they show via network experiments to lower the number of out-of-order packets. The scheduler estimates receiver arrival times, and sends redundant packets to cope with estimation errors. Video streaming is simulated via $iperf3$, and no application layer performance measures are used. Among all these works, our line of research has focused on application level performance measures in addition to network layer performance indicators such as throughput. In our previous works, we have also introduced multipath path scheduling generic principles, which can be applied in the design of various path schedulers to specifically improve video stream quality. Using these principles, we have introduced in [8] MPTCP path schedulers based on dynamically varying path characteristics, such as congestion window space and estimated path throughput. In addition, in Nagayama et al. [14], we have also proposed to enhance path schedulers with TCP state information, such as whether a path is in fast retransmit and fast recovery states. Finally, in Nagayama et al. [9], we have introduced a novel concept of sticky scheduling, where once a path switch is executed, the scheduler stays with the new path until the path bandwidth resources become exhausted. In this work, we evaluate multipath video streaming using only the default minimum RTT path scheduler, in combination with popular BBR TCP [1] over realistic Wi-Fi/Cellular multipath scenarios, focusing on video quality at the application layer. We seek to determine whether BBR delivers high performance in combination with the default path scheduler over challenging variable delay network scenarios. BBR performance in MPTCP transport is a novelty. Han et al. [15] have recently introduced one such study, where BBR is evaluated in combination with a new adaptive packet scheduling scheme (adaptive redundant + predictive). Multiple copies of a packet are injected in paths of low quality, for reliability improvement. The scheme also predicts packet delivery on paths in order to keep in order delivery, mitigating head-of-line blocking. Their evaluation, however, is limited to throughput and download time performance metrics of files. In our previous evaluation of BBR [16], we have provided an evaluation of BBR when transporting multipath video streams over lossy paths, which showed superior performance vis-a-vis popular TCP variants. In this study, we further evaluate BBR on various path delay characteristics. We show that performance degradation is indeed experienced in some scenarios.

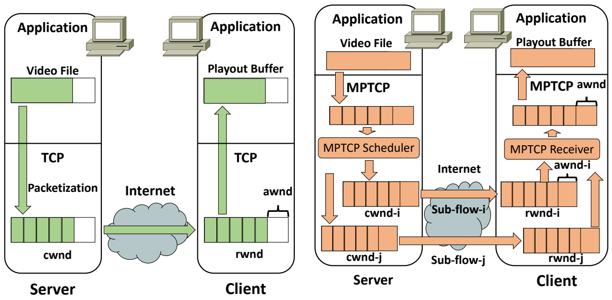
III. Video Streaming Over MPTCP

Video streaming over Hypertext Transfer Protocol/Transmission Control Protocol (HTTP/TCP) originates at a HTTP server storing video content, where video files can be streamed upon HTTP requests over the Internet to video clients. At the transport layer, a TCP variant provides reliable transport of video data over IP packets between the server and client end points (Figure 1). Upon an HTTP video request, a TCP sender is instantiated to transmit packetized data to the client machine, connected to the application via a TCP socket. At the TCP transport layer, a congestion window is used at the sender to control the amount of data injected into the network. The size of the congestion window ($cwnd$) is adjusted dynamically, according to the level of congestion experienced at the network path, as well as space available for data storage ($awnd$) at the TCP client receiver buffer. the congestion window space at the sender is freed only when data packets are acknowledged by the receiver. Lost packets are retransmitted by the TCP layer to ensure reliable data delivery. At the client end, in addition to acknowledging arriving packets, the TCP receiver informs the TCP sender about its current receiver available space, so that the $cwnd \leq awnd$ condition is enforced by the sender at all times to prevent receiver buffer overflow. At the client application layer, a video player extracts data from a playout buffer, which draws packets delivered by the TCP receiver from receiver TCP socket buffer. The playout buffer hence serves to smooth out variable network throughput.

A. MPTCP

MPTCP is an Internet Engineering Task Force (IETF) extension of TCP transport layer protocol to support data transport over multiple concurrent TCP sessions [17]. The network multipath transmission of the transport session is hidden from the application layer by a legacy TCP socket exposed per application session. At the transport layer, however, MPTCP coordinates concurrent TCP sessions on various sub-flows, each of which in itself is unaware of the multipath nature of the application session. In order to accomplish multipath

 ISBN: ISBNFILL



(a) TCP (b) MPTCP

Figure 1. Video Streaming over TCP/MPTCP

transport, a path scheduler connects the application socket with transport sub-flows, extracting packets from the application facing MPTCP socket, selecting a sub-flow for transmission, and injecting packets into the selected sub-flow. the MPTCP transport architecture is depicted in Figure 1 (b).

The first and most used path scheduler, called default scheduler, selects the path with the shortest RTT among paths with currently available congestion window space for new packets. Other path schedulers have appeared recently. These path schedulers can operate in two different modes: uncoupled, and coupled. In uncoupled mode, each sub-flow congestion window $cwnd$ is adjusted independently of other sub-flows. On the other hand, in coupled mode, the MPTCP scheduler couples the congestion control of the sub-flows, by adjusting the congestion window $cwnd_k$ of a sub-flow k according to the current state and parameters of all available sub-flows. Although many coupling mechanisms exist, we focus on the performance study of the Bottleneck Bandwidth and round trip [1] TCP variant over the shortest path RTT scheduler.

Regardless of the path scheduler used, the IETF MPTCP protocol supports the advertisement of multiple IP interfaces available between two endpoints via specific TCP option signalling. IP interfaces may be of diverse nature (e.g., Wi-Fi, LTE). A common signalling issue is caused by intermediate IP boxes, such as firewalls, blocking IP options. Paths that cross service providers with such boxes may require Virtual Private Network (VPN) protection so as to preserve IP interface advertising between endpoints. In addition, multipath transport requires MPTCP stack at both endpoints for the establishment and usage of multiple paths.

IV. TCP VARIANTS

TCP protocol nowadays has branched into different variants, implementing different congestion window adjustment schemes. TCP protocol variants can be classified into delay- and loss-based congestion control schemes. Loss-based TCP variants use packet loss as primary congestion indication signal, typically performing congestion window regulation as $cwnd_k = f(cwnd_{k-1})$, which is ack reception paced. Most f functions follow an Additive Increase Multiplicative Decrease (AIMD) window adjustment scheme, with various increase and decrease parameters. AIMD strategy relies on a cautious window increase (additive) when no congestion is detected, and fast window decrease (multiplicative) as soon as congestion is detected. TCP NewReno [18] and CUBIC [19] are examples of AIMD strategies. In contrast, delay based TCP variants use queue delay information as the congestion indication signal, increasing/decreasing the window if the delay is small/large, respectively. Compound [20] and Capacity and Congestion Probing (CCP) [21] are examples of delay based congestion control variants. Delay based congestion control does not suffer from packet loss undue window reduction due to random, not congestion, packet losses, as experienced in wireless links. Regardless of the congestion control scheme, TCP variants follow a phase framework, with an initial slow start, followed by congestion avoidance, with occasional fast retransmit, and fast recovery phases. BBR congestion control may be considered delay based, since BBR measures the bandwidth and RTT of the bottleneck which a flow goes through [1]. Based on such measurements, BBR adjusts the sending rate to make the best use of the bottleneck bandwidth without dropping its rate during wireless link random losses.

CUBIC TCP Congestion Avoidance: TCP CUBIC is a Loss-based TCP that has achieved widespread usage as the default TCP of the Linux operating system. During congestion avoidance, its congestion window is adjusted as follows (1):

$$\begin{aligned} AckRec: \quad cwnd_{k+1} &= C(t-K)^3 + Wmax \\ K &= (Wmax\frac{\beta}{C})^{1/3} \\ PktLoss: \quad cwnd_{k+1} &= \beta cwnd_k \\ Wmax &= cwnd_k \end{aligned} \tag{1}$$

where C is a scaling factor, Wmax is the cwnd value at time of packet loss detection, and t is the elapsed time since the last packet loss detection. K parameter drives the cubic increase away from Wmax, whereas β tunes how quickly cwnd is reduced on packet loss. This adjustment strategy ensures that its $cwnd$ quickly recovers after a loss event.

BBR TCP Congestion Avoidance: BBR is a bandwidth delay product based TCP that has achieved widespread usage as one of available TCP variants in the Linux operating system. BBR uses measurements of a connection delivery rate and RTT to build a model that controls how fast data may be sent and the maximum amount of unacknowledged data in the pipe. Delivery rate is measured by keeping track of the number of acknowledged packets within a defined time frame. In addition, BBR uses a probing mechanism to determine the maximum delivery rate within multiple intervals.

More specifically, BBR regulates the number of in-flight packets to match the bandwidth delay product of the connection, or $BDP = BtlBw \times RTprop$, where $BtlBw$ is the bottleneck bandwidth of the connection, and RTprop its propagation time, estimated as half of the connection RTT. These quantities are tracked during the lifetime of the connection, as per equations below (2):

 ISBN: ISBNFILL

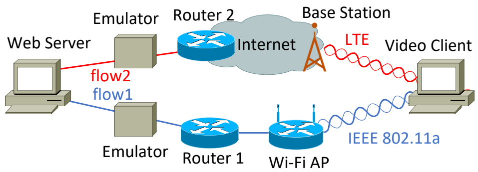


Figure 2. Video Streaming Emulation Network

TABLE I. EXPERIMENTAL NETWORK SETTINGS

Element	Value
Video size	113 MBytes
Video rate	5.24 Mb/s
Playout time	3 mins
Video Codec	H264 MPEG-4 AVC
MPTCP variants	BBR, CUBIC
MPTCP schedulers	Low RTT First (Default)

$$\begin{aligned} RTT_t &= RTprop_t + \eta_t \\ RT\hat{p}rop &= RTprop + min(\eta_t) \\ &= min(RTT_t) \forall t \in [T - W_R, T] \\ Btl\hat{B}w &= max(deliveryRate_t) \forall t \in [t - W_B, T] \end{aligned} \quad (2)$$

where η_t represents the noise of the queues along the path, W_R a running time window, of tens of seconds, and W_B a larger time window, of tens of RTTs. This adjustment strategy seeks to tune its $cwnd$ to a number of packets equivalent to the connection bandwidth delay product.

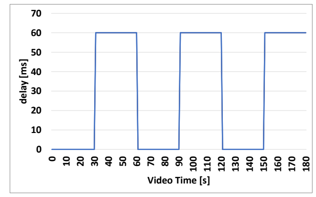
V. MPTCP with TCP BBR

A MPTCP scheduler selects a sub-flow to inject packets into the network on a packet by packet basis. The default strategy is to select the path with shortest average round trip packet delay, hereafter called LRF. If a short and non-congested path exists between the end points, it becomes the preferred path for data transport. Non-congested path is defined as a path for which its congestion window ($cwnd$) has available space among packets yet to be acknowledged by the receiver. Hence, a congested path will have no space for more unacknowledged packets to be injected.

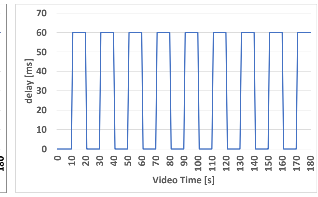
As BBR sizes its $cwnd$ according to path bandwidth delay product, paths with large delays and high bandwidth result in large $cwnd$. Even though the scheduler may favor a path with smaller bandwidth delay product, as it looks at the path RTT only, if such path has low bandwidth availability BBR will size its $cwnd$ to a small value as compared to high bandwidth paths, effectively blocking low bandwidth paths from scheduling selection, and forcing a high bandwidth delay product path to be used.

VI. Video Streaming Performance over Wi-Fi/LTE

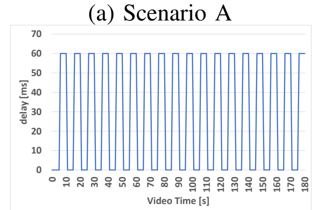
Figure 2 describes the network testbed used for emulating network paths with Wi-Fi and LTE wireless access links. An HTTP Apache video server is connected to two L3 switches, one of which directly connected to an 802.11a router, and the other connected to an LTE base station via a cellular network card via emulator boxes. Since the bandwidth of IEEE 802.11a is sufficiently large for the bit rate of video, we have adopted 802.11a as the wireless LAN interface. In this paper, the emulator boxes are used to vary each path RTT. No packet loss is injected. The simple topology and isolated traffic allow us to better understand the impact of differential delays on TCP variant's performance.



(a) Scenario A



(b) Scenario B



(b) Scenario C

Figure 3. Wi-Fi Delay Dynamic Change Scenario

TABLE II. EXPERIMENTAL NETWORK SCENARIOS

Scenario	Emulator (BW, Packets Loss)
A - Slow Wi-Fi delay cycle Scenario A delay pattern	LTE: BW 3Mbps, Loss 0% Wi-Fi: BW 3Mbps, Loss 0% DCycle 60secs
B - Baseline Wi-Fi delay cycle Scenario B delay pattern	LTE: BW 3Mbps, Loss 0% Wi-Fi: BW 3Mbps, Loss 0% DCycle 20secs
C - Fast Wi-Fi delay cycle Scenario C delay pattern	LTE: BW 3Mbps, Loss 0% Wi-Fi: BW 3Mbps, Loss 0% DCycle 10secs

Network settings and scenarios under study are described in Tables I and II, respectively. Video settings are typical of a video stream, with video playout rate of 5.24 Mb/s, and size short enough to run multiple streaming trials within a short amount of time. Three Wi-Fi packet delay pattern scenarios are used (Figure 3). Scenario A represents streaming sessions with steady Wi-Fi delays, with ON 60 msec cycles of 60 seconds. Scenario B represents a baseline Wi-Fi delay scenario, where 60 msec delay happen at 20 sec cycles. Scenario C represents a highly variable delay scenario, where 60 msec delay is experienced at a faster 10 second cycles. Emulator boxes are tuned to generate various multiple path network conditions, and have been selected as per Table II to represent LTE/Wi-Fi streaming situations at home. TCP variants used are: CUBIC and BBR. Performance measures are:

- **Picture discards:** number of frames discarded by the video decoder.
- **Buffer underflow:** number of buffer underflow events at video client buffer.
- **Sub-flow throughput:** TCP throughput on each sub-flow.
- **Sub-flow cwnd:** TCP cwnd value on each sub-flow.

We organize our video streaming experimental results in network scenarios summarized in Table II: A- A Wi-Fi-Cellular (LTE) scenario A high frequency delay cycle; B- A Wi-Fi-Cellular scenario B, where a slightly larger Wi-Fi delay cycles is assumed as baseline; C- A Wi-Fi-Cellular with

 ISBN: ISBNFILL

scenario C high cycle delay pattern.

A. Slow delay on Scenario A delay pattern

Scenario A delay represents slow varying path delays, emulating a transition between stable Wi-Fi low load connections. Figures 4(a) and (b) report on video streaming buffer underflow and picture discard performance when Wi-Fi delay is slow varying about 60 ms. Video performance is excellent for both BBR and CUBIC TCP variants (single buffer underflow event and zero picture discards). Figures 5(a) and (b) report of Wi-Fi $cwnd$ dynamics of BBR and CUBIC TCP variants. We can see that BBR enforces a much reduced Wi-Fi $cwnd$ than CUBIC, still delivering excellent video performance. BBR $cwnd$ size tracks nicely delay cycles, differently from CUBIC, which, being a loss based variant, is insensitive to delay variations. Moreover, Figures 5(c) and (d) show similar levels of LTE (cellular) and Wi-Fi path throughput for both TCP variants, showing an equal share of the two paths available, despite Wi-Fi path delay variability.

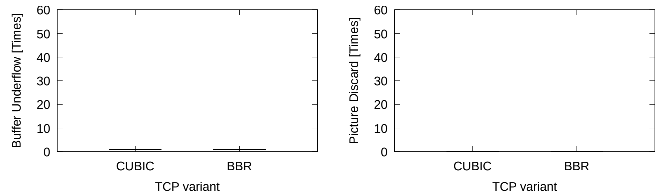
B. Baseline delay on Scenario B delay pattern

Scenario B delay represents medium varying path delays, taken as a baseline Wi-Fi scenario. Figures 6(a) and (b) report on video streaming buffer underflow and picture discard performance when Wi-Fi delay is varying on 20 seconds cycles of 60 ms delay. Video performance degrades for both TCP variants, with BBR delivering less buffer underflows and more picture discards than CUBIC. Figures 7(a) and (b) report of Wi-Fi $cwnd$ dynamics of BBR and CUBIC TCP variants. Again, BBR enforces a much reduced Wi-Fi $cwnd$ than CUBIC, still tracking delay cycles nicely. Moreover, Figures 7(c) and (d) show BBR delivering a reduced level of Wi-Fi throughput than CUBIC, using more LTE bandwidth than CUBIC.

C. Small delay on Scenario C delay pattern

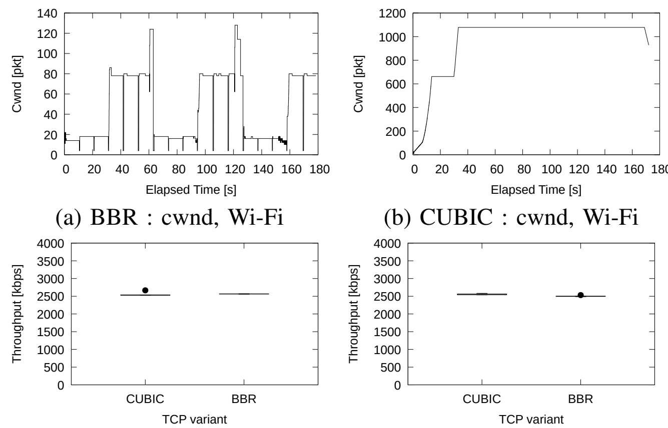
Scenario C delay represents highly varying path delay Wi-Fi scenario. Figures 8(a) and (b) report on video streaming buffer underflow and picture discard performance when Wi-Fi delay is varying on 10 seconds cycles of 60 ms delay. Video performance degrades significantly for BBR, whereas CUBIC delivers video performance comparable with previous scenario. Figures 9(a) and (b) report of Wi-Fi $cwnd$ dynamics of BBR and CUBIC TCP variants, and helps explain BBR performance degradation. BBR is no longer able to track delay cycles as before, remaining ”stuck” at a small $cwnd$ of 15 packets. Moreover, Figures 9(c) and (d) show BBR delivering a much reduced level of Wi-Fi throughput than CUBIC, and not being able to compensate enough with more LTE bandwidth than CUBIC. The overall video connection, therefore, gets starved when served by BBR.

Finally, Figure 10 reports on the number of path switches of both TCP variants on the three scenarios investigated. We see that the number of path switches when using BBR is as much as four times larger than CUBIC. This is because there is a stress between the delay sensitivity of BBR vs the default scheduler minimum delay path selection. That is, a delay variation causes BBR to shrink its $cwnd$, effectively blocking that path quickly, pushing the scheduler to switch to LTE path. However, as soon as there is any room in the Wi-Fi path, the scheduler switches paths back to Wi-Fi. This indicates that a scheduler that follows BBR bandwidth and delay estimation may be able to work more harmoniously with BBR variant.



(a) Buffer Underflow (b) Picture Discard

Figure 4. Scenario A - Video Performance



(a) BBR : cwnd, Wi-Fi (b) CUBIC : cwnd, Wi-Fi

(c) Throughput, LTE (d) Throughput, Wi-Fi

Figure 5. Scenario A - Transmission Performance

VII. Conclusion

We have studied BBR transport performance of video streaming on multipath cellular/Wi-Fi mixed scenarios. We have shown that on rapidly varying path delay scenario, BBR TCP variant delivers a degraded video streaming performance. Under this fast delay variation, BBR remains at a shrunk congestion window situation that effectively reduces considerably the path throughput. These early results seem to point to an opportunity of designing a path scheduler that is better tuned to delay and bandwidth delay product based TCP variants such as BBR. For instance, path scheduler could select the path with larger bandwidth delay product, rather than lower delay of default scheduler. We are currently designing one such scheduler, where BBR estimators are used to drive path selection.

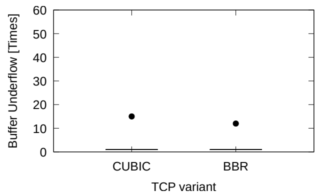
Acknowledgments

Work supported by JSPS KAKENHI Grant #21H03430, and National Institute of Information and Communication Technology (NICT).

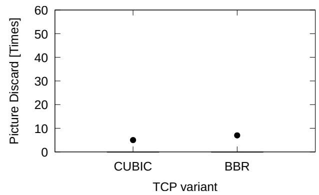
References

[1] N. Cardwell, Y. Cheng, I. Swett, and V. Jacobson, “BBR Congestion Control,” *IETF draft-cardwell-iccrg-bbr-congestion-control-01*, November 2021.

[2] M. R. Palash et al., “MPWiFi: Synergizing MPTCP Based Simultaneous Multipath Access and WiFi Network Performance,” IEEE Transactions on Mobile Computing, vol. 19, no. 1, pp. 142-158, Jan. 2020.

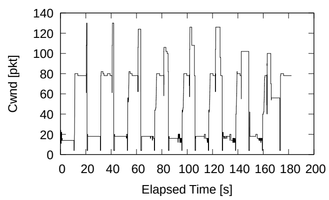


(a) Buffer Underflow

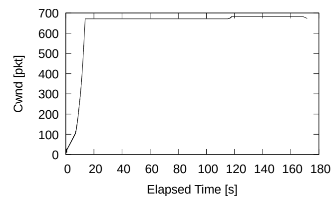


(b) Picture Discard

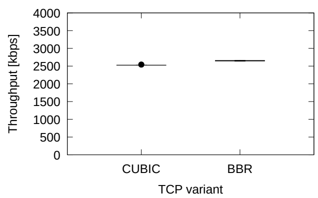
Figure 6. Scenario B - Video Performance



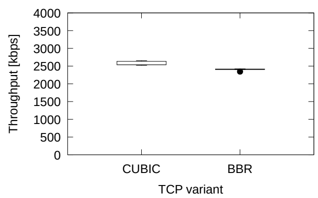
(a) BBR : cwnd, Wi-Fi



(b) CUBIC : cwnd, Wi-Fi

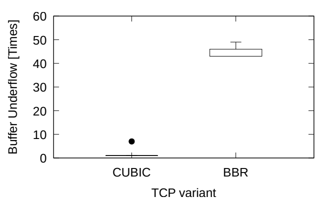


(c) Throughput, LTE

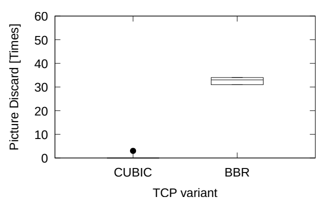


(d) Throughput, Wi-Fi

Figure 7. Scenario B - Transmission Performance

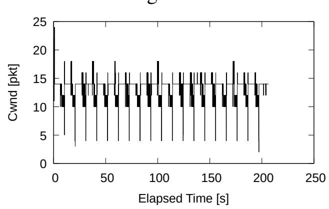


(a) Buffer Underflow

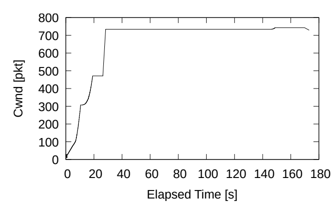


(b) Picture Discard

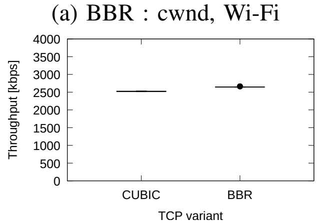
Figure 8. Scenario C - Video Performance



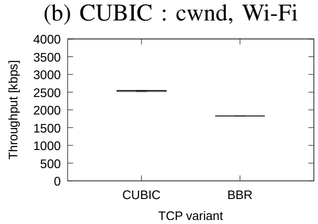
(a) BBR : cwnd, Wi-Fi



(b) CUBIC : cwnd, Wi-Fi

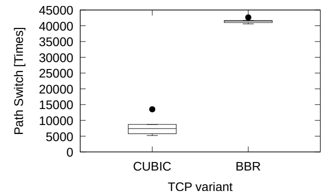


(c) Throughput, LTE

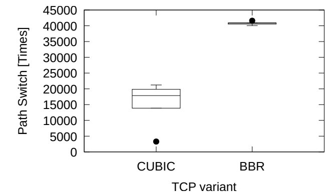


(d) Throughput, Wi-Fi

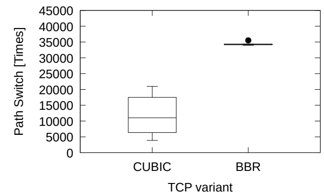
Figure 9. Scenario C - Transmission Performance



(a) Scenario A : Path Switch



(b) Scenario B : Path Switch



(c) Scenario C : Path Switch

Figure 10. All Scenarios - Path Switch

[3] P. Austria et al., “BBR Congestion Control Analysis with Multipath TCP (MPTCP) and Asymmetrical Latency Subflow,” In Proc. of IEEE 12th Annual Computing and Communication Workshop - CCWC, pp. 1065-1069, January 2022.

[4] I. Mahmud et al., “Coupled Multipath BBR (C-MPBBR): A Efficient Congestion Control Algorithm For Multipath TCP,” IEEE Access, vol. 8, pp. 165497-165511, Sept. 2020.

[5] C. Raiciu, M. Handly, and D. Wischik, “Coupled Congestion Control for Multipath Transport Protocols,” IETF RFC 6356, 2011.

[6] R. Khalili, N. Gast, and J-Y Le Boudec, “MPTCP Is Not Pareto-Optimal: Performance Issues and a Possible Solution,” IEEE/ACM Trans. on Networking, vol. 21, no. 5, pp. 1651-1665, Aug. 2013.

[7] A. Walid et al., “Balanced Linked Adaptation Congestion Control Algorithm for MPTCP,” IETF draft-walid-mptcp-congestion-control, 2014.

[8] R. Matsufuji et al., “Multipath TCP Packet Schedulers for Streaming Video,” IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM) , August 2017, pp. 1-6.

[9] S. Nagayama et al., “Path Switching Schedulers for MPTCP Streaming Video,” IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM) , August 2019, pp. 1-6.

[10] M. Morawski and P. Ignaciuk, “A Price to Pay for Increased Throughput in MPTCP Transmission of Video Streams,” In Proc. of 24th Intern. Conference on System Theory, Control and Computing - ICSTCC, pp. 673-678, October 2020.

[11] M. Amend et al., “Cost optimized multipath scheduling in 5G for Video-on-Demand traffic,” In Proc. of IEEE Wireless Communications and Networking Conference - WCNC 21, March 2021.

[12] M. F. Imaduddin et al., “Multipath TCP Scheduling Performance Analysis and Congestion Control on Video Streaming on the MPTCP Network,” In Proc. of Intern. Conference on Software Engineering & Computer Systems and 4th Intern. Conference on Computational Science and Information Management - ICSECS-ICOCSIM, pp. 562-567, August 2021.

[13] Y. Xing et al., “A Low-Latency MPTCP Scheduler for Live Video Streaming in Mobile Networks,” IEEE Transactions on Wireless Communications, vol. 20, no. 11, pp. 7230-7242, Nov. 2021.

[14] S. Nagayama et al., “TCP State Driven MPTCP Packet Scheduling for Streaming Video,” IARIA 10th International Conference on Evolving Internet, pp. 9-14, June 2018.

[15] J. Han et al., “Leveraging Coupled BBR and Adaptive Packet Scheduling to Boost MPTCP,” IEEE Transactions on Wireless Communications, vol. 20, no. 11, pp. 7555-7567, November 2021.

[16] M. Kondo et al., “Evaluation of MPTCP with BBR Performance on Wi-Fi/Cellular networks for Video Streaming,” IARIA 14th International Conference on Evolving Internet, pp. 6-11, May 2022.

[17] A. Ford et al., “Architectural Guidelines for Multipath TCP Development,” IETF RFC 6182, 2011.

[18] M. Allman, V. Paxson, and W. Stevens, “TCP Congestion Control,” IETF RFC 2581, April 1999.

[19] I. Rhee, L. Xu, and S. Ha, “CUBIC for Fast Long-Distance Networks,” Internet Draft, draft-rhee-tcpm-ctcp-02, August 2008.

[20] M. Sridharan, K. Tan, D. Bansal, and D. Thaler, “Compound TCP: A New Congestion Control for High-Speed and Long Distance Networks,” Internet Draft, draft-sridharan-tcpm-ctcp-02, November 2008.

[21] D. Cavendish, K. Kumazoe, M. Tsuru, Y. Oie, and M. Gerla, “Capacity and Congestion Probing: TCP Congestion Avoidance via Path Capacity and Storage Estimation,” IEEE Second International Conference on Evolving Internet, pp. 42-48, September 2010.